JYVÄSKYLÄ STUDIES IN COMPUTING
176

Lassi Paavolainen

Algorithms and Software for Biological Multiscale Image Analysis

JYVÄSKYLÄN YLIOPISTO

JYVÄSKYLÄ STUDIES IN COMPUTING 176

Lassi Paavolainen

Algorithms and Software for Biological Multiscale Image Analysis

Esitetään Jyväskylän yliopiston informaatioteknologian tiedekunnan suostumuksella
julkisesti tarkastettavaksi yliopiston Agora-rakennuksen auditoriossa 3
joulukuun 18. päivänä 2013 kello 12.

Academic dissertation to be publicly discussed, by permission of
the Faculty of Information Technology of the University of Jyväskylä,
in building Agora, Auditorium 3, on December 18, 2013 at 12 o'clock.

UNIVERSITY OF JYVÄSKYLÄ

JYVÄSKYLÄ 2013

Algorithms and Software for Biological Multiscale Image Analysis

JYVÄSKYLÄ STUDIES IN COMPUTING 176

Lassi Paavolainen

Algorithms and Software for Biological Multiscale Image Analysis

UNIVERSITY OF JYVÄSKYLÄ

JYVÄSKYLÄ 2013

Editors
Timo Männikkö
Department of Mathematical Information Technology, University of Jyväskylä
Pekka Olsbo, Sini Tuikka
Publishing Unit, University Library of Jyväskylä

URN:ISBN:978-951-39-5495-6
ISBN 978-951-39-5495-6 (PDF)

ISBN 978-951-39-5494-9 (nid.)
ISSN 1456-5390

Jyväskylä University Printing House, Jyväskylä 2013

ABSTRACT

Paavolainen, Lassi
Algorithms and Software for Biological Multiscale Image Analysis
Jyväskylä: University of Jyväskylä, 2013, 112 p.(+included articles)
(Jyväskylä Studies in Computing
ISSN 1456-5390; 176)
ISBN 978-951-39-5494-9 (nid.)
ISBN 978-951-39-5495-6 (PDF)
Finnish summary
Diss.

Microscopy imaging is essential for studying dynamic cellular processes, viruses and the organization of macromolecular complexes in nanometer and micrometer-scale. Recent developments in imaging techniques have increased the complexity of the imaging data in biological experiment. This has created a need for software and automatic image analysis methods that function in high-throughput fashion. In this work, an open-source BioImageXD software for bioimage processing, analysis and visualization is presented. The software was shown to fulfill general criteria for scientific image analysis software. Two novel image analysis methods were developed to fluorescence microscopy imaging data. A feature-based particle tracking method was developed and applied in the $\alpha 2\beta 1$ integrin clustering study. It was shown to give more accurate results in highly clustering dataset than a state-of-the-art single-particle tracking method. In addition, a novel object-based association method used to analyze interconnection of particles was developed. The studies showed the association method to be robust with different association ratios, and to give more accurate results than other tested object-based and pixelwise methods. Two new methods for structural studies were developed. Effects of the missing wedge in electron tomography was studied. Also, a new reconstruction method for biological limited-angle electron tomography was tested. The method improved the resolution and reduced the artifacts, as compared to the traditional reconstruction methods used in electron tomography. In another structural study, a novel framework for segmenting particles in extremely noisy cryo-electron microscopy imaging data was developed to reduce manual workload and to improve accuracy. The method showed high recall and low false detection rate in the study of segmenting Simian virus 40 particles.

Keywords: BioImageXD, bioimage analysis, bioimage informatics, single-particle tracking, colocalization, single-particle reconstruction, electron tomography, segmentation, simulated data, method validation, fluorescence microscopy, electron microscopy

Author

Lassi Paavolainen
Department of Biological and Environmental Science
Department of Mathematical Information Technology
University of Jyväskylä
Finland

Supervisors

Professor Tuomo Rossi
Department of Mathematical Information Technology
University of Jyväskylä
Finland

Docent Varpu Marjomäki
Department of Biological and Environmental Science
University of Jyväskylä
Finland

Reviewers

Professor Jari Hyttinen
Department of Electronics and Communications Engineering
Tampere University of Technology
Finland

Docent Lasse Lensu
Department of Mathematics and Physics
Lappeenranta University of Technology
Finland

Opponent

Professor Jean-Christophe Olivo-Marin
Department of Cell Biology and Infection
Quantitative Image Analysis Unit
Institut Pasteur, Paris
France

ACKNOWLEDGEMENTS

This work was carried out at the University of Jyväskylä, at the Department of Biological and Environmental Science and at the Department of Mathematical Information Technology. This work was funded by the Academy of Finland, Tekes, and COMAS graduate school, the Department of Biological and Environmental Science and the Department of Mathematical Information Technology at the University of Jyväskylä

I would like to thank my supervisors Professor Tuomo Rossi and Docent Varpu Marjomäki. I have been privileged to work in Varpu's research group for many years. The team spirit in the group has been excellent all the time, which is largely due to Varpu's personality. I would not be here without Tuomo who took me to work in his conference project almost 10 years ago. Tuomo was also my Master's thesis supervisor, and the one who introduced me to Varpu to work in the BioImageXD project.

I want to express my gratitude to the reviewers, Docent Lasse Lensu and Professor Jari Hyttinen, for reviewing this work in a tight schedule. Your comments and suggestions made this work much better. I am grateful to Professor Jean-Christophe Olivo-Marin for taking the responsibility to act as the opponent.

I have been fortunate to work with the best colleagues and collaborators. Professor R. Holland Cheng has worked as unofficial supervisor for my electron microscopy work. He was also kind to invite me to visit his lab at the UC Davis, California, USA. I want to thank Mikko and Moona for being great company in the office and also outside of the office. I want to acknowledge my closest collaborators Pasi and Pekka for great work, friendship, and also for sharing unforgettable conference trips with me to Iceland, to Barcelona, and to the West Coast of the USA. I am also deeply grateful to all other collaborators who have been part of this work. I would also like to thank Dr. Steve Legrand for checking the language of the dissertation.

Many great people have worked or are working in Varpu's group. I want to thank former group members Elina, Nina and Paula, and current group members Paula, Artur, Mari, Maria, Marie, Mira, Anni and Ganna. Also, the ”old gang” of corridor C2 was formed back in the year 2007. I want to thank especially Salla, Jarkko, Heikki, Elisa and Mikko for many enjoyable moments. Thanks to all other colleagues as well.

I want to thank my family for giving me the grounds to work for my interests. I am grateful to all my friends who have given me nice counterbalance to my work. Finally, I want to express my gratitude to Anita for everything. Without your support, I would not have been able to succeed. Thank You!

ACRONYMS

2D	Two-dimensional
3D	Three-dimensional
COM	Center of mass
CLSM	Confocal laser scanning microscopy
CryoEM	Cryo-electron microscopy
DoG	Difference of Gaussians
ET	Electron tomography
FBP	Filtered backprojection
ITK	Insight Segmentation and Registration Toolkit
ICP	Iterative Closest Point
MAP-EM	Maximum a posterior expectation maximization
PPM	Point-pattern matching
PSF	Point spread function
SNR	Signal-to-noise ratio
SPR	Single-particle reconstruction
SPT	Single-particle tracking
SIRT	Simultaneous iterative reconstruction technique
TEM	Transmission electron microscopy
VTK	Visualization Toolkit
WBP	Weighted backprojection

LIST OF FIGURES

LIST OF TABLES

LIST OF ALGORITHMS

CONTENTS

LIST OF INCLUDED ARTICLES

PI Pasi Kankaanpää, Lassi Paavolainen, Silja Tiitta, Mikko Karjalainen, Joacim Päivärinne, Jonna Nieminen, Varpu Marjomäki, Jyrki Heino & Daniel J White. BioImageXD: an open, general-purpose and high-throughput image-processing platform. *Nature Methods 9(7)*, 2012.

PII Lassi Paavolainen, Pasi Kankaanpää, Pekka Ruusuvuori, Gregory McNerney, Mikko Karjalainen & Varpu Marjomäki. Application independent greedy particle tracking method for 3D fluorescence microscopy image series. *9th IEEE International Symposium on Biomedical Imaging (ISBI)*, 2012.

PIII Pekka Ruusuvuori*, Lassi Paavolainen*, Kalle Rutanen*, Anita Mäki, Heikki Huttunen & Varpu Marjomäki. Quantitative analysis of dynamic association in live biological fluorescent samples. *Submitted manuscript*.

PIV Lassi Paavolainen*, Erman Acar*, Uygar Tuna, Sari Peltonen, Pan Soonsawad, Varpu Marjomäki, R. Holland Cheng & Ulla Ruotsalainen. Sequential statistical reconstruction for electron tomography with missing wedge. *Manuscript*.

*Equal contribution

1 INTRODUCTION

The size scale of biological populations, organisms and structures is huge. Macroscopic imaging modalities are used to study macroscopic objects, such as giant redwood forests, animals and human organs, in kilometer to millimeter-scale. Other imaging modalities are used to study microscopic objects, such as cells and viruses, visible only with microscopy imaging methods in micrometer and nanometer-scale. The scale in microscopy is a million-fold from structural studies measured in Ångströms (1 Å = 0.1 nm) to cell populations of hundreds of micrometers. In this work, research on algorithms and software for biological image analysis is done in multiple microscopic scales, from tens of micrometers to Ångströms, in cellular biology and in structural biology.

Imaging is an essential part of modern molecular and cellular biology research. Imaging offers great tools to understand dynamic cellular processes (Eils and Athale, 2003), such as material uptake and cell signaling, as well as overall organization of macromolecular structures in cells. Live cellular imaging enables research on the mechanisms of cellular functions (Stephens and Allan, 2003). This knowledge can be used in a wide variety of medical applications such as development of new drugs and treatments (Kitano, 2002).

Recent developments on microscopy imaging techniques, computer technology, and image analysis methods have turned bioimaging from qualitative to quantitative science. Quantitative bioimaging is an essential part of modern cellular biology research. However, turning images to quantitative measures still holds many challenges. Modern microscopes produce vast amounts of multidimensional image data, which is unfeasible for a human observer to analyze. Need for automatic and robust image analysis methods is clear in all areas of bioimaging, and especially in high-throughput applications (Pepperkok and Ellenberg, 2006; Zhou and Wong, 2006). Lack of *a priori* information of imaged cellular structure and large variance between images separate cell biological image analysis from many macroscale and medical image analysis applications. Low image quality and the fact that the resolution of diffraction-limited optical microscope is much worse than the size of many imaged particles make applications such as localization and tracking of particles challenging.

Bioimage informatics software has long been a limiting factor in quantitative bioimaging (Wilt et al., 2009). Over the years, analysis methods have been described in methodological sections of biological publications without release of the method implementation (Cardona and Tomancak, 2012). Reasons for this have been several. Often closed commercial products have been used or reproducibility of the image analysis results have not been considered as important as reproducibility of the biological experiment described thoroughly. Recently, attitude in the field has been changing towards more open and reproducible imaging also in the requirements of many top journals (Ince et al., 2012). Development of open-source general-purpose bioimage informatics software (Eliceiri et al., 2012) readily able to solve many bioimage analyses tasks have had its positive impact on change towards reproducible quantitative bioimaging.

1.1 Objectives of the research

The first goal is to develop a general-purpose bioimage informatics software for cellular and molecular biology research. The main requirements for the software are scientific accuracy, openness and applicability to a wide range of multidimensional bioimaging applications in different scales. The aim is that the software offers a framework to include new image analysis methods usable in high-throughput fashion. Also, the software is used in the research of all other objectives.

The second objective is to develop a particle tracking method, as a solution to the problem of tracking clustering particles. The main application of the method is on tracking clustering $\alpha 2\beta 1$ integrin clusters in multidimensional fluorescence microscopy image time series. For the application, the method has to enable tracking of clustering particles. In addition, the aim is to develop a general-purpose particle tracking algorithm that can be configured for the needs of the application.

The third goal is to study methods to analyze interconnection between particles in different imaging channels in fluorescence microscopy image data. Currently, no commonly used method exist to analyze particle association. Colocalization methods (Manders et al., 1992, 1993) are designed to analyze direct signal overlap. However, these methods are also used to analyze interaction between particles (Comeau et al., 2006). The aim is also to study the applicability of colocalization methods for association analysis.

The fourth objective is to study the effects of the missing wedge in widely used electron tomography (ET) methods and in a new method applied to ET. The aim is to develop ET reconstruction method to minimize the effects of the missing wedge.

The fifth objective is to develop a reference-free automatic segmentation method for particle selection problem in single-particle reconstruction (SPR). The aim of the method is to minimize false particle detections to make labor-intensive

and biased manual post-selection pruning of the particles unnecessary. The method should be applicable for a wide range of particles.

1.2 Structure of the thesis

In this work, software and algorithms for multiscale bioimage analysis are presented. The structure of the thesis is as follows. After introduction in Chapter 1, basics of fluorescence and electron microscopy imaging techniques in cellular biology are presented in Chapter 2. Two most common fluorescence microscopy techniques, widefield microscopy and confocal microscopy, are introduced and their strengths and weaknesses are compared in three-dimensional (3D) and live optical imaging. Next, a two-dimensional (2D) transmission electron microscopy (TEM) is described, followed by ET, an application of TEM to high resolution 3D imaging.

Chapter 3 elaborates on quantitative bioimage analysis with focus on automatic methods for fluorescence microscopy. First, description of image segmentation, which has a considerable role in many image analysis methods, is given. Segmentation methods for different scales from sub-resolution particles to whole cells are reviewed. The section about segmentation is concluded in Subsection 3.1.2 with descriptions of segmentation methods used in Publications PI–PIII. Next, widely used colocalization analysis is presented. Some weaknesses of existing pixelwise methods are described, followed by introduction and the main results of object-based particle association method of Publication PIII also usable for colocalization analysis is presented in Subsection 3.2.4. Next, the idea of tracking particles in live specimen is explained with literature review on single-particle tracking (SPT). Lastly, a general feature-based tracking method with the main results of Publication PII is presented in Subsection 3.3.3. Chapter 3 is concluded with discussion on method validation, which is an important part of automatic image analysis.

A SPR technique used to solve structures of macromolecular complexes is introduced in Chapter 4. Manual or semi-automatic particle selection is the most labor-intensive part of SPR. Next, published methods for particle selection are reviewed. The chapter ends with a presentation of novel particle selection framework and its application to simulated and experimental data of Simian virus 40 particles.

Chapter 5 discusses bioimage informatics software which has been the limiting factor in bioimage analysis and visualization. Common requirements and challenges of bioimage informatics software are discussed. An overview of BioImageXD software published in Publication PI is presented in Section 5.1 with a design criteria for BioImageXD development. The chapter is concluded with a review on research articles citing BioImageXD and a brief overview of other bioimage informatics software. Finally, the thesis is concluded in Chapter 6 with conclusions and consideration of future work.

1.3 Main contribution

I was responsible for the development of BioImageXD software presented in Publication PI. Since the beginning of year 2009, I have been the lead software engineer in the BioImageXD project being responsible for all software engineering activities. I developed new algorithms used in Publication PI, and together with Pasi Kankaanpää, defined analysis protocols and did most of the analyses. I created simulated data used to validate analysis methods in the publication. I did half of the software comparisons. I took part on designing the publication, and wrote the sections in supplementary materials and methods about the BioImageXD development, simulated data, and analysis methods.

I was responsible for designing the study in Publication PII. I wrote the article and was the main developer of the method. I also created the simulated data used to validate the method, and did analysis for simulated and experimental data.

I took part on designing the study in Publication PIII. I designed and created simulated data used to validate the method. I was responsible for designing and doing colocalization analyses with pixelwise methods. I also extracted the point-sets from the experimental data used to analyze association with the presented and compared methods. In addition, I took part on writing the sections of my work in the manuscript.

I took part on designing the study in Publication PIV. I wrote the manuscript, except the parts about the developed and compared tomography methods. I designed and created simulated data to test the methods. I did all analyses from the reconstructed volumes.

In addition to the Publications PI–PIV, a novel method for particle selection in SPR is presented with the results in Section 4.3. I was responsible for designing the study and developed the method. I also created all simulated data to validate the method, and did all analyses.

In addition to the publications included in this work, the author has used the presented methods and software in following publications: Björkbom et al. (2011); Rintanen et al. (2012); Wu et al. (2013); Kaakinen et al. (2014). These methods and software were also presented by the author at the international conferences of Annual Conference of the Nordic Microscopy Society (Copenhagen, Denmark, 2013) and Annual Conference of the Nordic Microscopy Society (Bergen, Norway, 2012), 9th IEEE International Symposium on Biomedical Imaging (Barcelona, Spain, 2012), Annual Conference of the Nordic Microscopy Society (Oulu, Finland, 2011), 3rd Annual Conference of the Nordic Network on Imaging in Biology and Medicine (Gothenburg, 2010), Annual Conference of the Nordic Microscopy Society (Reykjavik, Iceland, 2009), 8th European Light Microscopy Initiative meeting (Davos, Switzerland, 2008), and at the national meetings of Virus-Cell Interactions meeting (Konnevesi, 2012) and Workshop on Industrial Image Processing (Jyväskylä, 2008), and at the seminars in the Department of Biological and Environmental Science (University of Jyväskylä, Finland,

2009–2012) and the Department of Mathematical Information Technology (University of Jyväskylä, Finland, 2011–2012).

2 MICROSCOPY IMAGING TECHNIQUES AND 3-DIMENSIONAL RECONSTRUCTION

People tend to think of a basic optical transmission microscope with bright field illumination when they hear the word *microscope*. Even though the principles of optical microscope used in modern cell biological research are similar, the equipment is far more complex with advanced illumination techniques, optics and electronics. Instead of visible light, the specimen can also be interacted with electron beam, with other electromagnetic radiation such as X-rays, or with a probe. Altogether tens of imaging methods are available that all produce images in different contrast and resolution. All methods have their strengths and weaknesses, which are taken into account while considering the best possible, and available, imaging method for an application. In this work, optical, mainly fluorescence, and electron microscopy are used. These are the most used imaging modalities in cellular biology.

Cell biological samples are imaged either live or after fixation. In live imaging, cells are kept alive during imaging while damage to the sample is prevented by all means possible. Fixation means that the sample is killed by stopping its metabolism while preserving microstructures larger than the size defined by the fixation process (Bacallao et al., 2006). Fixation can be done in many ways, but careful consideration of the method is needed as a wrong method can create artifacts in interesting parts of the sample.

Image formation in microscope is a convolution of the object function $f(x)$, defining the imaged specimen, and the point spread function (PSF) $h(x)$ of the microscope. Imaging of each point-like subvolume of the specimen can be modeled as (van Kempen et al., 1997; Verveer et al., 1999):

$$g(x) = N\left(\int_X h(x-\chi)f(\chi)d\chi\right), \tag{1}$$

where $x \in X$ is a coordinate in n-dimensional space X, $g(x)$ is the formed image and $N(\cdot)$ the noise function. An ideal imaging system forms an infinitely small symmetrical spot in the image from an infinitely small symmetrical spot in the

specimen. In practice, the spot is blurred by the PSF of the imaging device, and final image is impacted by the specimen and different sources of noise.

In this chapter, two common imaging modalities, optical fluorescence microscopy and TEM, are introduced. First, both 2D and 3D fluorescence microscopy methods for studying fixed and live cellular structures and events (Yuste, 2005) are presented. Second, 2D TEM for studying fixed cellular structures in greater detail is presented. Third, introduction to ET (Lučić et al., 2005) for reconstructing a 3D volume from many TEM images is given, followed by preliminary results of Publication PIV.

2.1 Fluorescence microscopy

Fluorescence microscopy is a general term used for optical (light) microscopy methods where fluorophores are used to selectively mark interesting components for imaging. In cell biology, fluorophores are used as contrast agents to practically transparent cells and subcellular structures, and to make visible particles smaller than the resolution of the optical microscope (Vonesch et al., 2006). Nowadays, fluorescence microscopy is a fundamental part of modern biomedical research.

Physically, a fluorophore is a molecule that can absorb a photon of excitation light and almost instantly emits a photon (in nanoseconds) (Johnson and Spence, 2010). The event is generated as the particle raises into higher energy level by excitation. When the particle returns to the initial lower energy level, it emits a photon. During this process, energy is usually lost as heat vibration. According to Planck's relation $E = hc/\lambda$, where h is Planck constant, c and λ are the speed and the wavelength of light in vacuum (Lichtman and Conchello, 2005), emission wavelength is longer than excitation wavelength when energy is lost in fluorophore. Each fluorophore has characteristic excitation and emission spectra that defines the range of usable wavelengths to illuminate the sample and to detect the emission signal from fluorophores. Typically, an excitation light wavelength is selected to be the peak of excitation spectrum and wavelengths close to emission spectrum peak are collected to form an image. The difference in excitation and emission spectra peaks, called *Stokes shift*, makes simultaneous illumination and imaging of fluorophores possible. Even though excitation and emission spectra overlap, excitation light can be blocked from reaching the detector by correct selection of optics.

Common fluorophores used in cell biology are proteins, such as the famous green fluorescence protein (Chalfie et al., 1994), used in several ways to highlight specific subcellular components. Often, a specific target is labeled directly or by using immunofluorescence (Buchwalow and Böcker, 2010). In immunofluorescence, either a primary antibody of target protein is labeled with a fluorophore or the primary antibody is conjugated with a secondary antibody that is labeled (Buchwalow and Böcker, 2010). With these methods, subcellular structures of interest, such as nucleic acids, cell cytoplasm or integrin proteins,

can be selectively labeled for imaging.

It is common to label more than a single target in cell. For instance, two different markers to label endosomes and viruses are used in (Rintanen et al., 2012), and two different target proteins and the cell cytoplasm are labeled in Publication PI. Basically, it is possible to use as many different fluorophores as needed. However, in practice, rarely more than three or four different fluorophores are used in fluorescence microscopy experiment for several reasons. First, fluorescent microscope components, explained in Subsections 2.1.1 and 2.1.2, have restrictions on the number of different excitation wavelengths used in experiment. Second, imaging multiple fluorophores increases imaging time, as each channel needs to be imaged separately, which can be especially harmful in live specimen imaging (Stephens and Allan, 2003). Third, different fluorophores can and usually have overlapping excitation and emission spectra that may cause overlapping in image channels. Imaging two fluorophores with distinct emission spectra is usually a simple task, unless only specific fluorophores can be used in the experiment. When using three or more fluorophores in a specimen, careful selection of imaging components and fluorophores is needed to avoid spectral overlapping. Often, there is at least some emission spectra overlap that can be minimized with linear unmixing methods (Tsurui et al., 2000).

Both live and fixed samples can be imaged with fluorescence microscopy. In live experiment, preventing photobleaching by all means possible is an important aspect (Stephens and Allan, 2003). Photobleaching of the dyes causes damage to the sample and fades emission signal, which is undesirable for image analysis. Typically, fixation is done chemically with formaldehyde or glutaraldehyde (Bacallao et al., 2006). Fixed samples can also be used to study dynamic processes by statistical analyses. This is achieved by fixing samples at different time points from the start of the biological process.

Next, a traditional fluorescence microscopy technique, *widefield microscopy*, and more advanced *confocal laser scanning microscopy* (CLSM) are presented. Both methods can be used to image a 3D stack of optical sections of sample. However, computational methods are needed for signal restoration in widefield microscopy to improve image quality.

2.1.1 Widefield microscopy

A diagram of basic components of widefield microscope is presented in Figure 1 A. Different arc lamps and light-emitting diodes (LED) are typically used as light sources in widefield microscopes. The excitation filter is used to select excitation wavelengths from usually broad spectrum of wavelengths excited by light source. Next, a dichroic mirror is used to reflect excitation light to the objective. Dichroic mirrors are filters that pass some range of wavelengths while reflecting others. The objective is the last component, before the sample, used to define the magnification and resolution (Abramowitz et al., 2002). After some of the excitation light is absorbed by the fluorophores in the specimen, the fluorophores emit photons with a longer wavelength. Some emitted light goes through the objective

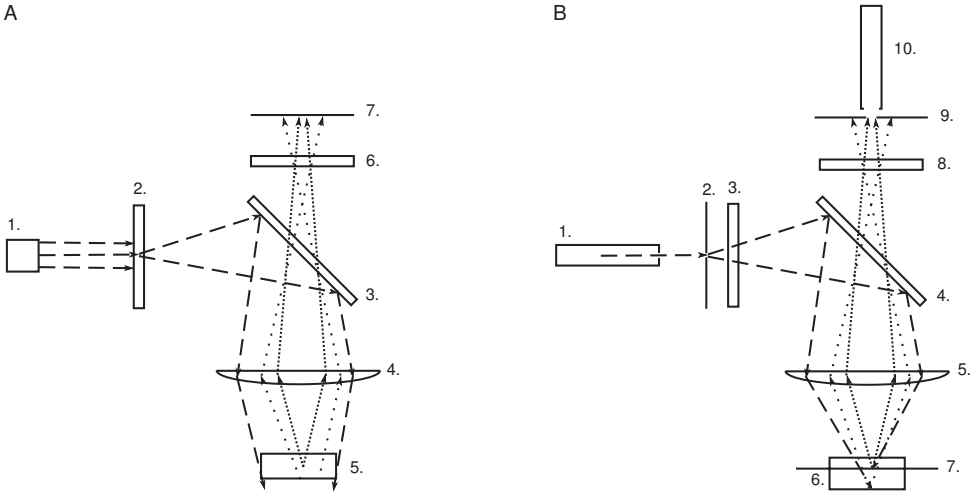


FIGURE 1 Diagram of the basic components of widefield (A) and confocal (B) microscopes. A. (1) light source, (2) excitation filter, (3) dichroic mirror, (4) objective, (5) specimen, (6) emission (barrier) filter, (7) detector (typically CCD camera). B. (1) light source (laser), (2) light source pinhole, (3) excitation filter, (4) dichroic mirror, (5) objective, (6) specimen, (7) focal plane, (8) emission (barrier) filter, (9) detector pinhole, (10) detector (photomultiplier tube). Edited from (Conchello and Lichtman, 2005; Stephens and Allan, 2003)

and passes the dichroic mirror. Finally, an emission filter is used to block any excitation light that has passed the dichroic mirror before detecting the fluorophore emission signal with a detector, usually a CCD camera (Lichtman and Conchello, 2005).

Resolution, the shortest distance between two distinguishable points, of a diffraction-limited microscope is defined by the objective, the wavelength of emission light λ (in fluorescence microscopy, excitation in general), and refractive index of immersion medium (Vonesch et al., 2006). Lateral resolution R_l is defined by Rayleigh limit as the radius of first airy disk formed in image of a point-like object (Inoué, 2006; Wallace et al., 2001):

$$R_l = \frac{0.61\lambda}{\mathrm{NA}} \tag{2}$$

where NA is the numerical aperture of the objective. The diffraction pattern of a point source in axial direction is not disk shaped, but instead hourglass shaped. Axial resolution is thus defined as the distance from maximum intensity of diffraction pattern to the first minimum intensity to axial direction (Inoué, 2006; Wallace et al., 2001):

$$R_a = \frac{2\lambda\eta}{(\mathrm{NA})^2} \tag{3}$$

where η is the refractive index of immersion medium.

Every objective has characteristic numerical aperture $\mathrm{NA} = \eta sin\theta$, where θ is half of the angular aperture of the objective (Abramowitz et al., 2002). As

$sin\theta$ cannot be larger than 1.0, and is less in practice, the refractive index of the imaging medium defines the maximum numerical aperture. The numerical aperture of a dry (air) objective is less than 1.0. However, common oil immersion objectives (refractive index approximately 1.515) can have numerical aperture of 1.4 (Abramowitz et al., 2002).

With common nucleic acid stain DAPI, which has its emission spectra peak in blue color wavelength ($\lambda = 455\,\text{nm}$), and a high numerical aperture objective (NA $= 1.4$), the maximum theoretical lateral resolution is $R_l = 0.61 \cdot 455\,\text{nm}/1.4 \approx 200\,\text{nm}$, and the axial resolution is $R_a = 2 \cdot 455\,\text{nm} \cdot 1.515/(1.4)^2 \approx 700\,\text{nm}$. In practice, the depth of a sample also has an effect on resolution as scattering of photons is directly proportional to the depth of the sample (Ntziachristos, 2010). Resolution of widefield microscopy can be increased with special superresolution methods such as structured illumination microscopy (Gustafsson, 2000). As superresolution widefield microscopy techniques are not yet widely available and as they are less easily applicable than traditional methods, they were not used in this work. Interested readers are pointed to the review by Huang et al. (2009).

Physical pixel and voxel size of image is set using the Nyquist sampling theorem. It specifies that at least two samples must be taken for each resolution unit (Bolte and Cordelières, 2006; Wallace et al., 2001) to prevent undersampling. In principle, the sampling rate can be anything larger than the Nyquist sampling rate. Typically, factor 2.3 is used in microscopy (Bolte and Cordelières, 2006). Oversampling more than that will increase the size of the data without adding any information. Using factor 2.3, suitable voxel dimensions for previous theoretical resolution example are $d_{x,y} = 200\,\text{nm}/2.3 \approx 87\,\text{nm}$ and $d_z = 700\,\text{nm}/2.3 \approx 304\,\text{nm}$. From Equations 2 and 3, and previously presented voxel dimensions, it can be seen that voxel is anisotropic in optical microscopy. Depending on the objective and used fluorophores, axial resolution is approximately 3.5–4 times worse than lateral resolution.

Widefield microscopy imaging is fast since the whole 3D sample is illuminated and the 2D image of a specified focus level is captured at one go. Also, imaging multiple channels is fast, as an automatic switch of excitation wavelength takes only milliseconds. The biggest restriction on image acquisition time for single channel imaging is the frame rate of the detector. Common CCD cameras used in widefield microscopy can image around 10 to 15 frames per second with the maximum field of view. However, the frame rate can be increased when imaging a smaller region of interest. For this reason, widefield microscopy is often preferred over CLSM, presented in Subsection 2.1.2, in live cell imaging of rapid processes (Stephens and Allan, 2003).

While imaging in widefield microscope, a single focus level in the sample is selected. As widefield microscope does not have any means to block out-of-focus light, typically as much as 90% of the signal in the image is blurry out-of-focus light (Conchello and Lichtman, 2005). As a result of this inherently 2D way to form an image of the whole sample, widefield microscopy is at its best for thin specimen (30 µm) (Swedlow et al., 2002). Widefield microscopy can be used also for 3D imaging by taking images with different focus levels through the sample.

This kind of volume is practically unusable as such because of the out-of-focus light in every slice. However, computational methods can be used to restore or remove the out-of-focus blur (Sarder and Nehorai, 2006).

Deconvolution is a computational method to solve the inverse of $f(x)$ from Equation 1. In theory, the deconvolution method removes, from the image, blur and out-of-focus signal caused by the microscope PSF as well as noise generated during imaging. In practice, solving $f(x)$ is an ill-posed problem (van Kempen et al., 1997) since the noise function is not known, and the PSF can only be approximated. In Equation 1, $g(x)$ is known, and traditional deconvolution methods expect that also the PSF $h(x)$ is known. PSF convolution kernel for deconvolution can be generated using a theoretical model or an experimental measurement. Theoretical diffraction-based models for PSF computations are presented by Gibson and Lanni (1991); Sarder and Nehorai (2006). As no optical system is perfect, deconvolution with the theoretical PSF model can fail badly. To model the PSF of a specific microscope, sub-diffraction beads can be imaged. From this image, beads are extracted and averaged to form the PSF model. It is important to image beads with same microscope settings as original specimen to get as good model of PSF of real experiment as possible (McNally et al., 1999), as the beads usually cannot be included in the actual specimen. Other issue with both theoretical and imaged PSF is that imaging conditions are not identical everywhere in the sample. To solve this issue, and to remove the need for a PSF model, blind deconvolution methods have been developed. These methods estimate also PSF during the deconvolution process (McNally et al., 1999). Both, a theoretically computed and an experimentally measured PSF are shown in Figure 2.

Deconvolution can be applied to a 2D single-focus level image or stack of images of different focus levels. It is to be noted that the risk of creating artifacts with deconvolution methods exists, which is avoidable with true optical 3D microscopy. Reviews and comparisons of deconvolution methods are presented in (McNally et al., 1999; Verveer et al., 1999; Sarder and Nehorai, 2006). Figure 2 presents an example image from a widefield microscope before and after deconvolution.

2.1.2 Confocal laser scanning microscopy

Many components of the confocal laser scanning microscope, presented in Figure 1 B, are used for the same purpose as in the widefield microscope. The excitation filter, dichroic mirror, objective and emission filter, respectively, have the same functionality as explained in Subsection 2.1.1. However, there are some fundamental differences between confocal and widefield microscopes. The main difference is that the confocal microscope has a detector pinhole to block most of out-of-focus and scattered light (Conchello and Lichtman, 2005). With the aid of this pinhole, the confocal microscope is used to image optical sections (slices) of the specimen to create a 3D volume without the need of a deconvolution method. Additionally, most confocal scanning microscopes exploit laser light sources to get point-like coherent excitation light. A photomultiplier tube is mostly used as

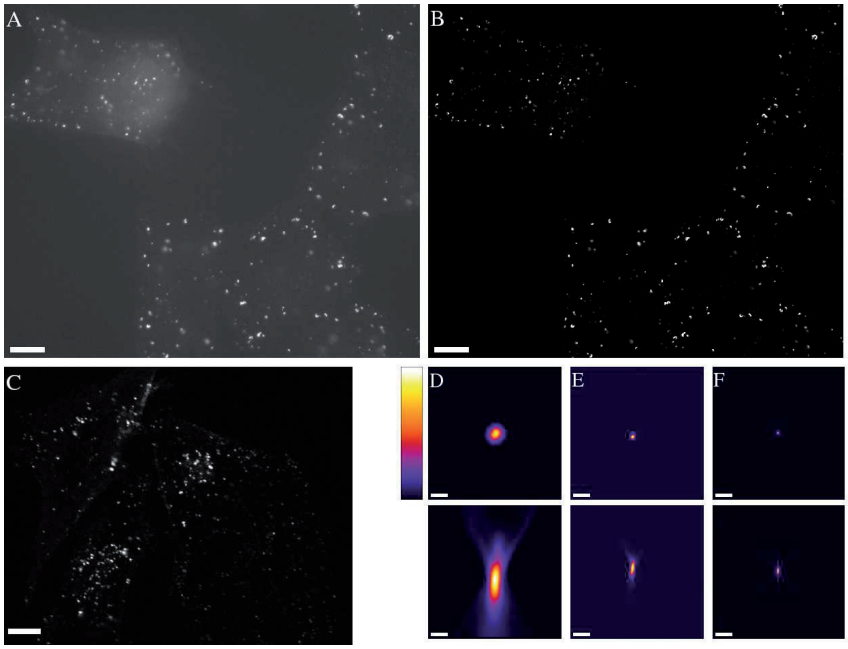


FIGURE 2 Examples of widefield (A), deconvolved widefield (B), and confocal images (C) with experimental widefield (D) and confocal (E), and theoretical PSF (F). The original widefield image (A) was deconvolved (B) with the Huygens software (Scientific Volume Imaging) using measured PSF (D). A slice from the confocal image stack (C) does not suffer from out-of-focus signal, as widefield image (A). PSF images (D, E, F) were measured with widefield (D), measured with confocal (E), and theoretically computed (F). Central x-y and x-z slices are shown of PSF images. Contrast is increased in images A–C for improved visualization. Scale bars in the example images A–C are 10 nm. Scale bars in the PSF images D–F are 1.5 nm.

a detector in the confocal laser scanning microscope.

The confocal laser scanning microscope forms an image by scanning through each focal plane of specimen pixel-by-pixel while recording the emission signal of each pixel with the photomultiplier tube. Scanning can be done either by moving the specimen stage or by moving the scanning spot. Since moving the stage has an effect on biological samples, it is more common to move the scanning spot by using two oscillating mirrors. One of the mirrors is used to move the scanning spot in horizontal direction while the other moves the spot in vertical direction (Conchello and Lichtman, 2005).

In addition to blocking out-of-focus light, a detector pinhole can be used to improve resolution. The resolution achievable with CLSM is reported to be approximately 1.4 times better than in widefield microscopy (Conchello and Lichtman, 2005; Inoué, 2006; Bolte and Cordelières, 2006). This is possible only by setting the detector pinhole diameter smaller than the diameter of the central airy disk (Conchello and Lichtman, 2005). The downside is that a smaller detector pinhole also causes a weaker signal and decreases signal-to-noise ratio (SNR). Using Equations 2, 3, the setup used in example in Subsection 2.1.1, and factor 1.4, the theoretical lateral resolution of CLSM is approximately 140 nm and axial resolution 500 nm. In practice, this is never achieved as a result of low SNR.

The resolution of CLSM can be slightly improved with deconvolution, but not to the extent of the resolution in widefield microscopy. To further improve resolution, a few super-resolution confocal techniques have been proposed. Recently methods such as stimulated emission depletion microscopy (Hell and Wichmann, 1994) and 4Pi microscopy (Hell et al., 1994) have gained momentum. These techniques were combined by Schmidt et al. (2008) who reported almost isotropic resolution of 40 nm. So far super-resolution CLSM has been available only to a few and limited to specific applications. However, in the near future, as methods mature and prices of devices decrease, it is expected that super-resolution techniques will become more widely used (Huang et al., 2009).

The biggest disadvantage of CLSM is its slow speed of image acquisition in live sample imaging. Scanning a single slice takes typically 1–2 seconds (Inoué, 2006). When imaging a typical stack of 30 slices with two channels, the imaging time is close to a minute. The same stack can be imaged in just a few seconds with widefield microscopy. However, post-processing deconvolution needed takes from minutes to hours. Speed of image acquisition can be improved by decreasing pixel time, which is the time spent to scan each pixel. However, a shorter pixel time also decreases SNR, which is already low in live imaging with low laser power to prevent photobleaching. Other means to improve acquisition time is to reduce resolution or to image a smaller region of interest. All of these have to be optimized to the needs of biological application when using CLSM in live imaging. Acquisition time and photobleaching are not significant issues when imaging fixed samples. With fixed samples, resolution and SNR can be maximized by long pixel time, high laser power, and even taking multiple scans over the sample and averaging the results with the Kalman filter (Conchello and Lichtman, 2005).

2.2 Transmission electron microscopy

The common transmission electron microscope works with same principle as the transmission optical brightfield microscope. Instead of light, transmission electron microscope uses electron beam to illuminate the sample and electromagnetic coils as lenses. Generally, illumination is generated with a field-emission gun and a CCD camera used as a detector (Lučić et al., 2005). The image is formed by measuring the amplitude of signal generated by electrons transmitted through the sample without elastic or inelastic scattering (Reimer and Kohl, 2008). Scattering of electrons is caused by dense matter that decreases the amount of electrons passing through the sample directly. A typical TEM image has an extremely low SNR. Contrast can be increased using contrast agents, such as gold particles, or by using scattered electrons to form a phase contrast image (Frank, 2006b).

TEM can be used to image only thin samples to prevent scattering of too many electrons. Usable thickness of the sample is dependent on the acceleration energy used in imaging. Generally, samples thinner than 100 nm can be imaged with 100 keV acceleration (Reimer and Kohl, 2008). If the sample depth is more than 200 nm, blurring is strong, even with high acceleration energy (Lučić et al., 2005). Changes in acceleration voltage change also the contrast transfer function (Fourier transformed equivalent of PSF), which defines the resolution of TEM. Atomic resolution of 3 Å is achievable with 100 keV equipment, and with 1 MeV, even resolution of 1 Å can be achieved (Reimer and Kohl, 2008). Contrast of a TEM image is defined by the defocus level used. Best resolution is reached when the focus level is close to so called *Scherzer defocus* (van Heel et al., 2000). With a small defocus, high frequency information is included in the image. However, the contrast is too low to detect any structures (Lučić et al., 2005; van Heel et al., 2000). Usually a larger defocus is used to improve the contrast at the expense of resolution (van Heel et al., 2000).

TEM can be used only to image fixed specimen, as no living structure would survive sample preparation, electron beam or near vacuum conditions (van Heel et al., 2000). Many sample preparation methods have been used (Afzelius and Maunsbach, 2004). Regardless of the fixation, radiation damage is an important question to consider in electron microscopy. The electron dosage needed to gain a good SNR is so high that serious damage is induced to the sample. With cryo-electron microscopy (cryoEM) (van Heel et al., 2000), the dosage can be increased 10-fold, but still the maximum density of only $10\mathrm{e}/\mathrm{Å}^2$ can be used to keep the atomic scale structures unharmed (van Heel et al., 2000). In cryoEM, samples are rapidly frozen in liquid methane and preserved in liquid nitrogen to form a snapshot of the sample in vitreous ice.

2.3 Electron tomography

ET is a technique for 3D imaging in great detail where many TEM projection images are used. ET is used from macromolecular to cellular structures to gather information of the organization of these structures (Koster et al., 1997). With the resolution approximated between 3 to 10 nanometers (McIntosh et al., 2005; Jonić et al., 2008), ET fills the gap between optical microscopy and structural methods such as SPR (Lučić et al., 2005). However, ET is also a much more complicated method for 3D reconstruction than, for instance, CLSM.

In tomography, either specimen or the illumination source and detector are tilted while transillumination images of the specimen are taken. These 2D projections are used to reconstruct a 3D volume of the sample with inverse tomographic reconstruction methods. To collect information from the whole sample, a full 180° tilt range should be used with small intervals. Typically 1–5 degree interval (Koster et al., 1997) is used. However, in ET, electron dosage should be kept to minimum to prevent damage to the sample. Additionally, in most cases, a sample cannot be imaged in the full 180° tilt range.

The maximum tilt angle in ET is typically restricted to $\pm 60°$–$70°$ (Frank, 2006a) as the specimen holder blocks the electron beam with higher angles. However, with a regular slab-shaped sample, images with higher tilt angles become almost useless as the electron beam passes many times more material than with low-tilt angles (Koster et al., 1997). With a specially-shaped specimen and a holder, even a larger tilt angle can be achieved (Kawase et al., 2007). However, the special-shaped holder and sample are designed for material science applications rather than for biological ET. The information gap between the minimum and maximum tilt angle is called the *missing wedge* (Figure 3) when single tilt direction is used. The missing wedge can be decreased to a missing pyramid by using two perpendicular tilt directions (Penczek et al., 1995). However, while missing information is reduced using two tilt series, reconstruction process becomes more complicated and the electron dosage is doubled if the same dosage level is used for each projection.

One of the requirements for good quality tomographic reconstruction is that projection images are aligned. However, TEM images of a tilted sample can include large misalignments as no equipment is stable enough for extremely small scale TEM imaging (Jonić et al., 2008). Alignment step is performed before reconstruction to correct any misalignment. Alignment can be done manually, which is both time-consuming and inaccurate (Brandt et al., 2001). Both manual and automatic alignment are often based on landmarks (Lučić et al., 2005), such as colloidal gold particles. Automatic alignment methods using landmarks are often based on the least squares minimization (Winkler and Taylor, 2006). Brandt et al. (2001) presents method based on epipolar geometry. When distinguishable landmarks are not used, alignment can be done with a marker-free method (Winkler and Taylor, 2006).

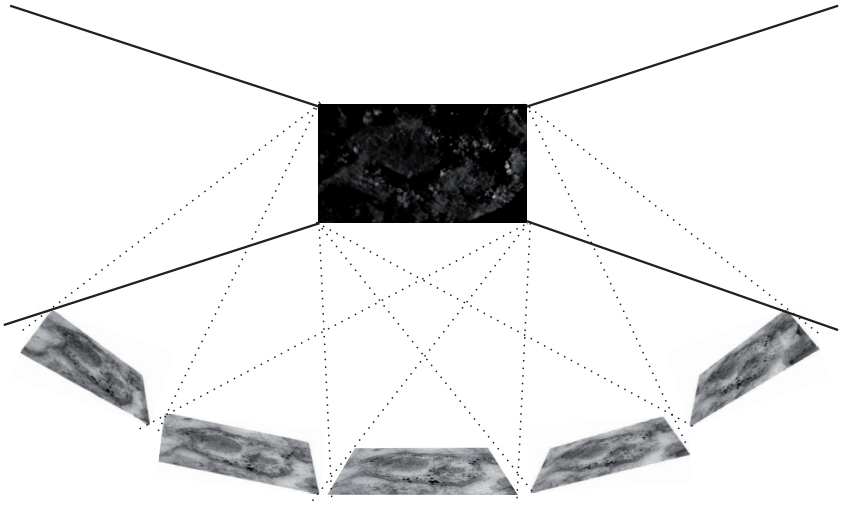


FIGURE 3 ET with the missing wedge. The sample is tilted and imaged in the transmission electron microscope to collect information for reconstructing a 3D structure of the sample. The regions between dark lines on both sides of the sample present the missing wedge.

2.3.1 Common reconstruction methods for electron tomography

Common reconstruction methods can be loosely grouped into three classes (Jonić et al., 2008): *backprojection* (Harauz and van Heel, 1986; Radermacher, 2006), *direct Fourier inversion* (Penczek et al., 2004; Sandberg et al., 2003), and *iterative and algebraic reconstruction* methods (Gilbert, 1972; Andersen and Kak, 1984; Marabini et al., 1998). The basic backprojection method is an inverse of a projection operation as it casts a "ray" from each pixel of every projection image to a volume in real space. The method is simple and computationally efficient as well as easy to use without the need for parameter adjustment (Penczek, 2010). The downside of simple backprojection is that it blurs objects and emphasizes the center of the volume (Frank, 2006b).

To find a solution for issues with the simple backprojection method, filtered and weighted backprojection methods (Harauz and van Heel, 1986; Radermacher, 2006) have been developed. Filtered backprojection (FBP) has a pre-filtering step with weighting function in Fourier space: $p_2 = F^{-1}(wF(p_1))$, where p_1, p_2, w, F, and F^{-1} are imaged tomogram, filtered tomogram, weighting function, Fourier transform, and inverse Fourier transform, respectively. A ramp filter is used as weighting function. WBP does weighting in real space, where weighting function is derived from the simple backprojection result (Radermacher, 2006; Penczek, 2010). Both, FBP and WBP, can provide a solution for blurring and non-uniform distribution of density. However, all backprojection methods are sensitive to the missing wedge showing as artifacts and elongation of structures (Frank, 2006b). Elongation of structures in axial direction can be theo-

retically calculated by (Plitzko and Baumeister, 2007):

$$e_{zy} = e_{zx} = \sqrt{\frac{\theta + \sin\theta\cos\theta}{\theta - \sin\theta\cos\theta}}, \tag{4}$$

where θ is the maximum tilt angle, and e_{zy}, e_{zx} is the elongation of the structure in axial direction. Using Equation 4, it can be calculated that with 60° maximum tilt angle, theoretical elongation is approximately 1.55.

Direct Fourier inversion methods are based on the central slice theorem. In the theory, projection images are Fourier transformed and organized in 3D Fourier space (Jonić et al., 2008). The reconstruction is the result of inverse Fourier transform. Unfortunately, an inverse 3D fast Fourier transform cannot be employed for non-uniform grid which is the result of a set of Fourier transformed projections (Frank, 2006b; Penczek, 2010). Generally, interpolation methods have been used to form a uniform grid. Currently the most accurate methods are gridding-based and use non-uniform Fourier transform (Penczek, 2010). The gridding-based method by Penczek et al. (2004) has been reported to produce as good results as algebraic reconstruction methods and as fast as weighted back-projection methods (WBP) (Frank, 2006b). A method by Sandberg et al. (2003) has been reported to take only half of the time required by WBP.

The simultaneous iterative reconstruction technique (SIRT) (Gilbert, 1972) is widely used method in SPR reconstruction (Penczek, 2010). The method improves the reconstruction estimate iteratively by minimizing the difference between measured and re-projected projection images. Iterative algebraic reconstruction methods solve the reconstruction problem by forming a set of linear equations of basis functions. Linear combination of these basis functions determines the reconstructed volume (Marabini et al., 1998). Approximation of the reconstructed volume is solved iteratively by minimizing the squared difference of projections of the reconstructed volume and the measured tomograms (Penczek, 2010). Projections of the reconstructed volume are formed using a weighted projection matrix which can be also used to incorporate constraints of *a priori* knowledge to the reconstructed sample. A mathematical theory of algebraic reconstruction methods is presented in Penczek (2010). Voxels are often used as basis functions (Andersen and Kak, 1984), but Marabini et al. (1998) use spherical blobs as basis functions to smooth borders and reduce artifacts in the reconstruction. In SPR, algebraic reconstruction methods outperform WBP (Sorzano et al., 2001). However, in ET, algebraic methods do not function as well due to the variation of the depth of the sample in different tilt angles (Penczek, 2010; Lučić et al., 2005).

2.3.2 Sequential statistical reconstruction method

A new reconstruction method for biological ET is studied in Publication PIV. The aim is to decrease artifacts due to missing wedge. The tested reconstruction method is a sequential statistical maximum a posterior expectation maximization (sMAP-EM) (Tuna et al., 2013). The method is compared to widely used WBP and SIRT reconstruction methods. All of these methods are directly applicable to biological ET, as none require prior knowledge of the objects in the sample.

All methods are tested with simulated and experimental data. Simulated data including small spherical gold particles and large spherical viruses is used to enable evaluation of the resolution of the reconstructed volumes. Experimental data is evaluated using elongation of manually selected gold particles from a sample including subcellular vesicle. In addition, the contrast ratio between a gold particle and the volume surrounding it, is evaluated to measure how well structures can be detected.

The preliminary results show (Publication PIV) that the sMAP-EM reconstruction method decreases elongation, as compared to the WBP and the SIRT reconstructions. Also, the resolution is better in the sMAP-EM reconstructions. No visible artifacts are present in the sMAP-EM reconstruction, as are in the WBP and SIRT reconstructions. The WBP reconstruction show smaller elongation than SIRT reconstruction in all datasets. However, the WBP reconstructions show deformation in lateral direction also in simulated data. The contrast ratio is similar in the WBP and the SIRT reconstructions of all datasets. The sMAP-EM method improves the contrast ratio approximately 2.5–4.5 times depending on the dataset.

3 BIOIMAGE ANALYSIS

Quantitative bioimaging has become an essential part of modern cellular and molecular biology research. Multidimensional bioimaging data embody structures and processes impossible to measure reliably by human observer. To quantitate these structures and events, advanced analysis methods are needed. The amount of bioimaging data is continuously growing, and with recent success in high-throughput imaging (Pepperkok and Ellenberg, 2006), need for effective automatic analysis methods is enormous. In this chapter, bioimage analysis methods used in light, mainly fluorescence, microscopy are presented, with methods used and presented in Publications PI, PII, and PIII.

An overview of segmentation methods commonly used in bioimage analysis is presented first. Segmentation is significant for bioimage analysis as it can be used as is to analyze objects or as a part of many other analysis methods. The segmentation methods used in this work are also briefly described. Next, pixelwise and object-based colocalization methods, used in the field to analyze closeness of different types of particles, are presented. After the introduction to colocalization, an object-based algorithm (Publication PIII), which is able to analyze association between particles, is introduced. Next, tracking methods used to analyze cellular dynamics are surveyed, followed by the presentation of a tracking method of Publication PII. Finally, an important subject on method validation completes the chapter.

3.1 Segmentation

Segmentation is a process that separates regions of interest from the background of an image. In a simple form, a segmentation method takes one image as input and produces output as binary image I_b:

$$I_b(x) = \begin{cases} 1, & \text{if } x \text{ in foreground} \\ 0, & \text{if } x \text{ in background,} \end{cases}$$

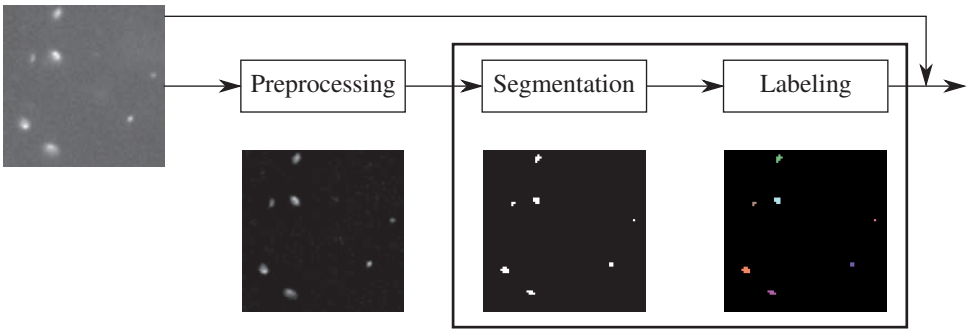


FIGURE 4 Common segmentation-based bioimage analysis pipeline. Raw image data is first preprocessed to make subsequent segmentation simpler and more accurate. After segmentation, individual objects are labeled unless identified in the segmentation process. Finally, the labeled and raw image data are passed on to analysis methods.

where $x \in \mathbb{N}^n$ and $n \in \{2,3\}$. In case the segmentation method is able to detect individual objects, it produces label image I_l as output:

$$I_l(x) = \begin{cases} k, & \text{if } x \text{ in object labeled } k \\ 0, & \text{if } x \text{ in background,} \end{cases}$$

where $k \in \mathbb{N}^+$.

Segmentation is used in some form in almost all bioimage analysis tasks. It can be used as is to study individual objects or to draw statistical conclusions from a collection of objects. In addition, segmentation is used as a part of bioimage analysis methods such as tracking methods or object-based colocalization methods.

It is common to combine image processing methods into pipeline to solve segmentation problem. A general segmentation pipeline in bioimage analysis includes a few steps with a broad range of methods selected to fit the application. Different steps are hard to categorize from the wide spectrum of methods and applications. However, the common segmentation pipeline (Figure 4) includes at least preprocessing before actual segmentation (binarization/labeling of the image), and labeling of individual objects if not done in the segmentation method and needed in the application.

Preprocessing of images is done to prepare images for segmentation. Typical activities are to reduce noise, compensate non-uniform illumination, smooth variance inside objects, and to enhance features of interest. Noise in fluorescence microscopy is mostly Poisson noise with additive Gaussian noise coming from the electronics (Vonesch et al., 2006; Smal et al., 2010). In addition, images are blurred with approximately Gaussian modeled point-spread function (Thomann et al., 2002). For this reason, often-used Gaussian smoothing, which is known to reduce resolution (Thomann et al., 2002) and to blur edges, is not the best option to reduce noise from sub-resolution particles. However, for larger objects,

Gaussian smoothing is useful to smooth intensity variation. Median filtering is a suitable noise reduction method for Poisson noise, and is commonly used with other edge-preserving methods such as anisotropic diffusion (Perona and Malik, 1990) and bilateral filtering (Tomasi and Manduchi, 1998).

Prior to the segmentation method, image features can be enhanced to improve segmentation accuracy. Common enhancement methods for spot-like particles are Laplacian of Gaussian (LoG) (Sage et al., 2005; Smal et al., 2010) and Difference of Gaussians (DoG) (Lowe, 2004), which can be used to approximate LoG (Lindeberg, 1994). Calculating gradient magnitude can be used to enhance edges for larger particles such as nuclei. However, objects rarely embody continuous edges in fluorescence microscopy.

3.1.1 Common segmentation methods for optical microscopy

Thresholding is the most simple segmentation method. In its basic form, a binary threshold method T is defined as:

$$T(x) = \begin{cases} 1, & \text{if } I(x) \geq t \\ 0, & \text{if } I(x) < t, \end{cases}$$

where t is user defined threshold value for scalar image I. Although global manual threshold is often the first thing to try on the segmentation of new images, it is not considered useful for image analysis on cell biological research. First, the threshold level set by a researcher may be biased or inconsistent. Second, a single threshold value is rarely usable for a batch of images in the same experiment, as intensity levels between images vary due to biological variation in samples. Also, image background can have non-uniform illumination, especially in widefield and electron microscopy.

User-defined parameters are often the only solution to configure a method. However, when applicable, automatic methods should be favored. Many methods have been developed for automatic selection of threshold level and, surveyed and tested for instance in Trier and Jain (1995) and Sezgin and Bülent (2004). It is fairly safe to say that the Otsu method (Otsu, 1979) is the most used of these methods as is and as built-in other methods (Trier and Jain, 1995; Ruusuvuori et al., 2010). The method defines optimal threshold to be the one that maximizes between class variance. Another widely used threshold selection method maximizes class entropies (Kapur et al., 1985), and another minimizes entropy between original and thresholded images (Li and Lee, 1993). Automatic thresholding eliminates problems with user bias and intensity level variation between images. However, global threshold is not usable with non-uniform background nor with images having large variance in intensity between objects. To solve these issues, adaptive thresholding is applied where threshold level is computed for each pixel from a pre-defined kernel size. However, adaptive thresholding can create new problems by giving emphasis to anything that points out from the background.

It is clear that thresholding used alone is not a very sophisticated solution

to the common segmentation problem. More advanced methods are often tuned to fit a specific application. Spot detection methods, reviewed recently by Ruusuvuori et al. (2010) and Smal et al. (2010), are used to segment sub-resolution particles. In Smal et al. (2010), the h-dome detection method (Smal et al., 2010, 2008b) was found to be the most accurate spot detection method. Similar results were presented in Ruusuvuori et al. (2010) for object detection measures. However, they reported h-dome detection not to be very accurate when comparing segmented pixels to ground-truth. This applied to other spot detection method also. The h-dome detection method starts by enhancing the image with the LoG filter (Gaussian smoothing is used in Smal et al. (2008b)) and then uses morphological h-dome transform (Vincent, 1993). After transformation, the image is sampled, divided into clusters with the mean shift algorithm (Comaniciu and Meer, 2002), and finally clusters are divided into foreground and background objects by their intensity features and size. Simpler and more common methods for spot detection include enhancement with LoG (or DoG) followed by manual or automatic thresholding, top-hat filtering (Bright and Steel, 1987; Soille, 2003), and wavelets (Olivo-Marin, 2002). These methods were found to give equally good or better results than many other methods in Ruusuvuori et al. (2010). Results of top-hat filtering and wavelets were not encouraging with synthetic images with SNR of 2 (Smal et al., 2010).

Other methods are used to segment larger structures such as nuclei. While combination of noise reduction, thresholding and morphological operations does work for binarization of DAPI-stained nuclei, methods are needed to separate touching nuclei. Watershed transform (Vincent and Soille, 1991; Soille, 2003) is a common approach to both to separate touching nuclei (Malpica et al., 1997) and to directly segment nuclei image. Touching nuclei in a binary image are traditionally separated by computing distance transform, inverting the result, and selecting local minima as markers for watershed transform. Maximally stable extremal regions (Matas et al., 2004) was recently used in Kaakinen et al. (2014) to segment cells in phase contrast microscopy. The method generates a set of maximally stable connected regions by going through all threshold levels (typically from 0 to 255 in 8-bit grayscale image), and selecting regions that are stable with more threshold levels than defined by the Δ parameter.

Active contours are used in many bioimage analysis problems to detect often complex and even irregular boundaries. Large objects do not typically exhibit continuous edge in fluorescence microscopy, making detection of object boundary by popular edge detection methods (Canny, 1986) impractical. Active contours provide a clean solution with the continuous deformable boundary model. These methods are based on energy minimization often defined by a contour internal force such as smoothness and an external force such as image gradients. The original parametric active contour model by Kass et al. (1988) was able to evolve only an initial contour without possibility to merge or split the contour. Inclusion of level sets (Malladi et al., 1995) made active contour model topology changes simple. Level set is a function that presents contour as zero level, the inside of the contour as negative (positive), and the outside of the contour as

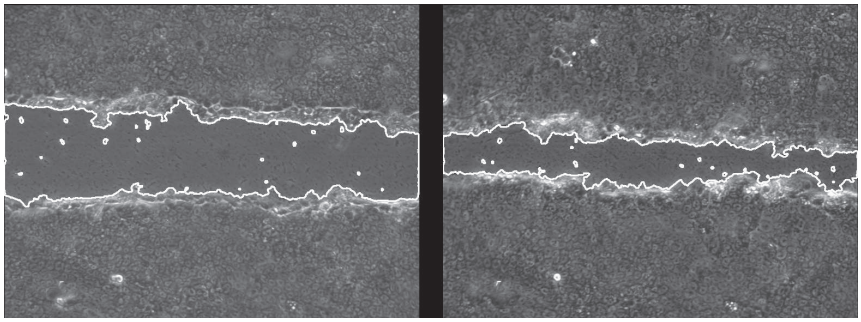


FIGURE 5 Tracking evolution of boundary in a wound healing assay with geodesic active contours (Caselles et al., 1997) from Pajari et al. (2013). Raw phase contrast image data courtesy of Anne-Maria Pajari.

positive (negative) values. This was used by Caselles et al. (1997) in geodesic active contours widely used in medical and bioimage analysis (Figure 5). Geodesic active contours are able to propagate only to a single direction and as is drawn by the gradient of the image. Chan and Vese (2001) developed an active contours method which can propagate in any direction without dependency on image gradients. Dufour et al. (2005) combined the edge force in Caselles et al. (1997) into the model of Chan and Vese (2001), and used the method to track cells in live fluorescence microscopy. Active contours provide built-in solution for tracking cells with deformable shape, and they have been recently used also by Dzyubachyk et al. (2010) and Maska et al. (2013).

After preprocessing and segmentation, individual objects are identified and labeled (Figure 4) to be used for analysis. This step is optional since many segmentation methods, such as the watershed segmentation and active contours, result directly in labeled image data. Connected component labeling is used to identify objects from binary image data. However, it is common that separate objects touch each other, especially when segmenting image of an nuclei or cells. For convex objects, such as nuclei, the previously presented method based on distance and watershed transforms (Malpica et al., 1997) works as long as the objects are not overlapping too much. For complex and oddly-shaped objects, it is often better to use another segmentation method.

3.1.2 Overview of the segmentation methods used in Publications

Various segmentation protocols for fluorescence image data are used in Publications PI, PII, and PIII to analyze particles, or as a part of another image analysis method. All segmentation protocols were created and ran in the BioImageXD software (Section 5.1, Publication PI). Both fixed and live cells are segmented in Publication PI. Cell membrane and $\alpha 2\beta 1$ integrin clusters were segmented from the fixed cells to analyze change in integrin cluster features (size, intensity, location) and to analyze internalization of $\alpha 2\beta 1$ integrins. Background offset was removed from images by subtracting the value of the highest peak of the image

intensity histogram. As the signal in the stained cell cytoplasm channel was extremely low after background subtraction, data was smoothed with a Gaussian filter, and all voxels having any signal were set as foreground voxels. The cell model was made whole with morphological operations followed by membrane extraction with the marching cubes algorithm (Lorensen and Cline, 1987).

Integrin clusters were segmented using adaptive thresholding in form:

$$T(x) = \begin{cases} 1, & \text{if I(x) >= M(x) + c} \\ 0, & \text{otherwise,} \end{cases}$$

where $M(x)$ is the median value from the neighborhood of radius of 5 pixel sizes and c is the constant value set larger than background variance. These choices were balanced between thousand image stacks, as the same segmentation method and parameters were used to segment integrin clusters in every time point and in every dataset with different treatments. Segmented integrin clusters were labeled with a previously defined watershed-based method. Live cell samples of $\alpha 2\beta 1$ integrin clusters for tracking analyses were segmented using global thresholding after median filtering of the data.

Spots are detected in Publications PII and PIII using DoG filtering to enhance signal followed by Otsu thresholding (Otsu, 1979). Background is also subtracted, as in Publication PI, before DoG filtering in Publication PIII, where images are from a widefield microscope with high background signal.

3.2 Colocalization and particle association

Colocalization is defined as overlapping signal in different imaging channels (Comeau et al., 2006). Biologically, this is generated by differently labeled closely located particles, which usually locate in the same structure (Costes et al., 2004). Thus the term *co-localize*. To study colocalization, at least two different types of particles in the sample are dyed with fluorophores and imaged as different channels in a fluorescence microscope. Colocalization is measured from these images either by analyzing signal colocalization with traditional pixelwise methods or by first segmenting the interesting objects from the images and then using object-based colocalization techniques.

By definition, colocalization analysis is inherently dependent on the resolution of the images. With a theoretical imaging device able to resolve atoms and to take images in infinite frequency, no colocalization would ever be measured as two particles cannot exist in the same location at the same time. However, in cell biology, colocalization is used to detect whether different particles are located in the same structure or in close structures (North, 2006). Size of these structures in cell biology vary from tens to hundreds of nanometers, which justifies the use of the existing fluorescence microscopy imaging techniques.

As pixelwise colocalization methods measure signal overlap or correlation between two images, pixel by pixel, it is clear that in addition to resolution, mea-

suring colocalization with pixelwise methods is dependent on the quality of the images (Zinchuk and Grossenbacher-Zinchuk, 2009). To remove the possibility of superimposition of axially separated particles, colocalization should be measured with 3D imaging techniques (Bolte and Cordelières, 2006) whenever possible. In addition, colocalization analysis is sensitive to noise, background offset, and overlap of excitation or emission spectra of fluorophores (Bolte and Cordelières, 2006; Zinchuk and Grossenbacher-Zinchuk, 2009). Spectral overlapping can be prevented with careful consideration of the fluorophores used. A problem with background offset can often be solved with post-processing by subtracting background before the colocalization analysis. However, subtracting the background offset does not improve image quality.

Deconvolution and super-resolution techniques may be used to improve image quality. Deconvolution has been shown to improve accuracy of the colocalization analysis (Landmann, 2002) by reducing noise and improving the resolution. However, deconvolution can create artifacts, which may have impact to the colocalization analysis. Also, deconvolution is burdensome task to apply for hundreds of images typically used in statistical colocalization analysis. Super-resolution imaging techniques can be used to improve the accuracy of colocalization analysis. However, not all available super-resolution techniques are easily usable for colocalization experiments, as they require special fluorescent probes instead of common fluorescent proteins (Huang et al., 2009). Also, super-resolution techniques might not yield overlapping signal from closely located particles typically considered as colocalizing (Lagache et al., 2013).

Taking all inaccuracies in sample preparation, imaging, and colocalization analysis into account, it must be noted that in a general case, colocalization analysis measurements should not be reported with absolute values (North, 2006). The same applies to many bioimage analysis problems, but in colocalization analysis, it is fairly simple to get measurements of high colocalization from noisy images with low levels of colocalization. Instead, results should be compared to a control sample, other samples, or to other time points of the same sample in the experiment.

Colocalization analysis is sensitive to alignment error in laser lines and on imaging delay in live experiment. Alignment error in the microscope causes systematic shift between channels and results in unreliable colocalization measurements (Waters (2009), Publication PIII). This affects all the samples imaged in the microscope, and to correct this, at least an additional registration step is needed before the colocalization analysis. Misalignment can be fixed also in hardware. However, this task requires special service. Live sample experiments are also prone to errors caused by imaging delay between channels. This is especially problematic when imaging samples, including vibrant particles in a confocal microscope where the delay between channels can be tens of seconds.

3.2.1 Pixelwise colocalization

Traditional pixelwise methods are the most used colocalization measurements. These include calculation of the percentage of overlapping pixels and intensity, the Pearson correlation coefficient (introduced to colocalization analyses by Manders et al. (1992)), and Manders colocalization coefficients (Manders et al., 1993) derived from the Pearson correlation coefficient.

Pixelwise colocalization is calculated between two 2/3D image channels, C_1 and C_2. Let $i \in \{1,2\}$, X be a set of pixels in image C_i, and t_i a threshold for channel C_i. The basic pixel and intensity overlap colocalization measurements for both channels are defined as:

$$P_{i,coloc} = \frac{|B|}{|T_i|} \tag{5}$$

$$I_{i,coloc} = \frac{\sum_{x \in B} C_i(x)}{\sum_{y \in T_i} C_i(y)}, \tag{6}$$

where $T_i \subseteq X$, $T_i = \{x | x \in X, C_i(x) > t_i\}$, and $B = T_1 \cap T_2$. The value range of both of the measurements is $P_{i,coloc}, I_{i,coloc} \in [0,1]$.

The Pearson correlation coefficient is used to measure correlation between channels. Let $\overline{C}_i$ be the average intensity of the image C_i. Pearson's correlation coefficient is defined as (Manders et al., 1992):

$$r_p = \frac{\sum_{x \in X} \left(C_1(x) - \overline{C}_1\right)\left(C_2(x) - \overline{C}_2\right)}{\sqrt{\sum_{x \in X} \left(C_1(x) - \overline{C}_1\right)^2 \sum_{x \in X} \left(C_2(x) - \overline{C}_2\right)^2}}. \tag{7}$$

The value range of the Pearson correlation coefficient is $r_p \in [-1,1]$, where 1 means perfect correlation, 0 no correlation, -1 perfect negative correlation, and values between either some level of positive or negative correlation (Bolte and Cordelières, 2006).

The Pearson correlation coefficient is an unbiased statistical measurement, as it does not need any parameters to calculate. However, Pearson correlation can be used to measure correlation also from the pixels above defined thresholds to remove the effect of the background noise and offset. The criticism towards Pearson correlation targets its inability to show colocalization for individual channels that can be considerably different (Comeau et al., 2006). Manders et al. (1993) derived new colocalization coefficients from the Pearson correlation coefficient by removing the averages and separating the channels. The original and modified Manders' colocalization coefficients (Manders et al., 1993) are defined as:

$$M_i = \frac{\sum_{x \in X} C_{i,\mathrm{coloc}}(x)}{\sum_{x \in X} C_i(x)} \tag{8}$$

$$M_i' = \frac{\sum_{x \in X} C'_{i,\mathrm{coloc}}(x)}{\sum_{x \in X} C_i(x)} \tag{9}$$

where $C_{i,\text{coloc}}$ and $C'_{i,\text{coloc}}$ are defined as:

$$C_{i,\text{coloc}}(x) = \begin{cases} C_i(x) & \text{if } C_j(x) > 0 \\ 0 & \text{if } C_j(x) = 0 \end{cases}$$

$$C'_{i,\text{coloc}}(x) = \begin{cases} C_i(x) & \text{if } C_j(x) > t_j \\ 0 & \text{if } C_j(x) \leq t_j, \end{cases}$$

where $i, j \in \{1, 2\}$, $i \neq j$. From the above equations, it can be seen that $M_i, M'_i \in [0, 1]$ and that both measure the ratio of intensity inside a mask defined by the other channel to whole intensity of the channel. The original Manders' colocalization coefficients M_1 and M_2 (Equation 8) differ from the modified Manders' colocalization coefficients M'_1 and M'_2 (Equation 9) in that they define all voxels $x \in X$ colocalized where $C_i(x)$ and $C_j(x)$ has any signal. The modified Manders' colocalization coefficients include only voxels $x \in X$ where the other channel $C_j(x) > t_j$. These coefficients are essentially the same if both image channels have been preprocessed by setting all the values below threshold values to zero. M'_i and $I_{i,coloc}$ differ in that M'_i does not take threshold t_i into account, and only uses t_j to define the mask from image C_j.

Computation of colocalization measurements is simple. However, defining the threshold values, t_1 and t_2 manually is complicated and requires expert biological knowledge of the experiment. While it is possible to set the thresholds manually using same principle for all images in the experiment, defining the thresholds automatically is unbiased and much more applicable for large-scale and high-throughput applications (Pepperkok and Ellenberg, 2006). Costes et al. (2004) present a method to define thresholds objectively by ruling out uncorrelated low intensity pixels. First, the method calculates a least-squares fit in the form of $C_2 = aC_1 + b$ by orthogonal regression. Then the threshold t_1 for C_1 is set to the maximum value and decreased until Pearson correlation (Equation 7) in pixels below the thresholds t_1 and $t_2 = at_1 + b$ is approximately zero.

Both the Pearson's correlation coefficient and Manders' colocalization coefficients are widely used for colocalization analysis. The advantage of the Pearson's coefficient is in its objectivity. The original Manders' colocalization coefficients are also parameter free. However, without any pre-processing, the original Manders' colocalization coefficients will always result in values close to 1 as a result of noise and background offset in the image. The advantage of the Manders' colocalization coefficients over the Pearson's correlation coefficient is in their ability to present colocalization for both channels separately. This is especially important when the amount of signal differs largely. Villalta et al. (2011) presented recently a method that combines the Manders overlap coefficient (Manders et al., 1993) and the Pearson correlation coefficient (Equation 7) with improved accuracy over individual methods. However, this combined measurement is not channel specific.

Adler et al. (2008) raise an important question about the image quality in pixelwise colocalization analysis, as the saying "garbage in, garbage out" is especially suitable for colocalization quantification (North, 2006). Adler et al. (2008)

use a correction term on correlation calculation to remove the effect of Poisson noise from colocalization analysis. The correction term is formed by taking duplicates of all images and calculating a Pearson correlation between the duplicates. Correlation should equal to 1, but as a result of Poisson noise, a correction term is needed to achieve this. The correction terms for both of the channels are finally combined into a single correction term used in correlation-based colocalization analysis. The biggest problem with the method is additional imaging time needed to take the duplicates. Also in live imaging, taking duplicates increases the chance for photobleaching.

Statistical significance tests are employed to evaluate the trustworthiness of the colocalization analyses. Often used methods are the Costes (Costes et al., 2004) and the van Steensel (van Steensel et al., 1996) statistical significance tests. Costes et al. (2004) tests the statistical significance by randomizing one of the input images, calculating the Pearson correlation with randomized image and the other input image, and then comparing whether this correlation is higher than with the original input images. The test is iterated hundreds of times (Costes uses 200 iterations) to measure if colocalization is real or generated by random processes. Costes sets the P-value of null hypothesis of colocalization being random to 0.05, meaning that if more than 10 iterations of total 200 generate larger correlation than the correlation between input images, the colocalization is not statistically significant. Randomization is performed by mixing small regions in the image. Regions are used instead of pixels since the neighboring pixels are not independent as a result of the PSF of the imaging setup (Costes et al., 2004). The statistical significance test of van Steensel et al. (1996) is simpler than the Costes test. It tests the statistical significance using the Pearson correlation by cross-correlation in a one-dimensional space. Cross-correlation generates the randomization by shifting one of the images in x-direction pixel-by-pixel. Because of the simplicity of the test, it is applicable only to isotropic particles (van Steensel et al., 1996).

3.2.2 Effects of preprocessing to pixelwise colocalization analysis and pixelwise colocalization used in Publications

An extreme example of the effect of low-quality images on colocalization analysis was created from the experiment that should not include much of colocalization. Instead of overlapping, particles are locating in close structures. In this example, two channels from a widefield microscope of live experiment having much out-of-focus signal were used. The costes automatic threshold method (Costes et al., 2004) has been designed for good-quality images. It can be seen that the thresholds set by the Costes method are too low when the background signal is high (Figure 6). The scatterplot of the original images shows a clear correlation that is mostly caused by the out-of-focus background signal. Thus the background should be removed before applying the Costes method.

Background subtraction, deconvolution, and segmentation-based approaches (Subsection 3.2.3) can be used to reduce the effect of noise and background sig-

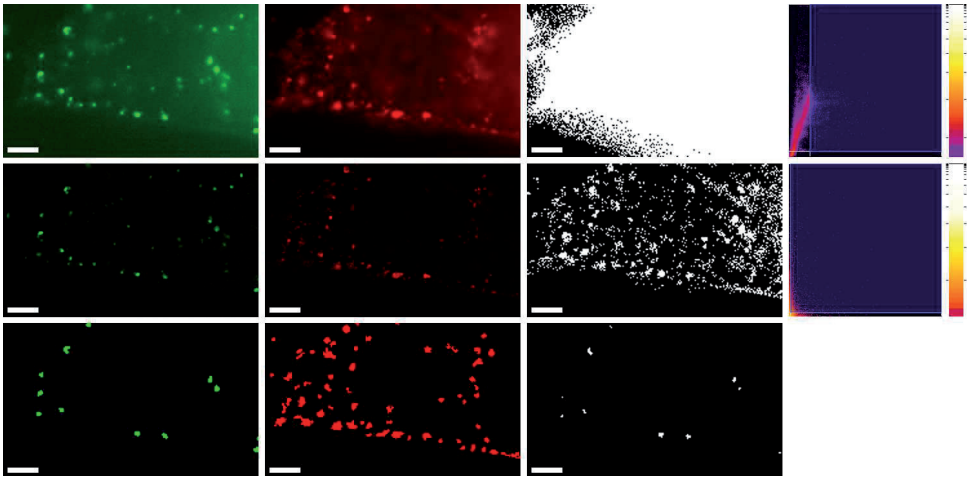


FIGURE 6 Effects of preprocessing on colocalization analysis. Top row: Unprocessed widefield images of two channels, the colocalization map after applying Costes automatic thresholding to unprocessed images, and scatterplot showing correlation between the channels (x-axis: green, y-axis: red). Middle row: Same widefield images after deconvolution with background subtraction, the colocalization map after applying Costes automatic thresholding for deconvoluted images, and scatterplot showing no correlation between the channels. Bottom row: The segmentation-based approach used in publication PIII showing the masks of the channels and the colocalization map. Results of the different cases are shown in Table 1. Scale bars are 5 µm

TABLE 1 Example showing the sensitivity of the pixelwise colocalization analysis. Colocalization statistics were calculated from the original two-channel images without any preprocessing, from images deconvoluted and background subtracted in Huygens software (Scientific Volume Imaging), and within region of interest masks explained in Publication PIII. Thresholds used for unprocessed and deconvoluted images were calculated with the Costes method (Costes et al., 2004). This live biological experiment was made so that the images should not include much of colocalization. M_i', r_p, $P_{i,coloc}$, $I_{i,coloc}$, C_{p-v}, and S_{p-v} are Manders' colocalization coefficients, Pearson correlation, ratio of colocalizing pixels, ratio of colocalizing intensity, Costes P-value, and van Steensel P-value for true colocalization, respectively. P-value calculations are not applicable to segmentation-based preprocessing in Publication PIII.

Preproc.	M_1'	M_2'	r_p	$P_{1,coloc}$	$P_{2,coloc}$	$I_{1,coloc}$	$I_{2,coloc}$	C_{p-v}	S_{p-v}
None	0.83	0.94	0.86	0.91	0.90	0.92	0.97	0.21	0.00
Deconv.	0.29	0.52	0.17	0.26	0.52	0.34	0.56	1.00	1.00
Pub PIII	0.27	0.05	-0.36	0.28	0.04	0.27	0.05		

nal to colocalization analyses, and can significantly improve the reliability of the results (Table 1). The results presented in Table 1 show that the two preprocessing approaches, deconvolution and segmentation-based, produce almost similar results for the first channel. The Costes method still sets a bit too low threshold value for the first channel, which can be seen from high colocalization of measurements on the second channel (M_2', $P_{2,coloc}$, and $I_{2,coloc}$). However, the segmentation-based preprocessing might underestimate the pixelwise colocalization, as all particles smaller than 8 pixels have been removed from the analysis. The particle size threshold was determined experimentally to remove particles generated by noise in the image. Most likely, the truth is somewhere between the two preprocessing approaches, showing only minor colocalization for both of the channels, as expected from the biological setup. The example shows the importance of performing colocalization analysis relatively, as relative changes in colocalization are often more meaningful than individual colocalization measurements, which are highly dependent on the selection of preprocessing and the thresholds (North, 2006).

In Publication PI, $\alpha 2\beta 1$ integrin colocalization with caveolin-1 is studied with the Manders colocalization coefficients (Equation 9). Colocalization changes are tracked relatively through the time points of integrin internalization on a fixed experiment. It is found that internalization of $\alpha 2\beta 1$ integrin involves caveolin-1 as suggested earlier by Karjalainen et al. (2008), showing as increase in colocalization as the internalization moves on. In Publication PIII, the Manders colocalization coefficients (Equation 9), pixelwise overlap (Equation 5), and the Pearson correlation (Equation 7) are compared to the proposed particle association method with experimental fixed and live imaging data as well as to simulated data expressing fixed and live samples with and without misalignment between the imaging channels. The method and the experiments are presented in next section.

3.2.3 Object-based colocalization and association

Object-based colocalization and association analysis are preceded with segmentation and labeling of individual objects from the input images. The difference between the two methods is that direct overlap of the objects is analyzed with colocalization methods whereas association methods are used to analyze interconnection of the objects. The difference is subtle, as some object-based colocalization methods consider objects colocalizing even though direct overlap between the objects does not exist. For instance, Lachmanovich et al. (2003) consider that the center of the masses of two objects can be separated at most the distance of the resolution of the image to be measured as colocalizing. However, when the objects have any larger displacement or local matching of the particles is applied, the occurrence should be called association.

Three common measurements to analyze the object-based colocalization used are: (1) colocalization of center of masses (COM) or centroids of the objects (Lachmanovich et al., 2003); (2) the ratio of COMs or centroids that locate inside the

objects of the other channel (Lachmanovich et al., 2003); (3) the ratio of pixels or intensity inside the objects of one channel that locate inside the objects of the other channel (Publication PIII). In addition, object-based colocalization measurements are often designed for specific application (Chessel et al., 2012). Wörz et al. (2010) define altogether six different object-based colocalization measurements for their application, including (1) defined above. Other measurements in their model-based method relate to studying whether a whole object is inside other bigger object, or whether there is a partial overlap.

Through the rest of the section, the center of mass and acronym COM are used to mean the center of mass, centroid or any other point used to define an individual object. Let $i, j \in \{1, 2\}, i \neq j$. Let also X be a set of pixels in image channels C_i. After segmentation and labeling of the objects, let $O_{i,pix}$ be a set of all pixels inside any object in segmented and labeled image C_i. Also let $O_{i,com}$ be a set of COMs of the objects in image C_i. The object-based colocalization of COMs $C_{i,comc}$ (1) (Lachmanovich et al., 2003), the COM overlap $C_{i,como}$ (2) (Lachmanovich et al., 2003), and the object-based pixel overlap $C_{i,pixo}$ (3) (Publication PIII) are defined as:

$$C_{i,comc} = \frac{\sum_{x \in O_{i,com}} N(x, O_{j,com})}{|O_{i,com}|} \tag{10}$$

$$C_{i,como} = \frac{|O_{i,com} \cap O_{j,pix}|}{|O_{i,com}|} \tag{11}$$

$$C_{i,pixo} = \frac{|O_{i,pix} \cap O_{j,pix}|}{|O_{i,pix}|}, \tag{12}$$

where $N(x, O_{j,com})$ is defined as

$$N(x, O_{j,com}) = \begin{cases} 1 & \text{if } \exists y \in O_{j,com} : ||x - y|| \leq R \\ 0 & \text{otherwise,} \end{cases} \tag{13}$$

where R is the resolution of the image. It must be noted that in common 3D fluorescence microscopy techniques, voxels are anisotropic, making the resolution definition more complicated. Lachmanovich et al. (2003) defined R = 200 nm that is approximately the maximum lateral resolution of fluorescence microscopy. Even though the above equations are for the calculation of global object-based colocalization measurements, all measurements can be calculated for each individual object as well.

Studying association of small particles accurately is practically impossible with pixelwise colocalization methods (Figure 7). Also, object-based colocalization methods often fail to produce reliable results on particle association, as these particles do not necessarily overlap at all and are often separated by a distance larger than the resolution of the images. Special methods are needed to measure the correspondence of associating particles. Not many methods able to measure association have been presented. Recently, Lagache et al. (2013), using Ripley's K function, presented a statistical method to measure whether two point-sets are spatially close to each other.

3.2.4 Point-pattern matching association method

In Publication PIII, we present an association method based on point-pattern matching (PPM). The association method includes a global transform (translation in this case) searched by PPM to compensate possible misalignment between the image channels and a local matching step to define association. In live experiment, local matching is also able to compensate the movement of the particles during an imaging lag between separate channels. Next, the association method and related concepts are presented briefly. A complete presentation of the method can be found in Publication PIII.

Let $O_{1,com}$, $|O_{1,com}| = n$ and $O_{2,com}$, $|O_{2,com}| = m$ be sets of COMs of the objects in the channels C_1 and C_2, respectively. Let $M \subset O_{1,com} \times O_{2,com}$ be a matching between the point sets, $M = \{(o_{1,1}, o_{2,1}), \ldots, (o_{1,|M|}, o_{2,|M|})\}$, with the following constraints:

$$\forall s \in \{1, \ldots, |M|\} : ||o_{i,s} - f(o_{j,s})|| \leq \delta$$
$$\forall p \in \{1, \ldots, |M|\}, \forall q \in \{1, \ldots, |M|\} \backslash \{p\} : o_{i,p} \neq o_{i,q},$$

where $i, j \in \{1, 2\}, i \neq j$, $\delta \in \mathbb{R}$, $\delta \geq 0$, is a user-defined maximum matching distance, and f is translation transformation, $f(x) = x + t$, where $t \in \mathbb{R}^d$ is global translation. Only translation transform is allowed, as translation is the main misalignment between microscopy images. Small rotations can be compensated with δ parameter. By restricting only to translations, the performance of the method is greatly improved. The association ratio, $\alpha_i \in \mathbb{R}$, $\alpha_i \in [0, 1]$, is the ratio of the number of objects in the matching to the number of objects in the channel. For channel C_1, $\alpha_1 = |M|/n$ and for channel C_2, $\alpha_2 = |M|/m$.

The PPM association algorithm, presented in detail in Publication PIII, is described shortly in Algorithm 1. The PPM algorithm is a global non-linear optimization method that searches the maximum matching between input point sets and returns this matching.

Algorithm 1 The point-pattern matching association algorithm (Publication PIII)

Input: Images C_1 and C_2
Segment and label images C_1 and C_2
$O_{1,com}$ = extract a set of COMs from C_1 labels
$O_{2,com}$ = extract a set of COMs from C_2 labels
$n = |O_{1,com}|$
$m = |O_{2,com}|$
M = PPM_association_algorithm($O_{1,com}, O_{2,com}$)
$\alpha_1 = |M|/n$
$\alpha_2 = |M|/m$
return (α_1, α_2)

Value range of association ratio $\alpha_1, \alpha_2 \in [0, 1]$ is the same as in the Manders colocalization coefficients and object-based colocalization measurements presented

TABLE 2 Ground-truth values of particle association simulations.

Level	α_1	α_2
High	$83/111 \approx 0.748$	$83/97 \approx 0.856$
Intermediate	$58/111 \approx 0.523$	$58/123 \approx 0.472$
Low	$23/111 \approx 0.207$	$23/82 \approx 0.280$

in this work. These are all comparable, as they measure ratio of colocalizing (associating) intensity or objects to intensity or objects of whole channel. With fixed cells, and without alignment error between channels, the association ratio measurement is similar to the object-based colocalization in Equation 10, with a difference in the allowed object distance. However, in the PPM association algorithm, the match between point sets is a bijection, a constraint not included in the object-based colocalization measurement. Association ratio can be used to approximate colocalization with a small distance parameter in fixed cells, as long as the particles are small. This is shown in the fixed cell experiment in Publication PIII that includes a high level of colocalization. It is clear that the presented association method does not fit for studies where both of the channels contain only large objects, as is the case with the object-based colocalization measurement of Equation 10. It is also shown in Publication PIII that the association method is able to detect association ratios where traditional colocalization methods fail in live experiment including closely located particles without much of overlap.

To validate the association method, simulated data including both colocalization and association was generated in Publication PIII. Three different levels of ground-truth association are generated to test different levels of association as exist in experimental data: high, intermediate, and low. The ground-truth association ratios for both of the channels are presented in Table 2. Using the generated simulated data, we compare the association method to traditional pixelwise colocalization methods. In addition, the PPM association method is also compared to another point-set matching method. The chosen matching method is the Iterative Closest Point (ICP) (Besl and McKay, 1992; Chen and Medioni, 1991), which is modified to include the matching distance parameter of the association method. From the various variants of ICP methods (Rusinkiewicz and Levoy, 2001), a recent method, showing accurate results, called the Biunique Correspondence ICP (Zhang et al., 2011) is tested. ICP is a local non-linear optimization method that iteratively improves the transformation to improve matching of the point sets. For comparison, only translations were allowed, as in PPM algorithm, and the same maximum distance of the objects was used to define association.

Altogether 111 objects were created in the first channel. The number, the volume, and the average intensity of the particles were extracted from the experimental data used in Publication PIII. As initialization, the same set of objects is set in both channels. To simulate association between the object sets, a predefined number of objects are removed from the second channel and new objects are created to get the association level defined in Table 2. To simulate misalignment be-

tween the channels, four levels of global translation are introduced to the objects in the second channel. The length of the global translation l is set to $l \in \{0, \ldots, 3\}$ of lateral pixel size, and the direction is randomized for every simulation. In addition, to simulate imaging lag, each object is moved in random direction of a distance d selected for each object from the normal distribution $N(0, \sigma)$ of lateral pixel size. Simulations of six levels of object movements are applied having $\sigma \in \{0, \ldots, 5\}$. For each (l, d) parameter combinations, 10 simulations are created, making the total number of simulations 720.

The simulated data is analysed with the PPM (Publication PIII) and ICP (Zhang et al., 2011) association methods, and with the Manders colocalization coefficient (Manders et al., 1993) (Equation 8). For the PPM and ICP methods, images are first segmented as described in Publication PIII and in Subsection 3.1.2, and the centroids of the objects are used for point-set matching. To be able to compare the association methods and the traditional colocalization method, segmented regions are used as masks for the colocalization method as presented in Figure 6.

Results from the intermediate association level show (Figure 7), as expected, that the Manders colocalization coefficient (as well as other traditional colocalization methods) is not usable on association studies. It is clear that colocalization methods are sensitive to even a small misalignment and imaging lag between channels seeable in global and particle translation association error plots. Both, the PPM and ICP association methods, are able to compensate the channel misalignment tested. Results similar to global translation were measured with a maximum of 3 pixel global translation.

The PPM method gives a more accurate translation between the point-sets than the ICP method, which can be seen in all the plots in Figure 7. This can be seen especially from the simulations without any particle translation. With perfect align error compensation, the methods should give an association error close to zero. However, ICP gives an association error of approximately 0.06. Interestingly, the association error is practically the same with both matching distances. The PPM method shows worse results with increased matching distance when there is no particle translation applied. However, the difference is subtle. More clear is the close-to linear increase in association error when particles move during imaging more than the matching distance is able to handle. A larger matching distance gives better results, having an association error less than 0.04 with $\delta = 8$ in all simulations in the PPM association method. However, it should be noted that matching distance is application dependent, and should be based on the maximum distance on which particles can still be assumed to associate with each other. Overall, the PPM association method produces more accurate results than the ICP association method in all parameter combinations. The results are similar with high and low association level simulations, with increased accuracy on the high association and decreased on low association level.

As the ground-truth point correspondences in the simulated datasets are known, the PPM and ICP methods can be compared by true positive, false positive, and false negative matches. Using these, F-measure was calculated for all pa-

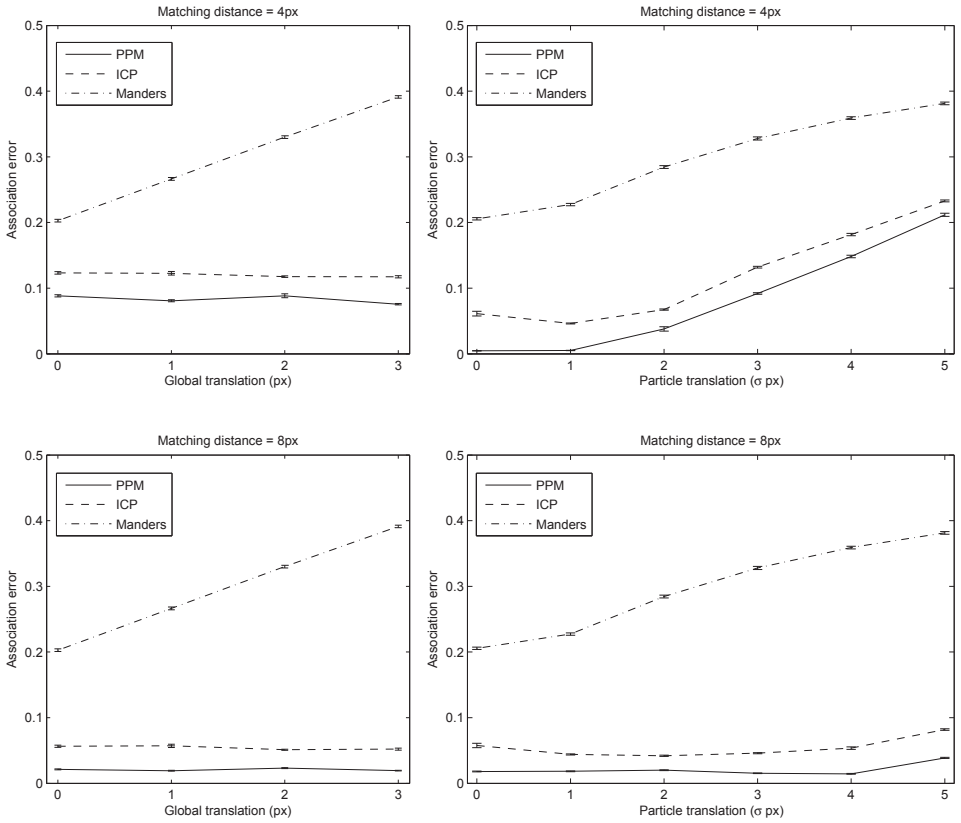


FIGURE 7 Absolute association error of intermediate association-level simulated data analyses with the PPM method (solid line), the ICP method (dashed line), and the Manders colocalization coefficient (dash-dot line). The presented results are averages of association or colocalization absolute error, including standard error, of all intermediate association level simulations for both channels as a function of either global translation (120 simulations for each step) or local particle translation (80 simulations for each step). The results of two different matching distances, $\sigma = 4$ and $\sigma = 8$, for the PPM and ICP methods are shown.

rameter combinations for both the PPM and ICP association results. The results, using matching distance $\delta = 6$ (approximately 1200 nm) are presented in Publication PIII, showing that the PPM association method outperforms ICP with all association levels and parameter combinations. An interesting detail not seeable in the association results is that ICP failed completely to produce any true positive matches on some low level association simulations. This makes F-measure undefined, which can be seen as missing ICP low-level F-measure plots. The result shows the sensitivity of the local ICP matching method, which can end up in a wrong local minimum, at least when the samples have a low association level. The same effect can be seen with matching distances $\delta = 4$ and $\delta = 8$. Only with matching distance $\delta = 10$, ICP is able to find at least one true positive match in all low association level simulations (Publication PIII, Supplementary figures). The global PPM association method is able to produce true positive matches in all 720 simulations.

3.3 Tracking

Studying living cells in their natural state is essential for understanding dynamic cellular processes. These processes can be analyzed statistically by drawing conclusions from average changes for instance in size, intensity, location, and shape of the particles detected during an experiment. However, the possibility to analyze individual particles, such as viruses or proteins involved in cell signaling, in living cells opens many new possibilities to understand cell functionalities. Cell dynamics at the single-particle level can be analyzed with tracking methods.

Tracking can be applied from small proteins to whole cells. The term SPT is used when tracking small particles (often smaller than the resolution limit of the imaging device) such as proteins and viruses in fluorescence microscopy (Meijering et al., 2006). Resulting trajectories of individual particles are used to study cell membrane dynamics (Saxton and Jacobson, 1997; Alcor et al., 2009), virus entry (Godinez et al., 2009; Ruthardt et al., 2011; Godinez et al., 2012), and different intra-cellular events (Sage et al. (2005), Publication PII, Publication PI). Tracking whole cells or nuclei is called cell tracking (Zimmer et al., 2006). Though the same tracking principles can be applied to solve both problems, the distinctive factor is that in SPT each particle can be approximated by a single point in space whereas a whole, large, and possibly complex object needs to be taken into account in cell tracking. For this reason, many cell tracking methods bind segmentation and tracking together, using evolutive models (Dufour et al., 2005; Dzyubachyk et al., 2010; Maska et al., 2013). It is more common in SPT methods to separate segmentation and tracking as is done in the method included in this work (Subsection 3.3.3).

3.3.1 Single-particle tracking

Meijering et al. (2006) defined the whole SPT process to include at least four steps: preprocessing of image data, segmentation and identification of individual particles, linking particles through time series, and analyzing the results. The preprocessing needed is dependent on the experiment and the quality of the images. In some cases that only includes denoising the image data when it can be considered as part of segmentation. However, it is common that a living cell or even the whole sample move during a live experiment. This needs to be compensated before it is possible to link the corresponding particles. For instance, in Publication PI (Supplementary Figure 11), we needed to use registration to compensate for the drifting of the whole sample, which was caused by changes in heat conditions in the microscope. This drift, many micrometers between some time points, would have made particle linking impossible without compensation. For SPT, segmentation is often done with spot detection methods (Ruusuvuori et al., 2010; Smal et al., 2010). However, segmentation and identification of individual particles, as well as analyzing the results, is dependent on the experiment. Various segmentation methods are presented in Section 3.1. Through the rest of the section, the focus is on linking corresponding particles through time series, which can be considered to be the core of the SPT process.

The SPT method tries to find the best possible trajectories of particles in time series by linking particles in consecutive time points together. Particle locations and features, extracted from the segmentation results, are given as input to the tracking method, which then tries to create links between the particles (Figure 8). Tracking a dense set of fluorescent particles is far from being a trivial task. Common problems are created already in imaging and in particle detection. Correct detection of particles is an extremely hard task, as SNR is especially low in live fluorescence microscopy. As a result, some particles are often missed. Also resolution of the images complicates identifying individual particles, as particles closer than the resolution are in general case unresolvable. Some methods such as Gaussian fitting have been used to detect location of particles of sub-resolution size in greater detail (Cheezum et al., 2001). However, applicability of these methods is dependent on the SNR.

Four common events can be identified while linking particles (Jaqaman et al., 2008): *merge*, *split*, *appear*, and *disappear*. Merging of two or more particles can be caused by real clustering of the particles (Karjalainen et al., 2011). However, a typical case is that the distance of the merging particles is unresolvable to identify individual particles. Merging is an even bigger problem in 2D microscopy where particles might be overlapping in an image but still be separated by micrometers in real space. Merge is often followed by a split event where two or more particles, identified as a single particle in an earlier time point, are separated. Appearing and disappearing of particles are often caused by problems in particle detection making the events consecutive. However, in living samples, physiological changes can make a particle to gain or lose fluorescence, and as such to be detected or undetected by the segmentation method during time series. Particles

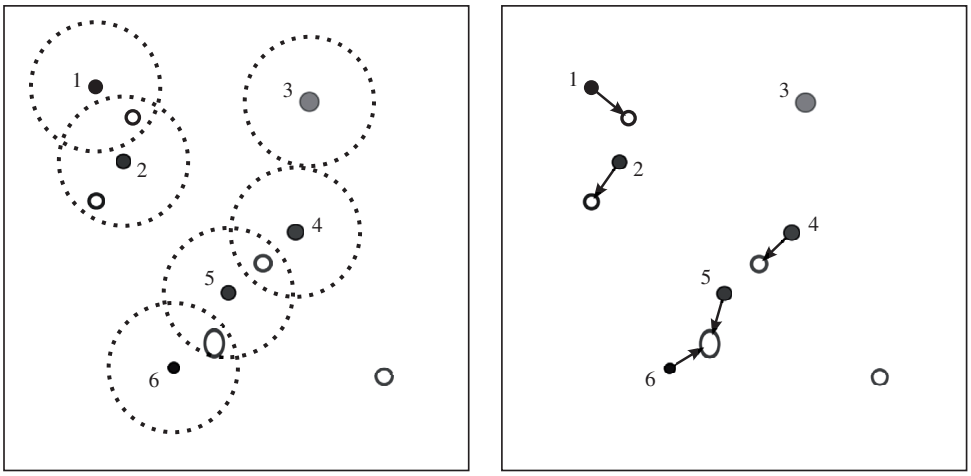


FIGURE 8 Basics of particle tracking between two time points. Six particles in time point t are visualized with filled objects of different size and intensity. There are five particles in time point $t + 1$ visualized with unfilled objects. The task of the particle tracking method is to create a link between each corresponding particle in both sets. A common maximum distance constraint is drawn around every particle in time point t in the left image. Particles 1, 4 and 6 have only one particle of time point $t + 1$ in the search range whereas particles 2 and 5 have two particles. Particle 3 does not have any particles in the search range, and cannot be linked at all. Correct link assignment is presented in the right image. Particles 1 and 2 are both competing over the same particle in time point $t + 1$. If particle 2 is linked wrongly, particle 1 cannot be linked at all. The same applies to particles 4 and 5. In this case, particles 5 and 6 are clustering into a single particle in time point $t + 1$.

can also appear in the field of view from outer regions of the sample as well as go out from the field of view.

3.3.2 Existing methods for single-particle tracking

The general SPT problem is NP-hard (Shafique and Shah, 2005). Without any constraints or assumptions, a problem with a few tens of particles and time points is unsolvable (Jaqaman et al., 2008). A typical SPT problem includes hundreds or even thousands of particles in tens or hundreds of time points. To find a solution to the problem, some constraints need to be introduced. The most common constraint is search range d_{max} (Figure 8). With the search range constraint, correspondences for particle p^k are searched only from particles in the next time point that fulfills $||p^{k+1} - p^k|| \leq d_{max}$. With this constraint, most of the particles can be ruled out without risking losing correct particle correspondence, as long as d_{max} is set using *a priori* knowledge of the application. Other similar constraints are also used, such as change in particle features, in methods that use also other than location information of the particle. Instead of finding a global solution to the problem directly, many methods (Veenman et al., 2001; Shafique and Shah, 2005) solve the problem with local greedy methods. These methods take many local optimal steps and combine these to a global solution. Jaqaman et al. (2008) also include a second global step to optimize the results from the first local step.

Tracking methods rely on different amounts of *a priori* knowledge. Some methods (Chenouard et al., 2013; Shafique and Shah, 2005; Veenman et al., 2001) assume no information about sub-resolution particles is known, and use only motion models to find point correspondence. Chenouard et al. (2013) raise an important question of whether intensity of a sub-resolution particle, measured with a process with random nature, can be used as a feature at all. However, when SNR is not too low, particle intensity can be assumed to be at a similar level, and thus be used as a feature in search for correspondence by methods (Jaqaman et al., 2008; Sage et al., 2005; Tvarusko et al., 1999). Also, as merge and split events have effects on the intensity level of a particle, Jaqaman et al. (2008) used particle intensity to identify these events. For larger particles, other features such as size (Tvarusko et al., 1999) and shape (Smal et al., 2008a) are used.

Particle motion modeling is a key element in tracking a dense population of sub-resolution particles. A simple nearest-neighbor correspondence assignment works only when the particle set is sparse and its particles are moving slowly. Often that does not apply in an SPT application. Veenman et al. (2001) and Shafique and Shah (2005) present methods to solve point correspondence based on motion models that emphasize constant velocity and smooth trajectory. These motion models designed for macro-world tracking problems rarely produce satisfactory results at a cellular level, as particles exhibit Brownian motion when diffusing freely and directed motion with small fluctuation when being actively transported (Saxton and Jacobson, 1997). As the direction of the movement is hard to estimate, Sage et al. (2005), Sbalzarini and Koumoutsakos (2005), and Jaqaman et al. (2008) all include distance of particles, with other measurements, in cost

functions to be minimized. This random walk motion model can capture Brownian motion. A probabilistic method presented by Godinez et al. (2009) also uses random walk motion model with Gaussian probability. A motion model used in Chenouard et al. (2013), called the conveyor-belt motion model (Saxton, 1994), is a mix of random walk and directed transport motion models able to capture both Brownian motion and actively transported motion of particle. A similar mix of random walk and directed transport was presented in Genovesio et al. (2006).

Existing methods handle merge, split, appear, disappear and temporal disappearance events with various success. Most of the methods cannot handle merge and split events explicitly. Some methods have been developed for applications where particles cannot merge or split (Veenman et al., 2001; Shafique and Shah, 2005) or where these events have been found unimportant to handle (Chenouard et al., 2013; Godinez et al., 2009; Sbalzarini and Koumoutsakos, 2005). A common way to handle these events is described in Smal et al. (2008a), where merge of two particles is handled as disappear of one particle, and split as appear of a new particle. Jaqaman et al. (2008) use a second step to handle merge and split events with the possibility to connect short trajectories together after the initial linking step. Particle intensity is used in the cost function to identify merge (increase in intensity) and split (decrease in intensity) events. Appear is intuitively handled as a start of a new trajectory when a particle is not assigned to any existing trajectory, whereas disappear is handled as a termination of trajectory to which no particle is assigned. However, there are differences in the handling of temporal disappearance, which in most cases leads to the termination of a trajectory and creation of a new one. Sbalzarini and Koumoutsakos (2005) proposed an idea of assigning a dummy particle in a trajectory which does not have any measured particle assigned in a certain time point. In this way, the trajectory is not terminated and could potentially continue on later time points. The same method is used in Godinez et al. (2009). In Jaqaman et al. (2008), an additional step of connecting track segments is used.

3.3.3 General feature-based tracking method

The method in Publication PII was designed as a general tracking method for fluorescence microscopy. The cost function and the constraints of the method can be configured to the needs of the application. With constraints and local greedy link assignments, the method is efficient and usable also in high-throughput applications. However, local optimization cannot guarantee that the method would find a global optimal solution to the tracking problem. The method includes both random walk and directed transport motion models. However, these cannot be combined together as is done in Chenouard et al. (2013) and Genovesio et al. (2006). Random walk can capture also directed motion. However, when only directed motion is interesting to the application, this has to be defined in constraint.

The method can handle appear and disappear of particle directly by creation or termination of trajectory. Split and merge events are handled as well. Depending on the constraints, a split of two particles is handled either as a creation of

two new trajectories or in a common case as a continuation of one trajectory and creation of a new one. Merge of particles is handled as a true merge event where the same particle is assigned to more than one trajectory. This is uncommon in SPT methods which typically include a constraint of a particle being assigned only to a single trajectory at a time. The only event that cannot be handled by the method is temporal disappearance. When disappearance is caused by a particle detection problem, then nothing can be done in the current method. An event of a temporal disappearance caused by merging of particles at an unresolvable distance, could be caught and resolved in a simple post-processing step.

Candidate particle search can be constrained with four parameters based on particle features and trajectory history. Let d_{min} and d_{max} be the application specific minimum and maximum search distance, and let σ_d, σ_s, σ_i, σ_a be deviations in distance, size, average intensity, and direction angle, respectively. Also let p^{t_k} be the particle of k:th time point in trajectory t, and p_l, p_s, p_i be the particle location, size, and average intensity features, respectively. Then all particles $p \in P^{k+1}$, where P^{k+1} is a set of particles detected in time point $k+1$, that fulfill the following constraints, are included in a set of candidate particles of trajectory t

$$\begin{aligned} d_{min} * (1 - \sigma_d) \leq ||p_l - p_l^{t_k}|| &\leq d_{max} * (1 + \sigma_d) \\ \frac{|p_s - p_s^{t_k}|}{p_s^{t_k}} &\leq \sigma_s \\ \frac{|p_i - p_i^{t_k}|}{p_i^{t_k}} &\leq \sigma_i \\ cos^{-1}\left(\frac{p_l - p_l^{t_k}}{||p_l - p_l^{t_k}||} \cdot \frac{p_l^{t_k} - p_l^{t_{k-1}}}{||p_l^{t_k} - p_l^{t_{k-1}}||}\right) &\leq \sigma_a. \end{aligned} \tag{14}$$

All constraints, except the angle constraint, are usable in every trajectory. For the angle constraint, the trajectory needs to have at least two particles assigned. It is common that the set of candidate particles include more than one particle. To determine the goodness of fit for all candidate particles, a function

for each feature is defined

$$\begin{aligned}
\Psi_d(t,p) &= \begin{cases} 1 & \text{if } d_{min} \leq ||p_l - p_l^{t_k}|| \leq d_{max} \\ \left| \frac{d_{min} - ||p_l - p_l^{t_k}||}{d_{max} * \sigma_d} - 1 \right| & \text{if } ||p_l - p_l^{t_k}|| < d_{min} \\ \left| \frac{||p_l - p_l^{t_k}|| - d_{max}}{d_{max} * \sigma_d} - 1 \right| & \text{if } ||p_l - p_l^{t_k}|| > d_{max} \end{cases} \\
\Psi_s(t,p) &= \left| \frac{|p_s - p_s^{t_k}|}{p_s^{t_k} * \sigma_s} - 1 \right| \\
\Psi_i(t,p) &= \left| \frac{|p_i - p_i^{t_k}|}{p_i^{t_k} * \sigma_i} - 1 \right| \\
\Psi_a(t,p) &= \left| \frac{|p_a - (t_{p_t-1})_a|}{(t_{p_t-1})_a * \sigma_a} - 1 \right|.
\end{aligned} \tag{15}$$

For all particles $p \in P^{k+1}$ that passed constraints (Equation 14), a goodness of fit for track t is calculated as a weighted sum of feature functions (Equation 15)

$$\Omega(t,p) = \sum_k w_k * \Psi_k(t,p), \tag{16}$$

where $k \in \{d, s, i, a\}$, $w_k \in [0,1]$, and $\sum_k w_k = 1$. The particle tracking algorithm tries to assign a single particle only in a single trajectory. However, in case of merging, and when there are no better candidates, one particle can be assigned to more than one trajectory. One additional parameter, $\alpha \in [0,1]$, is introduced to control merge event. The probability of a merge event increases with a high α value. The main points of the particle tracking method is presented in Algorithm 2.

The method described in Algorithm 2 is implemented in BioImageXD software presented in Section 5.1. The method is used to track $\alpha 2\beta 1$ integrin clusters in Publications PI and PII (Figure 9). For this application, use of a tracking method able to detect merge events is especially important. Even though the movements of similar particles are tracked in both experiments, there were significant differences in the experiments. In Publication PII, clustering of $\alpha 2\beta 1$ integrin is induced with echovirus 1 which makes the movement of particles more rapid than in Publication PI where clustering is induced with antibodies. However, almost the same set of parameters are used to solve both tracking problems with increased size constraint σ_s and adjusted size and average intensity weights, w_s, w_i, in Publication PII. The results were qualitatively analysed and approved by expert biologists in both experiments, and in Publication PI we are able to see significant differences in average speed and average directional persistence (how directly particles are moving) in cells under different treatments. However, as validation of the results of complex tracking task is impractical even for an experienced biologist, the tracking method was validated with simulated datasets.

Algorithm 2 The particle tracking algorithm (Publication PII)

Input: Particle sets P^k, where $k \in [1, m]$ is time point of time series.
$T = \emptyset$ // Set of tracks
for all $p \in P^1$ **do**
  $T = T \cup NewTrack(p)$
for $k = 2 \ldots m$ **do**
  $G = \emptyset$ // Set for track, particle, and goodness value triple
  for all non-terminated tracks $t \in T$ **do**
    $C^{t_k} = ParticlesPassConstraints(t, P^k)$ // (Equation 14)
    for all $p \in C^{t_k}$ **do**
      $G = G \cup (t, p, \Omega(t, p))$ // (Equation 16)
  Order G by goodness value
  for all $(t, p, g) \in G$ **do**
    $h_t =$ the highest goodness value for track t
    if t and p not used, t not terminated, and $g > \alpha * h_t$ **then**
      $AddToTrack(t, p)$
      Mark other tracks with particle $p^{t_{k-1}}$ as terminated
  for all Track t not used and not terminated **do** // Merge
    $p =$ Particle with the highest goodness value
    $AddToTrack(t, p)$
  for all $p \in P^k$ not used yet **do**
    $T = T \cup NewTrack(p)$
return T

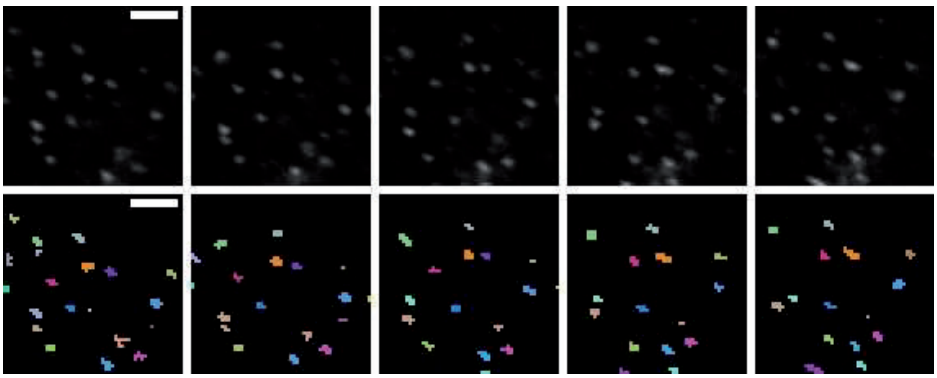


FIGURE 9 Five consecutive image stacks from an $\alpha 2\beta 1$ integrin clustering sample imaged with 30 second interval. Top row: Single slice from each of the image stacks. Bottom row: Segmented and tracked particles. Particles belonging to same trajectory are visualized in the same color. Two particles merge in the fourth time point (orange trajectory). Scale bar 3 µm.

TABLE 3 Ground-truth and tracking results of simulated data as statistical measurements. For each simulation, four parameters are shown: number of tracks, average number of time points in tracks, average speed of particles in tracks and average directional persistence.

Ground-truth	# tracks	Avg. tpts	Avg. speed (nm/sec)	Avg. direct. persistence
Simulation 1	103	20.0	17.4 ± 0.3	0.25 ± 0.02
Simulation 2	154	18.1	34.5 ± 0.5	0.30 ± 0.02
Simulation 3	146	16.2	69.3 ± 1.1	0.36 ± 0.02
Tracking method results	# tracks	Avg. tpts	Avg. speed (nm/sec)	Avg. direct. persistence
Simulation 1	104	19.8	17.9 ± 0.3	0.25 ± 0.02
Simulation 2	164	16.9	35.5 ± 0.6	0.32 ± 0.02
Simulation 3	177	13.3	68.5 ± 1.1	0.39 ± 0.02

Simulated live datasets are created in Publications PI and PII to validate the tracking method. They were created to resemblance, as close as possible, the experimental data used in the studies. Average speed and average directional persistence are compared between the ground-truth and tracking result in Publication PI and no significant difference is found. In Publication PII, simulations with three levels of complexity are created (Table 3). Particles are set to move at the same speed as analyzed from the experimental data without any clustering in Simulation 1. The average speed is doubled in Simulation 2 with 15% clustering probability of closely located particles. Clustering was defined to increase particle size and average intensity. In Simulation 3, the speed of particles is set to be four times faster than the speed analyzed from the experimental dataset, and the clustering probability is increased to 30%. The tracking results are compared to a ground-truth with four measures: number of trajectories, average time points in trajectories, average speed of particles, and average directional persistence of particle movement. The results (Table 3) show average speed and directional persistence to be within of standard error. However, with increasing speed of the particles, the number of trajectories is increased, and the average number of time points in the trajectory is decreased. This could implicate the use of a too tight maximum distance constraint, d_{max}, or that the method is making wrong decisions leading to premature termination of some trajectories.

To get exact information on tracking method accuracy, all links need to be verified. The results presented in Table 4 show linking accuracy in different simulation scenarios. In Simulation 1, without any clustering, the method gives near perfect results. As expected, with increasing particle speed and clustering, the number of perfect tracks decreases. However, almost all links are correct in tracking of Simulations 1 and 2, with almost 95% of correct links in the tracking results of Simulation 3. In the tracking results of Simulation 1, only a single wrong link (merge event) was enough to make two existing trajectories imperfect and to create one new imperfect trajectory, showing the sensitivity of perfect tracks

TABLE 4 Linking accuracy of the tracking method on simulated data. The ratio of perfect tracks to the total number of tracks, the ratio of perfect tracks and tracks in ground-truth, and the ratio of correct particle links to total particle links, are shown for each analyzed series.

Simulation	Tracks (perfect / total)	Tracks (perfect / truth)	Links (correct / total)
Simulation 1	101 / 104 = 97.1%	101 / 103 = 98.1%	1956 / 1957 = 99.9%
Simulation 2	94 / 164 = 57.3%	94 / 154 = 61.0%	2579 / 2609 = 98.9%
Simulation 3	46 / 177 = 26.0%	46 / 146 = 31.5%	2067 / 2183 = 94.7%

measurement. From the results of link accuracy (Table 4), it can be deduced that statistical measurements, such as those presented in Table 3, can be drawn from all simulations. In Publication PII, a method by Jaqaman et al. (2008) is compared with the presented method in all three simulation scenarios. The presented method gave more accurate results in all of these.

3.4 Method validation

Validation of image analysis methods is imperative in bioimage analysis. However, validation is anything but a straightforward matter, as the ground-truth to compare is generally unknown. Ground-truth phantom can be generated either by simulating data or by having an application area expert to analyze data and form the ground-truth. Both of these approaches have their strengths and weaknesses.

Validation by simulated data is a common approach in bioimage analysis. The strength of using a simulated ground-truth is clear, as the ground-truth is perfectly known. However, simulating data that resembles the characteristics and variation in experimental data is a difficult task (Warfield et al., 2004). This problem can be handled to some extent by creation of different sets of simulated data with different signal-to-noise levels as well as changes in other characteristics typical in imaged data. Another question is raised about what characteristics can be compared with simulated data. For instance, when simulating ground-truth spots by convoluting with point-spread function to form a realistic simulated image with blurry spots, it is clear that the accuracy of spot detection methods can be compared by their success in finding the correct spot locations. Yet, can these methods be expected to segment the ground-truth or blurry particles the way a human expert would? The answer to this question need to be considered whenever simulated data is used to measure method accuracy. As an example, Ruusuvuori et al. (2010) show how big the difference there can be in the accuracy of spot detection methods when comparing locations to segmented pixels.

Another possibility is to use human experts to define the ground-truth from real image data. This can be done directly by the expert or after applying an auto-

matic method the results of which are then edited by the expert. As ground-truths defined by different experts, or by the same expert at different times, vary (Nattkemper et al., 2003; Ruusuvuori et al., 2008), methods (Kamarainen et al., 2012; Raykar et al., 2010; Warfield et al., 2004) have been designed to combine many results to a single ground-truth. In bioimaging, researchers often believe in their eyes more than to automatic image analysis methods. Human brain is a magnificient tool for object detection. However, expert results can be fooled easily with things such as having a non-uniform background in the image (Eagleman, 2001). Culverhouse et al. (2003) noticed on an ecological classification task that experts make mistakes as a result of fatigue, short memory, and possible bias. Though the study is from a different field of science, it can be generalized to experts in cellular biology. In addition, expert validation is in many cases impossible when analyzing hundreds or thousands of small particles in 3D time series.

Simulated data are used to validate and test analysis methods in Publications PI, PII and PIII. All simulations were created in the BioImageXD software (Section 5.1, Publication PI). Decision to use simulated data instead of expert-defined ground-truths were made since real image data are too complex to be analyzed by a human expert (Publications PI and PII) or method is tested in a setup for which no real data was available (Publication PIII). Altogether nine different combinations of signal-to-noise levels and clustering of objects are used in both fixed and live simulated data in Publication PI to validate segmentation and tracking analyses (Fig. 10). A number of objects, average volume and average intensity are set according to real data analyses. In addition, objects are set close to the membrane extracted from real image data, and defined to internalize at the same speed as in real data. Similarly, parameters are extracted from real live data analyses to setup simulated live data. However, objects are not segmented from live data for tracking method validation. Instead, ground-truth particles are used to confirm that possible problems in segmentation will not affect the testing of particle linking accuracy. The methods used for both fixed and live sample analyses are validated successfully. Simulated data for testing the analysis methods in Publications PII and PIII are also setup according to experimental data (details in Sections 3.3 and 3.2).

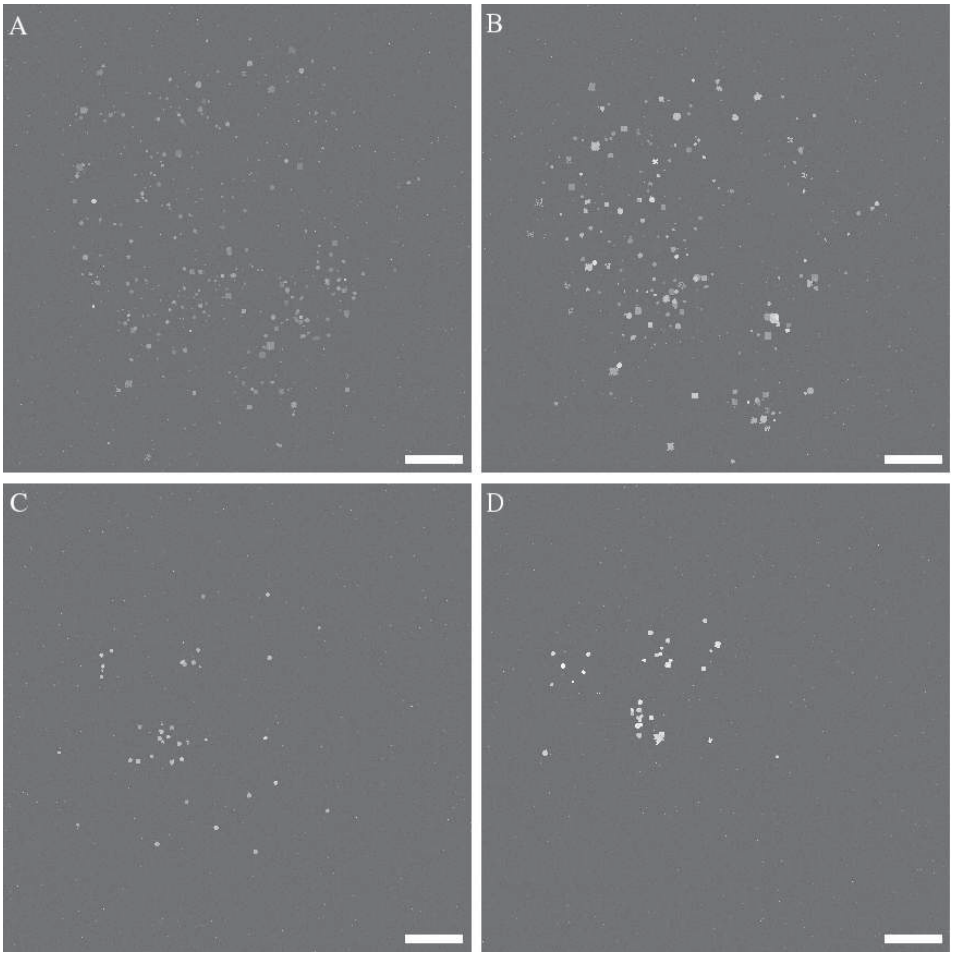


FIGURE 10 Examples of simulated fixed (top row) and live (bottom row) data in Publication PI from a cell bottom slice. All examples are from the middle noise level and clustering simulations. Simulations representing the 15-minute time point (left) and the 45-minute time point (right). Linear enhancements in the images are for visualization purposes. Scale bars are 5 µm.

4 SINGLE-PARTICLE RECONSTRUCTION

SPR is a method used to solve structures of large macromolecular complexes in near-atomic resolution, typically in resolution between 6 to 15 Ångströms (Jonić et al., 2008). Even though a resolution as good as that with crystallography techniques (Jonić et al., 2008) has not been achieved with SPR, it has many advantages over crystallography: mainly, its usability to solve structures in its original state and ability to solve complex structures that are hard to crystallize (Frank, 2002).

SPR is based on taking 2D projection images of numerous individuals of the same particle in random orientations. This differs from typical tomography-based reconstruction methods in that one sample includes tens or hundreds of particles and that the sample is usually not tilted. Instead, particles are expected to be in random orientations in the sample. Thousands of particle images are needed for reconstruction for two reasons: (1) to effectively average out noise from the high resolution particles imaged with a low electron dose, and (2) to get good-quality particle averages from many orientations. Henderson (1995) calculated that theoretically about 10 000 particle images are needed to achieve atomic scale resolution, but Glaeser (1999) estimates that, in practice, more than million are needed to average out noise and distorted particles. Even with typical resolutions achieved with SPR, tens of thousands of particle images are used. However, the number of particle images needed is greatly dependent on particle symmetry and complexity, and therefore the universal number of particle images needed to achieve a specific resolution cannot be given.

SPR is a computationally demanding iterative process. The main parts of the SPR pipeline are presented in Figure 11. An SPR process starts by sample preparation and imaging of samples in TEM. Typically, samples are fixed on vitreous ice (cryoEM) to get a stable sample for imaging (Frank, 2006b). Imaging is carried out with a small defocus to reach the best possible resolution. However, small defocus complicates the subsequent steps in the SPR pipeline, affecting the quality of particle selection, classification, and reconstruction. Often, a large defocus results in more usable particles in the later steps of SPR, and as such, has an effect on the number of samples needed. Thus, a balance between resolution and usable images for reconstruction must be found. Imaging is followed by a particle

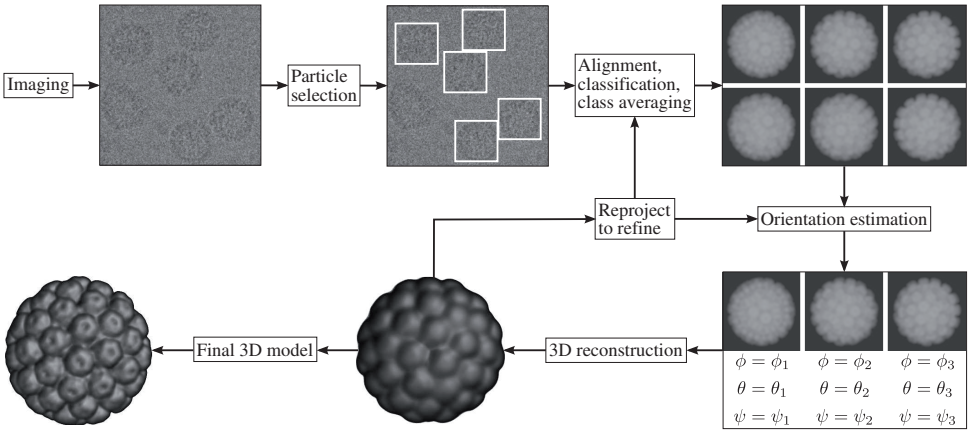


FIGURE 11 Outline of the SPR process (Ruprecht and Nield, 2001; Jonić et al., 2008; van Heel et al., 2000). The SPR method is highly iterative (the main parts in the figure are iterative as well) for gradual reconstruction of a high resolution model from projections of particle in extremely low SNR electron micrographs. A Simian virus 40 model edited from (Shen et al., 2011).

selection step to extract individual particle images from the micrographs. Particle selection is presented in Section 4.1, and literature review on particle selection methods in Section 4.2, followed by a presentation of a novel particle selection method in Section 4.3. Next, the steps following particle selection are presented shortly.

After particle selection, particle images are aligned and grouped into similarity classes. Alignment is carried out to translate all particles to the image centers and to rotate to the same orientation as the other images in the class. Initial alignment is often done by finding a maximum correlation of a particle image and an average image of all particles (Ruprecht and Nield, 2001; van Heel et al., 2000). Any classification method can be used to define the classes of particle images. Typically, the process is started by computing a set of eigenimages, using a multivariate statistical analysis by correspondence analysis or a principal component analysis (Frank, 2006b). To ease the computational burden, covariance matrix is not computed. Instead, eigenimages are generated iteratively by calculating the inner product of eigenimages and particle images. The initial eigenimages for the iterative process are generated randomly (van Heel et al., 2000). After multivariate statistical analysis (detailed description of the process can be found in Frank (2006b); Borland and van Heel (1990); van Heel et al. (2000)), each particle image is represented by a j-dimensional vector in $\mathbb{R}^j$ factor space, where j is the number of eigenimages. Typically the number of eigenimages used is between 10 to 100 (van Heel et al., 2000).

A compressed representation of particle images as a linear combination of eigenimages is used to create a set of classes. Classes can be formed with any clustering method, for instance with k-means clustering (MacQueen, 1967). The most used method is the hierarchical ascendant classification (van Heel et al.,

2000). It starts by defining all particle images as individual classes. These classes are merged together by minimizing the intra-class variance, measured by the distance of particle images in factor space, until only a single class exists. Merging of classes forms a tree presentation. From the merging tree, a single level with a pre-defined number of classes is selected to present the final classification of the particle images (van Heel et al., 2000; Frank, 2006b).

After classification, particle images in a class are averaged to form a class average that has SNR that is significantly higher than that of each single particle image (van Heel et al., 2000). After the first class averages are calculated, the whole alignment and classification process is refined using class averages instead of average of all particle images as references (Ruprecht and Nield, 2001). This process is typically refined a few times until class averages do not change significantly.

Before the initial 3D model of the particle can be reconstructed, orientations of each class average have to be estimated. A method called *angular reconstitution* (van Heel, 1987; Serysheva et al., 1995; van Heel et al., 2000), based on the *common line* projection theorem (De Rosier and Klug, 1968; Crowther et al., 1970; van Heel, 1987), is used to estimate the initial Euler angles of the class average images. The common line theorem states that two 2D projections of a 3D object have a one-dimensional common line (van Heel et al., 2000). To find common lines between class averages, first a sinogram of each class average is formed. Sinogram is a Radon transform (Radon, 1986) of a class average including 360 one-dimensional projections of a class average rotated by one degree intervals. Next, the maximum correlation coefficient of sinograms of two class averages that defines the common line is search (van Heel et al., 1997). Finally, the angles between the common lines of different class averages are used to estimate the orientation of each class average. During the angular reconstitution process, bad class averages that show low correlation with other class averages, can be removed.

Once particles have been selected from the micrographs, classified into classes, class averages calculated, and Euler angles of class averages estimated, the first 3D model of the particle can be reconstructed. Any 3D reconstruction method that can handle random tilts is usable. Typically, backprojection methods are used for their efficiency and simplicity (Frank, 2006b; Penczek, 2010). Details of backprojection and other reconstruction methods are described in Section 2.3.

After the initial 3D model of the particle is reconstructed, the model is reprojected to create a set of projections with known orientations. This set, including tens of projections, is used to refine the Euler angles of class averages estimated with the common lines method, until the model is stable (van Heel et al., 2000). Many methods have been developed for reference-based orientation estimation that employ for instance correlation of images to polar Fourier transformed (Baker and Cheng, 1996) or wavelet transformed (Sorzano et al., 2004) projections to find the closest reference orientation. Once the orientations of class averages have been refined, the whole process from alignment to reconstruction is refined iteratively until the reconstructed 3D model converges. During the it-

erative refinement process, hundreds of projections of the reconstructed model are used as references. The density of reference projections is increased at every iteration to increase the accuracy of class averages. The number of reference projections needed depends on the complexity of the particle (Serysheva et al., 1995). Finally, the highly iterative SPR process results in a final 3D model of the particle.

4.1 Particle selection

The first task in the SPR process is extracting particles from stained or cryoEM micrographs to be used in the reconstruction process. Particle selection is generally also the most labor-intensive part. Traditionally, particle selection (boxing) was done manually (Ramani Lata et al., 1995). However, boxing hundred thousand particles can take a month; it has poor repeatability and user bias on manual selection, and, therefore, automatic or semi-automatic methods are favored nowadays.

Boxing particles from extremely low SNR electron micrographs is an intractable task. High noise and low contrast are the result of the low electron dose used in imaging to prevent damage to the sample (van Heel et al., 2000). Since particles in micrographs can be in any orientation, it is often easiest to detect symmetrical particles whose shape in a projection is the same in all orientations. Asymmetric particles are more complicated as the size and shape of the particle can vary considerably depending on the orientation of the particle. Small asymmetric particles are the most difficult particles to detect, and even for experts to find manually. Fortunately, all of the particles do not have to be detected from micrographs. According to Glaeser (2004), more than 75% of particles should be picked up by a particle selection method. It is clear that from the particle reconstruction point of view, it does not matter what percentage of particles are selected as long as all orientations are included, although more micrographs are needed when fewer particles are detected from each micrograph. A more important measure is the number of wrong selections (false positives): typically background noise, damaged or overlapping particles, and contaminations (Figure 12). Non-particle images can have significant impact on the accuracy of the reconstruction process. However, when the number is sufficiently low, false images are mostly averaged out during reconstruction process. Glaeser (2004) suggests that the number of non-particle images should be less than 10%. The exact number is mostly dependent on the particle, number of class averages, and how non-particles are distributed in different classes. In any case, the particle selection method should pick as many true particles as possible while minimizing the number of false particles. Often minimization of false particles is more important.

Accuracy of particle selection methods is often measured with a few simple statistical measures. True positives (TP) is the number of particles correctly found from micrographs while false positives (FP) is the number of wrongly detected particles, and false negatives (FN) is the number of real particles not selected.

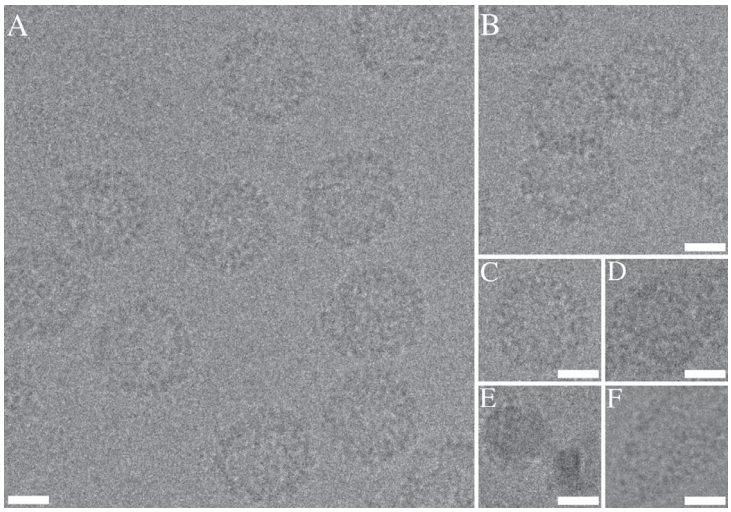


FIGURE 12 Raw cryoEM micrograph (A) and common problematic regions to particle selection methods. Only single isolated particles (C) should be picked. The common problematic regions are particles touching (B) and overlapping (D), as well as contamination (E) and other particles in the micrograph (F). Scale bar of 20 nm in all images.

From these measures, simple statistics can be drawn to measure the rate of particles detected and the rate of wrong particles detected. The statistical measures of *precision* (P) and *recall* (R) are defined as (Langlois and Frank, 2011):

$$P = \frac{TP}{FP + TP} \qquad 1 - P = \frac{FP}{FP + TP} \tag{17}$$

$$R = \frac{TP}{FN + TP} \qquad 1 - R = \frac{FN}{FN + TP} \tag{18}$$

It can be seen that precision (Equation 17) measures the ratio of correct particle picks (TP) to all picks ($FP + TP$), whereas recall (Equation 18) measures the ratio of correct particle picks (TP) to all existing particles ($FN + TP$). $1 - P$ is known as *false discovery rate* and $1 - R$ as *false negative rate* (Langlois and Frank, 2011). For most of the methods, these measures are interconnected. It is common that while recall increases precision decreases and vice versa, an effect often presented as a precision-recall graph (Arbeláez et al., 2011). The functionality of the classifier in particle selection method can be also presented in ROC graph (Fawcett, 2006) showing false positive rate ($FP/(FP + TN)$) and recall.

Several methods use separable particle detection and particle classification or clustering. The first step is to detect the candidate particles with the aim of detecting all real particles. A set of candidate particles will in principle always include also false particles. Next, classification or clustering is used to increase precision by removing as many false particle images as possible. If only false particles are removed in the classification or clustering step, recall is the same as after the detection step while precision increases. In practice, when precision

increases, recall decreases, as also true particles are removed.

4.2 Literature review on particle selection methods

Many different strategies have been used to select particles from electron micrographs (Zhu et al., 2004; Nicholson and Glaeser, 2001). The methods typically have separate particle picking and classification steps. However, for instance, widely used semi-automatic methods rely on the user to do final classification by ruling out ”junk” from detected images. Since methods cannot be clearly classified into specific classes, the review of existing methods will first focus on candidate particle picking schemes and then to classification and clustering strategies.

Template matching is a traditional method used to pick particles from electron micrographs (Nicholson and Glaeser, 2001). Reference-based methods of this kind start by creating a set of 2D template images from different angles of the particle. These images can be created from a previously solved model of the particle or by reconstructing a scarce model from a few hundred or thousand manual selections. A raw force method is to cross-correlate all templates with micrograph to find high peaks in the correlation coefficient. Precision and performance are problems in template matching with traditional cross-correlation of tens of template images with micrograph.

Many variations of general template matching have been proposed to improve performance and accuracy. Roseman (2003) suggested improving accuracy of template matching by using local correlation that eliminates the effect of non-uniform background illumination common in electron micrographs. Chen and Grigorieff (2007) present spectrum correlation with local correlation to include both intensity and shape in correlation function. To improve the performance of template matching, modified representation of templates can be used. Local correlation was used by Huang and Penczek (2004), who clustered a set of 98 template images to five by k-means clustering (MacQueen, 1967) and rotated to a final set of 30 templates. Wong et al. (2004) did similar clustering of templates, but instead of using cross-correlation, they use convolution of templates and micrograph. Sigworth (2004) reduced the number of templates from 64 to 14 using principal component analysis to create a set of eigenimages from templates.

To improve performance in orders of magnitude, template matching can be done with a model. Volkmann (2004) defined reduced representation of template image with less than 100 binary points. With this representation of templates, they reported a 1000-fold improvement in performance compared to normal cross-correlation (Volkmann, 2004). Some methods use even a simple Gaussian profile for template matching (Hall and Patwardhan, 2004; Ramani Lata et al., 1995) or Difference of Gaussian filter (Voss et al., 2009) to get candidate particles. It is clear that these methods generate many false candidate particles, and as such, do not give satisfactory results without classification or clustering to remove false positives.

The results of template matching are correlation coefficient images, one for each template. From these images, a search for correlation peaks is performed to find candidate particles. Selecting correct peaks is not an unambiguous task. Many methods use user defined threshold values to select peaks (Volkmann, 2004; Huang and Penczek, 2004), but the information from the original model can also be used (Wong et al., 2004). These methods proceed differently after a set of peaks has been selected. Some methods accept all peaks as is while many compute the distance between peaks. All peaks that are closer than the expected size of the particles can be removed (Chen and Grigorieff, 2007; Volkmann, 2004) or the highest peak kept while others are removed (Sigworth, 2004). Finally, candidate particles are extracted using peaks that have been left as centers of particle images. Template matching methods typically result in a high recall measure, as particles create a high peak in the correlation images. However, also low precision is common since artifacts with the size of a particle will often also create a high correlation coefficient (Zhu et al., 2004).

Some research has been done on edge detection (Yu and Bajaj, 2004; Zhu et al., 2002) and segmentation-based (Singh et al., 2004; Adiga et al., 2005) methods used regularly to solve many macroscopic and light microscopy image analysis problems. In general, these methods do not work robustly enough since particles in low-SNR cryoEM micrographs rarely have distinguishable edges or shape. A thorough search of suitable parameters and even filters needs to be done for each different particle selection case. However, working protocols have been reported, and the advantage of these methods is that no initial model or manual selection of particles is needed. Methods based on edge detection start by searching a large continuous gradient from the image, typically with the Canny edge detection method (Canny, 1986). In (Zhu et al., 2002), edge detection is followed by either circular or rectangular Hough transform (Duda and Hart, 1972). The method does not distinguish contamination of the size of true particle, and cannot be used for asymmetric particles. Yu and Bajaj (2004) replaced Hough transform with distance transform, and in case of rectangular particles, also used the Voronoi diagram (Aurenhammer, 1991) to estimate the pose of rectangle. Peaks of distance transform were, after several refinement steps, defined as centers of either circular or rectangular particles.

Segmentation-based methods differ from edge detection-based methods in that a certain geometrical shape is not expected. Segmentation-based methods typically use variations of filtering, thresholding, and morphological operations to segment the particle, and in the end filter the found objects by a predefined size range. Singh et al. (2004) proposed a segmentation-based method that uses a hidden Markov random field initialized with the Otsu threshold (Otsu, 1979) and refined by an expectation maximization algorithm (Dempster et al., 1977). Singh et al. (2004) reported about 17% for false discovery rate. A high percentage of false positives is expectable with segmentation-based methods without a final particle pruning step. A similar false positive rate was reported by Adiga et al. (2005), who applied a combination of global and local thresholding and morphological operations to detect particles.

Ogura and Sato (2004a) presented an interesting approach to particle picking. In their method, a specified number of windows is initialized in a micrograph and then moved around by Simulated Annealing optimization (Kirkpatrick et al., 1983). This holds potential to effectively reduce the number of false positives, as only a specified number of particles are detected. It seems that the metric used in optimization is not robust enough, and the precision from 50% to 95% was reported with different particles (Ogura and Sato, 2004a). According to Ogura and Sato (2004a), a final classification with supervised neural networks could be used to improve precision. The downside is that supervised classification needs manual labeling of training images.

Both supervised (trained with labeled data) and unsupervised (untrained or trained with unlabeled data) classification/learning methods are often used after candidate particle picking. Some methods use classification to remove some of the false positives from already a good set of candidate particles, while others rely almost completely to classification after picking all regions differentiating from background noise. The classification methods as well as features used vary significantly.

A common use case of supervised learning method is to manually classify a set of initial particle images to particles and non-particles (Arbeláez et al., 2011; Ramani Lata et al., 1995; Ogura and Sato, 2004b) used to train the method. It is also possible to train a classifier to recognize different kinds of particles or non-particles such as contamination and background. After a training phase, the classifier is used to classify particles automatically using calculated feature vectors from particle images. Texture features (Arbeláez et al., 2011; Ramani Lata et al., 1995) are typically used in feature vectors.

Different kinds of neural networks, both supervised and unsupervised, have been used to solve the classification problem (Hall and Patwardhan, 2004; Ogura and Sato, 2004b). Ogura and Sato (2004b) used supervised neural network trained with hundreds of manually picked particle and background images. They reported as high as 90–95% of precision without any particle detection method, but testing all pre-defined size regions from the micrograph. An unsupervised neural network called the self-organizing map (Kohonen, 1990) was used by Hall and Patwardhan (2004) after initial detection of candidate particles with the correlation of Gaussian profile. From a self-organizing map, different clusters were defined with k-means clustering (Hall and Patwardhan, 2004). Hall and Patwardhan (2004) did not report the precision of the method. However, 3D reconstructions were made with manual and automatic particle selection, showing that the automatic method with more particles selected gave improved resolution.

Supervised classification methods other than neural networks have been recently used. These include support vector machines (Arbeláez et al., 2011) and naive Bayesian classifier (Sorzano et al., 2009). Arbeláez et al. (2011) trained a support vector machine with hundred to thousand manually selected particles, and reached a precision of 80–90% with texture-based selection of candidate particles. Sorzano et al. (2009) reported as high as 90% precision using an ensemble of naive Bayesian classifiers without candidate particle picking. However, a recall

of less than 70% was reported with high precision values (Sorzano et al., 2009).

It is clear that a particle selection method cannot give satisfactory results without an additional classification step if both precision and recall measures should be high. Traditionally, template matching methods have been the most used. Recently, supervised classification methods without any candidate particle picking have shown good results. For supervised methods, a combination of candidate particle picking and classification trained with manually selected particles is most likely to give the best results. Research on unsupervised methods with clustering of candidate particle picks has shown promising results. The advantage of unsupervised methods is that no initial model or selection of training particles is needed. Even though common measures for the goodness of particle selection methods have not been established, Langlois et al. (2011) compiled a table including the precision and recall values of the methods that have been used on the labeled Keyhole Limpet Hemocyanin dataset (Zhu et al., 2004). On average, the precision of the methods was around 80–90% and their recall about 70–90%.

4.3 Novel particle selection framework

A novel framework was developed as a basis to reference-free particle selection in cryoEM micrographs. Detection and recognition of correct particles from cryoEM micrographs is a challenging task especially when no reference is available. For this reason, a special focus on both candidate particle search and unsupervised classification is needed. Next, details of the particle selection framework are presented with application to boxing spherical Simian virus 40 particles. The method details are followed by the results of an extensive study with simulated data and results with experimental data. The framework is implemented in the BioImageXD software presented in Section 5.1.

4.3.1 Details of the particle selection framework

Our particle selection framework (Algorithm 3) is based on iteratively moving boxes and classification of final boxes. Previously, only Ogura and Sato (2004a) have suggested a method which utilizes iteratively moving boxes. However, their method relies on user-defined maximum number of particles, uses randomized movement by simulated annealing optimization with different box goodness evaluation, and does not include any classification of the final set of boxes.

The particle selection framework is based on image features computed within every box. In principle, the same features could be evaluated in every pixel of an image. However, this would lead to an intractable computational task with large box sizes and nonlinear features, such as many texture features, and to obscure peak selection from the feature image.

A box is defined by its centroid and size in an image grid. A box is always

rectangular but can hold a mask of an arbitrary shape. The framework utilizes image features calculated inside of a mask in the box. *Feature vector* is defined as vector $v = [v_1, v_2, \ldots, v_k]$ of computed features, where $k \in \mathbb{N}$, and $v_i \in \mathbb{R}$. Different features can be weighted with *feature weights*, $w = [w_1, w_2, \ldots, w_k]$, where $w_i \in \mathbb{R}$. Using weights and features, a *feature value* is defined as $f_v = \sum_{i=1}^{k} w_i * v_i$, where $f_v \in \mathbb{R}$. Finally, a *feature function* is a function that returns a feature value computed from a box in given coordinates, $f : \mathbb{N}^2 \rightarrow \mathbb{R}$.

Initialization. The method is initialized with a sparse grid of boxes. The distance between adjacent boxes is not fixed. Smaller distance increases computation time and accuracy of the method. Larger distance has opposite effects. Distance of approximately half of the maximum diameter of the box is used to balance between efficiency and accuracy. However, research on the optimal distance have not been made. Feature function is evaluated in every box, using defined features, feature weights, and mask. Once all the initial boxes have been created, the functions to normalize feature distribution to zero mean with the standard deviation of one are created. Normalization is done to get a similar impact from each feature on particle search.

Minimization of boxes' feature values. After initialization, local minima of feature function are searched by iteratively moving boxes. The minimization is done by a gradient descent method. To estimate the gradient of a feature function, the complete feature function is first approximated with multilevel cubic B-splines (Lee et al., 1997), using previously evaluated locations as known values. The gradient is calculated from a feature function approximation, and normalized to have a magnitude in the range of 0.0–1.0. After the gradient approximation, each box is moved in the gradient descent direction by a length of the gradient magnitude by a user-defined step size. If the maximum step size is too large, the box might fluctuate around the local minima without ever reaching it. Typically, the maximum step size is set to be smaller than the particle diameter, often half of the diameter. However, the optimal value depends on the data. With a larger step size, the method might converge faster, but to a different solution than with the smaller step size. As the number of boxes decreases during iterations (explained later), and the movement of some boxes stops in local minima, the number of new evaluations decreases in every iteration. This also decreases the change in the feature value approximation in every iteration. To prevent unnecessary computation, the feature function is approximated only every 2^k, where $k \in \mathbb{N}$, iterations.

Fusion of boxes and weighted feature sum. It is common that during iterations, many boxes will move to the same local minimum. To ease computational burden, and classification after a candidate particle search, boxes can be fused together during iterations. A search for closely located boxes is done at the end of every iteration. First, boxes are ordered by their feature values. Next, starting from the box with the highest feature value, a search for boxes locating closer than the minimum distance $||b^i_{x,y} - b^j_{x,y}|| < d_{min}$, where $b^k_{x,y}$ is the location of box b^k, is conducted. The box with a higher feature value is fused to the other box. The method convergence is checked after every iteration by calculating a weighted

Algorithm 3 Outline of the particle selection framework

Input: TEM micrograph I of size (x_{max}, y_{max})
$B = \{\}$ // Set of boxes
$d =$ Maximum diameter of searched particle
$k =$ Initial distance between boxes
$F =$ Image used to store known feature values
for all $x \in \{d/2,\ d/2+k, \ldots,\ x_{max} - d/2\}$ **do**
    for all $y \in \{d/2,\ d/2+k, \ldots,\ y_{max} - d/2\}$ **do**
        $f = EvaluateFeatureFunction(I, (x, y))$
        $F(x, y) = f$
        $b = CreateBox((x, y), f)$
        $B = B \cup \{b\}$
$f_n = CreateFeatureNormalizationFunctions(B)$
$i = 1$
repeat
    if $log_2 i \bmod 1 = 0$ **then**
        $F_{approx} = ApproximateFeatureFunction(F)$
        $G = Normalize(ComputeGradient(F_{approx}))$
    for all $b \in B$ **do**
        $b_{x,y} = b_{x,y} - G(x, y) * max\_stepsize$
        $b_f = EvaluateFeatureFunction(b_{x,y}, f_n)$
        $F(b_{x,y}) = b_f$
    $B = FuseCloseBoxes(B)$
    $f^i_{sum} = WeightedFeatureSum(B)$
    if $i = 1$ **then**
        $f_{change} = f_{min} + 1$
    else
        $f_{change} = |f^i_{sum} - f^{i-1}_{sum}|$
    $i = i + 1$
until $i > iter_{max}$ or $f_{change} < f_{min}$
for all $b \in B$ **do**
    $b_{x,y} = AlignBox(b_{x,y})$
$B = RemoveOverlapping(B)$
$C = ClassifyBoxes(B)$
return C

feature sum of boxes B, $f^i_{sum} = \sum_{b \in B} b_f * (1 + b_{fused})$, where b_f is a box feature value, and the weight is defined by the number of boxes fused in the box. The iterations of the candidate particle search are stopped when either the change in the weighted feature sum between adjacent iterations is smaller than the defined convergence value, or a maximum number of iterations have been done.

Final alignment and overlap removal. A feature-based particle search performs well on finding a particle. However, a fine-tuned alignment should be done with other methods. Centering is done by cross-correlating a mask in a small region around the box centroid to find the maximum correlation between the particle and the mask. After alignment, overlapping boxes are searched and removed with the method used as the boxes were fused during iterations.

Unsupervised classification of boxes. The final set of boxes needs to be classified to remove false positives such as contamination, overlapping particles, and background. Unsupervised classification is done by clustering in feature space. The common k-means clustering method is not optimal for clustering of candidate particles for many reasons. First, the initial number of clusters is not known *a priori*. Next, initialization of clusters is not straightforward. Finally, k-means clustering does not detect outliers unless set into their own cluster. To find a solution to these problems, a density-based clustering method DBSCAN (Ester et al., 1996) was used. DBSCAN has many advantages over k-means clustering. It does not need any initialization or *a priori* knowledge of clusters, and it can detect and mark outliers even with very different measurements. DBSCAN takes only two parameters: ϵ used to define the search range of measurements and a minimum number of measurements needed to be found in the search range to be considered as part of the same cluster. Typically, the second parameter can be fixed to some percentage of the total number of measurements, and configure the method only with the ϵ parameter. To be able to prune touching or slightly overlapping particles, the box size is enlarged by 50% before clustering. Finally, particle images of 25% enlarged box size (needed for later SPR steps) are extracted from the micrograph to their respective classes.

4.3.2 Particle selection results

A particle selection method was tested with simulated and experimental micrographs of Simian virus 40 (Stehle et al., 1996) particles. Simulated data with a known ground-truth was created to test the accuracy of the proposed particle selection framework with varying overlap scenarious and different contrast levels. CryoEM micrographs were simulated with the TEM-simulator (Rullgård et al., 2011). The main parameters defined for the simulations are presented in Table 5. Additionally, the parameters related to the optical system, spherical and chromatic aberration, and focal length were set to known values of TEM used in experimental data imaging. Detector quantum efficiency and modulation transfer function parameters, used to define noise in the micrographs, were adjusted experimentally to get micrographs with a histogram as similar as possible to the histogram of the experimental data.

TABLE 5 Main parameters used in SV40 micrograph simulations.

Parameter	Value
Acceleration voltage	200 kV
Defocus	1.2 µm and 2.5 µm
Electron dosage	10 e/Å^2
Ice thickness	100 nm
Image dimensions	4096 x 4096
Image pixel size	2 Å
Magnification	50000
Physical pixel size	10 µm

Particles were set to touch or overlap with three different levels: no overlap, maximum of 15% of particles overlapping, and maximum of 30% of particles overlapping. Each particle (approximately 50 nm diameter) was embedded in a random depth in ice, and the orientation of the particle was randomized. A number of particles in each micrograph was drawn from normal distribution $N(\mu, \sigma)$, where $\mu \in \{50, 60, 70\}$ was increased with the overlap level, and $\sigma = 10$. An example of the simulated data is shown in Figure 13. Ten repetitions of all defocus and overlap combinations were made, making the total number of simulated micrographs 60.

Texture features from the gray-level run-length matrix (Galloway, 1975; Chu et al., 1990; Dasarathy and Holder, 1991), Haralick's texture features (Haralick et al., 1973), and mean intensity were tested as features for particle selection in SV40 micrographs. As the searched particle is spherical, a circle mask was used inside the boxes to define the region of interest for feature computation. Also, for the final alignment, a thin circle ring mask was used to take advantage of the distinguishable fringe of the particle.

Run-length features (Galloway, 1975; Chu et al., 1990; Dasarathy and Holder, 1991) are computed from the run-length matrix, which is a $M \times N$ matrix, where M is the number of different intensity levels used and N is the number of run-lengths. Floating point intensity values were binned to 16 intensity levels, and the maximum run-length was defined as particle diameter. Each element (i, j) in the run-length matrix includes information on the number of runs of j length of i intensity level found in the image. Run-lengths are dependent on the direction they are measured. Element (i, j) of the run-length matrix computed in θ direction is defined as $p(i, j, \theta)$. Haralick's texture features (Haralick et al., 1973) are computed from the co-occurrence matrix, a $M \times M$ matrix, where M is the different intensity levels used as in the computation of the run-length matrix.

Different features were tested in the particle selection framework with a different set of simulated and experimental data from the one that was used to produce the results presented later. It turns out that in low SNR cryoEM micrographs many features either fail to distinguish particles from the background or are correlated and produce exactly the same results. Finally, two run-length features, grey level non-uniformity (gln) (Galloway, 1975) and low grey level run empha-

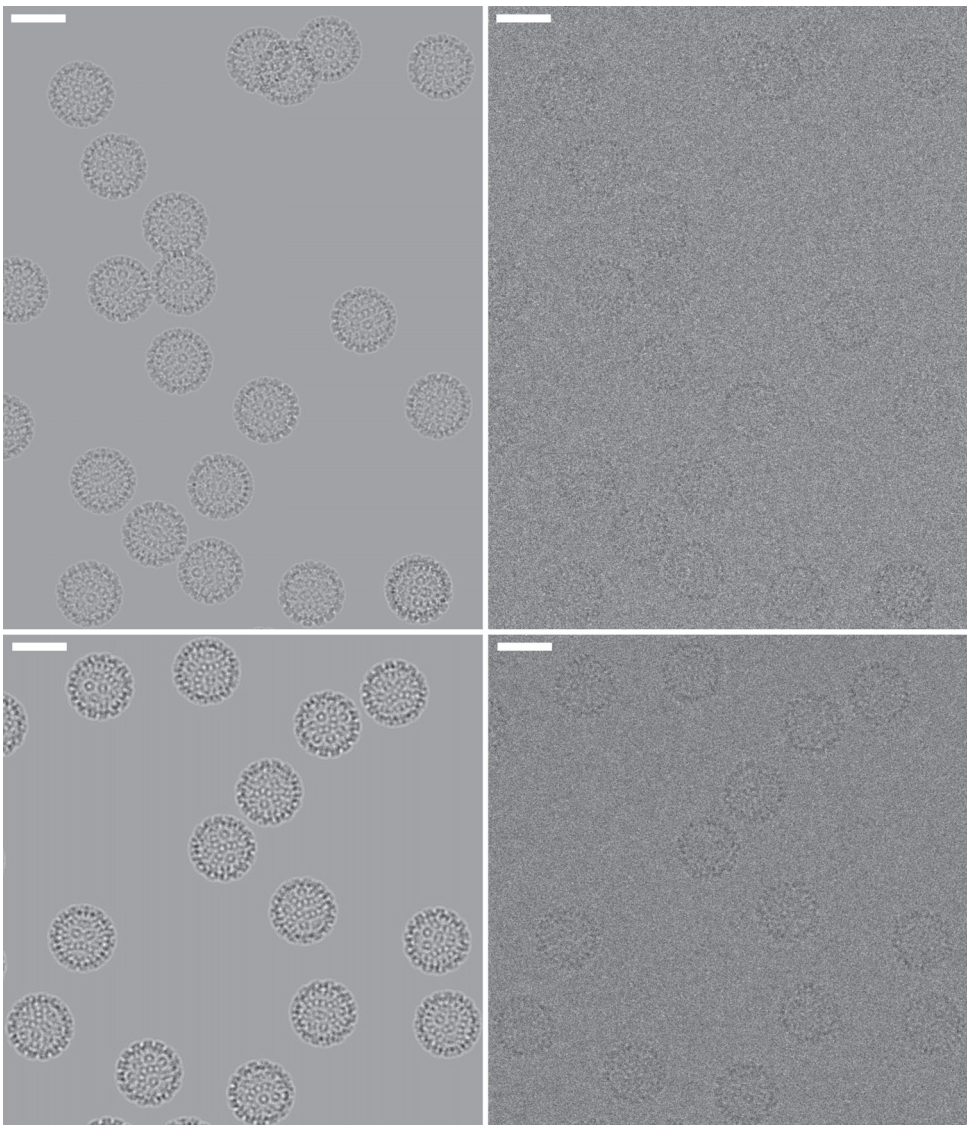


FIGURE 13 Examples of simulated micrographs with two different defocus levels including SV40 particles. Top row: Defocus 1.2 µm with a third overlap level includes many touching or overlapping particles. Bottom row: Defocus level 2.5 µm with only isolated particles. Noiseless images (left) and final simulations (right). Scale bar 40 nm.

sis (lgre) (Chu et al., 1990), were used in candidate particle search and box classification. The gln and lgre features are defined as (Galloway, 1975; Chu et al., 1990):

$$gln(\theta) = \frac{1}{n}\sum_{i=1}^{M}\left(\sum_{j=1}^{N} p(i,j,\theta)\right)^2$$

$$lgre(\theta) = \frac{1}{n}\sum_{i=1}^{M}\sum_{j=1}^{N}\frac{p(i,j,\theta)}{i^2}$$

where $n = \sum_{i=1}^{M}\sum_{j=1}^{N} p(i,j,\theta)$ is the total number of runs. Texture features were computed in all four principal directions $\theta \in \{0°, 45°, 90°, 135°\}$ to make the features rotationally invariant. The final gln and lgre texture features were averages over all directions. The features were weighted to define a single feature value $f_v = 1.0 * gln - 1.0 * lgre$ used for particle search.

The particle selection results on simulated data with defined features are presented in Table 6. The results of candidate particles are presented with the final results using two different values of ϵ in DBSCAN clustering. The aim of the initial candidate particle search is to find as many particles as possible by keeping false particles at minimum. The number of false particles can be further decreased in classification. More than thousand boxes that were initialized were reduced to approximately 90 by the end of iterations. It can be seen that the candidate particle search was able to find almost all particles with only a few false negatives. The number of false negatives increased with smaller defocus simulation, as expected. The number of true positives was close to the false positive boxes before classification. The results show that the method was able to find particles with a great accuracy, as shown by recall values greater than 0.93 in all simulations. Interestingly, recall was highest with simulations having the highest rate of overlapping or touching particles, as fewer true positives were removed in classification. False discovery rate (1-P), important for later SPR steps, was low with 0.055 being the highest false discovery rate and 0.023 with simulations with no overlapping or medium rate of overlapping particles. The smaller epsilon value for classification made false discovery rate slightly smaller while also the recall was decreased as expected.

As can be seen from the results presented in Table 6, the epsilon parameter in classification can be used to decrease the number of false positive particles. The simulation result of the worst case scenario was used to study the effects of epsilon parameter (Figure 14). It can be seen that smaller epsilon will decrease recall. However, as epsilon can be used to decrease false discovery rate to some extent in general, it does not necessarily have huge impact as the number of true positives decreases while false positives are removed. With the worst case simulation, including many particles which barely touch, the current setup was not sensitive enough to prune out all false positives, as can be seen from the ground-truth in feature space (Figure 14). To solve the problem, either different features or larger box size for classification would be needed.

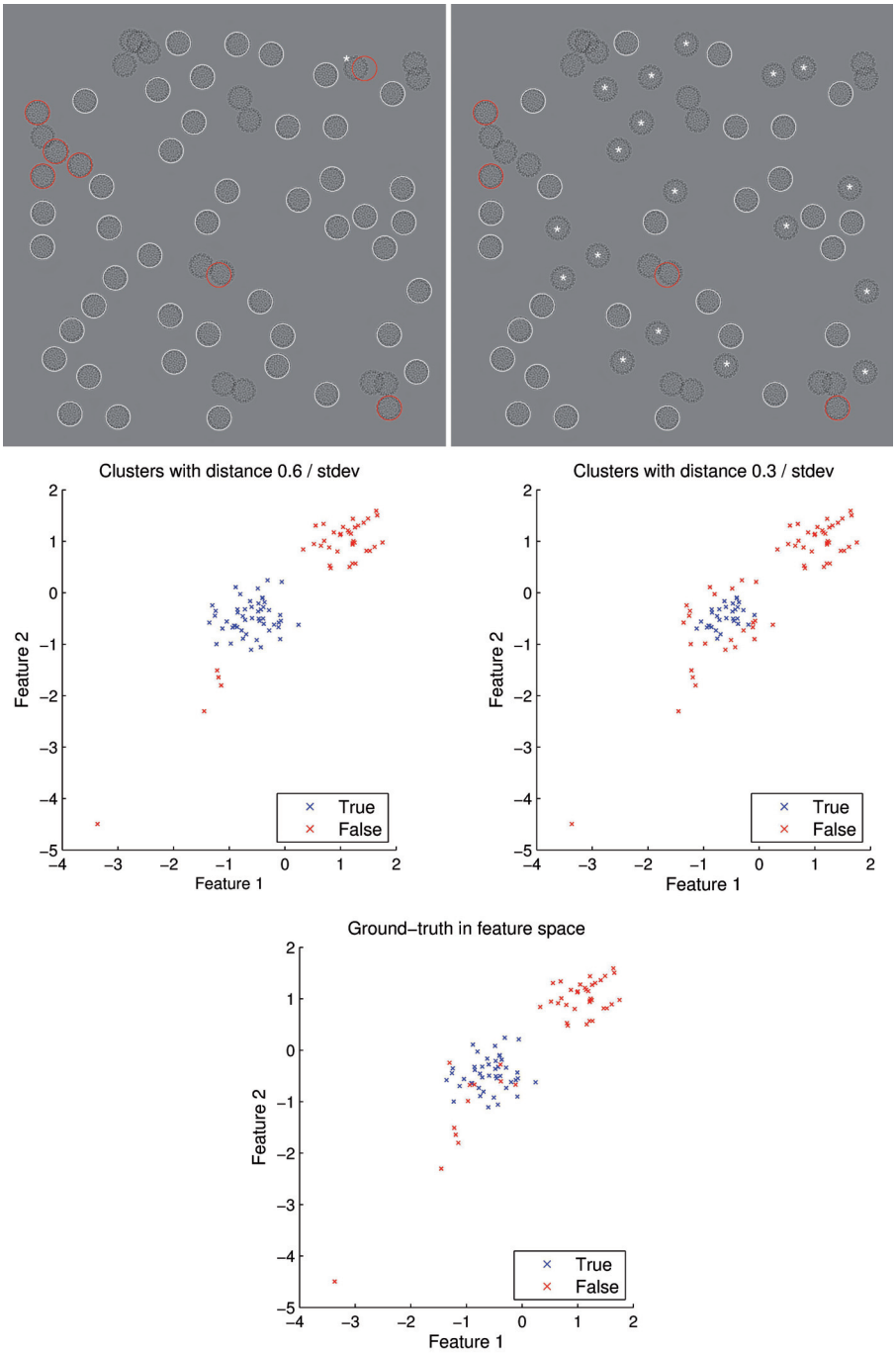


FIGURE 14 Particle selection results with different ϵ values in clustering. Particle selection results with $\epsilon = 0.6/\text{stdev}$ (left) and $\epsilon = 0.3/\text{stdev}$ (right) drawn in noiseless micrographs (1.2 µm defocus) for better visual interpretation, and presented in feature space. Bottom row: Ground-truth result in feature space.

TABLE 6 Particle selection results for simulated data shown for each defocus and overlap level combination. The total number of true particles in simulations is 556 (no overlap), 484 (mid), and 470 (high). The number of true and false positives after candidate particle selection (no classification), and the results with two different epsilon values for classification, including recall and false discovery rate (1-P), are shown.

Simulation		No class.		$\epsilon = 0.5$ / stdev				$\epsilon = 0.6$ / stdev			
Def.	Overlap	TP	FP	TP	FP	R	1-P	TP	FP	R	1-P
1.2	No	545	383	518	4	0.932	0.008	536	4	0.964	0.007
	Mid	478	427	451	8	0.932	0.017	465	10	0.961	0.021
	High	467	444	456	25	0.970	0.052	461	27	0.981	0.055
2.5	No	554	366	543	1	0.977	0.002	551	2	0.991	0.004
	Mid	483	434	473	7	0.977	0.015	476	11	0.983	0.023
	High	469	448	467	20	0.994	0.041	469	22	0.998	0.045

The results on experimental data are shown in Figure 15. Only visual evaluation of the experimental data results is possible, which is why there is no way to define if the selected particle is actually a "good" one. However, at least clusters of particles, contamination, and touching particles are almost all removed from the results. Only two clearly false positives were found. One of those was caused by a misaligned box, and the other was in a particle touching another one. Overall, it can be said that the framework and selected features are functional with low SNR experimental data of SV40 particles.

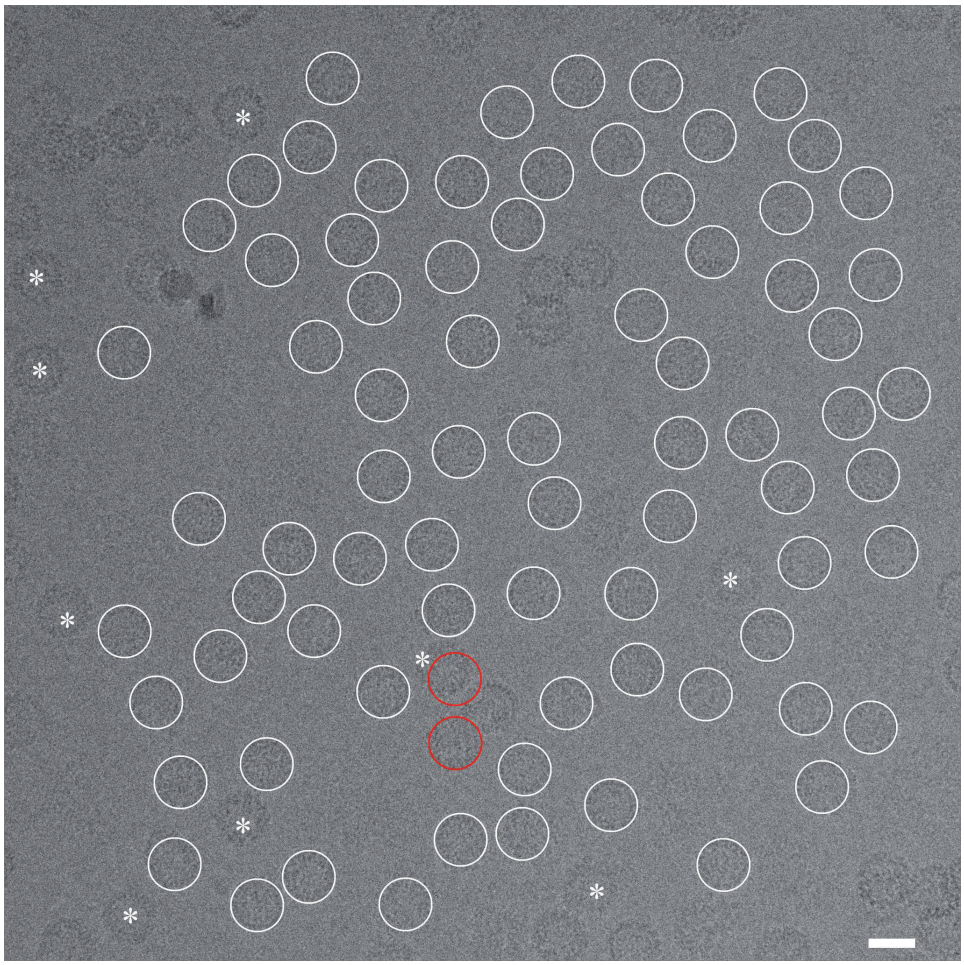


FIGURE 15 Particle selection results on experimental data. Only two clear false positives (red), of which one is a badly aligned box, were found. Altogether nine false negatives (marked with *) were found. It seemed that 76 of the selected particles were true positives. However, as there is no ground-truth available, this number is only an expert's opinion. Scale bar 40 nm.

5 BIOIMAGE INFORMATICS SOFTWARE

From the bioinformatics field of science, a subfield called *bioimage informatics* has emerged during last 15 years (Peng, 2008; Peng et al., 2012). From the bioimaging pipeline (Figure 16), bioimage informatics covers areas related to data management and analysis as well as parts of data acquisition when automated microscopes are used in high-throughput applications (Pepperkok and Ellenberg, 2006). Broad needs in bioimage informatics collect together specialists from biology, image analysis, and software engineering.

There has been a constantly growing need for bioimage processing and analysis methods and software since bioimaging moved from film to digital imaging devices and turned from qualitative science to quantitative. Modern imaging devices are capable of producing vast amounts of complex multidimensional image data (Swedlow et al., 2009), which is utilized in bioimaging by increasing sample sizes in experiments to gain more accurate results. Until recently, methods and software for analyzing the data have been the limiting factor in bioimage informatics (Wilt et al., 2009). Traditionally, the methods used have been described in biological publications (Cardona and Tomancak, 2012) without the method implementation being released at all, or released as an unmaintained stand-alone program or a plugin to frameworks such as ImageJ (Abràmoff et al., 2004; Schneider et al., 2012) or MATLAB (Mathworks) making reuse by other biologists intractable task. Recently, the bioimage informatics community has awaken to the software problem in the field which has resulted in maintained and supported open-source platforms such as BioImageXD (Publication PI), CellProfiler (Carpenter et al., 2006; Kamentsky et al., 2011), Fiji (Schindelin et al., 2012), and Icy (de Chaumont et al., 2011, 2012).

Reproducibility is one of the key requirements in scientific research. To fulfill this requirement, methods used to analyze bioimage data should be made available to the research community. Many closed commercial products, such as Imaris (Bitplane), MATLAB, Metamorph (Molecular Devices), and Volocity (Perkin-Elmer) are widely used in the field. It is clear that the exact implementation details are not known to the user of closed software even though many commercial products do include method details at some level of specificity in

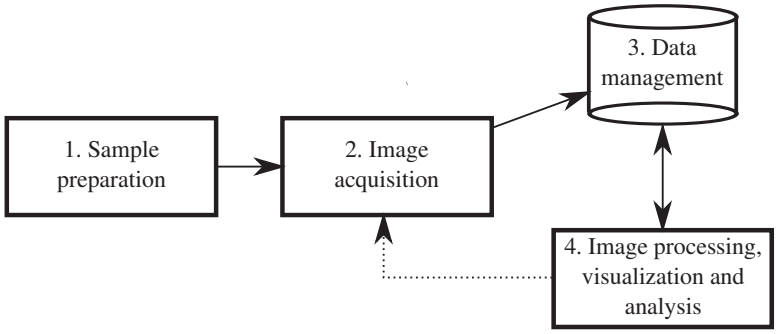


FIGURE 16 Bioimaging pipeline. First, samples are prepared by a biologist for imaging. These samples are then imaged in a software controlled microscope and saved with metadata. Postprocessing software is used to process, visualize and analyze data.

their manuals. It is possible to write a new open-source method as a script to many commercial products. This is especially common in the MATLAB user community. However, running this script would still require users to acquire an expensive license to the software (MATLAB being an exception, with a lower price range and availability in many research institutions) to be able to repeat the results. Also, as Cardona and Tomancak (2012) pointed out, open-source software is essential to enable continuous development in the growing bioimage informatics field. Lately, journals and research communities have started to require open software for reproducibility (Ince et al., 2012).

Bioimage informatics software development still holds many challenges. As the imaging data is constantly growing and becoming more complex, software need to be able to handle data as a whole, not just as a collection of images. For instance, it is important that bioimaging software processes anisotropic voxels correctly and can take multiple images in time series into account. Software architecture is the key element that holds individual processing methods together and defines how new methods can be included in the software. Most of modern bioimage informatics software include versatile plugin architecture where new tools are easily added. However, there is a risk that the continuously increasing number of plugins can turn the software into an unusable mess for basic users. To prevent this, strong management is also needed but rarely identified in scientific open-source projects.

Carpenter et al. (2012) draw attention to bioimage informatics software usability. This important matter is not on the top of the list of things to take care in scientific software for a few reasons. First, most of the work is done to create new methods to provide a solution for a specific biological problem. Additionally, few labs have funding for a full-time software engineer to focus on testing and usability of the software. For these reasons, scientific software are mostly developed by application area or methodology experts without time or special interest to focus on software engineering issues (Cardona and Tomancak, 2012). To take better advantage of scarce resources in the field, interoperability between

software packages and the usage of suitable libraries and toolkits is an increasingly important aspect. Carpenter et al. (2012) also raises interoperability as one of the challenges in the field. It is clear that even general bioimage informatics software should not reinvent the wheel, but rather work in collaboration with other projects.

5.1 BioImageXD

BioImageXD is an open-source post-processing software for bioimage informatics. It is a general-purpose platform for processing, analysis and visualization of multidimensional bioimaging data. BioImageXD is being developed by a multidisciplinary development team at the Universities of Jyväskylä and Turku in Finland, and has formerly been under development in the Max Planck Institute CBG in Dresden, Germany, with collaborators worldwide. BioImageXD version 1.0 was released for publication in the 2012 July issue of Nature Methods (Publication PI) after more than five years of development.

BioImageXD is a modular software, the core and modules of which are written in Python while C++ is used for more demanding computations. Its graphical user interface utilizes wxPython (a wrapper for wxWidgets) library. Visualization Toolkit (VTK) (Schroeder et al., 2006) and Insight Segmentation and Registration Toolkit (ITK) (Yoo et al., 2002) are used in many of its processing, analysis and visualization modules. Also, many proprietary microscopy file format readers and image processing, analysis and visualization filters developed in the project, are implemented as VTK or ITK C++ filters.

The BioImageXD features are presented in detail in Publication PI Supplementary material, as the features of early 2012. Shortly, BioImageXD can read some proprietary and open image file formats. The open OME-TIFF (Linkert et al., 2010) format is read and written to support data interchange with other bioimage informatics software. BioImageXD includes regular visualization modes for 2D slices and projections. In addition, data can be visualized with a full modular 3D visualization mode where modules, such as volume and surface rendering, can be combined in a common view to visualize the same data or different channels with multiple modules. A versatile animator can be used to create videos of 3D visualizations of the data. Image processing is always done in 3D unless specified otherwise or when processing 2D image data. BioImageXD includes many noise filtering methods, morphological operations, and registration methods to compensate sample movement during live imaging. Many segmentation and labeling methods can be used to identify different types of objects for object-based analyses. A tracking method enables analyses of movement of individual particles in live imaging. Colocalization analysis tools can be used to draw many statistical measurements from signal overlap between channels. All processing and analysis methods can be combined into processing pipelines and run in a batch mode.

5.1.1 BioImageXD design criteria

BioImageXD was designed with six design criteria: open, extensive, usable, adjustable, applicable, and extendable. Next, ideas and requirements for general bioimage informatics post-processing software behind these criteria are presented.

Openness and transparency of the software are the most important criteria from the scientific aspect, as pointed-out above. As the development was, and mostly still is, driven by the needs and ideas of the main development team and collaborators, it was clear that the software was licensed with the GNU GPL v2 license to make it available to users and other developers. Also, for the development of BioImageXD, many other open-source packages, such as Python, VTK and ITK, were used, which makes releasing the software with an open license self-evident. Unlike proprietary and black-box commercial software, open software is transparent. However, as most of the users are not familiar with reading source code, it was important from the scientific aspect to make methods to carry out only a single functionality. Thus there are no complex compound methods which are not understandable to users without reading the source code. However, there is still work to do to improve the software documentation.

Extensive set of features is needed for bioimage informatics software to be considered as general-purpose. As many other software in the field (Eliceiri et al., 2012), also BioImageXD was initially developed for special purposes, mainly visualization, animation, and colocalization analysis. During many years of development and different projects, the feature list of BioImageXD has grown to be extensive and general for many bioimage informatics needs. Turning images into quantitative measurements is the most important task in bioimage informatics. To support this for a wide range of imaging modalities, such as fluorescence, phase contrast, and electron microscopy, a large set of processing and analysis methods are needed. In addition, as the field is all about the images, it is imperative to have extensive visualization tools. We felt that these tools should be coupled with the analysis tools. Furthermore, 3D visualization tools can be used to create animations of datasets through user-defined flypaths. Eliceiri et al. (2012) regarded BioImageXD visualization and animation tools as the best in the field.

Usability, recently raised by Carpenter et al. (2012), was also one design criteria for BioImageXD. It was clear from the beginning that BioImageXD should run on all three major operating systems (Linux (many variants), Microsoft Windows, and Mac OS X (Apple)) as opposed to many commercial software which are only available to Windows and maybe to Mac OS X (Publication PI, Supplementary material). We also felt strongly for distributing BioImageXD in a single package without the need to search for various websites for plugins. It is clear that this requirement is easier to fulfill in a smaller and younger project than for instance in ImageJ, which has been developed for 25 years (Schneider et al., 2012) and has huge user community. BioImageXD is run on single GUI (also command line processing is possible) which is designed for scientific applications. One example of this is that manipulation or processing is not saved until the user decides

to do so, to prevent mistakes and unmeant manipulations to the original imaging data. Carpenter et al. (2012) also lists support for reproducible research as one usability criteria. In BioImageXD, this is handled by saving all processing information to saved datasets, which can be used later to trace methods and parameters used to analyze data. Finally, the bioimage informatics software should be usable with datasets much larger than the available system memory. BioImageXD was designed to load into memory only what is needed. Often this means a single 3D stack from one time point and channel, but some features are usable even with datasets that have 3D stack larger than the available system memory.

Adjustability, along with openness, is one of the key questions to enable accurate scientific research. As mentioned above, BioImageXD was designed not to include any unscientific black-box processing. To make adjustability possible, the parameters of the methods are tunable by users. It is clear that too many options can confuse biologists and new users (Carpenter et al., 2012) who are not familiar with image processing and analysis nor understand how different parameters affect processing. Unfortunately, research in image analysis and computer vision is not yet in a state where everything could be automated. Human expertise is still needed to solve general image analysis problems. However, default values and recommendations are given to make the learning curve easier. Also, to increase usability, parameters requiring more advanced knowledge are clearly marked. Combining methods into an image processing pipeline is needed to solve most image analysis problems (e.g segmentation pipeline in Figure 4). Construction of pipelines is possible in BioImageXD without any macro-writing or programming skills. Pipelines, along with the parameters used, can be saved for later use.

Applicability for a wide range of imaging modalities as well as image data from single-channel 2D images to 3D multi-channel timeseries is important for a general bioimage informatics platform. In addition, as high-throughput applications (Peng, 2008; Zhou and Wong, 2006) are becoming more popular, and are a must in systems biology (Kitano, 2002; Pepperkok and Ellenberg, 2006), it is imperative that the analysis methods are validated and that bioimage informatics software are applicable to high-throughput analyses. BioImageXD is not yet applicable for studies where the automatic microscope is controlled by the analysis software. However, BioImageXD includes a versatile batch processor usable for analyses of thousands of images without user interaction. Additionally, validation tools in BioImageXD have been developed and used to validate segmentation, tracking, and internalization analyses in Publications PI–PIII.

Extendability is a must in modern software architectures, and especially important in bioimage informatics, where new methods are constantly published. BioImageXD, as most other software in the field, is modular and utilizes plugin architecture (raised also as criteria by Carpenter et al. (2012)). Modularity enables rapid development and testing of new methods within an extensive platform offering data input and visualization. Through the development of BioImageXD, the project has leaned to other open-source toolkits, mainly to VTK for visualization and ITK for processing and analysis. The developer and user communities for these toolkits are huge, which guarantees continuous development. BioIm-

ageXD can be extended with filters in these libraries with a simple module to handle communication between the software and the filter. To give back to these communities, all C++ processing code is written as filters to VTK or ITK and is thus directly usable by others.

5.1.2 BioImageXD in research articles

BioImageXD has been cited in numerous research articles. Approximately 75 articles were found in Google Scholar search that either cite or mention of using BioImageXD. BioImageXD has been cited in research articles from various fields. It can be seen that the software truly is a general-purpose bioimage informatics platform as indicated in Publication PI. BioImageXD has been recently surveyed for light microscopy applications in systems biology (Antony et al., 2013), profiling cellular phenotypes (Laksameethanasan et al., 2013), bioimage informatics software (Eliceiri et al., 2012) for systems pharmacology (Li et al., 2013), and for open-source bioimage informatics for cellular biology (Swedlow and Eliceiri, 2009) as well as listed in many other review articles.

From the information extracted from individual research articles (not including articles where the author of this work was involved), it can be concluded that pixelwise colocalization tools are the most used analysis tools in BioImageXD, followed by 3D visualization and object-based quantification. Colocalization tools have been used by numerous groups in many fields, including drug delivery by nanoparticles (Bergman et al., 2013), neurology (Harrison et al., 2013), experimental eye research (Mueller et al., 2013), as well as in many studies in virology (Khan et al., 2011; Quattrocchi et al., 2012; Turkki et al., 2013), and cellular and molecular biology (Karjalainen et al., 2008; Laakkonen et al., 2009; Lemnitzer et al., 2013). Object-based measurements are other common analyses done in BioImageXD. Statistical analyses of object features such as average intensity and size are preceded by segmentation and identification of individual objects in the processing pipeline. Different processing pipelines have been used to measure nanoparticles (Bergman et al., 2013), cell membrane proteins and/or subcellular vesicles (Karjalainen et al., 2008; Siljamäki et al., 2013; Turkki et al., 2013), nuclei (Turkki et al., 2013), and to compute the amount of infected cells (Siljamäki et al., 2013). Most of the analyses were done from 3D fluorescence data.

3D visualization tools have been used for a wider range of imaging domains than analysis tools. In addition to visualization of fluorescence microscopy data of cellular structures (Harrison et al., 2013), the software has been used to create animations (Connors et al., 2007) and visualize blood vessels (Korpisalo et al., 2008) from fluorescence data. BioImageXD has also been used for visualization of untraditional imaging data such as scanning transmission ion microscopy (STIM) data (Whitlow et al., 2007), coherent anti-Stokes Raman scattering (CARS) microscopy data (Downes et al., 2009), single plane illumination microscopy data (SPIM) (Ejsmont et al., 2009), stimulated emission depletion (STED) microscopy data (Hämälistö et al., 2013), and 3D holographic light patterns (Yang et al., 2011). To conclude, BioImageXD is applicable to a wide range of image analysis and vi-

sualization task in many imaging domains in bioimage informatics.

5.2 Overview of other bioimage informatics software, tools and libraries

Many applications, tools, and software have been developed and published during years of quantitative bioimaging. Development of many of the applications has stopped and many of the applications are made only for a single special purpose. In this section, software, tools and libraries which are actively developed, and are at least partly in the same field as BioImageXD or offer methods of interest to BioImageXD, are presented shortly. Reader is advised to consult Eliceiri et al. (2012) for a more complete review of optical imaging software in bioimage informatics.

ImageJ (Abràmoff et al., 2004) and its predecessor NIH Image have been under development for 25 years (Schneider et al., 2012) as general-purpose image processing software. Due to its openness, long development history, and early adaptation of plugins, it has gain popularity in many fields of scientific imaging, including bioimaging. ImageJ core processes and visualizes 2D images in a low-throughput fashion. Over the years, numerous plugins have been developed to ImageJ by bioimaging community to add new processing tools as well as to offer 3D visualization. However, most of these plugins have been distributed by individual research groups in their own websites. Fiji (Schindelin et al., 2012) was developed to modernize the ImageJ architecture and to collect plugins of interest to bioimaging community in a single package, with the ImageJ functionality as its core. Fiji's improved architecture with native support for multidimensional image data has made many developers to move from ImageJ to Fiji.

Icy (de Chaumont et al., 2012) is a general-purpose bioimage informatics software that offers native 3D visualization with VTK and a graphical tool for creating complex processing pipelines. The Icy project takes community driven development into a new level by offering anyone a chance to develop plugins and to distribute these to others in a repository hosted by the project. In addition, Icy connects directly to ImageJ and MATLAB. Vaa3D (Peng et al., 2010) is another native 3D software taking advantage of VTK for visualization and ITK for image analysis. However, Vaa3D is designed for visualization-based analysis and is mainly for larger-scale biological applications such as embryo anatomy atlases. CellProfiler (Carpenter et al., 2006; Kamentsky et al., 2011) offers tools for completely different types of applications. It is designed for high-throughput 2D applications of cell segmentation and phenotyping by classification.

As mentioned, many bioimage informatics software use VTK (Schroeder et al., 2006) for visualization (BioImageXD, Icy, Vaa3D) and ITK (Yoo et al., 2002) for analysis (BioImageXD, Vaa3D). Both of the toolkits are written in C++, but wrapping to Python and Java is supported. Most microscopy imaging data is in microscope manufacturers' proprietary formats. The Bio-Formats (Linkert et al., 2010)

Java library was developed for the community to read image formats (currently approximately hundred), including all popular microscopy formats. The library is used directly in Java-based software: ImageJ, Fiji, Icy, and CellProfiler. Also a wrapper to include Bio-Formats to ITK pipeline has been developed but not yet utilized in BioImageXD. Microscopes are also directly controllable through the API of open source software μManager (Edelstein et al., 2010). μManager gives the same advantages for controlling microscopes as Bio-Formats for reading proprietary file formats.

The Open Microscopy Environment (http://www.openmicroscopy.org, OME), the present developers of the Bio-Formats library, support open microscopy data interchange. OME-TIFF (Linkert et al., 2010) is a widely-adopted open file format for microscopy. Recently, many microscope manufacturers have also added support for OME-TIFF into their software, making file interchange between software easier. OME also develops the OMERO server used to manage microscopy data (Figure 16, 3. Data management).

Many technically oriented researchers and developers in bioimage informatics have become committed to academic open source software. However, commercial software are still widely used in the field. Imaris (Bitplane) was found to be the most applicable in cellular image analysis application in Publication PI (Supplementary material), with Volocity (Perkin-Elmer) having extensive 3D visualization and animation tools. Columbus (Perkin-Elmer) is a commercial product for data management and 2D image analysis based on the open OMERO server. Other products often cited in research articles are Metamorph for analysis (Molecular Devices), Amira for visualization (FEI), and Huygens for deconvolution (Scientific Volume Imaging).

6 CONCLUSIONS AND FUTURE WORK

This work presented algorithms and software for biological image analysis at multiple scales from macromolecular structures imaged with electron microscopy to whole cells imaged with optical microscopy. The main conclusions are summarized followed by suggestions for future work.

6.1 Main conclusions

The main results and conclusions of the work are:

- An open-source bioimage informatics software, BioImageXD, was developed (Publication PI, Section 5.1). The software enables image analysis, such as internalization analyses, previously unavailable in open-source bioimaging software. BioImageXD also offers progressive tools for 3D visualization and animation to make the structure of the imaged data more understandable. All processing is done in multiple dimensions, and no background processing, such as resampling data for faster visualization are present. These are important aspects in scientific image analysis. Six general criteria for bioimage informatics software were defined. Finally, research articles citing BioImageXD were checked to find the most used methods in the software. It seems that colocalization as well as object-based and internalization analyses with 3D visualization were the most used features. The aim to develop general-purpose bioimage informatics software was fulfilled. The defined criteria guaranteed that the requirements for the software were also fulfilled.

- A feature-based tracking method was developed (Publication PII, Subsection 3.3.3) and used to track $\alpha 2\beta 1$ integrin clusters. The method was shown to give a good linking accuracy ($0.947 - 0.999$) and moderate perfect tracks accuracy ($0.260 - 0.97$) with different simulations. The method was also able to detect the clustering of particles. However, the method is dependent

on good pre-segmentation of the image data. The objective of the particle tracking method was fulfilled, as proven by the comparison to a state-of-the-art SPT method with the same input segmentation and a similar parameter configuration. The developed method gave more accurate results than the compared method in the tracking of clustering $\alpha 2\beta 1$ integrin cluster data. The method is highly configurable, and as such, fulfills the aim of general-purpose particle tracking method. The particle tracking method is available in the BioImageXD software.

- The third objective was to study new methods and existing colocalization methods for analyzing particle association. A novel idea for particle association analysis and a novel PPM association method was presented (Publication PIII, Subsection 3.2.4) and compared to existing pixelwise colocalization methods as well as to the ICP matching method. The objective was fulfilled, and the PPM association method was proven more accurate than the methods compared. For association analyses in various simulation conditions, the global PPM association method was shown to be more accurate than the local ICP method. The PPM association method was shown to produce results similar to those of pixelwise colocalization methods in fixed imaging experiment with large amount of colocalization. Our method was able to detect more association than traditional colocalization methods in live imaging experiment with closely located particles without direct overlap. The PPM association method is currently available as MATLAB implementation. However, the plan is to include the method in BioImageXD in near future.

- Effects of the missing wedge in limited angle ET were studied in Publication PIV. The missing wedge results in anisotropic resolution. A new MAP-EM reconstruction method for biological ET was studied to solve problems due to the missing wedge. The method was shown to decrease elongation of colloidal gold particles, as compared to common WBP and SIRT methods in simulated and experimental studies. Also, the contrast ratio was increased in both, experimental and simulated data, reconstructions. The developed method reduces the artifacts caused by the missing wedge, and as such, the method fulfilled the objective of the study.

- A novel particle selection framework for SPR was developed (Section 4.3) to maximize the quality of selected particle images for reconstruction. The framework was tested with simulated and experimental micrographs including Simian virus 40 particles. Run-length and Haralick's texture features were tested in the framework. The particle selection method gave a high recall (over 0.93) and low false discovery rate (less than 0.055) even with the most complex simulation scenario tested. The research on particle selection framework mostly fulfilled the objective of minimizing false particle detection. However, the method has not been tested and configured for detecting small asymmetric particles. The particle selection method is

available in the BioImageXD software.

6.2 Plans for future work

During the research presented in this work, several directions for new research have emerged. In the BioImageXD project, the future plans concern the usability of high-throughput applications and the interoperability with other bioimage informatics software. Usability is an important aspect in all software development, but especially in scientific software: finding a balance between simple usage and applicability to complex applications is hard. For the next generation of BioImageXD, different features are planned with the purpose of making them work together more seamlessly without unnecessary intermediate processing. Interoperability will be improved by adopting the Bio-Formats library (Linkert et al., 2010) for microscopy image data import and export. A general API for using BioImageXD from other applications is planned to be designed in the next generation BioImageXD. Also direct access to other bioimage informatics software will be researched.

Criteria for automatic parameter selection for the tracking method is an interesting topic. This would relieve the user from sometimes intractable search of suitable parameters. In addition, the plan is to include more features to be used in search for particle correspondence, usable when tracking larger objects such as nuclei or cells. Manually adjustable parameters for each feature is not a sound solution. For this, automatic parameter selection would be needed. Idea of dummy particles as a solution to temporal disappearance is another subject worth of research.

SPR is another field of research with open questions. It would be interesting to test many other features, such as wavelet features (Aydogan et al., 2009), in the particle selection framework, and to develop the presented particle selection method for the selection of small particles with their molecular weight of only a few hundred kilodalton. CryoEM images of these particles are almost invisible for naked eye, so there are few means for manual inspection to check whether the selected particles are ”good”. A method for truly automatic selection and classification of good particles would have a huge impact on the SPR community. Another interesting direction would be to test the reconstruction method, presented for ET to SPR, with particles that favor some orientations over others.

YHTEENVETO (FINNISH SUMMARY)

Kuvantaminen on merkittävä osa nykypäivän solubiologian tutkimusta. Solubiologiassa tutkittavat kohteet ovat nanometreistä kymmeniin mikrometreihin. Tämän kokoisten objektien kuvantaminen vaatii mikroskopiaa. Solubiologiassa turvaudutaan usein fluoresenssimikroskopian menetelmiin, jotka mahdollistavat tiettyjen rakenteiden merkkaamisen muuten käytännössä läpinäkyvistä soluista. Valomikroskopialla ei kuitenkaan yleisesti pystytä erottelemaan rakenteita 200 nanometriä tarkemmin. Elektronimikroskopiaa käytetään, kun tarvitaan parempaa erottelukykyä muun muassa tutkittaessa kuinka vain kymmenien nanometrien kokoiset virukset menevät soluun sisälle.

Tämä väitöskirja "Moniskaalaisen biologisen kuva-analyysin algoritmeja ja ohjelmisto" käsittelee mikroskopian kuva-analyysiongelmia sekä analyysiohjelmistoja useassa kokoluokassa. Työllä on viisi päämäärää. Ensimmäisenä on yleiskäyttöisen ohjelmiston kehittäminen moniulotteisen mikroskopiadatan visualisointiin ja analysointiin. Kuvantamismenetelmien kehittymisen myötä nykyiset mikroskoopit tuottavat valtavat määrät moniulotteista kuvadataa, jonka kvalitatiivinen analysointi ei ole mahdollista, eikä tieteelliseltä kannalta edes suotavaa. Tästä syystä tarve tieteelliset vaatimukset täyttäville analyysiohjelmistoille on ollut valtava. Tutkimustyön aikana on kehitetty avoimen lähdekoodin ohjelmisto, BioImageXD, joka vastaa osaltaa biokuvantamisen haasteisiin. Ohjelmiston vertailu muihin tarjolla oleviin sekä kaupallisiin että avoimiin ohjelmistoihin osoittavat BioImageXD:n tarjoavan monipuolisia visualisointi- ja analysointiominaisuuksia mitä ei ole saatavilla muissa vastaavissa ohjelmistoissa. Osoituksena BioImageXD:n monipuolisuudesta, ohjelmistoa käytettiin vaativan solubiologian kuva-analyysiongelman ratkaisemiseen. Tutkimus on esitetty väitöskirjan ensimmäisessä osajulkaisussa.

Toinen ja kolmas tavoite liittyvät fluoresenssimikroskopiadatan analysointiin. Toisena päämääränä on kehittää seurantamenetelmä ratkaisemaan klusteroituvien partikkeleiden seurantaongelmaa kolmiulotteisessa aikasarjassa. Partikkeleiden seuranta elävissä soluissa mahdollistaa solun toiminnan yksityiskohtaisen tutkimisen. Työssä kehitettyä menetelmää sovelletaan $\alpha 2\beta 1$-integriiniklusterien seurantaan. Tulokset sekä simuloidulla että kokeellisella datalla ovat tarkkoja. Menetelmä ja tulokset esitetään tämän työn toisessa osajulkaisussa.

Kolmantena päämääränä on tutkia menetelmiä eri partikkeleiden yhteyksien analysoimiseen, jolla on lukuisia käyttökohteita solun sisäisten vuorovaikutusten ymmärtämisessä. Yleisesti käytetyt kolokalisaatiomenetelmät eivät kuitenkaan kykene tunnistamaan lähekkäin sijaitsevia objekteja elleivät ne sijaitse alle valomikroskoopin resoluution etäisyydellä toisistaan. Tätä vaikutusta tutkittiin myös tässä työssä. Partikkeleiden yhteyden analysoimiseen kehitettiin menetelmä täysin uudenlaisen idean pohjalta. Menetelmä todettiin tarkaksi ja vakaaksi useilla simuloiduilla ja kokeellisilla datoilla. Menetelmät ja tulokset on esitetty tämän työn osajulkaisussa kolme.

Neljäs ja viides päämäärä liittyvät elektronimikroskopiaan, joka tarjoaa mo-

nisatakertaisen erottelukyvyn valomikroskopiaan verrattuna, mutta sisältää myös merkittäviä ongelmia. Osatyössä neljä tutkitaan elektronitomografian puuttuvista kuvakulmista aiheutuvia ongelmia. Perinteisesti käytetyillä menetelmillä luoduissa rekonstruoinneissa on selkeästi nähtävissä heikentynyt resoluutio ja erinäiset artefaktat. Ongelman ratkaisemiseen sovellettiin Tampereen teknillisessä yliopistossa toimivan yhteistyöryhmän lääketieteen tomografiasovelluksiin kehitettyä menetelmää, joka vähentää puuttuvista kuvakulmista aiheutuvia ongelmia merkittävästi. Alustavat tulokset on esitetty osatyössä neljä.

Viidentenä tutkimuskohteena on pienten, vain nanometrien tai kymmenien nanometrien kokoisten partikkelien rakenteiden tutkiminen nanometrien osien tarkkuudella elektronimikroskopialla. Tällaisia partikkeleita ovat esimerkiksi virukset ja niiden rakenteet, joiden toimintaa voidaan ymmärtää paremmin niiden rakenteet tuntemalla. Niin sanotussa yksittäisen partikkelin rekonstruoinnissa luodaan kolmiulotteinen malli tuhansista kaksiulotteisista kuvista. Partikkeleiden tunnistaminen näistä kaksiulotteisista kuvista on haasteellisuuden takia yksi menetelmän merkittävimmistä rajoitteista. Tässä työssä esitellään menetelmä, jolla partikkeleita voidaan tunnistaa automaattisesti sekä karsia niin sanotusti ”huonoja” partikkeleita pois, joka yleensä vaatii päivien tai jopa viikkojen manuaalista työtä.

REFERENCES

Abramowitz, M., Spring, K. R., Keller, H. E. & Davidson, M. W. 2002. Basic principles of microscope objectives. Biotechniques 33 (4), 772–781.

Abràmoff, M. D., Magalhães, P. J. & Ram, S. J. 2004. Image processing with ImageJ. Biophotonics international 11 (7), 36–42.

Adiga, U., Baxter, W. T., Hall, R. J., Rockel, B., Rath, B. K., Frank, J. & Glaeser, R. 2005. Particle picking by segmentation: a comparative study with SPIDER-based manual particle picking. Journal of Structural Biology 152 (3), 211. doi:10.1016/j.jsb.2005.09.007.

Adler, J., Pagakis, S. N. & Parmryd, I. 2008. Replicate-based noise corrected correlation for accurate measurements of colocalization. Journal of Microscopy 230 (1), 121–133. doi:10.1111/j.1365-2818.2008.01967.x.

Afzelius, B. A. & Maunsbach, A. B. 2004. Biological ultrastructure research; the first 50 years. Tissue and Cell 36 (2), 83 - 94. doi:10.1016/j.tice.2003.11.001.

Alcor, D., Gouzer, G. & Triller, A. 2009. Single-particle tracking methods for the study of membrane receptors dynamics. European Journal of Neuroscience 30 (6), 987–997. doi:10.1111/j.1460-9568.2009.06927.x.

Andersen, A. H. & Kak, A. C. 1984. Simultaneous algebraic reconstruction technique (SART): a superior implementation of the ART algorithm. Ultrasonic imaging 6 (1), 81–94. doi:10.1016/0161-7346(84)90008-7.

Antony, P. M. A., Trefois, C., Stojanovic, A., Baumuratov, A. S. & Kozak, K. 2013. Light microscopy applications in systems biology: opportunities and challenges. Cell Communication and Signaling 11 (1), 24. doi:10.1186/1478-811X-11-24.

Arbeláez, P., Han, B. G., Typke, D., Lim, J., Glaeser, R. M. & Malik, J. 2011. Experimental evaluation of support vector machine-based and correlation-based approaches to automatic particle selection. Journal of Structural Biology 175 (3), 319–328. doi:10.1016/j.jsb.2011.05.017.

Aurenhammer, F. 1991. Voronoi diagrams–a survey of a fundamental geometric data structure. ACM Computing Surveys 23 (3), 345–405. doi:10.1145/116873.116880.

Aydogan, D. B., Hannula, M., Arola, T., Dastidar, P. & Hyttinen, J. 2009. 2D texture based classification, segmentation and 3D orientation estimation of tissues using DT-CWT feature extraction methods. Data & Knowledge Engineering 68 (12), 1383–1397. doi:10.1016/j.datak.2009.07.009.

Bacallao, R., Sohrab, S. & Phillips, C. 2006. Guiding principles of specimen preservation for confocal fluorescence microscopy. In J. B. Pawley (Ed.) Handbook of biological confocal microscopy. Springer US, 368–380. doi:10.1007/978-0-387-45524-2_18.

Baker, T. S. & Cheng, R. H. 1996. A model-based approach for determining orientations of biological macromolecules imaged by cryoelectron microscopy. Journal of Structural Biology 116 (1), 120–130. doi:10.1006/jsbi.1996.0020.

Bergman, L., Kankaanpää, P., Tiitta, S., Duchanoy, A., Li, L., Heino, J. & Lindén, M. 2013. Intracellular degradation of multilabeled poly (ethylene imine)–mesoporous silica–silica nanoparticles: Implications for drug release. Molecular Pharmaceutics 10 (5), 1795–1803. doi:10.1021/mp3005879.

Besl, P. J. & McKay, N. D. 1992. A method for registration of 3-D shapes. IEEE Transactions on Pattern Analysis and Machine Intelligence 14 (2), 239–256. doi:10.1109/34.121791.

Björkbom, A., Róg, T., Kankaanpää, P., Lindroos, D., Kaszuba, K., Kurita, M., Yamaguchi, S., Yamamoto, T., Jaikishan, S., Paavolainen, L., Päivärinne, J., Nyholm, T. K. M., Katsumura, S., Vattulainen, I. & Slotte, J. P. 2011. N- and o-methylation of sphingomyelin markedly affects its membrane properties and interactions with cholesterol. Biochimica et Biophysica Acta (BBA) - Biomembranes 1808 (4), 1179–1186. doi:10.1016/j.bbamem.2011.01.009.

Bolte, S. & Cordelières, F. P. 2006. A guided tour into subcellular colocalization analysis in light microscopy. Journal of Microscopy 224 (3), 213–232. doi:10.1111/j.1365-2818.2006.01706.x.

Borland, L. & van Heel, M. 1990. Classification of image data in conjugate representation spaces. Journal of the Optical Society of America A 7 (4), 601–610. doi:10.1364/JOSAA.7.000601.

Brandt, S., Heikkonen, J. & Engelhardt, P. 2001. Multiphase method for automatic alignment of transmission electron microscope images using markers. Journal of Structural Biology 133 (1), 10–22. doi:10.1006/jsbi.2001.4343.

Bright, D. S. & Steel, E. B. 1987. Two-dimensional top hat filter for extracting spots and spheres from digital images. Journal of Microscopy 146 (2), 191–200. doi:10.1111/j.1365-2818.1987.tb01340.x.

Buchwalow, I. B. & Böcker, W. 2010. Immunohistochemistry: Basics and Methods (1st edition). Springer. doi:10.1007/978-3-642-04609-4.

Canny, J. 1986. A computational approach to edge detection. IEEE Transactions on Pattern Analysis and Machine Intelligence 8 (6), 679–698. doi:10.1109/TPAMI.1986.4767851.

Cardona, A. & Tomancak, P. 2012. Current challenges in open-source bioimage informatics. Nature Methods 9 (7), 661–665. doi:10.1038/nmeth.2082.

Carpenter, A. E., Jones, T. R., Lamprecht, M. R., Clarke, C., Kang, I. H., Friman, O., Guertin, D. A., Chang, J. H., Lindquist, R. A., Moffat, J. et al. 2006. CellProfiler: image analysis software for identifying and quantifying cell phenotypes. Genome Biology 7 (10), R100. doi:10.1186/gb-2006-7-10-r100.

Carpenter, A. E., Kamentsky, L. & Eliceiri, K. W. 2012. A call for bioimaging software usability. Nature Methods 9 (7), 666–670. doi:10.1038/nmeth.2073.

Caselles, V., Kimmel, R. & Sapiro, G. 1997. Geodesic active contours. International Journal of Computer Vision 22 (1), 61–79. doi:10.1023/A:1007979827043.

Chalfie, M., Tu, Y., Euskirchen, G., Ward, W. W. & Prasher, D. C. 1994. Green fluorescent protein as a marker for gene expression. Science 263 (5148), 802–805. doi:10.1126/science.8303295.

Chan, T. F. & Vese, L. A. 2001. Active contours without edges. IEEE Transactions on Image Processing 10 (2), 266–277. doi:10.1109/83.902291.

de Chaumont, F., Dallongeville, S. & Olivo-Marin, J.-C. 2011. ICY: A new open-source community image processing software. In IEEE International Symposium on Biomedical Imaging: From Nano to Macro, 234–237. doi:10.1109/ISBI.2011.5872395.

de Chaumont, F., Dallongeville, S., Chenouard, N., Hervé, N., Pop, S., Provoost, T., Meas-Yedid, V., Pankajakshan, P., Lecomte, T., Le Montagner, Y. et al. 2012. Icy: an open bioimage informatics platform for extended reproducible research. Nature Methods 9 (7), 690–696. doi:10.1038/nmeth.2075.

Cheezum, M. K., Walker, W. F. & Guilford, W. H. 2001. Quantitative comparison of algorithms for tracking single fluorescent particles. Biophysical Journal 81 (4), 2378–2388. doi:10.1016/S0006-3495(01)75884-5.

Chen, J. Z. & Grigorieff, N. 2007. SIGNATURE: a single-particle selection system for molecular electron microscopy. Journal of Structural Biology 157 (1), 168–173. doi:10.1016/j.jsb.2006.06.001.

Chen, Y. & Medioni, G. 1991. Object modeling by registration of multiple range images. In Proceedings of the 1991 IEEE International Conference on Robotics and Automation, 2724–2729. doi:10.1109/ROBOT.1991.132043.

Chenouard, N., Bloch, I. & Olivo-Marin, J. 2013. Multiple hypothesis tracking for cluttered biological image sequences. IEEE Transactions on Pattern Analysis and Machine Intelligence 35 (11), 2736-3750. doi:10.1109/TPAMI.2013.97.

Chessel, A., Dodgson, J. & Carazo-Salas, R. E. 2012. Spherical spatial statistics for 3D fluorescence video-microscopy. In Biomedical Imaging (ISBI), 2012 9th IEEE International Symposium on, 1747-1750. doi:10.1109/ISBI.2012.6235918.

Chu, A., Sehgal, C. & Greenleaf, J. F. 1990. Use of gray value distribution of run lengths for texture analysis. Pattern Recognition Letters 11 (6), 415–419. doi: 10.1016/0167-8655(90)90112-F.

Comaniciu, D. & Meer, P. 2002. Mean shift: a robust approach toward feature space analysis. IEEE Transactions on Pattern Analysis and Machine Intelligence 24 (5), 603-619. doi:10.1109/34.1000236.

Comeau, J. W. D., Costantino, S. & Wiseman, P. W. 2006. A guide to accurate fluorescence microscopy colocalization measurements. Biophysical Journal 91 (12), 4611–4622. doi:10.1529/biophysj.106.089441.

Conchello, J.-A. & Lichtman, J. W. 2005. Optical sectioning microscopy. Nature Methods 2 (12), 920–931. doi:10.1038/nmeth815.

Connors, W. L., Jokinen, J., White, D. J., Puranen, J. S., Kankaanpää, P., Upla, P., Tulla, M., Johnson, M. S. & Heino, J. 2007. Two synergistic activation mechanisms of $\alpha 2\beta 1$ integrin-mediated collagen binding. Journal of Biological Chemistry 282 (19), 14675–14683. doi:10.1074/jbc.M700759200.

Costes, S. V., Daelemans, D., Cho, E. H., Dobbin, Z., Pavlakis, G. & Lockett, S. 2004. Automatic and quantitative measurement of protein-protein colocalization in live cells. Biophysical Journal 86 (6), 3993–4003. doi:10.1529/biophysj.103.038422.

Crowther, R. A., DeRosier, D. J. & Klug, A. 1970. The reconstruction of a three-dimensional structure from projections and its application to electron microscopy. Proceedings of the Royal Society of London. A. Mathematical and Physical Sciences 317 (1530), 319–340. doi:10.1098/rspa.1970.0119.

Culverhouse, P. F., Williams, R., Reguera, B., Herry, V. & González-Gil, S. 2003. Do experts make mistakes? a comparison of human and machine identification of dinoflagellates. Marine Ecology Progress Series 247, 17–25. doi:10.3354/meps247017.

Dasarathy, B. V. & Holder, E. B. 1991. Image characterizations based on joint gray level-run length distributions. Pattern Recognition Letters 12 (8), 497–502. doi: 10.1016/0167-8655(91)80014-2.

Dempster, A. P., Laird, N. M. & Rubin, D. B. 1977. Maximum likelihood from incomplete data via the EM algorithm. Journal of the Royal Statistical Society. Series B (Methodological) 39 (1), 1–38.

De Rosier, D. J. & Klug, A. 1968. Reconstruction of three dimensional structures from electron micrographs. Nature 217 (5124), 130–134. doi:10.1038/217130a0.

Downes, A., Mouras, R. & Elfick, A. 2009. A versatile CARS microscope for biological imaging. Journal of Raman Spectroscopy 40 (7), 757–762. doi:10.1002/jrs.2249.

Duda, R. O. & Hart, P. E. 1972. Use of the Hough transformation to detect lines and curves in pictures. Communications of the ACM 15 (1), 11–15.

Dufour, A., Shinin, V., Tajbakhsh, S., Guillén-Aghion, N., Olivo-Marin, J.-C. & Zimmer, C. 2005. Segmenting and tracking fluorescent cells in dynamic 3-d microscopy with coupled active surfaces. IEEE Transactions on Image Processing 14 (9), 1396–1410. doi:10.1109/TIP.2005.852790.

Dzyubachyk, O., van Cappellen, W. A., Essers, J., Niessen, W. J. & Meijering, E. 2010. Advanced level-set-based cell tracking in time-lapse fluorescence microscopy. IEEE Transactions on Medical Imaging 29 (3), 852–867. doi:10.1109/TMI.2009.2038693.

Eagleman, D. M. 2001. Visual illusions and neurobiology. Nature Reviews Neuroscience 2 (12), 920–926. doi:10.1038/35104092.

Edelstein, A., Amodaj, N., Hoover, K., Vale, R. & Stuurman, N. 2010. Computer control of microscopes using μmanager. Current Protocols in Molecular Biology, 14–20. doi:10.1002/0471142727.mb1420s92.

Eils, R. & Athale, C. 2003. Computational imaging in cell biology. The Journal of Cell Biology 161 (3), 477–481. doi:10.1083/jcb.200302097.

Ejsmont, R. K., Sarov, M., Winkler, S., Lipinski, K. A. & Tomancak, P. 2009. A toolkit for high-throughput, cross-species gene engineering in drosophila. Nature Methods 6 (6), 435–437. doi:10.1038/nmeth.1334.

Eliceiri, K. W., Berthold, M. R., Goldberg, I. G., Ibáñez, L., Manjunath, B. S., Martone, M. E., Murphy, R. F., Peng, H., Plant, A. L., Roysam, B., Stuurman, N., Swedlow, J. R., Tomancak, P. & Carpenter, A. E. 2012. Biological imaging software tools. Nature Methods 9 (7), 697–710. doi:10.1038/nmeth.2084.

Ester, M., Kriegel, H.-P., Sander, J. & Xu, X. 1996. A density-based algorithm for discovering clusters in large spatial databases with noise. In Proceedings of the Second International Conference on Knowledge Discovery and Data Mining, Vol. 96, 226–231.

Fawcett, T. 2006. An introduction to ROC analysis. Pattern recognition letters 27 (8), 861–874. doi:10.1016/j.patrec.2005.10.010.

Frank, J. 2002. Single-particle imaging of macromolecules by cryo-electron microscopy. Annual Review of Biophysics and Biomolecular Structure 31 (1), 303–319. doi:10.1146/annurev.biophys.31.082901.134202.

Frank, J. 2006a. Introduction: Principles of electron tomography. In J. Frank (Ed.) Electron Tomography: Methods for Three-Dimensional Visualization of Structures in the Cell (2nd edition). New York: Springer, 1–16.

Frank, J. 2006b. Three-dimensional electron microscopy of macromolecular assemblies: visualization of biological molecules in their native state (2nd edition). New York: Oxford University Press.

Galloway, M. M. 1975. Texture analysis using gray level run lengths. Computer Graphics and Image Processing 4 (2), 172–179. doi:10.1016/S0146-664X(75)80008-6.

Genovesio, A., Liedl, T., Emiliani, V., Parak, W., Coppey-Moisan, M. & Olivo-Marin, J.-C. 2006. Multiple particle tracking in 3-D+t microscopy: method and application to the tracking of endocytosed quantum dots. IEEE Transactions on Image Processing 15 (5), 1062-1070. doi:10.1109/TIP.2006.872323.

Gibson, S. F. & Lanni, F. 1991. Experimental test of an analytical model of aberration in an oil-immersion objective lens used in three-dimensional light microscopy. Journal of the Optical Society of America A 8 (10), 1601–1613. doi:10.1364/JOSAA.8.001601.

Gilbert, P. 1972. Iterative methods for the three-dimensional reconstruction of an object from projections. Journal of Theoretical Biology 36 (1), 105–117.

Glaeser, R. M. 1999. Review: Electron crystallography: Present excitement, a nod to the past, anticipating the future. Journal of Structural Biology 128 (1), 3–14. doi:10.1006/jsbi.1999.4172.

Glaeser, R. M. 2004. Historical background: Why is it important to improve automated particle selection methods? Journal of structural biology 145 (1), 15–18. doi:10.1016/j.jsb.2003.09.005.

Godinez, W. J., Lampe, M., Koch, P., Eils, R., Muller, B. & Rohr, K. 2012. Identifying virus-cell fusion in two-channel fluorescence microscopy image sequences based on a layered probabilistic approach. IEEE Transactions on Medical Imaging 31 (9), 1786–1808.

Godinez, W. J., Lampe, M., Wörz, S., Müller, B., Eils, R. & Rohr, K. 2009. Deterministic and probabilistic approaches for tracking virus particles in time-lapse fluorescence microscopy image sequences. Medical Image Analysis 13 (2), 325–342. doi:10.1016/j.media.2008.12.004.

Gustafsson, M. G. L. 2000. Surpassing the lateral resolution limit by a factor of two using structured illumination microscopy. Journal of Microscopy 198 (2), 82–87. doi:10.1046/j.1365-2818.2000.00710.x.

Hall, R. J. & Patwardhan, A. 2004. A two step approach for semi-automated particle selection from low contrast cryo-electron micrographs. Journal of Structural Biology 145 (1), 19–28. doi:10.1016/j.jsb.2003.10.024.

Haralick, R. M., Shanmugam, K. & Dinstein, I. 1973. Textural features for image classification. IEEE Transactions on Systems, Man, and Cybernetics 3 (6), 610–621. doi:doi:10.1109/TSMC.1973.4309314.

Harauz, G. & van Heel, M. 1986. Exact filters for general geometry three dimensional reconstruction. Optik 73 (4), 146–156.

Harrison, B. J., Flight, R. M., Gomes, C., Venkat, G., Ellis, S. R., Sankar, U., Twiss, J. L., Rouchka, E. C. & Petruska, J. C. 2013. IB4-binding sensory neurons in the adult rat express a novel 3'UTR-extended isoform of CaMK4 that is associated with its localization to axons. In press in Journal of Comparative Neurology. doi:10.1002/cne.23398.

van Heel, M., Gowen, B., Matadeen, R., Orlova, E. V., Finn, R., Pape, T., Cohen, D., Stark, H., Schmidt, R., Schatz, M. & Patwardhan, A. 2000. Single-particle electron cryo-microscopy: towards atomic resolution. Quarterly Reviews of Biophysics 33 (4), 307–369.

van Heel, M., Orlova, E. V., Harauz, G., Stark, H., Dube, P., Zemlin, F. & Schatz, M. 1997. Angular reconstitution in three-dimensional electron microscopy: historical and theoretical aspects. Scanning Microscopy 11, 195–210.

van Heel, M. 1987. Angular reconstitution: a posteriori assignment of projection directions for 3D reconstruction. Ultramicroscopy 21 (2), 111–123. doi:10.1016/0304-3991(87)90078-7.

Hell, S. W. & Wichmann, J. 1994. Breaking the diffraction resolution limit by stimulated emission: stimulated-emission-depletion fluorescence microscopy. Optics Letters 19 (11), 780–782. doi:10.1364/OL.19.000780.

Hell, S. W., Stelzer, E. H. K., Lindek, S. & Cremer, C. 1994. Confocal microscopy with an increased detection aperture: type-B 4Pi confocal microscopy. Optics Letters 19 (3), 222–224. doi:10.1364/OL.19.000222.

Henderson, R. 1995. The potential and limitations of neutrons, electrons and X-rays for atomic resolution microscopy of unstained biological molecules. Quarterly Reviews of Biophysics 28, 171–171. doi:10.1017/S003358350000305X.

Huang, B., Bates, M. & Zhuang, X. 2009. Super resolution fluorescence microscopy. Annual Review of Biochemistry 78, 993–1016. doi:10.1146/annurev.biochem.77.061906.092014.

Huang, Z. & Penczek, P. A. 2004. Application of template matching technique to particle detection in electron micrographs. Journal of Structural Biology 145 (1), 29–40. doi:10.1016/j.jsb.2003.11.004.

Hämälistö, S., Pouwels, J., de Franceschi, N., Saari, M., Ivarsson, Y., Zimmermann, P., Brech, A., Stenmark, H. & Ivaska, J. 2013. A zo-1/α5β1-integrin complex regulates cytokinesis downstream of pkcϵ in nci-h460 cells plated on fibronectin. PLoS ONE 8 (8), e70696. doi:10.1371/journal.pone.0070696.

Ince, D. C., Hatton, L. & Graham-Cumming, J. 2012. The case for open computer programs. Nature 482 (7386), 485–488. doi:10.1038/nature10836.

Inoué, S. 2006. Foundations of confocal scanned imaging in light microscopy. In J. B. Pawley (Ed.) Handbook of biological confocal microscopy. Springer US, 1–19. doi:10.1007/978-0-387-45524-2_1.

Jaqaman, K., Loerke, D., Mettlen, M., Kuwata, H., Grinstein, S., Schmid, S. L. & Danuser, G. 2008. Robust single-particle tracking in live-cell time-lapse sequences. Nature Methods 5 (8), 695–702. doi:10.1038/nmeth.1237.

Johnson, I. & Spence, M. T. Z. (Eds.) 2010. Molecular Probes Handbook, A Guide to Fluorescent Probes and Labeling Technologies (11th edition). Invitrogen.

Jonić, S., Sorzano, C. O. S. & Boisset, N. 2008. Comparison of single-particle analysis and electron tomography approaches: an overview. Journal of Microscopy 232 (3), 562–579. doi:10.1111/j.1365-2818.2008.02119.x.

Kaakinen, M., Huttunen, S., Paavolainen, L., Marjomäki, V., Heikkilä, J. & Eklund, L. 2014. Automatic detection and analysis of cell motility in phase-contrast time-lapse images using a combination of maximally stable extremal regions and Kalman filter approaches. Journal of Microscopy 253 (1), 65–78. doi:10.1111/jmi.12098.

Kamarainen, J.-K., Lensu, L. & Kauppi, T. 2012. Combining multiple image segmentations by maximizing expert agreement. In Machine Learning in Medical Imaging, 193–200. doi:10.1007/978-3-642-35428-1_24.

Kamentsky, L., Jones, T. R., Fraser, A., Bray, M.-A., Logan, D. J., Madden, K. L., Ljosa, V., Rueden, C., Eliceiri, K. W. & Carpenter, A. E. 2011. Improved structure, function and compatibility for CellProfiler: modular high-throughput image analysis software. Bioinformatics 27 (8), 1179–1180. doi:10.1093/bioinformatics/btr095.

Kapur, J., Sahoo, P. K. & Wong, A. 1985. A new method for gray-level picture thresholding using the entropy of the histogram. Computer Vision, Graphics, and Image Processing 29 (3), 273–285. doi:10.1016/0734-189X(85)90125-2.

Karjalainen, M., Kakkonen, E., Upla, P., Paloranta, H., Kankaanpää, P., Liberali, P., Renkema, G. H., Hyypiä, T., Heino, J. & Marjomäki, V. 2008. A raft-derived, Pak1-regulated entry participates in $\alpha 2\beta 1$ integrin-dependent sorting to caveosomes. Molecular Biology of the Cell 19 (7), 2857–2869. doi:10.1091/mbc.E07-10-1094.

Karjalainen, M., Rintanen, N., Lehkonen, M., Kallio, K., Mäki, A., Hellström, K., Siljamäki, V., Upla, P. & Marjomäki, V. 2011. Echovirus 1 infection depends on biogenesis of novel multivesicular bodies. Cellular microbiology 13 (12), 1975–1995. doi:10.1111/j.1462-5822.2011.01685.x.

Kass, M., Witkin, A. & Terzopoulos, D. 1988. Snakes: Active contour models. International Journal of Computer Vision 1 (4), 321–331. doi:10.1007/BF00133570.

Kawase, N., Kato, M., Nishioka, H. & Jinnai, H. 2007. Transmission electron microtomography without the ”missing wedge” for quantitative structural analysis. Ultramicroscopy 107 (1), 8–15. doi:10.1016/j.ultramic.2006.04.007.

van Kempen, G. M. P., van Vliet, L. J., Verveer, P. J. & van der Voort, H. T. M. 1997. A quantitative comparison of image restoration methods for confocal microscopy. Journal of Microscopy 185 (3), 354–365. doi:10.1046/j.1365-2818.1997.d01-629.x.

Khan, A. G., Pickl-Herk, A., Gajdzik, L., Marlovits, T. C., Fuchs, R. & Blaas, D. 2011. Entry of a heparan sulphate-binding HRV8 variant strictly depends on dynamin but not on clathrin, caveolin, and flotillin. Virology 412 (1), 55–67. doi:10.1016/j.virol.2010.12.042.

Kirkpatrick, S., Gelatt, C. D. & Vecchi, M. P. 1983. Optimization by Simulated Annealing. Science 220 (4598), 671–680. doi:10.1126/science.220.4598.671.

Kitano, H. 2002. Computational systems biology. Nature 420 (6912), 206–210. doi:10.1038/nature01254.

Kohonen, T. 1990. The self-organizing map. Proceedings of the IEEE 78 (9), 1464–1480. doi:10.1109/5.58325.

Korpisalo, P., Karvinen, H., Rissanen, T. T., Kilpijoki, J., Marjomäki, V., Baluk, P., McDonald, D. M., Cao, Y., Eriksson, U., Alitalo, K. & Ylä-Herttuala, S. 2008. Vascular endothelial growth factor-a and platelet-derived growth factor-b combination gene therapy prolongs angiogenic effects via recruitment of interstitial mononuclear cells and paracrine effects rather than improved pericyte coverage of angiogenic vessels. Circulation Research 103 (10), 1092–1099. doi:10.1161/CIRCRESAHA.108.182287.

Koster, A. J., Grimm, R., Typke, D., Hegerl, R., Stoschek, A., Walz, J. & Baumeister, W. 1997. Perspectives of molecular and cellular electron tomography. Journal of Structural Biology 120 (3), 276–308. doi:10.1006/jsbi.1997.3933.

Laakkonen, J. P., Mäkelä, A. R., Kakkonen, E., Turkki, P., Kukkonen, S., Peränen, J., Ylä-Herttuala, S., Airenne, K. J., Oker-Blom, C., Vihinen-Ranta, M. et al. 2009. Clathrin-independent entry of baculovirus triggers uptake of E. coli in non-phagocytic human cells. PLoS One 4 (4), e5093. doi:10.1371/journal.pone.0005093.

Lachmanovich, E., Shvartsman, D. E., Malka, Y., Botvin, C., Henis, Y. I. & 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. doi:10.1046/j.1365-2818.2003.01239.x.

Lagache, T., Meas-Yedid, V. & Olivo-Marin, J.-C. 2013. A statistical analysis of spatial colocalization using Ripley’s K function. In Biomedical Imaging (ISBI),

2013 IEEE 10th International Symposium on, 896-901. doi:10.1109/ISBI.2013.6556620.

Laksameethanasan, D., Tan, R., Toh, G.-L. & Loo, L.-H. 2013. cellXpress: a fast and user-friendly software platform for profiling cellular phenotypes. BMC Bioinformatics 14 (16), 1-12. doi:10.1186/1471-2105-14-S16-S4.

Landmann, L. 2002. Deconvolution improves colocalization analysis of multiple fluorochromes in 3D confocal data sets more than filtering techniques. Journal of Microscopy 208 (2), 134–147. doi:10.1046/j.1365-2818.2002.01068.x.

Langlois, R. & Frank, J. 2011. A clarification of the terms used in comparing semi-automated particle selection algorithms in Cryo-EM. Journal of Structural Biology 175 (3), 348–352. doi:10.1016/j.jsb.2011.03.009.

Langlois, R., Pallesen, J. & Frank, J. 2011. Reference-free particle selection enhanced with semi-supervised machine learning for cryo-electron microscopy. Journal of Structural Biology 175 (3), 353–361. doi:10.1016/j.jsb.2011.06.004.

Lee, S., Wolberg, G. & Shin, S.-Y. 1997. Scattered data interpolation with multilevel B-splines. IEEE Transactions on Visualization and Computer Graphics 3 (3), 228-244. doi:10.1109/2945.620490.

Lemnitzer, F., Raschbichler, V., Kolodziejczak, D., Israel, L., Imhof, A., Bailer, S. M., Koszinowski, U. & Ruzsics, Z. 2013. Mouse cytomegalovirus egress protein pM50 interacts with cellular endophilin-A2. Cellular Microbiology 15 (2), 335–351. doi:10.1111/cmi.12080.

Li, C. H. & Lee, C. 1993. Minimum cross entropy thresholding. Pattern Recognition 26 (4), 617–625. doi:10.1016/0031-3203(93)90115-D.

Li, F., Yin, Z., Jin, G., Zhao, H. & Wong, S. T. 2013. Bioimage informatics for systems pharmacology. PLoS Computational Biology 9 (4), e1003043. doi:10.1371/journal.pcbi.1003043.

Lichtman, J. W. & Conchello, J.-A. 2005. Fluorescence microscopy. Nature Methods 2 (12), 910–919. doi:10.1038/nmeth817.

Lindeberg, T. 1994. Scale-space theory: A basic tool for analyzing structures at different scales. Journal of Applied Statistics 21 (1-2), 225–270. doi:10.1080/757582976.

Linkert, M., Rueden, C. T., Allan, C., Burel, J.-M., Moore, W., Patterson, A., Loranger, B., Moore, J., Neves, C., MacDonald, D. et al. 2010. Metadata matters: access to image data in the real world. The Journal of Cell Biology 189 (5), 777–782. doi:10.1083/jcb.201004104.

Lorensen, W. E. & Cline, H. E. 1987. Marching cubes: A high resolution 3D surface construction algorithm. In ACM Siggraph Computer Graphics, Vol. 21, 163–169. doi:10.1145/37402.37422.

Lowe, D. G. 2004. Distinctive image features from scale-invariant keypoints. International Journal of Computer Vision 60 (2), 91–110. doi:10.1023/B:VISI.0000029664.99615.94.

Lučić, V., Förster, F. & Baumeister, W. 2005. Structural studies by electron tomography: From cells to molecules. Annual Review of Biochemistry 74, 833–865. doi:10.1146/annurev.biochem.73.011303.074112.

MacQueen, J. 1967. Some methods for classification and analysis of multivariate observations. In Proceedings of 5th Berkeley Symposium on Mathematical Statistics and Probability, 281–297.

Malladi, R., Sethian, J. & Vemuri, B. 1995. Shape modeling with front propagation: a level set approach. IEEE Transactions on Pattern Analysis and Machine Intelligence 17 (2), 158-175. doi:10.1109/34.368173.

Malpica, N., Ortiz de Solorzano, C., Vaquero, J. J., Santos, A., Vallcorba, I., GarcÃa-Sagredo, J. M. & Pozo, F. d. 1997. Applying watershed algorithms to the segmentation of clustered nuclei. Cytometry Part A 28 (4), 289–297.

Manders, E. M., Stap, J., Brakenhoff, G. J., Van Driel, R. & Aten, J. A. 1992. Dynamics of three-dimensional replication patterns during the S-phase, analysed by double labelling of dna and confocal microscopy. Journal of Cell Science 103 (3), 857–862.

Manders, E. M. M., Verbeek, F. J. & Aten, J. A. 1993. Measurement of colocalization of objects in dual-colour confocal images. Journal of Microscopy 169 (3), 375–382. doi:10.1111/j.1365-2818.1993.tb03313.x.

Marabini, R., Herman, G. T. & Carazo, J. M. 1998. 3D reconstruction in electron microscopy using ART with smooth spherically symmetric volume elements (blobs). Ultramicroscopy 72 (1), 53–65. doi:10.1016/S0304-3991(97)00127-7.

Maska, M., Danek, O., Garasa, S., Rouzaut, A., Munoz-Barrutia, A. & Ortiz-de Solorzano, C. 2013. Segmentation and shape tracking of whole fluorescent cells based on the chan-vese model. IEEE Transactions on Medical Imaging 32 (6), 995–1006. doi:10.1109/TMI.2013.2243463.

Matas, J., Chum, O., Urban, M. & Pajdla, T. 2004. Robust wide-baseline stereo from maximally stable extremal regions. Image and Vision Computing 22 (10), 761–767. doi:10.1016/j.imavis.2004.02.006.

McIntosh, R., Nicastro, D. & Mastronarde, D. 2005. New views of cells in 3D: an introduction to electron tomography. Trends in Cell Biology 15 (1), 43–51. doi:j.tcb.2004.11.009.

McNally, J. G., Karpova, T., Cooper, J. & Conchello, J. A. 1999. Three-dimensional imaging by deconvolution microscopy. Methods 19 (3), 373–385. doi:10.1006/meth.1999.0873.

Meijering, E., Smal, I. & Danuser, G. 2006. Tracking in molecular bioimaging. IEEE Signal Processing Magazine 23 (3), 46–53. doi:10.1109/MSP.2006.1628877.

Mueller, B. H., Park, Y., Daudt, D. R., Ma, H.-Y., Akopova, I., Stankowska, D. L., Clark, A. F. & Yorio, T. 2013. Sigma-1 receptor stimulation attenuates calcium influx through activated l-type voltage gated calcium channels in purified retinal ganglion cells. Experimental Eye Research 107, 21–31. doi:10.1016/j.exer.2012.11.002.

Nattkemper, T. W., Twellmann, T., Ritter, H. & Schubert, W. 2003. Human vs. machine: evaluation of fluorescence micrographs. Computers in Biology and Medicine 33 (1), 31–43.

Nicholson, W. V. & Glaeser, R. M. 2001. Review: automatic particle detection in electron microscopy. Journal of Structural Biology 133 (2-3), 90–101. doi:10.1006/jsbi.2001.4348.

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. doi:10.1083/jcb.200507103.

Ntziachristos, V. 2010. Going deeper than microscopy: the optical imaging frontier in biology. Nature Methods 7 (8), 603–614. doi:10.1038/nmeth.1483.

Ogura, T. & Sato, C. 2004a. Auto-accumulation method using simulated annealing enables fully automatic particle pickup completely free from a matching template or learning data. Journal of Structural Biology 146 (3), 344–358. doi:10.1016/j.jsb.2004.01.007.

Ogura, T. & Sato, C. 2004b. Automatic particle pickup method using a neural network has high accuracy by applying an initial weight derived from eigenimages: a new reference free method for single-particle analysis. Journal of Structural Biology 145 (1), 63–75. doi:10.1016/S1047-8477(03)00139-4.

Olivo-Marin, J.-C. 2002. Extraction of spots in biological images using multiscale products. Pattern Recognition 35 (9), 1989–1996. doi:10.1016/S0031-3203(01)00127-3.

Otsu, N. 1979. A threshold selection method from gray-level histograms. IEEE Transactions on Systems, Man and Cybernetics 9 (1), 62–66. doi:10.1109/TSMC.1979.4310076.

Pajari, A.-M., Päivärinta, E., Paavolainen, L., Vaara, E., Törrönen, R., Mutanen, M., Marjomäki, V. & Ridley, A. 2013. Cloudberry inhibits hepatocyte growth factor-induced cell motility and phosphatidylinositol 3-kinase/AKT activation in colon carcinoma cells and tumors in Min mice. Manuscript.

Penczek, P. A., Renka, R. & Schomberg, H. 2004. Gridding-based direct Fourier inversion of the three-dimensional ray transform. Journal of the Optical Society of America A 21 (4), 499–509. doi:10.1364/JOSAA.21.000499.

Penczek, P. A. 2010. Fundamentals of three-dimensional reconstruction from projections. Methods in Enzymology 482, 1–33. doi:10.1016/S0076-6879(10)82001-4.

Penczek, P., Marko, M., Buttle, K. & Frank, J. 1995. Double-tilt electron tomography. Ultramicroscopy 60 (3), 393–410. doi:10.1016/0304-3991(95)00078-X.

Peng, H., Bateman, A., Valencia, A. & Wren, J. D. 2012. Bioimage informatics: a new category in bioinformatics. Bioinformatics 28 (8), 1057–1057. doi:10.1093/bioinformatics/bts111.

Peng, H., Ruan, Z., Long, F., Simpson, J. H. & Myers, E. W. 2010. V3D enables real-time 3D visualization and quantitative analysis of large-scale biological image data sets. Nature Biotechnology 28 (4), 348–353. doi:10.1038/nbt.1612.

Peng, H. 2008. Bioimage informatics: a new area of engineering biology. Bioinformatics 24 (17), 1827–1836. doi:10.1093/bioinformatics/btn346.

Pepperkok, R. & Ellenberg, J. 2006. High-throughput fluorescence microscopy for systems biology. Nature Reviews Molecular Cell Biology 7 (9), 690–696. doi:10.1038/nrm1979.

Perona, P. & Malik, J. 1990. Scale-space and edge detection using anisotropic diffusion. IEEE Transactions on Pattern Analysis and Machine Intelligence 12 (7), 629–639. doi:10.1109/34.56205.

Plitzko, J. M. & Baumeister, W. 2007. Cryoelectron tomography (cet). In P. W. Hawkes & J. C. H. Spence (Eds.) Science of Microscopy. New York: Springer, 535–604.

Quattrocchi, S., Ruprecht, N., Bönsch, C., Bieli, S., Zürcher, C., Boller, K., Kempf, C. & Ros, C. 2012. Characterization of the early steps of human parvovirus b19 infection. Journal of Virology 86 (17), 9274–9284. doi:10.1128/JVI.01004-12.

Radermacher, M. 2006. Weighted back-projection methods. In J. Frank (Ed.) Electron Tomography: Methods for Three-Dimensional Visualization of Structures in the Cell (2nd edition). New York: Springer, 245–273.

Radon, J. 1986. On the determination of functions from their integral values along certain manifolds. IEEE Transactions on Medical Imaging 5 (4), 170–176. doi:10.1109/TMI.1986.4307775.

Ramani Lata, K., Penczek, P. & Frank, J. 1995. Automatic particle picking from electron micrographs. Ultramicroscopy 58 (3), 381–391. doi:10.1016/0304-3991(95)00002-I.

Raykar, V. C., Yu, S., Zhao, L. H., Valadez, G. H., Florin, C., Bogoni, L. & Moy, L. 2010. Learning from crowds. The Journal of Machine Learning Research 11, 1297–1322.

Reimer, L. & Kohl, H. 2008. Transmission Electron Microscopy: Physics of Image Formation (5th edition). New York: Springer.

Rintanen, N., Karjalainen, M., Alanko, J., Paavolainen, L., Mäki, A., Nissinen, L., Lehkonen, M., Kallio, K., Cheng, R. H., Upla, P., Ivaska, J. & Marjomäki, V. 2012. Calpains promote a2b1 integrin down-regulation after integrin clustering triggered internalization. Molecular Biology of Cell 23 (3), 448–463. doi:10.1091/mbc.E11-06-0548.

Roseman, A. M. 2003. Particle finding in electron micrographs using a fast local correlation algorithm. Ultramicroscopy 94 (3), 225–236. doi:10.1016/S0304-3991(02)00333-9.

Rullgård, H., Öfverstedt, L.-G., Masich, S., Daneholt, B. & Öktem, O. 2011. Simulation of transmission electron microscope images of biological specimens. Journal of Microscopy 243 (3), 234–256. doi:10.1111/j.1365-2818.2011.03497.x.

Ruprecht, J. & Nield, J. 2001. Determining the structure of biological macromolecules by transmission electron microscopy, single particle analysis and 3d reconstruction. Progress in Biophysics and Molecular Biology 75 (3), 121–164. doi:10.1016/S0079-6107(01)00004-9.

Rusinkiewicz, S. & Levoy, M. 2001. Efficient variants of the ICP algorithm. In Proceedings of the Third International Conference on 3-D Digital Imaging and Modeling, 145–152. doi:10.1109/IM.2001.924423.

Ruthardt, N., Lamb, D. C. & Bräuchle, C. 2011. Single-particle tracking as a quantitative microscopy-based approach to unravel cell entry mechanisms of viruses and pharmaceutical nanoparticles. Molecular Therapy 19 (7), 1199–1211. doi:10.1038/mt.2011.102.

Ruusuvuori, P., Lehmussola, A., Selinummi, J., Rajala, T., Huttunen, H. & Yli-Harja, O. 2008. Benchmark set of synthetic images for validating cell image analysis algorithms. In Proceedings of the 16th European Signal Processing Conference, EUSIPCO.

Ruusuvuori, P., Äijö, T., Chowdhury, S., Garmendia-Torres, C., Selinummi, J., Birbaumer, M., Dudley, A., Pelkmans, L. & Yli-Harja, O. 2010. Evaluation of methods for detection of fluorescence labeled subcellular objects in microscope images. BMC Bioinformatics 11 (1), 248. doi:10.1186/1471-2105-11-248.

Sage, D., Neumann, F. R., Hediger, F., Gasser, S. M. & Unser, M. 2005. Automatic tracking of individual fluorescence particles: application to the study of chromosome dynamics. IEEE Transactions on Image Processing 14 (9), 1372–1383. doi:10.1109/TIP.2005.852787.

Sandberg, K., Mastronarde, D. N. & Beylkin, G. 2003. A fast reconstruction algorithm for electron microscope tomography. Journal of Structural Biology 144 (1), 61–72. doi:10.1016/j.jsb.2003.09.013.

Sarder, P. & Nehorai, A. 2006. Deconvolution methods for 3-d fluorescence microscopy images. IEEE Signal Processing Magazine 23 (3), 32–45. doi:10.1109/MSP.2006.1628876.

Saxton, M. J. & Jacobson, K. 1997. Single-particle tracking: applications to membrane dynamics. Annual Review of Biophysics and Biomolecular Structure 26 (1), 373–399. doi:10.1146/annurev.biophys.26.1.373.

Saxton, M. J. 1994. Single-particle tracking: models of directed transport. Biophysical Journal 67 (5), 2110–2119. doi:10.1016/S0006-3495(94)80694-0.

Sbalzarini, I. F. & Koumoutsakos, P. 2005. Feature point tracking and trajectory analysis for video imaging in cell biology. Journal of Structural Biology 151 (2), 182–195. doi:10.1016/j.jsb.2005.06.002.

Schindelin, J., Arganda-Carreras, I., Frise, E., Kaynig, V., Longair, M., Pietzsch, T., Preibisch, S., Rueden, C., Saalfeld, S., Schmid, B. et al. 2012. Fiji: an open-source platform for biological-image analysis. Nature Methods 9 (7), 676–682. doi:10.1038/nmeth.2019.

Schmidt, R., Wurm, C. A., Jakobs, S., Engelhardt, J., Egner, A. & Hell, S. W. 2008. Spherical nanosized focal spot unravels the interior of cells. Nature Methods 5 (6), 539–544. doi:10.1038/nmeth.1214.

Schneider, C. A., Rasband, W. S. & Eliceiri, K. W. 2012. NIH Image to ImageJ: 25 years of image analysis. Nature Methods 9 (7), 671–675. doi:10.1038/nmeth.2089.

Schroeder, W., Martin, K. & Lorensen, B. 2006. The Visualization Toolkit. An object-oriented approach to 3D graphics. Kitware, Inc.

Serysheva, I. I., Orlova, E. V., Chiu, W., Sherman, M. B., Hamilton, S. L. & Van Heel, M. 1995. Electron cryomicroscopy and angular reconstitution used to visualize the skeletal muscle calcium release channel. Nature Structural & Molecular Biology 2 (1), 18–24. doi:10.1038/nsb0195-18.

Sezgin, M. & Bülent, S. 2004. Survey over image thresholding techniques and quantitative performance evaluation. Journal of Electronic Imaging 13 (1), 146–168. doi:10.1117/1.1631316.

Shafique, K. & Shah, M. 2005. A noniterative greedy algorithm for multiframe point correspondence. IEEE Transactions on Pattern Analysis and Machine Intelligence 27 (1), 51–65. doi:10.1109/TPAMI.2005.1.

Shen, P. S., Enderlein, D., Nelson, C. D. S., Carter, W. S., Kawano, M., Xing, L., Swenson, R. D., Olson, N. H., Baker, T. S., Cheng, R. H., Atwood, W. J., Reimar, J. & Belnap, D. M. 2011. The structure of avian polyomavirus reveals variably sized capsids, non-conserved inter-capsomere interactions, and a possible location of the minor capsid protein VP4. Virology 411 (1), 142–152. doi:10.1016/j.virol.2010.12.005.

Sigworth, F. J. 2004. Classical detection theory and the cryo-EM particle selection problem. Journal of Structural Biology 145 (1), 111–122. doi:10.1016/j.jsb.2003.10.025.

Siljamäki, E., Rintanen, N., Kirsi, M., Upla, P., Wang, W., Karjalainen, M., Ikonen, E. & Marjomäki, V. 2013. Cholesterol dependence of collagen and echovirus 1 trafficking along the novel $\alpha 2\beta 1$ integrin internalization pathway. PLoS ONE 8 (2), e55465. doi:10.1371/journal.pone.0055465.

Singh, V., Marinescu, D. C. & Baker, T. S. 2004. Image segmentation for automatic particle identification in electron micrographs based on hidden markov random field models and expectation maximization. Journal of Structural Biology 145 (1), 123–141. doi:10.1016/j.jsb.2003.11.028.

Smal, I., Draegestein, K., Galjart, N., Niessen, W. & Meijering, E. 2008a. Particle filtering for multiple object tracking in dynamic fluorescence microscopy images: Application to microtubule growth analysis. IEEE Transactions on Medical Imaging 27 (6), 789-804. doi:10.1109/TMI.2008.916964.

Smal, I., Niessen, W. & Meijering, E. 2008b. A new detection scheme for multiple object tracking in fluorescence microscopy by joint probabilistic data association filtering. In Biomedical Imaging: From Nano to Macro, 2008. ISBI 2008. 5th IEEE International Symposium on, 264-267. doi:10.1109/ISBI.2008.4540983.

Smal, I., Loog, M., Niessen, W. & Meijering, E. 2010. Quantitative comparison of spot detection methods in fluorescence microscopy. IEEE Transactions on Medical Imaging 29 (2), 282–301. doi:10.1109/TMI.2009.2025127.

Soille, P. 2003. Morphological Image Analysis: Principles and Applications (2nd edition). Springer-Verlag New York, Inc.

Sorzano, C. O. S., Jonic, S., El-Bez, C., Carazo, J. M., De Carlo, S., Thévenaz, P. & Unser, M. 2004. A multiresolution approach to orientation assignment in 3D electron microscopy of single particles. Journal of Structural Biology 146 (3), 381–392. doi:10.1016/j.jsb.2004.01.006.

Sorzano, C. O. S., Marabini, R., Boisset, N., Rietzel, E., Schröder, R., Herman, G. T. & Carazo, J. M. 2001. The effect of overabundant projection directions on 3D reconstruction algorithms. Journal of Structural Biology 133 (2), 108–118. doi:10.1006/jsbi.2001.4338.

Sorzano, C. O. S., Recarte, E., Alcorlo, M., Bilbao-Castro, J. R., San-Martín, C., Marabini, R. & Carazo, J. M. 2009. Automatic particle selection from electron micrographs using machine learning techniques. Journal of Structural Biology 167 (3), 252–260. doi:10.1016/j.jsb.2009.06.011.

van Steensel, B., van Binnendijk, E. P., Hornsby, C. D., van der Voort, H. T., Krozowski, Z. S., de Kloet, E. R. & van Driel, R. 1996. Partial colocalization of glucocorticoid and mineralocorticoid receptors in discrete compartments in nuclei of rat hippocampus neurons. Journal of Cell Science 109 (4), 787–792.

Stehle, T., Gamblin, S. J., Yan, Y. & Harrison, S. C. 1996. The structure of simian virus 40 refined at 3.1 Å resolution. Structure 4 (2), 165–182. doi:10.1016/S0969-2126(96)00020-2.

Stephens, D. J. & Allan, V. J. 2003. Light microscopy techniques for live cell imaging. Science 300 (5616), 82–86. doi:10.1126/science.1082160.

Swedlow, J. R., Hu, K., Andrews, P. D., Roos, D. S. & Murray, J. M. 2002. Measuring tubulin content in toxoplasma gondii: A comparison of laser-scanning confocal and wide-field fluorescence microscopy. PNAS 99 (4), 2014–2019. doi:10.1073/pnas.022554999.

Swedlow, J. R. & Eliceiri, K. W. 2009. Open source bioimage informatics for cell biology. Trends in Cell Biology 19 (11), 656–660. doi:10.1016/j.tcb.2009.08.007.

Swedlow, J. R., Goldberg, I. G., Eliceiri, K. W. et al. 2009. Bioimage informatics for experimental biology. Annual Review of Biophysics 38, 327–346. doi:10.1146/annurev.biophys.050708.133641.

Thomann, D., Rines, D., Sorger, P. & Danuser, G. 2002. Automatic fluorescent tag detection in 3D with super-resolution: application to the analysis of chromosome movement. Journal of Microscopy 208 (1), 49–64. doi:10.1046/j.1365-2818.2002.01066.x.

Tomasi, C. & Manduchi, R. 1998. Bilateral filtering for gray and color images. In Sixth International Conference on Computer Vision, 839–846. doi:10.1109/ICCV.1998.710815.

Trier, O. D. & Jain, A. K. 1995. Goal-directed evaluation of binarization methods. IEEE Transactions on Pattern Analysis and Machine Intelligence 17 (12), 1191–1201. doi:10.1109/34.476511.

Tsurui, H., Nishimura, H., Hattori, S., Hirose, S., Okumura, K. & Shirai, T. 2000. Seven-color fluorescence imaging of tissue samples based on fourier spectroscopy and singular value decomposition. Journal of Histochemistry and Cytochemistry 48 (5), 653–662. doi:10.1177/002215540004800509.

Tuna, U., Sohlberg, A. & Ruotsalainen, U. 2013. Can we reduce SPECT acquisition time using MAP-EM reconstruction. Journal of Pattern Recognition and Intelligent Systems 1 (3), 54–63.

Turkki, P., Makkonen, K.-E., Huttunen, M., Laakkonen, J. P., Ylä-Herttuala, S., Airenne, K. J. & Marjomäki, V. 2013. Cell susceptibility to baculovirus transduction and echovirus infection is modified by protein kinase c phosphorylation and vimentin organization. Journal of Virology 87 (17), 9822–9835. doi:10.1128/JVI.01004-13.

Tvarusko, W., Bentele, M., Misteli, T., Rudolf, R., Kaether, C., Spector, D., Gerdes, H. & Eils, R. 1999. Time-resolved analysis and visualization of dynamic processes in living cells. Proceedings of the National Academy of Sciences 96 (14), 7950–7955. doi:10.1073/pnas.96.14.7950.

Veenman, C. J., Reinders, M. J. & Backer, E. 2001. Resolving motion correspondence for densely moving points. IEEE Transactions on Pattern Analysis and Machine Intelligence 23 (1), 54–72. doi:10.1109/34.899946.

Verveer, P. J., Gemkow, M. J. & Jovin, T. M. 1999. A comparison of image restoration approaches applied to three-dimensional confocal and wide-field fluorescence microscopy. Journal of Microscopy 193 (1), 50–61. doi:10.1046/j.1365-2818.1999.00421.x.

Villalta, J. I., Galli, S., Iacaruso, M. F., Arciuch, V. G. A., Poderoso, J. J., Jares-Erijman, E. A. & Pietrasanta, L. I. 2011. New algorithm to determine true colocalization in combination with image restoration and time-lapse confocal microscopy to map kinases in mitochondria. PLoS ONE 6 (4), e19031. doi:10.1371/journal.pone.0019031.

Vincent, L. 1993. Morphological grayscale reconstruction in image analysis: applications and efficient algorithms. IEEE Transactions on Image Processing 2 (2), 176-201. doi:10.1109/83.217222.

Vincent, L. & Soille, P. 1991. Watersheds in digital spaces: an efficient algorithm based on immersion simulations. IEEE Transactions on Pattern Analysis and Machine Intelligence 13 (6), 583–598. doi:10.1109/34.87344.

Volkmann, N. 2004. An approach to automated particle picking from electron micrographs based on reduced representation templates. Journal of Structural Biology 145 (1), 152–156. doi:10.1016/j.jsb.2003.11.026.

Vonesch, C., Aguet, F., Vonesch, J.-L. & Unser, M. 2006. The colored revolution of bioimaging. IEEE Signal Processing Magazine 23 (3), 20-31. doi:10.1109/MSP.2006.1628875.

Voss, N. R., Yoshioka, C. K., Radermacher, M., Potter, C. S. & Carragher, B. 2009. DoG Picker and TiltPicker: software tools to facilitate particle selection in single particle electron microscopy. Journal of Structural Biology 166 (2), 205–213. doi:10.1016/j.jsb.2009.01.004.

Wallace, W., Schaefer, L. H. & Swedlow, J. R. 2001. A workingperson's guide to deconvolution in light microscopy. Biotechniques 31 (5), 1076–1097.

Warfield, S., Zou, K. & Wells, W. 2004. Simultaneous truth and performance level estimation (STAPLE): an algorithm for the validation of image segmentation. IEEE Transactions on Medical Imaging 23 (7), 903-921. doi:10.1109/TMI.2004.828354.

Waters, J. C. 2009. Accuracy and precision in quantitative fluorescence microscopy. The Journal of Cell Biology 185 (7), 1135–1148. doi:10.1083/jcb.200903097.

Whitlow, H. J., Ren, M., van Kan, J. A., Watt, F. & White, D. 2007. Exploratory nuclear microprobe data visualisation using 3- and 4-dimensional biological volume rendering tools. Nuclear Instruments and Methods in Physics Research Section B: Beam Interactions with Materials and Atoms 260 (1), 28–33. doi:10.1016/j.nimb.2007.01.316.

Wilt, B. A., Burns, L. D., Ho, E. T. W., Ghosh, K. K., Mukamel, E. A. & Schnitzer, M. J. 2009. Advances in light microscopy for neuroscience. Annual Review of Neuroscience 32. doi:10.1146/annurev.neuro.051508.135540.

Winkler, H. & Taylor, K. A. 2006. Accurate marker-free alignment with simultaneous geometry determination and reconstruction of tilt series in electron tomography. Ultramicroscopy 106 (3), 240–254. doi:10.1016/j.ultramic.2005.07.007.

Wong, H., Chen, J., Mouche, F., Rouiller, I. & Bern, M. 2004. Model-based particle picking for cryo-electron microscopy. Journal of Structural Biology 145 (1), 157–167. doi:10.1016/j.jsb.2003.05.001.

Wu, X., Pertovaara, H., Dastidar, P., Vornanen, M., Paavolainen, L., Marjomäki, V., Järvenpää, R., Eskola, H. & Kellokumpu-Lehtinen, P.-L. 2013. ADC measurements in diffuse large B-cell lymphoma and follicular lymphoma: a DWI and cellularity study. European Journal of Radiology 82 (4), 158–164. doi:10.1016/j.ejrad.2012.11.021.

Wörz, S., Sander, P., Pfannmöller, M., Rieker, R., Joos, S., Mechtersheimer, G., Boukamp, P., Lichter, P. & 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. doi:10.1109/TMI.2010.2049857.

Yang, S., Papagiakoumou, E., Guillon, M., de Sars, V., Tang, C.-M. & Emiliani, V. 2011. Three-dimensional holographic photostimulation of the dendritic arbor. Journal of Neural Engineering 8 (4), 046002. doi:10.1088/1741-2560/8/4/046002.

Yoo, T. S., Ackerman, M. J., Lorensen, W. E., Schroeder, W., Chalana, V., Aylward, S., Metaxas, D. & Whitaker, R. 2002. Engineering and algorithm design for an image processing Api: a technical report on ITK–the Insight Toolkit. Studies in Health Technology and Informatics 85, 586–592.

Yu, Z. & Bajaj, C. 2004. Detecting circular and rectangular particles based on geometric feature detection in electron micrographs. Journal of Structural Biology 145 (1), 168–180. doi:10.1016/j.jsb.2003.10.027.

Yuste, R. 2005. Fluorescence microscopy today. Nature Methods 2 (12), 902–904. doi:10.1038/nmeth1205-902.

Zhang, L., Choi, S. I. & Park, S. Y. 2011. Robust ICP registration using biunique correspondence. In International Conference on 3D Imaging, Modeling, Processing, Visualization and Transmission (3DIMPVT), 80–85. doi:10.1109/3DIMPVT.2011.18.

Zhou, X. & Wong, S. T. 2006. Informatics challenges of high-throughput microscopy. IEEE Signal Processing Magazine 23 (3), 63–72. doi:10.1109/MSP.2006.1628879.

Zhu, Y., Carragher, B., Glaeser, R. M., Fellmann, D., Bajaj, C., Bern, M., Mouche, F., Haas, F., Hall, R. J., Kriegman, D., Ludtke, S. J., Mallick, S. P., Penczek, P. A., Roseman, A. M., Sigworth, F. J., Volkmann, N. & Potter, C. S. 2004. Automatic particle selection: results of a comparative study. Journal of Structural Biology 145 (1), 3–14. doi:10.1016/j.jsb.2003.09.033.

Zhu, Y., Carragher, B. & Potter, C. S. 2002. Fast detection of generic biological particles in cryo-EM images through efficient Hough transforms. In Proceedings of IEEE International Symposium on Biomedical Imaging, 205–208.

Zimmer, C., Zhang, B., Dufour, A., Thébaud, A., Berlemont, S., Meas-Yedid, V. & Olivo-Marin, J.-C. 2006. On the digital trail of mobile cells. IEEE Signal Processing Magazine 23 (3), 54–62. doi:10.1109/MSP.2006.1628878.

Zinchuk, V. & Grossenbacher-Zinchuk, O. 2009. Recent advances in quantitative colocalization analysis: focus on neuroscience. Progress in Histochemistry and Cytochemistry 44 (3), 125–172. doi:10.1016/j.proghi.2009.03.001.

ORIGINAL PAPERS

PI

BIOIMAGEXD: AN OPEN, GENERAL-PURPOSE AND HIGH-THROUGHPUT IMAGE-PROCESSING PLATFORM

by

Pasi Kankaanpää, Lassi Paavolainen, Silja Tiitta, Mikko Karjalainen, Joacim Päivärinne, Jonna Nieminen, Varpu Marjomäki, Jyrki Heino & Daniel J White
2012

Nature Methods 9(7)

BioImageXD: an open, general-purpose and high-throughput image-processing platform

Pasi Kankaanpää[1], Lassi Paavolainen[2], Silja Tiitta[1], Mikko Karjalainen[2], Joacim Päivärinne[1], Jonna Nieminen[1], Varpu Marjomäki[2], Jyrki Heino[1] & Daniel J White[2,3]

BioImageXD puts open-source computer science tools for three-dimensional visualization and analysis into the hands of all researchers, through a user-friendly graphical interface tuned to the needs of biologists. BioImageXD has no restrictive licenses or undisclosed algorithms and enables publication of precise, reproducible and modifiable workflows. It allows simple construction of processing pipelines and should enable biologists to perform challenging analyses of complex processes. We demonstrate its performance in a study of integrin clustering in response to selected inhibitors.

Bioimaging has emerged as one of the key tools in biomedical research. New imaging devices are appearing rapidly, producing large multidimensional images with several data channels in medium-to-high throughput. Specialized software is required to extract interpretable results from such data. Despite recent and ongoing progress in both commercial and academic software, availability of user-friendly tools often remains a limiting factor, especially in complex and high-throughput applications[1–3]. We formed a multidisciplinary consortium to develop bioimage processing software (**Supplementary Methods**) that is (i) open: has open-source code; (ii) extensive: has many features; (iii) usable: is inexpensive, easy to use and fast; (iv) adjustable: all features are adjustable and combinable into pipelines; (v) applicable: especially for validated batch processing of multidimensional time-lapse data; and (vi) extendable: with a modular design that allows addition of new functionality.

We now report the release of version 1.0 of BioImageXD (http://www.bioimagexd.net/). We explain for each of our design criteria why we consider them important and describe how BioImageXD fulfills them. We then compare BioImageXD to other software. Finally we demonstrate the capabilities of BioImageXD in practice with user-friendly, automated, multiparametric image analyses of the movement, clustering and internalization of integrins.

Criteria in BioImageXD development

Open. Our goal was to develop software that is transparent, with open and documented source code and no undisclosed algorithms or hidden processing the user is unaware of or cannot control. It is our view that software for scientific data analysis, in contrast to engineering, is best when it is not a 'black box'. Different implementations of the same method can often give different results, the reasons for which only become clear by comparing the source codes. Increasingly (and correctly in our view), disclosure of source code for image-processing methods and algorithms is recommended in scientific research[4–6]. Publication of scientific software tools also often requires the source code (the method) to be open, as is the norm for all other protocols in sample preparation, experimental procedure and data analysis. BioImageXD (**Fig. 1**) is released under a GNU General Public License version 2 (http://www.gnu.org/licenses/gpl-2.0.html); the source code is fully open.

Subsampling of image data in software may occur without the user's knowledge, for instance in the interest of user-interface responsiveness, and may affect data interpretations. Although BioImageXD can also subsample image data, subsampling is never done automatically and is always clearly indicated. All processed image data are written into new files by user action, with processing information automatically saved, making workflows retraceable and minimizing inadvertent processing. A concise user guide is provided with the software.

Extensive. We sought to develop software with a comprehensive collection of features for processing,

[1]Department of Biochemistry and Food Chemistry, University of Turku, Turku, Finland. [2]Department of Biological and Environmental Science, and Nanoscience Center, University of Jyväskylä, Jyväskylä, Finland. [3]Max Planck Institute of Molecular Cell Biology and Genetics, Dresden, Germany. Correspondence should be addressed to P.K. (pasi.kankaanpaa@utu.fi).

PUBLISHED ONLINE 28 JUNE 2012; DOI:10.1038/NMETH.2047

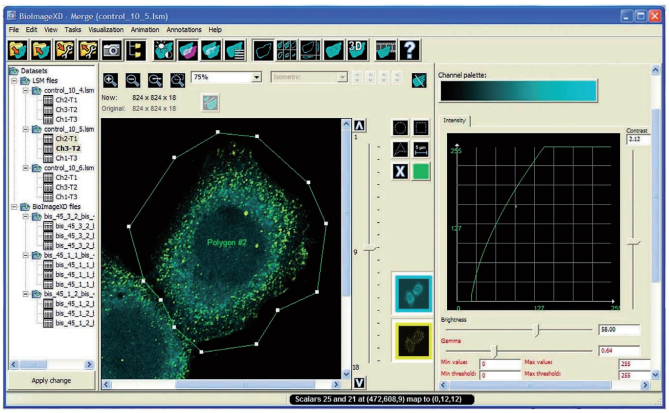


Figure 1 | BioImageXD GUI. Screenshot shows toolbars and menus (top), a file tree of the loaded images (left), and the currently active visualization (middle), showing here a two-channel merged image with a user-defined region of interest. On the right are settings for the currently active task, showing here the graphical intensity transfer function, a central feature in accurate image adjustment.

visualization and analysis. Bioimaging applications today are greatly varied, and a single software package should be able to meet as many needs as possible, without requiring extra programming. BioImageXD has been designed to be a general-purpose processing program, featuring approximately 220 tools (**Supplementary Table 1**). Processing tools include deconvolution, registration, 32 tools for basic mathematical and logical processing, and six noise-reduction algorithms. Visualization tools include an interactive three-dimensional (3D) mode with 11 simultaneously visible modules for creating volume renderings, surface renderings, 3D measurements and so on (**Fig. 2**, **Supplementary Fig. 1** and **Supplementary Video 1**). Movies can be created with an animator supporting two different combinable techniques: keyframes and virtual camera flight paths (**Supplementary Fig. 2** and **Supplementary Videos 2** and **3**). Analysis tools feature both voxel-based and object-based approaches. For the latter, 17 methods are available for segmenting image data into quantifiable objects (**Supplementary Fig. 3** and **Supplementary Video 4**) based on thresholding, watershed, region growing and active contours. A 3D motion tracking algorithm can follow the movement of different features, such as a few cells, or hundreds of segmented objects in a cell (**Supplementary Video 5**).

Many scientific journals have guidelines for setting up the intensity-transfer function (used to adjust, for instance, brightness and contrast)[5]. The function is represented graphically in BioImageXD (**Fig. 1**), enabling the user to check this graph and evaluate possible data loss (which happens in principle every time brightness or contrast, for example, is increased).

Studies of colocalization (whether two markers occupy the same spatiotemporal location) are very common but are often poorly executed in part owing to tool inaccessibility[7,8]. In BioImageXD, colocalization can be visualized with maps and histograms, and quantified with 54 parameters. Three statistical significance methods[9–11], automatic threshold determination[9], and both object-based and pixel-based colocalization are supported.

Usable. Our aim was to make BioImageXD inexpensive and simple to install and run on any common computer. One of the bottlenecks of bioimage processing is that the required software often comes with an expensive and restrictive license policy, limiting availability to isolated workstations that may also control microscopes. BioImageXD can be used free of charge on any computer with either 32-bit or 64-bit Windows (XP, Vista or 7), Mac OSX (10.6 or 10.7) or modern Linux operating systems. The software consists of a single package that is simple to install. Most features work without specific system requirements, but if processor-intensive tasks progress slowly, BioImageXD can subsample images on loading

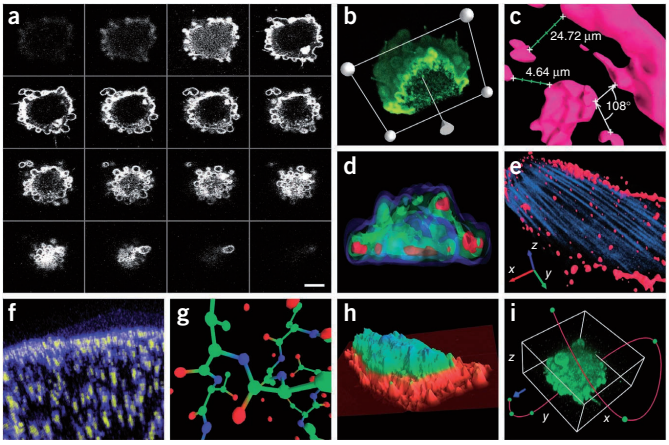


Figure 2 | Examples of visualizations created with BioImageXD. (**a**) Gallery mode for viewing 3D images as 2D slices. Scale bar, 10 µm. (**b**) Volume rendering of 2D slices into a 3D image, shown here with a cutting plane. (**c**) Surface rendering, shown here with 3D distance and angle measurements. (**d**) Several semitransparent surfaces created at the same time. (**e**) Volume rendering (blue actin filaments) and surface rendering (red integrins) displayed together, shown here with a 3D axis marker. (**f**) Volume rendered cell-surface proteins. (**g**) A 3D visualization of molecular models; detail of an integrin I-domain model[37]. (**h**) Warp scalar rendering of a cell showing atomic force microscopy data in three dimensions. (**i**) The 3D renderings can be animated in various ways, such as by specifying a 'flight path' (red line) for a virtual camera.

(this is clearly indicated and easily undone). Data sets larger than the computer's RAM can be visualized and processed.

Bioimaging software should be easy and efficient to use, with an intuitive graphical user interface (GUI)[12]. The GUI of BioImageXD is based on a single large window (**Fig. 1**) with an unchanged layout. Advanced settings are always available but are colored red to warn novice users (**Fig. 1**). Additional assistance is offered in simple on-screen instructions (examples are shown in **Supplementary Figs. 3** and **4**). For added performance, computationally demanding parts of the code have been implemented as platform-native optimized libraries, written in C++. However, the software architecture and implementation still leave room for improvement in performance.

Adjustable. To enable transparent and reproducible science, our goal was to develop software in which parameters for all methods and algorithms are individually adjustable in the GUI, identified by their correct technical name, but not at the expense of making the GUI too cluttered. In BioImageXD, the layout and on-screen color-coding and instructions (which include accepted inputs and outputs) also aid in pipeline assembly, which extends the flexibility of software for more complex image-processing needs. Such procedure lists can be created in BioImageXD without programming skills, assembling many methods and algorithms into tunable, human-readable pipelines.

Applicable. Wide-field and confocal fluorescence microscopy are among the most commonly used methods in bioimaging. Combinations of multiple channels, three dimensions, time series and images acquired in high throughput are becoming the norm, as systems biology imaging approaches gather momentum[13–16]. We therefore wished to develop software that is applicable to these situations and also to general 2D image processing. In BioImageXD, we integrated three-dimensional, multichannel and time-lapse support into the implementations of all processing tools as well as the possibility to automate batch processing of large amounts of data. Images are not loaded into memory upon opening them. Instead, image data are read only on demand, for efficient memory use when switching between data sets or handling large time series. Channels are processed separately to avoid confusion. Any command pipeline created in BioImageXD can be used for batch processing. The batch processor has a GUI (**Supplementary Fig. 4**), can run several pipelines simultaneously and aggregates results into spreadsheets, thus facilitating high-throughput data processing. BioImageXD is already used for multiwell applications in some laboratories, but pipelines cannot exchange data, and the retention of spatial information on multiwell plates is not yet supported. We are working on these aspects. High-throughput applications are a major development area for BioImageXD, with ongoing projects aiming at performing complex analyses interactively in real time with high-throughput image-acquisition platforms.

Validations tools are an important aspect of bioimaging software applicability. For complex and high-throughput applications, in particular, it is often difficult to verify results manually. BioImageXD can create simulated images, with a tool that can in its current iteration simulate particles with various distributions in relation to presegmented cell shapes and various levels of noise. The simulated data can be used to validate segmentation, motion tracking and other procedures in many applications.

Extendable. As scientific research continuously spawns new questions, requiring implementation of new or modified tools to answer them, bioimaging software should build upon and engineer compatibility with existing libraries and tools[17]. We therefore designed BioImageXD to have modular, extendable software architecture, primarily written in Python for simplicity and extendibility, and to leverage the open-source libraries Visualization Toolkit (VTK)[18] and Insight Segmentation and Registration Toolkit (ITK)[19]. It has a large adjustable feature set, with input from common microscopy and generic image file formats, including the Open Microscopy Environment (OME) TIFF format[20], and output in open standard formats (VTK XML image, TIFF, OME-TIFF and comma-separated value spreadsheets). Ideally, the architecture of BioImageXD should also facilitate widespread use of new algorithms.

Comparing BioImageXD to other software

We feel it is useful to compare the main characteristics of BioImageXD to those of other biological image-processing software. However, we note that such comparisons are difficult to make completely fairly and accurately: the tools may have fundamentally different goals, and in these comparisons we could only cover a subset of all the available software packages. Despite these limitations, we defined a testing protocol based on our six design criteria and compared BioImageXD to other open-source software, BioImage Suite[21], ImageJ (Fiji, http://fiji.sc/), Vaa3D (ref. 22) and VisBio[23], and also to proprietary software, Imaris (Bitplane), Fluoview (Olympus), Volocity (PerkinElmer), ZEN Lite (Zeiss) and Matlab plus Image Processing Toolbox (MathWorks). For these comparisons, we listed qualitative descriptions of features and characteristics of the software for each of our six criteria, assessed the number of features on a four-point scale and measured the time it took to perform certain basic operations (**Table 1** and **Supplementary Tables 2–4**).

Closed-source software products have a different design philosophy than that of BioImageXD. By definition, they cannot fulfill our first criterion of openness; furthermore, their usability, adjustability and especially extendibility are often restricted. BioImageXD compared favorably to the proprietary software we considered in the number of features and performance speed, but the proprietary software may offer more stability and maturity.

Other open-source software projects are as open and extendable as BioImageXD but are often less extensive or applicable. However, comparisons are difficult, as different projects have different aims. For instance, ImageJ is a generic platform for processing individual images, with an easy plugin architecture, resulting in many tools developed by a large community, whereas BioImageXD is a single package, combining the algorithmic and 3D visualization power of ITK and VTK with a user-friendly GUI.

Matlab, in contrast, has a different philosophy from all the other software, as its use requires programming. We included it in the comparisons to make them more representative of the different approaches available.

The open-source programs we considered may be better suited than BioImageXD for some specific applications, such as filament tracing, image annotation, manual manipulation of individual images or processing of bright-field images. However, BioImageXD often compares favorably to them in speed and has more well-integrated features, especially in areas such as 3D visualization, segmentation and tracking. BioImageXD also offers

Table 1 | Summary for comparison of BioImageXD to other software

	Open	Extensive	Usable	Adjustable	Applicable	Extendable
BioImageXD (1.0RC3)	••••	••••	••••	••••	••••	••••
BioImage Suite (3.01)	••••	••	•••	•••	•	••••
ImageJ (Fiji) (Madison, Wisconsin, USA, updated 02/2012)	••••	••	••	•••	••	••••
Vaa3D (previously V3D) (2.707)	•••	••	•••	•••	•	••••
VisBio (3.40RC1)	••••	•	••	••	•	•••
Matlab (7.9.0.529) plus IPT (6.4) (MathWorks)	••	••	•••	•••	•	••
Imaris (7.4.0) (Bitplane)	•	•••	•••	••	••	••
FluoView (3.01.01.09) (Olympus)	••	••	••	••	•	•
Volocity (6.0.1) (PerkinElmer)	•	••	•	•••	••	•
ZEN Lite (2011) (Zeiss)	••	•	••	•	••	•

For each of our six software design criteria, each software was given a score from 1 to 4 (indicated by dots), based on how many subcategories it fulfilled for that criterion. These comparisons have limitations: they were carried out for a limited set of software and were done from the point of view of the design criteria used for BioImageXD. Other software may have fundamentally different design criteria. IPT, Image Processing Toolbox.

simulation tools and user-friendly support for high-throughput command pipelines, unlike many other open-source platforms. We also compared the performance of BioImageXD to that of other software for a specific practical example of image analysis.

Practical example: analyzing cell-surface receptors

As a proof of concept for BioImageXD, we undertook a challenging quantitative imaging study of integrin clustering, movement and internalization. Integrins anchor cells to their surroundings, and their intracellular domains are connected to the cytoskeleton, signaling networks and the endocytosis machinery[24]. Integrin α2β1, a human collagen receptor, forms clusters after binding to multivalent ligands, such as collagen fibrils[25]. Lateral movement of α2β1 on the cell surface may also contribute to collagen fibril organization[26]. Furthermore, α2β1 is often internalized upon binding to soluble multivalent ligands or antibodies[27]. Internalization of α2β1 seems to take place with a unique mechanism compared to other integrins, but the entry pathway is incompletely known[28].

We induced clustering of α2β1 integrin with fluorescently conjugated antibodies,

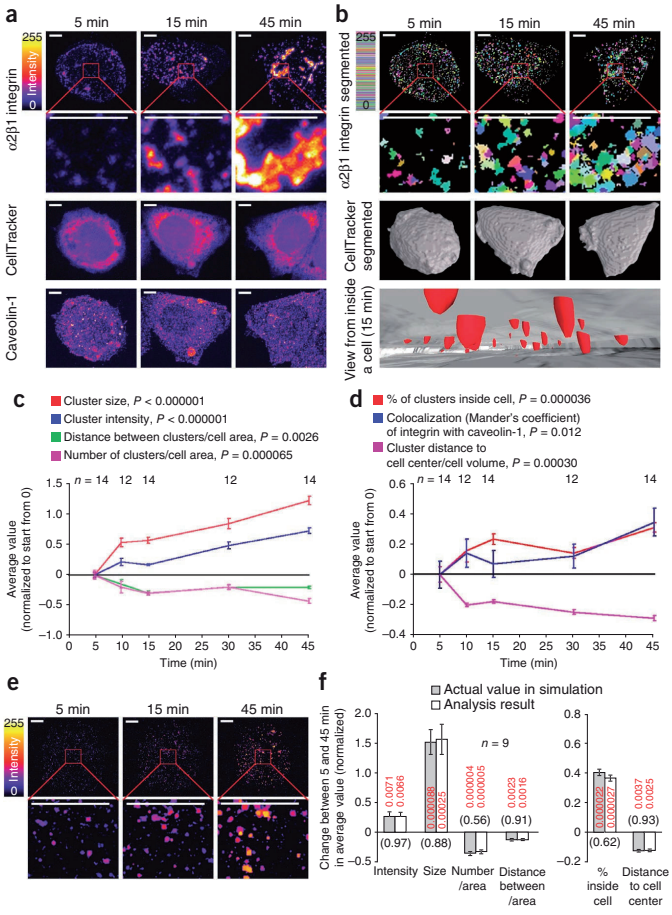


Figure 3 | Analyzing cell-surface receptor clustering and internalization. (**a**) Maximum intensity projections of confocal microscopy images of sample cells. Each data channel is shown separately, as indicated. (**b**) Maximum intensity projections of segmented integrin clusters (top two rows, color bar shows arbitrary colors assigned to segmented objects), 3D surface renderings of the cell membrane defined from CellTracker staining, smoothed with Gaussian and median filtering (middle), and 3D surface rendering from inside the cell, illustrating internalization of integrin clusters (red; bottom). (**c**) Average integrin cluster size, intensity, number and distance to each other. (**d**) Integrin internalization, colocalization with caveolin-1 and distance to cell center. In **c** and **d** all parameters were normalized to begin with 0, and *P* values are shown for changes between 5 min and 45 min. (**e**) Sample images of simulated integrin clustering used to validate the analyses. (**f**) Using simulated data, changes between 5 min and 45 min of the indicated parameters. *P* values in red are for change over time, and *P* values in parentheses indicate statistical significance between actual values and corresponding analysis results. Error bars, s.e.m. (*n* values indicate number of cells or simulations analyzed). Scale bars, 5 μm.

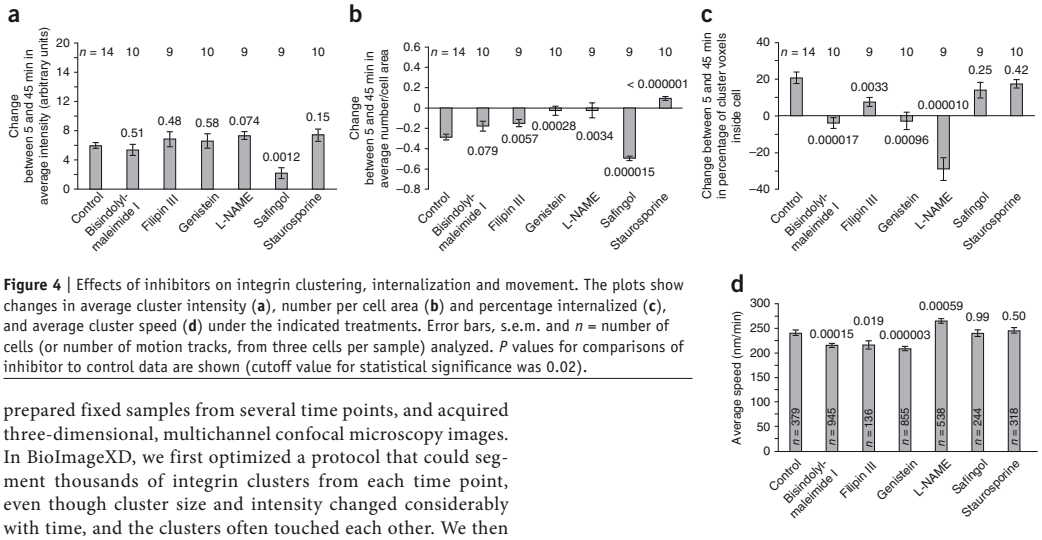


Figure 4 | Effects of inhibitors on integrin clustering, internalization and movement. The plots show changes in average cluster intensity (**a**), number per cell area (**b**) and percentage internalized (**c**), and average cluster speed (**d**) under the indicated treatments. Error bars, s.e.m. and *n* = number of cells (or number of motion tracks, from three cells per sample) analyzed. *P* values for comparisons of inhibitor to control data are shown (cutoff value for statistical significance was 0.02).

prepared fixed samples from several time points, and acquired three-dimensional, multichannel confocal microscopy images. In BioImageXD, we first optimized a protocol that could segment thousands of integrin clusters from each time point, even though cluster size and intensity changed considerably with time, and the clusters often touched each other. We then calculated four central parameters: size, intensity, number and distance between clusters. The latter two depended on cell size (**Supplementary Fig. 5**), so we developed an additional protocol for using a CellTracker fluorescent stain to calculate cell surface area. As expected, cluster size and intensity increased over time, whereas cluster number and distance between them decreased (**Fig. 3a–c**).

To analyze internalization of the clusters, we developed a protocol that, simply put, used the previously segmented cell membrane and calculated how many of the integrin clusters were inside it. Internalization increased over time (**Fig. 3d**). We confirmed the result in several ways. First, we quantified cluster distance to the center of the cell, normalized for cell volume. As internalization increased, this distance decreased. Second, we compared the results to differential staining before and after cell permeabilization, and examined the effect of different membrane definition thresholds, all supporting our analysis (**Supplementary Fig. 6**). Finally, as the internalization of $\alpha 2\beta 1$ integrin has been suggested to involve caveolin-1 (ref. 27), we included a third fluorescent marker for caveolin-1, and quantified the colocalization between it and $\alpha 2\beta 1$. As expected, the colocalization increased over time (**Fig. 3d**).

As there can be thousands of integrin clusters in a complex three-dimensional arrangement in every cell, checking analysis results manually, even for a single cell, is impossible. Therefore we created simulated images of integrin clustering and internalization with the simulation tools of BioImageXD and validated our analysis protocols with these data (**Fig. 3e,f** and **Supplementary Fig. 7**).

Using our validated analysis methods, we repeated all experiments with a selection of chemical inhibitors for various cellular signaling pathways (**Fig. 4a–c** and **Supplementary Figs. 8–10**).

Integrins move while clustering, and we analyzed this movement in living cells and tested the effects of the inhibitors. We applied tools for concatenating time series and registering them to compensate for focus drift and cell movement, and enhanced the BioImageXD motion tracking algorithm to be able to track clustering objects (**Fig. 4d** and **Supplementary Figs. 11** and **12**). We confirmed the tracking functionality by repeating all analyses with a segmentation protocol producing much higher cluster numbers, with consistent results. Again, we created simulations to validate the analysis (**Supplementary Fig. 13**).

We carried out all analyses with the BioImageXD batch processor and aggregated the results into a single matrix (**Supplementary Table 5**), which revealed that the clustering, internalization and lateral movement of integrins are regulated by different molecular mechanisms. Sample data and instructions are available for reproducing the analyses, which are widely applicable in biomedical research (**Supplementary Data 1** and **2**).

To attempt the analyses with other software packages, we listed the critical steps in performing the analyses and checked whether it was possible to perform them with the other software. If yes, we calculated sample results. Obtaining accurate results was difficult owing to a lack or inflexibility of features (**Supplementary Tables 6** and **7**), and we thus conclude that BioImageXD can be used to conduct complex analyses that might otherwise be considered impractical. However, we note that this comparison was limited to the features included in the tested packages and did not cover all possible plugins or features that might be available to users. Also, expert users of different software might be able to obtain better results than we did.

Discussion

The ethics of image processing is an important topic for the biological imaging community. With modern hardware and software, inaccurately manipulating image data is simple and tempting to the less scrupulous or poorly informed[5,6,29,30]. Openly described software implementations and full disclosure of the performed processing steps are probably the best ways to bring verifiability and transparency to the discipline[4]. Open-source projects have their challenges, such as maintaining continuity and rigorous organization, but these could be considered acceptable in order to achieve scientific reliability. Indeed, open-source programs

seem to be gaining popularity not only in science[31,32], but also in general, as exemplified by the Linux operating system[33]. To fully participate in the open-source bioimaging community, a software must implement open standards for image data and numerical result formats, and be interoperable with existing tools. BioImageXD currently leverages VTK[18] and ITK[19], and can read and write many image data formats, including the open standard OME-TIFF[20]. The BioImageXD consortium is open to collaborations, and we hope that in the future BioImageXD will be interoperable also with Java-based approaches and projects such as Icy[34], CellProfiler[35], ImageJ2 and Fiji[36].

Other developers may define the criteria for a bioimaging platform differently than we do, but we believe our criteria are reasonable. BioImageXD is among the most versatile single software packages in areas such as 3D visualization and animation, colocalization analysis and segmentation. It also has fairly unique features, such as its operational logic, reading of image data only on demand, simulated data generation and fully integrated support for batch processing. BioImageXD can serve as core software where the need is greatest, in complex and high-throughput applications. It already has a strong user base, with 400 downloads of beta versions per month.

In our application of BioImageXD to the analysis of integrin clustering, we showed that BioImageXD can deliver in practice. The extensive feature set of BioImageXD, coupled to its GUI design, enabled the creation of complex protocols by new combinations of existing tools. Its extendable, open-source architecture enabled modifications that could, at best, be developed to the point of use within mere hours. The new analyses, combined with traditional colocalization analysis, could be carried out virtually simultaneously for hundreds of files with pipelines in the BioImageXD batch processor. We validated all new analyses with simulated image data, also created with BioImageXD. Simulated data are useful for more than just method validation. In complex cases, numerical results can be hard to understand in a biological context. If a model can be constructed that simulates the experimental data, different parameters become comprehensible because their effects can be explored. It is recommendable for researchers working with complex analyses to always validate their results with such simulation tools. BioImageXD should enable biologists who are not experts in computer science to efficiently carry out and verify a wide range of scientific image-processing tasks.

Note: Supplementary information is available in the online version of the paper.

ACKNOWLEDGMENTS

We acknowledge all our former programmers, scientists and collaborators for their contributions to the development of BioImageXD. This research was funded by the Academy of Finland (projects 127168 and 114727), the FinNano nanoscience research programme, the Finnish Funding Agency for Technology and Innovation, the Sigrid Juselius Foundation, the National Doctoral Programme in Informational and Structural Biology (ISB), and the European Union 7th framework program (MetaFight). We thank the staff and users of the core imaging facilities who provided feedback and test data: Turku BioImaging and the Cell Imaging Core of the Turku Centre for Biotechnology (University of Turku and Åbo Akademi University), Jyväskylä Imaging Facility (University of Jyväskylä), Light Microscopy Facility of the Max Planck Institute of Molecular Cell Biology and Genetics, Dresden, and BioTec Zentrum, and Medical Theoretical Centre (Technical University of Dresden).

AUTHOR CONTRIBUTIONS

P.K. designed software and experiments, and wrote the manuscript; L.P. was head programmer; L.P. and J.P. programmed the software; P.K. and L.P. defined analysis protocols and did most analyses; S.T. and J.N. prepared samples, imaged and did some analyses; M.K. helped with software development; J.H. and V.M. designed scientific applications and software strategies; and D.J.W. supervised software development.

COMPETING FINANCIAL INTERESTS

The authors declare no competing financial interests.

Published online at http://www.nature.com/doifinder/10.1038/nmeth.2047. Reprints and permissions information is available online at http://www.nature.com/reprints/index.html.

1. Walter, T. *et al.* Visualization of image data from cells to organisms. *Nat. Methods* **7**, S26–S41 (2010).
2. Wilt, B.A. *et al.* Advances in light microscopy for neuroscience. *Annu. Rev. Neurosci.* **32**, 435–506 (2009).
3. Peng, H. Bioimage informatics: a new area of engineering biology. *Bioinformatics* **24**, 1827–1836 (2008).
4. Rueden, C.T. & Eliceiri, K.W. Visualization approaches for multidimensional biological image data. *Biotechniques* **43**, 31–36 (2007).
5. Rossner, M. & Yamada, K.M. What's in a picture? The temptation of image manipulation. *J. Cell Biol.* **166**, 11–15 (2004).
6. Anonymous. Microscopic marvels. *Nature* **459**, 615 (2009).
7. Bolte, S. & Cordelières, F.P. A guided tour into subcellular colocalization analysis in light microscopy. *J. Microsc.* **224**, 213–232 (2006).
8. French, A.P., Mills, S., Swarup, R., Bennett, M.J. & Pridmore, T.P. Colocalization of fluorescent markers in confocal microscope images of plant cells. *Nat. Protoc.* **3**, 619–628 (2008).
9. Costes, S.V. *et al.* Automatic and quantitative measurement of protein-protein colocalization in live cells. *Biophys. J.* **86**, 3993–4003 (2004).
10. van Steensel, B. *et al.* Partial colocalization of glucocorticoid and mineralocorticoid receptors in discrete compartments in nuclei of rat hippocampus neurons. *J. Cell Sci.* **109**, 787–792 (1996).
11. Fay, F.S., Taneja, K.L., Shenoy, S., Lifshitz, L. & Singer, R.H. Quantitative digital analysis of diffuse and concentrated nuclear distributions of nascent transcripts, SC35 and poly(A). *Exp. Cell Res.* **231**, 27–37 (1997).
12. O'Donoghue, S.I. *et al.* Visualizing biological data now and in the future. *Nat. Methods* **7**, S2–S4 (2010).
13. Pepperkok, R. & Ellenberg, J. High-throughput fluorescence microscopy for systems biology. *Nat. Rev. Mol. Cell Biol.* **7**, 690–696 (2006).
14. Neumann, B. *et al.* Phenotypic profiling of the human genome by time-lapse microscopy reveals cell division genes. *Nature* **464**, 721–727 (2010).
15. Jaensch, S., Decker, M., Hyman, A.A. & Myers, E.W. Automated tracking and analysis of centrosomes in early *Caenorhabditis elegans* embryos. *Bioinformatics* **26**, i13–i20 (2010).
16. Walsh, E.G. *et al.* High content analysis to determine cytotoxicity of the antimicrobial peptide, melittin and selected structural analogs. *Peptides* **32**, 1764–1773 (2011).
17. Eliceiri, K.W. & Rueden, C. Tools for visualizing multidimensional images from living specimens. *Photochem. Photobiol.* **81**, 1116–1122 (2005).
18. Schroeder, W., Martin, K. & Lorensen, B. *The Visualization Toolkit. An object-oriented approach to 3D graphics* (Kitware, Inc., 2006).
19. Yoo, T.S. *et al.* Engineering and algorithm design for an image processing API: a technical report on ITK—The Insight Toolkit. *Proc. of Medicine Meets Virtual Reality*. (ed. Westwood, J.) 586–592 (2002).
20. Linkert, M. *et al.* Metadata matters: access to image data in the real world. *J. Cell Biol.* **189**, 777–782 (2010).
21. Joshi, A. *et al.* Unified framework for development, deployment and robust testing of neuroimaging algorithms. *Neuroinformatics* **9**, 69–84 (2011).
22. Peng, H., Ruan, Z., Long, F., Simpson, J.K. & Myers, E.W. V3D enables real-time 3D visualization and quantitative analysis of large-scale biological image data sets. *Nat. Biotechnol.* **28**, 348–353 (2010).
23. Rueden, C., Eliceiri, K.W. & White, J.G. VisBio: a computational tool for visualization of multidimensional biological image data. *Traffic* **5**, 411–417 (2004).
24. Hynes, R.O. The extracellular matrix: not just pretty fibrils. *Science* **326**, 1216–1219 (2009).
25. Kawaguchi, S., Bergelson, J.M., Finberg, R.W. & Hemler, M.E. Integrin alpha 2 cytoplasmic domain deletion effects: loss of adhesive activity parallels ligand-independent recruitment into focal adhesions. *Mol. Biol. Cell* **5**, 977–988 (1994).
26. Meshel, A.S., Wei, Q., Adelstein, R.S. & Sheetz, M.P. Basic mechanism of three-dimensional collagen fibre transport by fibroblasts. *Nat. Cell Biol.* **7**, 157–164 (2005).
27. Upla, P. *et al.* Clustering induces a lateral redistribution of alpha 2 beta 1 integrin from membrane rafts to caveolae and subsequent protein kinase C-dependent internalization. *Mol. Biol. Cell* **15**, 625–636 (2004).

28. Karjalainen, M. *et al.* A Raft-derived, Pak1-regulated entry participates in alpha2beta1 integrin-dependent sorting to caveosomes. *Mol. Biol. Cell* **19**, 2857–2869 (2008).
29. Cromey, D.W. Avoiding twisted pixels: ethical guidelines for the appropriate use and manipulation of scientific digital images. *Sci. Eng. Ethics* **16**, 639–667 (2010).
30. Pearson, H. The good, the bad and the ugly. *Nature* **447**, 138–140 (2007).
31. Gordon, A. *et al.* Single-cell quantification of molecules and rates using open-source microscope-based cytometry. *Nat. Methods* **4**, 175–181 (2007).
32. Goldberg, I.G. *et al.* The Open Microscopy Environment (OME) data model and XML file: open tools for informatics and quantitative analysis in biological imaging. *Genome Biol.* **6**, R47 (2005).
33. Tchantchaleishvili, V. & Schmitto, J.D. Preparing a scientific manuscript in Linux: Today's possibilities and limitations. *BMC Res. Notes* **4**, 434 (2011).
34. de Chaumont, F. *et al.* Icy: an open bioimage informatics platform for extended reproducible research. *Nat. Methods* **9**, 690–696 (2011).
35. Lamprecht, M.R., Sabatini, D.M. & Carpenter, A.E. CellProfiler: free, versatile software for automated biological image analysis. *Biotechniques* **42**, 71–75 (2007).
36. Schindelin, J. *et al.* Fiji: an open-source platform for biological-image analysis. *Nat. Methods* **9**, 676–682 (2012).
37. Nymalm, Y. *et al.* Jararhagin-derived RKKH peptides induce structural changes in alpha1I domain of human integrin alpha1beta1. *J. Biol. Chem.* **279**, 7962–7970 (2004).

BioImageXD: open general purpose and high throughput image processing platform

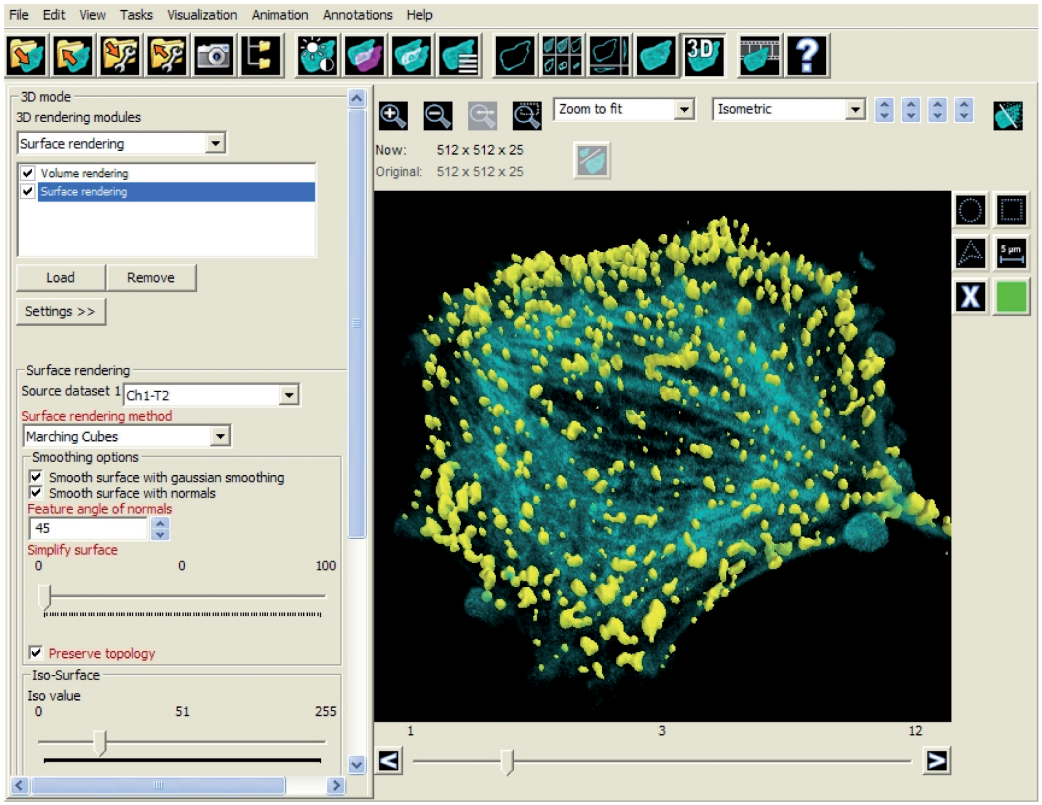
Pasi Kankaanpää, Lassi Paavolainen, Silja Tiitta, Mikko Karjalainen, Joacim Päivärinne, Jonna Nieminen, Varpu Marjomäki, Jyrki Heino, Daniel White

Supplementary figures, tables and methods

Contents

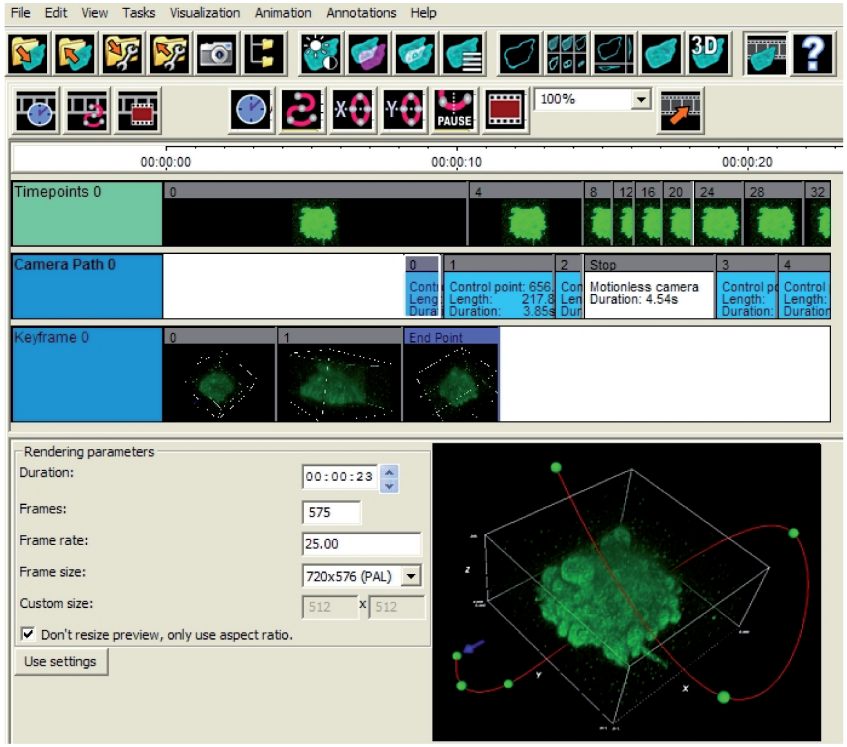
Note: There are in addition five supplementary videos and two sets of supplementary data with instructions.

Nature Methods: doi:10.1038/nmeth.2047



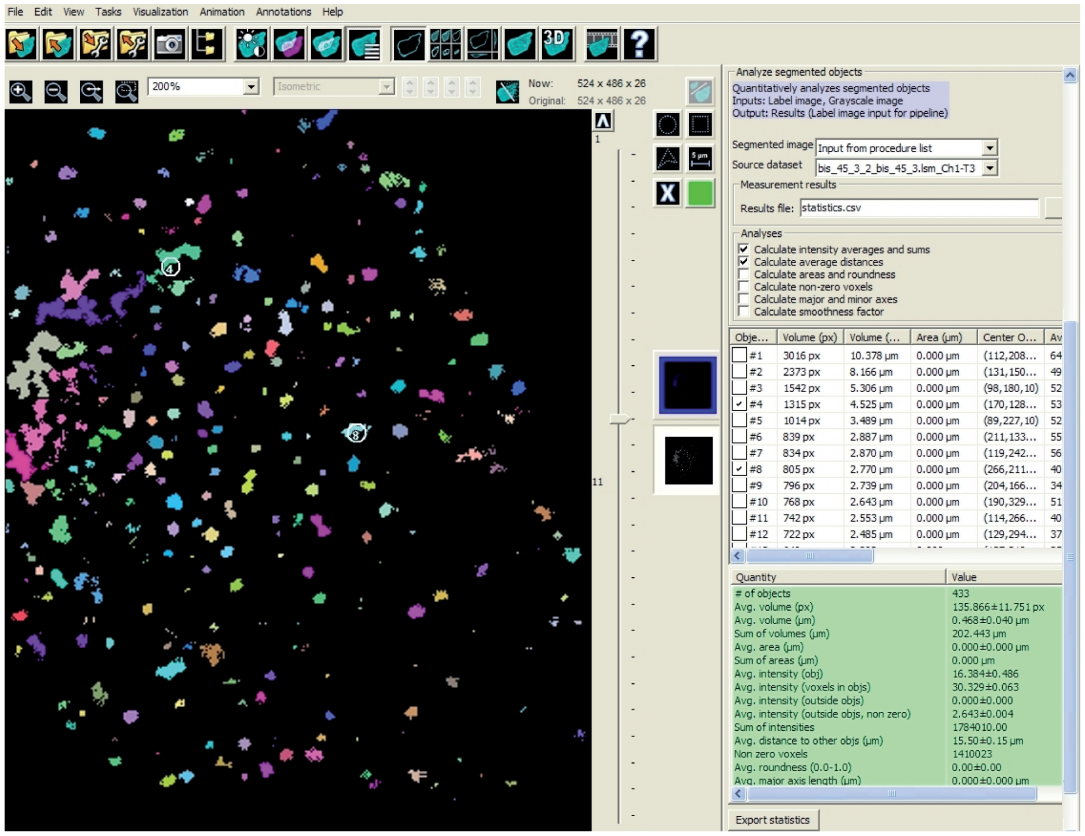
Supplementary Figure 1. Screen capture of 3D rendering with BioImageXD. In 3D mode several rendering modules can be active at the same time. Here one data channel is rendered as a translucent volume (cyan) while another channel is surface rendered (yellow). A list of the modules and their settings appears on the left. Elements in the 3D rendering can be panned, zoomed and rotated freely, often at interactive speeds. See also Supplementary Video 1.

Nature Methods: doi:10.1038/nmeth.2047



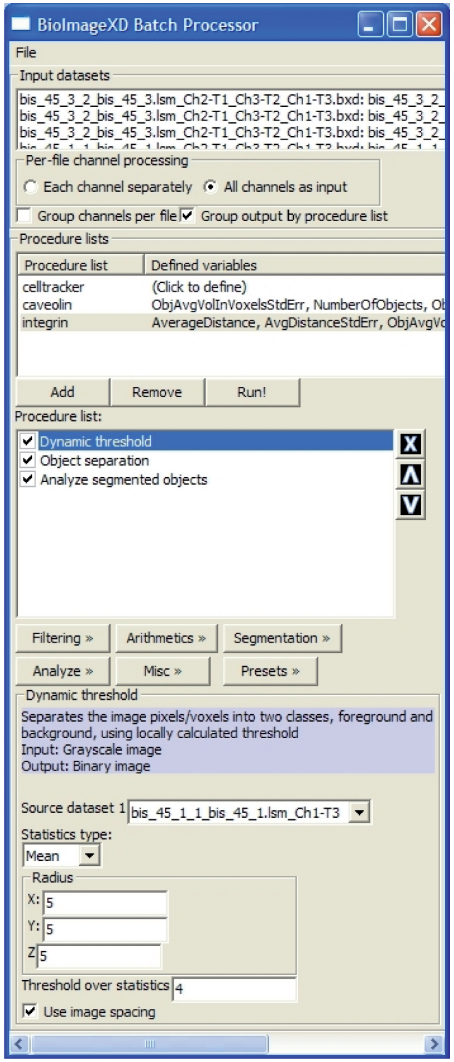
Supplementary Figure 2. Screen capture of animation creation with BioImageXD. The animator has adjustable tracks for timepoint changes, camera flight paths and keyframes. Here the animation begins with keyframe-coded movement and continues as a camera path as also timepoints start changing. When the timepoints change fast, the camera movement is temporarely paused. A sample rendering (bottom right) is used to set up keyframes and camera paths, showing here a camera path (red line), adjustable in 3D space from its control nodes (green spheres). The final animation is created automatically with the desired resolution and movie format, based on rendering settings set up in the 3D mode. See also Supplementary Videos 2 and 3.

Nature Methods: doi:10.1038/nmeth.2047



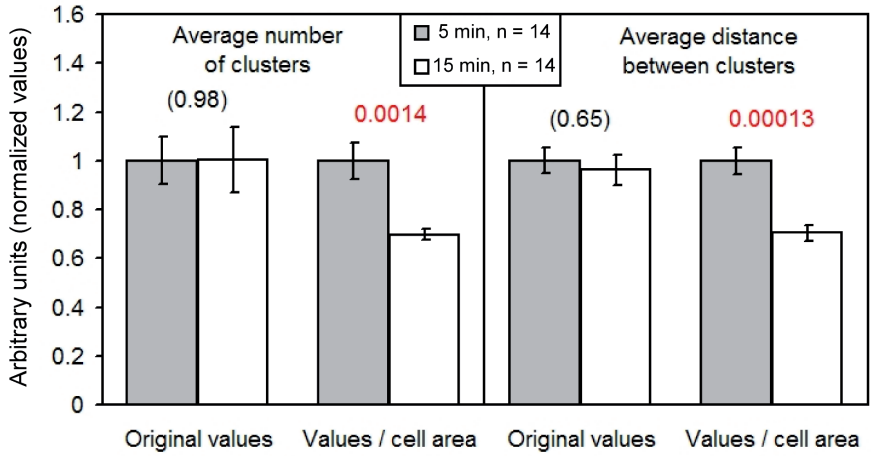
Supplementary Figure 3. Screen capture of segmentation with BioImageXD. Here the *Procedure list* task is used to segment α2β1 integrin clusters with dynamic thresholding and object separation. (Integrins are cell surface receptors studied in the practical analysis example of this paper.) The view panel on the left shows the result with each segmented object colored with a different arbitrary color. The task settings on the right show analysis results for each object separately, and averages for all objects (highlighted in green). Two objects have been selected from the list and are shown with circles in the image. See also Supplementary Video 4, and for trying out segmentation, see Supplementary Data 1.

Nature Methods: doi:10.1038/nmeth.2047



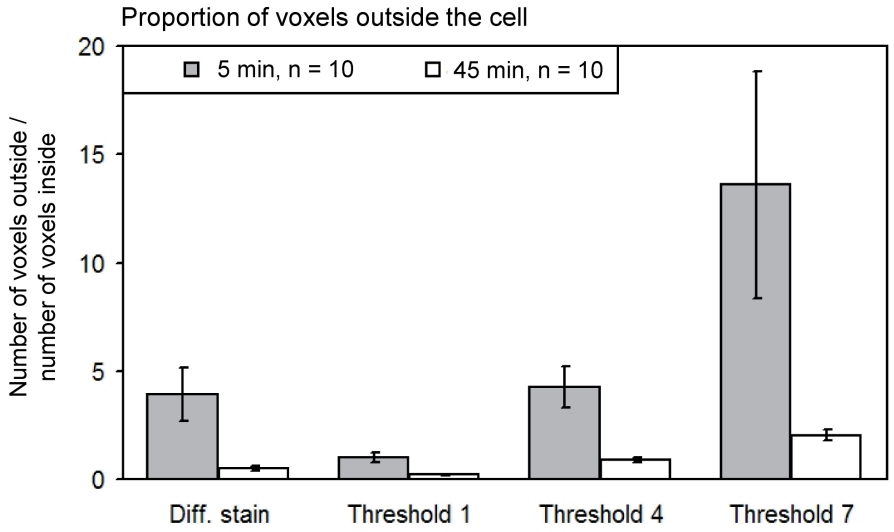
Supplementary Figure 4. Screen capture of the BioImageXD Batch Processor. The *Batch Processor* can process a very large number of files (in principle unlimited) in one go. It uses procedure lists created in the same way as in the *Procedure list* task (see Supplementary Video 4), but there can be several of them. Here there are three, one for each data channel. The list for integrin channel has been selected, and its contents are shown (it is similar to the list used in Supplementary Video 4). The settings for every procedure in the *Procedure list* task and the *Batch Processor* contain a simple description with input and output specification. These appear highlighted in light blue, as shown here for the selected *Dynamic threshold* procedure. When the batch processing is run, **csv** files are automatically created that contain aggregated results of selected variables. For trying out batch processing, see Supplementary Data 1.

Nature Methods: doi:10.1038/nmeth.2047



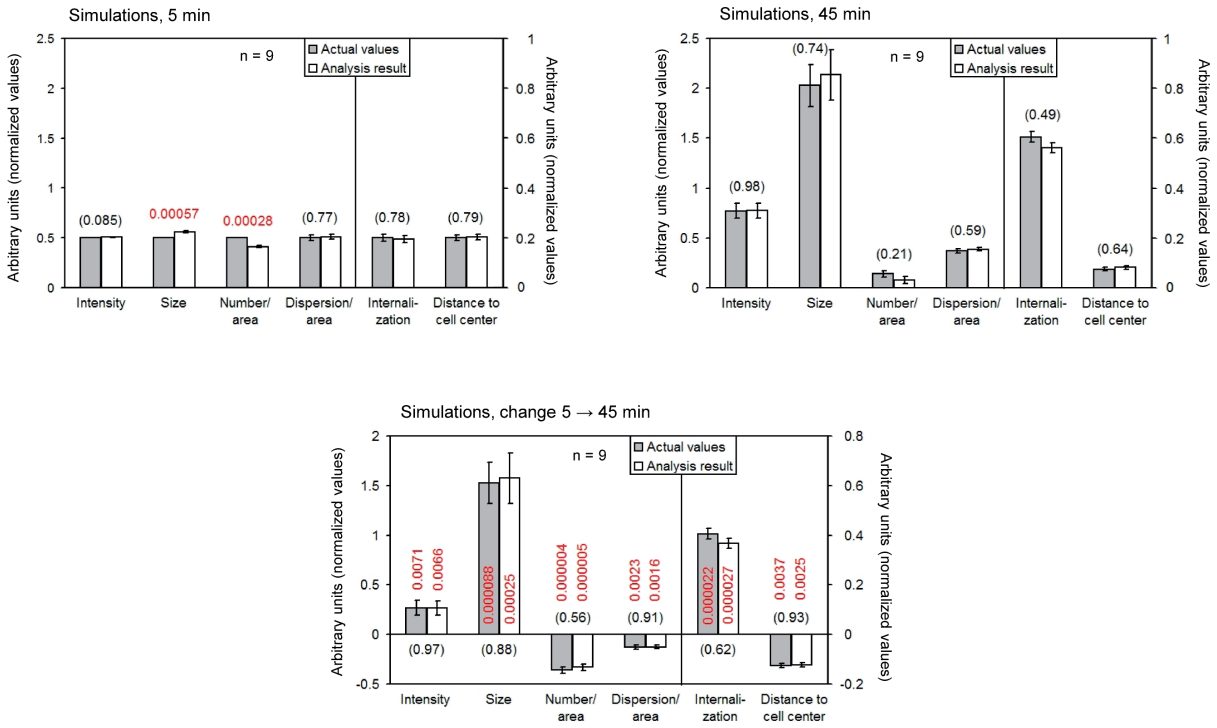
Supplementary Figure 5. The importance of considering cell area in the analysis of cell surface objects from 3D cell images. Because of the variations in cell size and shape that normally occur, analyzing features such as the number or dispersion of cell surface objects can produce misleading results. Differences between samples can erratically appear to be in the wrong direction or be non-existent even with fairly large sample sizes, as in this example, where 5 and 15 minute timepoints of α2β1 integrin clustering were compared. (Integrins are cell surface receptors studied in the practical analysis example of this paper, they form clusters as part of their normal functioning.) Based on the images, a difference between the timepoints was expected for both integrin cluster number and distance between clusters, but none was detected when analyzing the original values. After segmenting the cell membrane in 3D, calculating the surface area of the cell and dividing the original values with it, significant differences could be observed as expected: as integrins cluster together, the number of clusters decreases, as does the average distance between them. (Variation within samples also decreased especially in the later timepoint, as indicated by the smaller error bars.) BioImageXD features tools that make the somewhat complicated task of cell area calculation and its consideration in analyses straightforward. Values in the graph have been normalized. Error bars show standard errors, and p-values for the differences between timepoints are shown above bars, with statistically significant values in red and non-significant values in parentheses.

Nature Methods: doi:10.1038/nmeth.2047



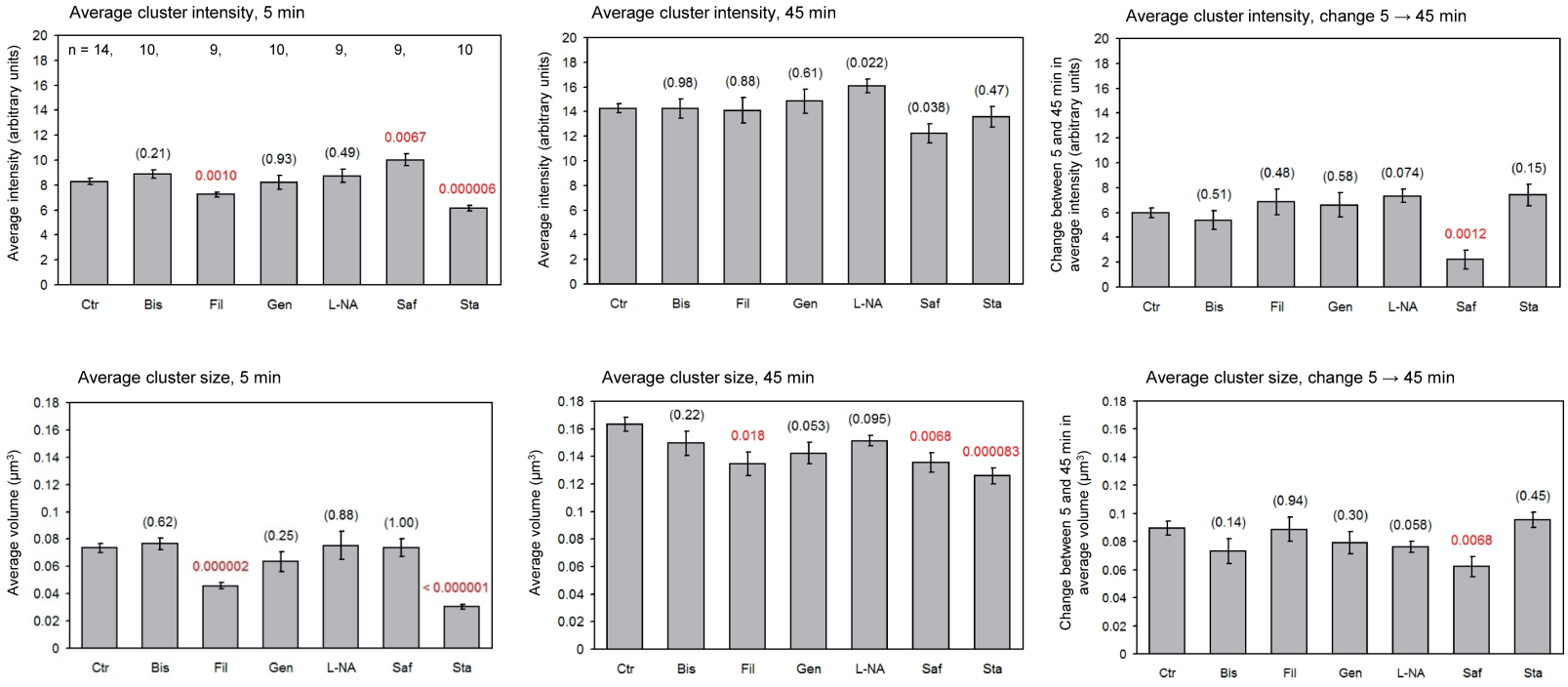
Supplementary Figure 6. Confirming the results of the internalization analysis. Because the analysis of internalization based on the definition of the cell membrane from the fairly low resolution images of a light microscope can only be an approximation, the reliability of the analysis was confirmed in numerous ways. One of these was to create samples that were labeled for integrins and the cell as usual, but integrins were labeled twice, before and after cell permeabilization, with different fluorophores (differential staining). A simple algorithm developed into the BioImageXD colocalization tool was then used to calculate the proportion of voxels that had not been internalized (bars on the left in the graph). As expected, a clear decrease in outside voxels was observed as the integrin clusters internalize over time. The result was compared to the internalization analysis based on cell membrane definition (other bars in the graph). Another reliability control was analyzing the effect of varying a manual threshold value, which is a critical part of the cell membrane segmentation procedure. As the graphs show, different thresholds produced results proportionately similar to each other and the differential staining, indicating that the internalization analysis was working reliably, and it was not sensitive to the threshold used. Error bars show standard errors.

Nature Methods: doi:10.1038/nmeth.2047



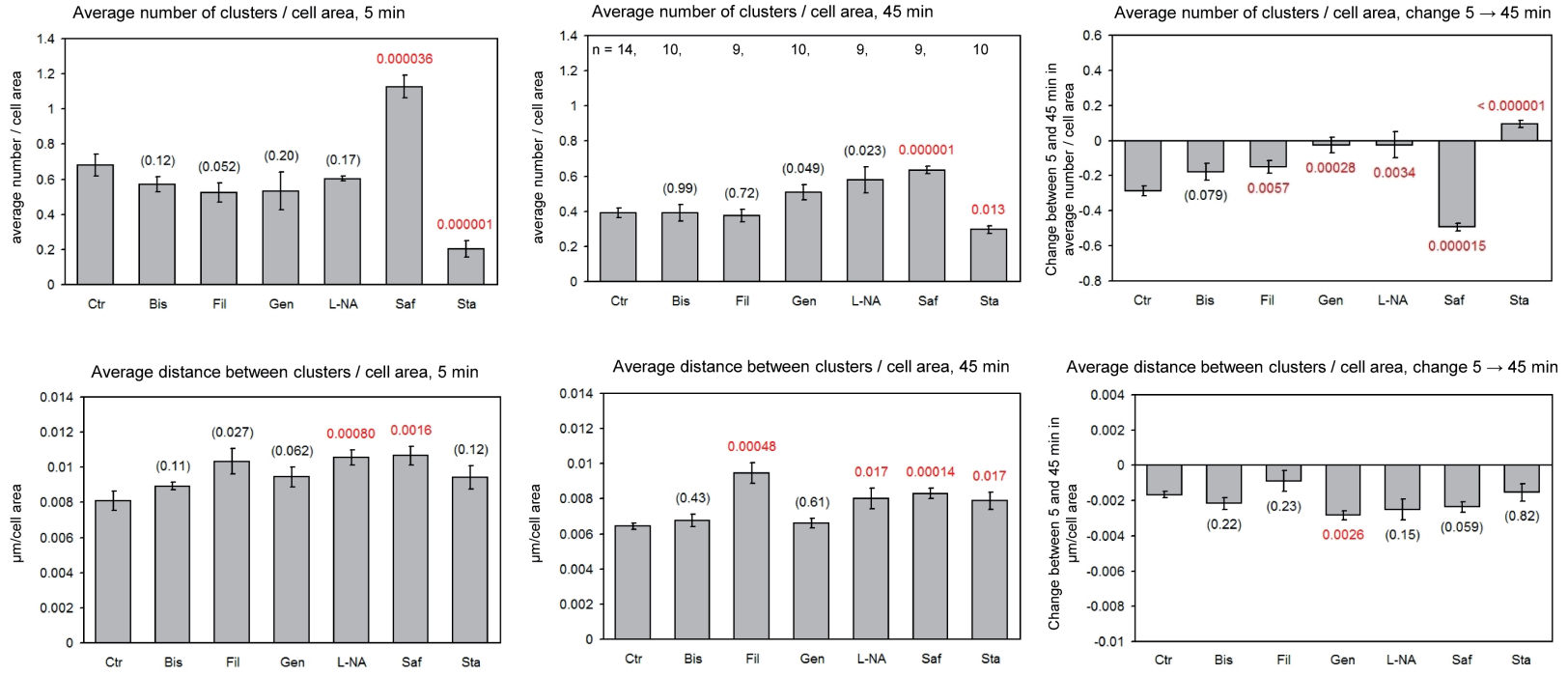
Supplementary Figure 7. Validation of fixed sample analyses with simulations. Error bars show standard errors, and n = number of simulations analyzed. For each bar, p-values of comparing inhibitor to control are shown, with statistically significant values in red and non-significant values in parentheses. All values have been normalized in order for the graphs to be comparable to Figure 3 (c) and (d). The changes in all parameters in the simulations were similar to changes in the actual samples and statistically significant, p-values in red. There were no significant differences between the analysis results of the simulated data and the actual values of the simulated data, p-values in parentheses, with the exception of size and number at 5 min (this was only due to all simulations having the same average, resulting in 0 variance). All three graphs have the same scale range in the Y-axis.

Nature Methods: doi:10.1038/nmeth.2047



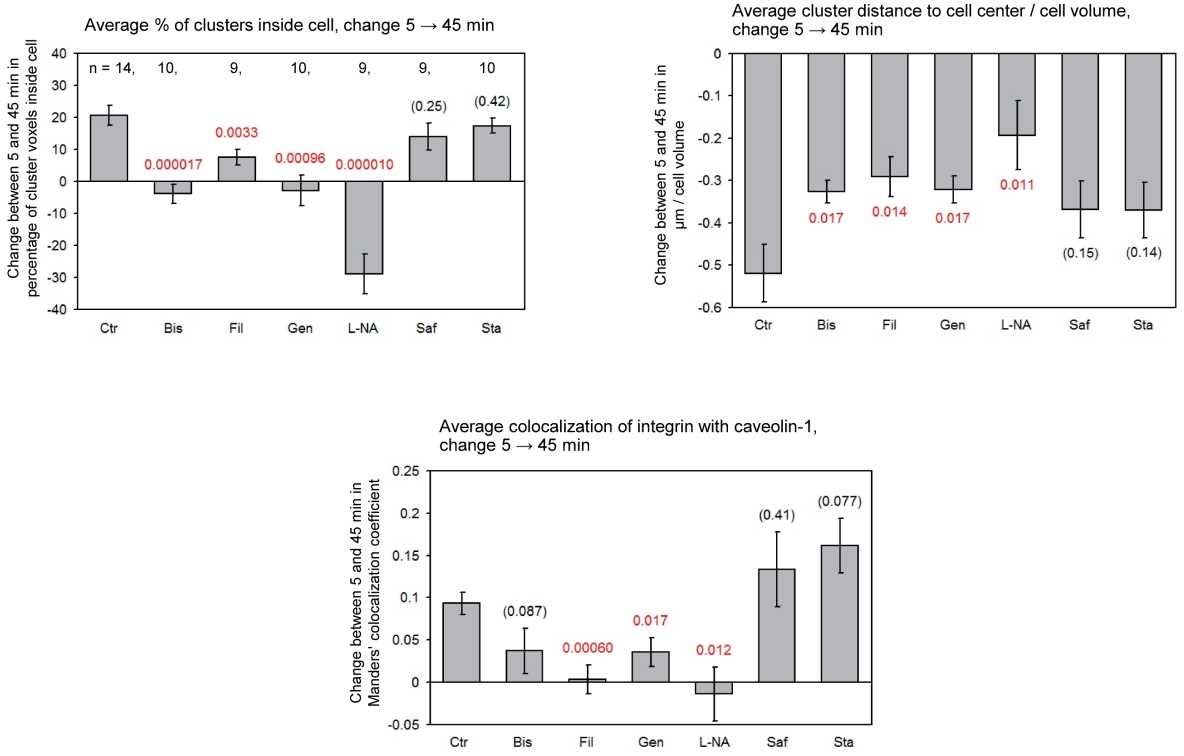
Supplementary Figure 8. Effects of inhibitors on integrin cluster intensity and size. Samples: Ctr = Control, Bis = Bisindolylmaleimide I, Fil = Filipin III, Gen = Genistein, L-NA = L-NAME, Saf = Safingol, Sta = Staurosporine. Error bars show standard errors, and n = number of cells analyzed (same for all graphs). For each bar, p-values of comparing inhibitor to control are shown, with statistically significant values in red and non-significant values in parentheses. All three graphs showing the same variable have the same scale range in the Y-axis.

Nature Methods: doi:10.1038/nmeth.2047



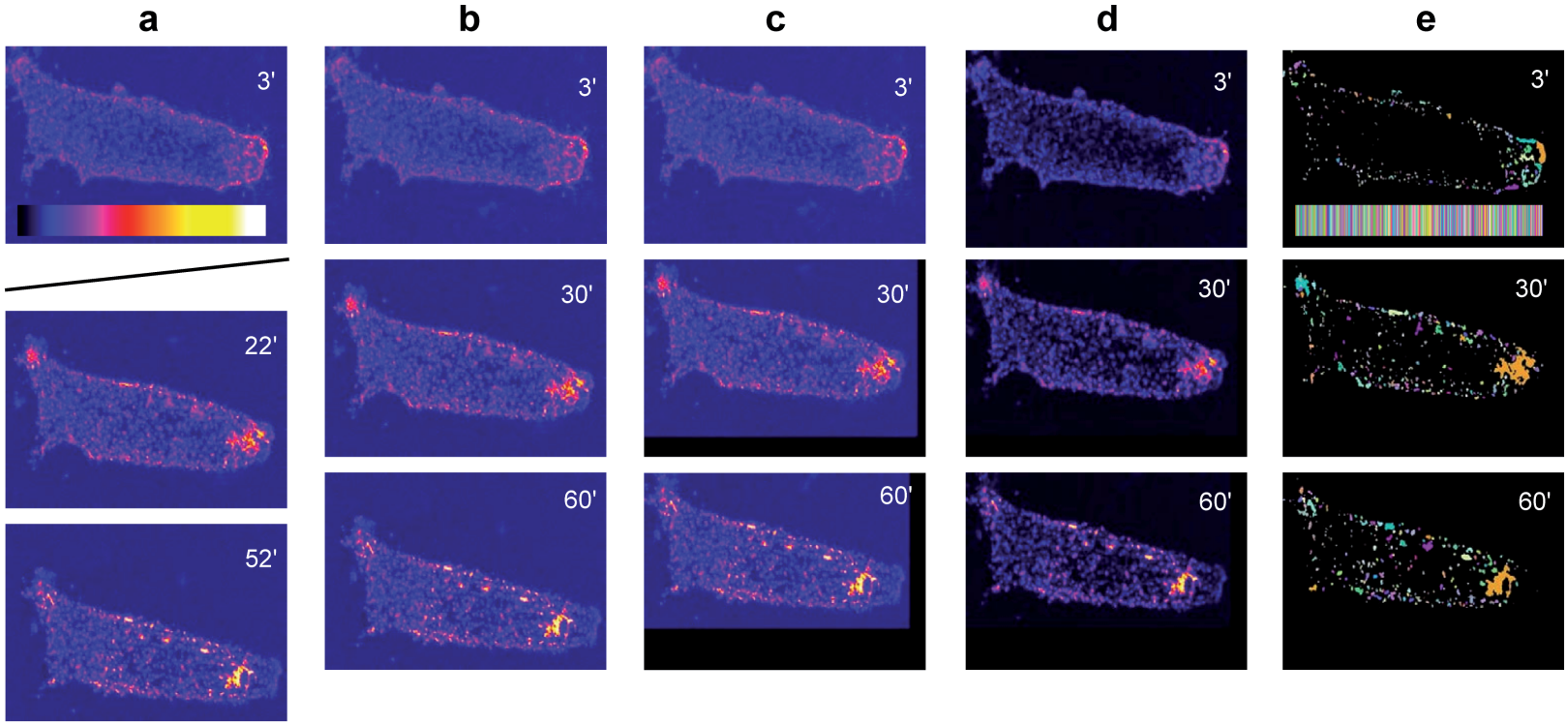
Supplementary Figure 9. Effects of inhibitors on integrin cluster number and dispersion. Samples: Ctr = Control, Bis = Bisindolylmaleimide I, Fil = Filipin III, Gen = Genistein, L-NA = L-NAME, Saf = Safingol, Sta = Staurosporine. Error bars show standard errors, and n = number of cells analyzed (same for all graphs). For each bar, p-values of comparing inhibitor to control are shown, with statistically significant values in red and non-significant values in parentheses. All three graphs showing the same variable have the same scale range in the Y-axis.

Nature Methods: doi:10.1038/nmeth.2047



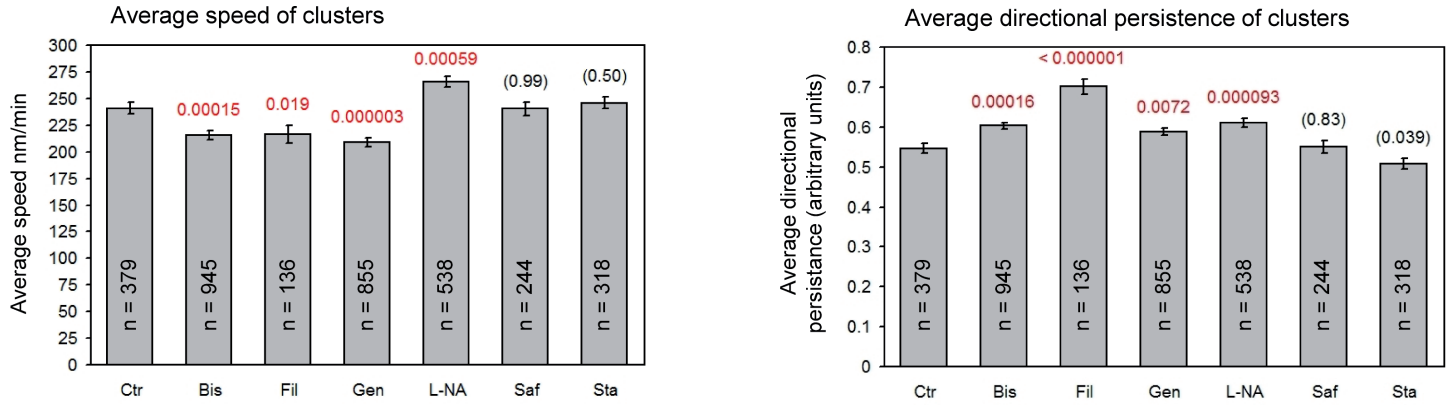
Supplementary Figure 10. Effects of inhibitors on integrin cluster internalization and colocalization with caveolin-1. Samples: Ctr = Control, Bis = Bisindolylmaleimide I, Fil = Filipin III, Gen = Genistein, L-NA = L-NAME, Saf = Safingol, Sta = Staurosporine. Error bars show standard errors, and n = number of cells analyzed (same for all graphs). For each bar, p-values of comparing inhibitor to control are shown, with statistically significant values in red and non-significant values in parentheses.

Nature Methods: doi:10.1038/nmeth.2047



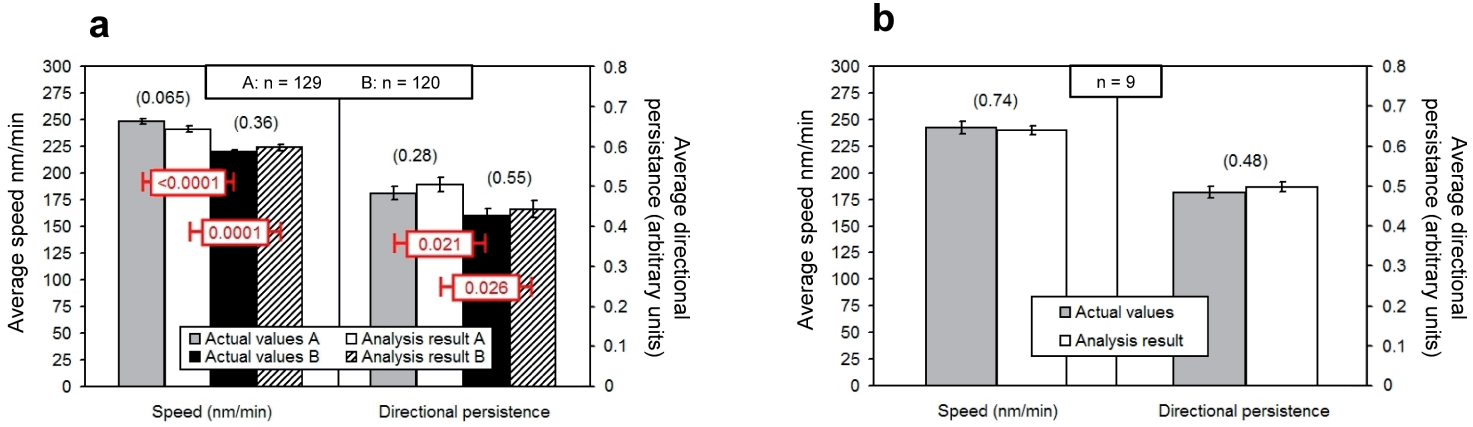
Supplementary Figure 11. Illustration of the processing steps of live cell data. 3 timepoints are shown from a time series of 21, each a maximum intensity projection of a 3D image. (**a**) The original series is in two different datasets, because due to excessive focal drift the image acquisition had to be stopped in between and restarted with different settings. The first processing step is to combine the two datasets together, so that the time stamps are also correct. (**b**) The time series is now one continuous dataset, and the next step is to compensate for the remaining thermal and focal drift with an automatic registration algorithm. (**c**) The cell is now in the same location within the image volume throughout the time series. Next, noise is removed and background subtracted. In addition, an intensity transfer function could be interpolated at this stage to compensate for unwanted intensity variations such as bleaching. (**d**) The time series is ready for analysis. (**e**) The time series is segmented and information of the hundreds of individual objects is available. Motion tracking for the objects can now be performed.

Nature Methods: doi:10.1038/nmeth.2047



Supplementary Figure 12. Effects of inhibitors on integrin cluster speed and directional persistence.
Analysis results of live cell data. Samples: Ctr = Control, Bis = Bisindolylmaleimide I, Fil = Filipin III, Gen = Genistein, L-NA = L-NAME, Saf = Safingol, Sta = Staurosporine. For each bar, p-values of comparing inhibitor to control are shown, with statistically significant values in red and non-significant values in parentheses. Error bars show standard errors, and n = number of motion tracks analyzed (from 3 cells per sample). See Supplementary Video 5 for a sample video of motion tracking.

Nature Methods: doi:10.1038/nmeth.2047



Supplementary Figure 13. Validation of live cell analyses with simulations. Simulations of live cell data. (**a**) Two simulations compared, with the differences between them in both parameters being similar to differences between actual samples, and statistically significant, p-values in red. There were no significant differences between the analysis results of the simulated data and the actual values of the simulated data, p-values in parentheses. Error bars show standard errors, and n = number of motion tracks analyzed. (**b**) 9 different simulations (= n) analyzed together, confirming that no significant differences between the actual values and the analysis results could be detected (p-values shown in parentheses).

Nature Methods: doi:10.1038/nmeth.2047

Supplementary Table 1. BioImageXD features (page 1/3). An overview of the main features of BioImageXD.

1/6

General
- Open source code
- Free
- Supports both 32-bit and 64-bit operating systems: Win, Mac OSX, Linux
- Single installable package
- 3D timelapse support in all functionality
- Features extensively adjustable
- Image data read only on demand (not upon file loading)
- Extendable program architecture, supports selected libraries (VTK, ITK)

Graphical user interface
- Single large window with re-sizeable panes/parts that are also switchable visible/invisible
- Main panes/parts: main tool bar, file tree, view panel (maximizable), dataset info, task panel
- Other panes/parts: view panel top toolbar, view panel side toolbar, view panel configuration, histograms, Python shell
- Color coding (expert features & settings are colored red)
- Short explanation texts always on screen for most features

File input/output features
- Internal file tree (switchable visible/invisible, contents savable for next program start)
- Open dataset (supports 10 common bioimaging file formats)
- Save dataset (supports BioImageXD file format; timepoints selectable upon saving)
- Open/save settings
- Save snapshot image (supports 3 common image file formats)
- Open/save 3D view scene
- Import/export images (supports 5 common image file formats)

View panel features
- Operates in any of 5 visualization modes: Slices, Gallery, Orthogonal, Projection, 3D
- Sliders for choosing timepoints/z-slices when applicable

2/6

- Zoom in/out/to fit/to user-selected region
- View angle/axis selection/rotation buttons (in 3D mode)
- Changeable interaction modes (trackball/joystick/camera/actor) (in 3D mode)
- Interpolation method adjustable (4 options)
- Displays current/original dataset dimensions
- Hold-down button for switching between original/processed dataset
- Button for switching between resized/original dataset
- Eyedropper tool for retrieving pixel/voxel information
- Regions of interest (ROIs) can be drawn with 3 tools: circle, rectangle, polygon
- Scalebar drawing
- Annotations (such as ROIs) can be deleted, moved, resized and their color changed
- Convert ROI to mask / mask selection

Slices visualization mode
- Displays chosen slice

Gallery visualization mode
- Displays all slices of chosen timepoint or all timepoints of chosen slice as small images

Orthogonal visualization mode
- Displays orthogonal sections (positions selectable) through the dataset along the three spatial axes (X, Y, Z)

Projection visualization mode
- Displays either a maximum intensity projection or an average intensity projection

3D visualization mode
- Displays 3D renderings interactively, can be rotated, panned and zoomed with mouse
- General settings for lighting, background color and view angle
- 3D mode settings can be saved/opened

Nature Methods: doi:10.1038/nmeth.2047

Supplementary Table 1. BioImageXD features (page 2/3). An overview of the main features of BioImageXD.

3/6

- Renderings are created by 11 different modules that can be loaded/unloaded and switched on/off
- Several rendering modules can be combined, each rendering also different data channels
- Angle measurement module (for measuring/drawing angles in 3D with mouse)
- Axes module (draws X/Y/Z axes, with adjustable lengths)
- Clipping box module (clips all or selected modules inside/outside defined box)
- Clipping plane module (clips all or selected modules on one side of defined plane)
- Distance measurement module (for measuring/drawing distances in 3D with mouse)
- Orthogonal slices module (displays adjustable orthogonal slices along all three spatial axes)
- Surface rendering module (creates one or several surfaces at selectable iso-values, adjustable surface transparency and intensity, 4 different surface rendering methods, several smoothing and simplifying options)
- Visualize protein databank file module (input selection for PDB file to visualize, adjustable tube & sphere radii)
- Visualize motion tracks module (input selection for tracks file to visualize, tracks can be individually chosen from a list and visualized with same/actual starting points, minimum track length specifiable, adjustable tube & sphere radii)
- Volume rendering module (graphical color and opacity transfer function editor (RGBA) with feehand/node adjustment and opening/saving of look-up-tables, 4 different volume rendering methods, adjustable interpolation and rendering quality, shading)
- Warp scalar module (displays 2D slice as 3D map, selectable slice, adjustable scale factor and adjustable surface smoothing)

Adjust task
- Graphical intensity transfer function, adjustable timepointwise
- Controllers for brightness, contrast, gamma, min/max values & thresholds, smooth start/end

4/6

- Interpolation with up to 5 specifiable timepoints, for bleaching correction etc.

Merge task
- Combines gray scale datasets into an RGB color image (which can subsequnetly be volume rendered in 3D mode)
- Each channel adjustable with graphical intensity transfer function (same as in Adjust task)
- Alpha channel construction method selectable (3 options)

Colocalization task
- Analyzes and visualizes colocalization between two channels
- Both manual and automatic threshold determination
- 2D histogram (savable as image)
- 3 methods for P-value calculation
- Calculates approximately 40 colocalization parameters (savable as CSV file)

Procedure list task
- Creates command pipelines (savable as presets) from approximately 90 different available procedures (each fully adjustable by clicking them)
- To help assembling procedure lists, each procedure has on-screen explanation text with acceptable inputs and outputs
- Outputs both images and analysis results (latter exportable as CSV files)
- Noise removal (Gaussian smooth, Gradient anisotropic diffusion, Median, Anisotropic diffusion, Hybrid median 2D, Mean)
- Iterative deconvolution and PSF generation
- Feature detection (Gradient magnitude, Find local maxima, Sobel, Difference of Gaussians, Canny edge detection, Sobel edge detection, City block distance, Danielsson distance map, Euclidean distance)
- Morphological operations (Dilate, Erode, Variance, Range, Binary thinning, Grayscale fill hole)
- Fourier transforms (Butterworth low pass, Inverse FFT, Ideal low pass, Ideal high pass, Butterworth high pass, FFT)
- Non-uniform illumination correction and Sigmoid filter
- Image arithmetics (Subtract, Add, Divide, Multiply, Sin, Cos, Exp, Log, Sqrt, Invert, Shift and scale)

Nature Methods: doi:10.1038/nmeth.2047

Supplementary Table 1. BioImageXD features (page 3/3). An overview of the main features of BioImageXD.

5/6

- Logical operations (And, Or, Xor, Not, Nor, Nand)
- Automatic background subtraction (3 methods)
- Thresholding (Threshold for maximum object number, Otsu threshold, Otsu multiple thresholds, Dynamic threshold, Threshold)
- Morphological watershed segmentation
- Region growing (Confidence connected threshold, Connected threshold, Neighborhood connected threshold)
- Active contours
- Segmented object processing (Re-label image, Connected component labeling, Object separation, Filter objects)
- Segmentation-based analyses (Analyze polydata, Analyze motion tracks, Analyze segmented objects), calculate in total approx. 70 parameters
- Voxel-based analyses (FRAP analysis, Timepoint correlation, Analyze region of interest), calculate in total approx. 20 parameters
- Colocalization analyses (Calculate thresholds for colocalization, Analyze colocalization, Object colocalization), calculate in total approx. 55 parameters
- Region of interest processing (Extract a subset, Subtract ROI, Use ROI mask)
- Simulations (4D particle simulations)
- Registration (Translation registration, Versor rigid registration, Rigid registration 2D, Deformable registration, Translation registration 2D, Multi resolution translation registration, Slices translation registration 2D)
- Tracking (Create motion tracks, Visualize motion tracks 2D)
- Dataset conversions (Combine time series, Convert data type, Convert to polygonal data, Combine images, Extract components, Combine different intensity range images)

Batch Processor

- Processes all selected datatsets, outputs both images and analysis results
- Supports multiple procedure lists (either channel specific or for all channels)
- Aggregates selected results (of approximately 120 available parameters) into spreadsheet format (CSV file)
- Presets from Procedure list task available
- Analysis settings can be saved/opened

6/6

Animator

- Creates movies of the scene rendered in 3D mode
- Adjustable frame size, frame rate and duration
- Two combinable animation techniques: keyframe and camera path (can be set to share a frame for smooth transition between techniques)
- Animation movements adjusted in 3D preview window
- Camera path features: free motion path, circular paths and movement pause
- Video-editor type interface based 3 types of tracks (timepoint changes, keyframes, camera path nodes)
- Tracks items can be resized and moved individually or in batches from menus
- Output as series of still images (5 formats) and/or videos (8 formats) with adjustable quality

Other

- Preferences dialogue (for setting up default paths, performance options, automatic resampling etc.)
- View panel updating switchable on/off/automatic
- Editor for look-up-tables (palettes) with freehand/node editing of RGB color transfer function and saving/opening of palettes
- Resize dataset (in X/Y/Z; one button switching original/resized)
- Convert dataset to 8-bit (with adjustable mapping of values)
- Previous path locations are remembered
- Mouse tooltips
- Getting Started Guide (viewable from software)
- Automatic bug reporting feature (activates after program crash)
- Manual bug reporting feature (activatable by user at any time)

Nature Methods: doi:10.1038/nmeth.2047

Supplementary Table 2. Protocol for comparing BioImageXD to other software (page 1/3).
(See Supplementary Tables 3 and 4 for comparison results).

The protocol on the following pages was used to compare BioImageXD to other software. Due to the large number of programs available, it is impossible to compare them all. A representative set was therefore chosen, covering popular alternatives of both open source and proprietary approaches, and both larger and smaller programs, and programs supplied with microscopes as well as stand-alone packages: BioImage Suite, ImageJ (Fiji), Vaa3D, VisBio, MATLAB + Image Processing Toolbox, Imaris (Bitplane), Fluoview (Olympus), Volocity (PerkinElmer), and ZEN Lite (Zeiss). It should be noted that MATLAB has a different philosophy from all the other software packages, and its use requires programming skills. It was nonetheless included in the comparisons to make them as a whole more representative of the different approaches available.

Different bioimaging programs are made for somewhat different purposes and have very different sets of tools. Therefore a detailed comparison between them can be difficult, as there can be very few tools or settings that are directly comparable. Different programs are also made with different philosophies and goals, the differences being perhaps largest between open source and proprietary software. Open source programs are often academic, aimed at specific scientific problem solving and openness, whereas proprietary programs aspire to be slick and easy-to-use commercial products. There are also notable differences between open source programs. For instance, ImageJ is a generic platform for processing individual images, with an easy plugin architecture resulting in many tools developed by its very large user community, whereas BioImageXD is a single package for processing multidimensional images with an integrated, large set of validated tools based on VTK and ITK.

Comparing BioImageXD to other programs was done from the point of view of the six criteria defined for bioimaging software in this paper. It is intended to illustrate differences between the programs from this point of view, not to function as an exhaustive review. Some of the criteria are more suitable for open source software, but proprietary software are also widely used within the scientific community, and have therefore been compared using the same criteria, to provide some common ground for comparisons.

The comparison results for open source software are shown in Supplementary Table 3, and those for proprietary software in Supplementary Table 4. They are approximations and do not show that one program would be better than another in all situations. Rather, they illustrate how BioImageXD compares to other software in terms of criteria that the authors believe to be generally advantageous for scientific bioimaging.

Also see Supplementary Tables 6 and 7.

Supplementary Table 2. Protocol for comparing BioImageXD to other software (page 2/3).
(See Supplementary Tables 3 and 4 for comparison results).

1/4

1. ***Qualitatively briefly describe the following software features, in each case comparing BioImageXD and the other program in question. When assessing numbers of features characterize them on a scale: none, one, some (<5), many (>5).***

Open
- Source code availability (is the source code open)
- Clarity of algorithms and other procedures (is it clear what they do, can the exact mathematics behind them be inspected)
- Background processing (are there indications of undisclosed background processing or other processing not controllable by the user, for instance resampling or smoothing of data to improve rendering or processing speed)
- Documentation available (user guides or similar, documentation of the code itself, if available)

Extensive
- Basic image processing possibilities (how many methods in resampling, rescaling, basic mathematical and logical processing, region of interest processing, channel merging etc. and what is the execution of the intensity transfer function)
- Noise reduction possibilities (how many different methods)
- Colocalization visualization and analysis tools (how comprehensive, also statistics)
- 3D visualization (how many visualization methods or modules and their combinability, how can images be interacted with, are there both volume and surface

2/4

rendering methods, is rendering quality full and/or controllable)
- Possibilities for creating animations and movies (what animation techniques are available, how are they controlled and how versatile results can be created with them)
- Segmentation into objects (how many methods, are they available in all basic classes: thresholding, watershed, region growing)
- Motion tracking possibilities (is 3D object tracking possible, what is the execution and its adjustability)

Usable
- Price of the program (free or $ hundreds/ thousands)
- Supported operating systems
- Program structure from the point of view of installation (single package or several modules, how are they acquired and installed)
- Basic design of the graphical user interface (single window or several windows etc.)

Adjustable
- Feature adjustability (in general throughout the software; are there features with limited adjustability, such as rendering, segmentation or tracking)
- Possibility to create lists of commands (are there any possibilities to combine commands into procedure lists and re-use such lists, do such possibilities cover the entire software or only limited parts, and are they usable without programming skills)

Nature Methods: doi:10.1038/nmeth.2047

Supplementary Table 2. Protocol for comparing BioImageXD to other software (page 3/3).
(See Supplementary Tables 3 and 4 for comparison results).

3/4

Applicable
- 3D time lapse support (does the program fully support multi-channel timelapse 3D data in all functionality, is this built in into the core architecture or only obtainable in some features or modules)
- File handling (is it possible to load several files simultaneously into an internal file tree or the like, is image data read into memory directly or on demand, channel/file format/metadata support, saving and opening data and settings)
- High throughput possibilities (is it doable through the user interface on the level of a single dataset, is there some kind of graphical interface for batch processing, or is high throughput only obtainable through programming, are all or only some software features covered, does file and memory handling support high throughput)
- Validation tools (are there methods for validating algorithms and other results, for instance by creating realistic simulated data)

Extendable
- Software architecture from the point of view of extendibility (is there a plugin architecture or similar, are libraries supported that make extending easier, what are the general possibilities for modifying and extending the program for varying scientific needs)
- Management & support (is the project well managed in a way that supports extendibility, are there ways for users and especially developers to get support, are bugs managed in some organized manner)

4/4

- Ongoing development (is the project actively developed, facilitating extendibility, and from what point of view, and what are the main characteristics of the software and is there anything else noteworthy about future plans or otherwise)

2. ***Measure the time it takes to carry out the following procedures, categorizing the results as follows: instant (< 2 seconds), seconds (< 20 seconds), tens of seconds (> 20 seconds). Always measure both BioImageXD and the other program on the same computer. If a program does not support a procedure, mark the result as N/A.***

- Open both test datasets.
- Perform median noise filtering, separately for both datasets.
- Create full quality volume rendering, separately for both datasets.
- Create full quality surface rendering, separately for both datasets.

Nature Methods: doi:10.1038/nmeth.2047

Supplementary Table 3. Open source software comparison. (Colors are for assisting in reading only. See also Supplementary Tables 2 and 4.)

		BioImageXD (1.0RC3)	**BioImage Suite** (3.01)	**ImageJ (Fiji)** (Madison, upd. 02/2012)	**Vaa3D (previously V3D)** (2.707)	**VisBio** (3.40RC1)
Open	**Source code**	Open source code.	Open source code.	Open source code.	Open source code, some limitations.	Open source code.
	Algorithms	Algorithms fully disclosed.	Algorithms fully disclosed.	Algorithms fully disclosed.	Algorithms fully disclosed.	Algorithms fully disclosed.
	Background processing	No undisclosed background processing.	No undisclosed background processing.	No undisclosed background processing.	No undisclosed background processing.	No undisclosed background processing.
	Documentation	User guide & documented code.	Thorough documentation.	Documentation available.	User guide and tutorials.	User guide.
Extensive	**Basic image processing**	Many methods, incl. graphical intensity transfer function.	Many methods, incl. graphical intensity transfer function.	Many methods, no graphical intensity transfer function.	Some methods, no graphical intensity transfer function.	Some methods, incl. graphical intensity transfer function.
	Noise reduction	Many methods.	Many methods.	Many methods.	Some methods.	Not found.
	Colocalization analysis	Both voxel and object based, graphical & quantitative + statistics.	Not found.	Voxel based, graphical & quantitative + statistics (somewhat limited).	Image color blending, no quantitative colocalization.	Not found.
	3D visualization	Interactive 3D rendering with full quality; many modules, including volume & surface rendering.	Interactive 3D rendering with full quality; some modules, including volume & surface rendering.	Interactive 3D volume and surface rendering possible with the 3D viewer plugin.	Interactive 3D rendering with full quality; many modules, including volume & surface rendering.	Interactive 3D rendering with some-what reduced quality; some modules, incl. volume rend., no surface rend.
	Animation / movie generation	Two combineable animation methods: keyframe & camera path.	Manual interactions with images recordable as series of still images.	Manual interactions with images re-cordable + axial rotation movie.	Manual interactions with images recordable.	Keyframe animation.
	Segmentation into objects	Many methods in 4 classes, such as watershed and region growing	Some methods, no watershed segmentation.	Many methods in 3 classes, limited in some cases.	Some methods, optimized specifically for neuron tracing.	Not found.
	Motion tracking	Object tracking in 3D.	Vessel tracking possible, no tracking of objects in 3D.	Object tracking in 3D possible with a combination of separate plugins.	None.	Not found.
Usable	**Price**	Free.	Free.	Free.	Free.	Free.
	Operating systems	Win, Mac, Linux.	Win, Mac, Linux.	Win, Mac, Linux.	Win, Mac, Linux.	Win, Mac, Linux.
	Program structure	Single installable module.	Single installable module.	Main bundle + separate plugins.	Main module + separate plugins.	Single installable module.
	User interface	Single large window contains everything in unchanging layout; help texts and color coded settings.	Single menu window is used to load different application windows, data loaded separately to each.	Main toolbar window with separate windows for each plugin, tool, image dataset and result output.	Single window with menu only, loaded files and results appear as new windows inside the main window.	One window for file handling, every visualization opened in a new window with settings.
	Speed test: loading files	Seconds.	N/A.	Tens of seconds.	Seconds.	Tens of seconds.
	Speed test: noise filtering	Seconds.	Instant.	Seconds.	Seconds.	N/A.
	Speed test: volume rend.	Instant.	Instant.	Seconds.	Seconds.	Instant.
	Speed test : surface rend.	Instant.	N/A.	Seconds.	Seconds.	N/A.
Adjustable	**Feature adjustability**	Features well adjustable in the graphical user interface.	Features well adjustable, especially through scripting.	Features generally well adjustable, depending on the plugin.	Features well adjustable in the graphical user interface.	Features well adjustable in the graphical user interface.
	Possibility to create lists of commands	Lists of commands can be assembled with a graphical Procedure list task that supports nearly all features.	No direct support for lists of commands, but similar functionality is obtainable through scripting.	Fiji script editor gives access to ImageJ features and plugins from scripts using commmon languages.	No direct support for lists of commands, plugin architecture may enable some type of command aggregation.	Not found.
Applicable	**3D timelapse support**	Integrated 3D timelapse support.	Integrated 3D timelapse support.	Support for 3D timelapse, with some limitations in individual features.	Limited timelapse support and 3D support needs Z-spacing input.	Integrated 3D timelapse support.
	File handling	File tree, image data read into memory only on demand.	No file tree, image data read into memory upon loading.	No file tree, image data read into memory upon loading; virtual stack.	File tree, image data read into memory upon loading.	File tree, image data read into memory upon loading.
	High throughput possibilities	Graphical batch processor, covers nearly all features, + command line.	Batch processing possible with command line scripts.	Some batch processing tools, especially with command line scripts.	Not found.	Not found.
	Validation tools	Simulated data creation of particles, their cellular location & movement.	Not found.	Not found, but model images can be created with scripts.	Some validation tools for neuronal tracing.	Not found,
Extendable	**Program architecture**	Extendable architecture, supports VTK and ITK libraries.	Based on scripts, supports VTK and ITK libraries.	Plugin architecture, creating plugins is straightforward, licences vary.	Plugin architecture, supports ITK library.	Extendable with ImageJ plugins.
	Management & support	Organized project management, mailing lists, bug management, email support.	Support forum, comprehensive manual.	Organized project management, active mailing lists and active bug tracker.	Mailing list, bug management.	Website with information, otherwise limited.
	Ongoing development	Developed actively, steered by scientific applications of users.	Developed actively, tailored for medic-al image visualization & registration.	Developed actively by a large community. Many plugins available.	Developed actively, a general purpose tool but tailored for neuronal imaging.	Primarily a visualization tool. Integration plans with ImageJ.

Supplementary Table 4. Proprietary software comparison. (Colors are for assisting in reading only. See also Supplementary Tables 2 and 3.)

		MATLAB (7.9.0.529) + **IPT*** (6.4)	**Imaris** (7.4.0) (Bitplane)	**FluoView** (3.01.01.09) (Olympus)	**Volocity** (6.0.1) (PerkinElmer)	**ZEN Lite** (2011) (Zeiss)
Open	**Source code**	Closed source code.	Closed source code.	Closed source code.	Closed source code.	Closed source code.
	Algorithms	Algorithms not fully disclosed, but explained in documentation.	Algorithms not fully disclosed.	Algorithms not fully disclosed, but most explained in user guide.	Algorithms not fully disclosed.	Algorithms not fully disclosed, but most explained in user guide.
	Background processing	No indications of background processing.	Rendering quality indicates possible background processing.	No indications of background processing, but cannot be confirmed.	Rendering quality indicates possible background processing.	No indications of background processing, but cannot be confirmed.
	Documentation	Thorough documentation.	User guide.	User guide.	Descriptive inline help desk.	User guide.
Extensive	**Basic image processing**	Many methods, no graphical intensity transfer function for 3D.	Many methods, no graphical intensity transfer function.	Many methods, incl. graphical intensity transfer function.	Some methods, no graphical intensity transfer function.	Many methods, incl. graphical intensity transfer function.
	Noise reduction	Many methods.	Some methods.	Some methods.	Some methods.	Some methods.
	Colocalization analysis	Some calculations possible, but no proper colocalization tools found.	With separate module, graphical & quantitative, limited statistics.	Simple colocalization tool with some statistics.	Some quantitative colocalization tools, also object-based, limited statistics.	Not found.
	3D visualization	Interactive 3D rendering of surfaces and other plots, volume rendering support through Viewer3D module.	Interactive 3D rendering with separate module, reduced quality, volume & surface rendering.	Interactive 3D rendering with two volume rendering modules, no surface rendering. Reduced quality.	3D rendering with volume & surface rendering, sometimes limited performance and interactivity.	No interactive 3D rendering in Lite version
	Animation / movie generation	Possibility to create movie from any matrix data, but no interactive tools.	Keyframe and rotation animations.	Simple rotations and time lapse animations.	Advanced keyframe animation with visual transitions.	Timelapse animations only.
	Segmentation into objects	Many methods.	Many methods (details unclear) and object separation filters.	Manual thresholding only.	Many methods (details unclear) and object separation filters.	Not found.
	Motion tracking	Not found, but some user-created methods can exist.	Object tracking in 3D with separate module, many options.	Not found.	Object tracking in 3D.	Not found.
Usable	**Price**	$ hundreds per core & per toolbox.	$ thousands per module.	With microscope purchase.	$ thousands per module.	Free (registration required).
	Operating systems	Win, Mac, Linux.	Win, Mac.	Win.	Win, Mac.	Win.
	Program structure	Separately purchased toolboxes.	Many modules, purchased separately.	Installed with microscope purchase.	Many modules, purchased separately.	Single installable module.
	User interface	Programming-based. Single large window for basic operations, plots & images in separate windows.	Single large multi-pane main window with detachable toolbars and some separate windows.	Single large window contains many organizeable subwindows for each file or control.	Single large multi-pane main window with resizeable subwindows. Data library for various types of items.	Single large window contains everything in unchanging layout.
	Speed test: loading files	Instant.	N/A.	N/A.	Tens of seconds.	Tens of seconds.
	Speed test: noise filtering	Seconds.	Seconds.	Seconds.	Tens of seconds.	Seconds.
	Speed test: volume rend.	N/A.	Seconds.	Instant.	Tens of seconds.	N/A.
	Speed test : surface rend.	Seconds.	Seconds.	N/A.	Instant.	N/A.
Adjustable	**Feature adjustability**	Features generally well adjustable using parameters to functions.	Features generally well adjustable, limited in some cases.	Features generally well adjustable in the graphical user interface.	Features generally well adjustable, limited in some cases.	Features generally well adjustable, limited in some cases.
	Possibility to create lists of commands	Possible through scripting, no graphical method except through Simulink.	Separate modules enable some commands to be assembled into lists, more flexibility through programming.	Not found.	Some modules enable some features to be assembled into lists of commands.	Not found.
Applicable	**3D timelapse support**	Somewhat limited.	Integrated 3D timelapse support.	Integrated 3D timelapse support.	Integrated 3D timelapse support.	Integrated 3D timelapse support.
	File handling	No file tree, images read into memory upon loading, workspace showing different variables.	No file tree, image data read into memory upon loading; only one file at a time, but it can be large.	File explorer and data manager, image data read into memory upon loading.	Data library with graphical interface, initial data loading may take time, but fast after that.	File tree, image data read into memory upon loading, but utilizing disk cache.
	High throughput possibilities	Possible through programming.	Some batch processing with separate module, more through programming.	Not found	Some batch processing, more with separately purchased server & software.	Graphical batch processing, but basically only for exporting files.
	Validation tools	Possible through programming.	Not found.	Not found.	Not found.	Not found.
Extendable	**Program architecture**	Core & toolboxes closed, but good extendability through scripts.	Closed, some user modifiability with a separate programming module.	Closed, possibly extendable with macros, more features can be purchased.	Closed, not extendable by users.	Closed, not extendable by users, purchasable version with more features.
	Management & support	Documentation, examples, email and phone support, extensive user community in MATLAB Central.	Customer portal, responsive email support, online training sessions.	Olympus support.	Email and web page support.	Email and phone support, web page contains hotfixes and information.
	Ongoing development	Developed actively, stable, mature and popular.	Developed actively, stable, mature and popular.	Developed as part of microscopes, limited viewer vers. available for free.	Being developed, also integrated into various platforms, mature & stable.	Developed as part of microscopes, full version not free.

*Image Processing Toolbox

	Bis	Fil	Gen	L-NA	Saf	Sta
Average cluster intensity, 5 min		-			+	-
Average cluster intensity, 45 min				+	-	
Average cluster intensity, change 5 → 45 min					- - -	
Average cluster size, 5 min		- -				- -
Average cluster size, 45 min		-			-	-
Average cluster size, change 5 → 45 min					-	
Average number of clusters / cell area, 5 min					+ + +	- - -
Average number of clusters / cell area, 45 min			+	+	+ + +	-
Average number of clusters / cell area, change 5 → 45 min		- -	- - -	- - -	+ + +	- - -
Average distance between clusters / cell area, 5 min		+		+	+	
Average distance between clusters / cell area, 45 min		+ +		+	+	+
Average distance between clusters / cell area, change 5 → 45 min			+ + +			
Average % of clusters inside cell, change 5 → 45 min	- - -	- - -	- - -	- - -		
Average cluster distance to cell center / cell volume, change 5 → 45 min	-	-	-	- -		
Average colocalization (Manders' coefficient) of integrin with caveolin-1, change 5 → 45 min		- - -	- -	- - -		
Average speed of clusters	-	-	- -	+		
Average directional persistence of clusters	+	+ + +	+	+ +		-

Supplementary Table 5. Aggregated analysis results of integrin clustering and internalization. BioImageXD can process and analyze hundreds of image data files in a batch session. Here, approximately 1.2 million integrin clusters were analyzed from approximately 1,300 three-dimensional cell images. Inhibitors: Bis = Bisindolylmaleimide I, Fil = Filipin III, Gen = Genistein, L-NA = L-NAME, Saf = Safingol, Sta = Staurosporine. Green color and plusses indicate a stronger effect than that observed with the control sample, and red color and minuses a weaker effect. Color intensity and the number of signs indicate strength of the effect, based on statistical significance and magnitude of differences in mean values (see Supplementary Methods for details). Safingol and Staurosporine, inhibitors of protein kinases, affected the clustering most, but had little effect on movement or internalization of the clusters. On the other hand, Filipin II and L-NAME, inhibitors of caveolar endocytosis and nitric oxide synthesis, respectively, affected clustering less, but strongly inhibited internalization and colocalization with caveolin-1. The results indicate that the clustering, internalization and lateral movement of integrins are controlled by different molecular mechanisms.

Nature Methods: doi:10.1038/nmeth.2047

Supplementary Table 6. Possibilities to conduct proof of concept integrin analyses with other software (page 1/2).

To illustrate that the integrin analyses done for this paper would have been difficult to do with other software, the critical steps needed to conduct them were first listed, and then each of the programs BioImageXD was previously compared to (Supplementary Tables 2-4) was checked for whether it was possible to conduct these steps with them in practise. For each step, this possibility was expressed with one of three alternatives:

- Possible (not all features might be exactly the same as those used in BioImageXD, but close enough for obtaining roughly the same result)
- Limited (the most critical features are available, but there are limitations and/or features missing, that might make getting correct results difficult, or require further non-trivial programming)
- Not possible (necessary features for conducting the step cannot be found, or require further non-trivial programming).

The results were gathered into the table on the following page. As different programs are made for somewhat different purposes and have different sets of features, the comparisons are only approximate. Also, to make the comparison fairer for other software, it was not very strict: features or parameters identical to those of BioImageXD were not required for the ”Possible” marking – it was sufficient that the outcome of the step seemed approximately right. Similarly, the ”Limited” marking was given if at least something in the right direction was found, even if the quantitative result might not necessarily be correct. Still, the results show that the proof of concept integrin study would have been difficult to conduct with other software. This does not mean that it would have been completely impossible (certainly by programming additional modules or scripts it might have been possible with other programs), but demonstrates the capabilities of BioImageXD in conducting complex analyses by end-users without programming skills.

It should be noted that in the case of MATLAB and ImageJ our comparisons only cover the packages used and the plugins they contain. Individual users may have more features at their disposal, but considering all such options would have been beoynd the scope of this paper.

In cases where it was possible to conduct at least somewhat similar analyses as those done with BioImageXD, such analyses were attempted with sample data and the results compared to those obtained with BioImageXD (see Supplementary Table 7).

Supplementary Table 6. Possibilities to conduct proof of concept integrin analyses with other software (page 2/2).

A. Prepare datasets for reliable analysis (polygonal cropping, automatic background subtraction, noise removal).
B. Make smooth, uniform segmentation of cell membrane (gaussian smoothing or similar, morphological dilate and erode or similar).
C. Segment integrin clusters that touch eachother and differ considerably in size, intensity and number, with a single protocol being able to handle different time points (dynamic thresholding or similar, object separation or similar).
D. Quantify segmented objects (volume, surface area, intensity, number, distance to each other).
E. Quantify cellular internalization (analyze whether segmented objects in one channel are inside segmented object in another channel, and calculate their distance to center of mass of second channel object).
F. Quantify colocalization automatically between images with varying intensities (autothresholding for colocalization, calculation of Manders' coefficients, "differential staining").
G. Prepare time series data for reliable particle tracking (time series concatenation, automatic registration for unwanted movement compensation).
H. Perform 3D motion tracking for particles changing considerably in movement direction, shape, size and intensity, and undergoing splitting and merging.
I. Create multi-channel 3D surface and volume renderings necessary for setting up and validating processing protocols.
J. Process large numbers of datasets with complex lists of commands in batches without needing programming skills.
K. Validate analyses with simulated data creation.
L. General adjustability and flexibility allow determining and adjusting the type of analysis patterns needed here.

Possibilities to carry out each step are marked as follows: X = possible, (X) = limited, - = not possible. See previous page for more information.

Software	A	B	C	D	E	F	G	H	I	J	K	L
BioImageXD (1.0RC3)	X	X	X	X	X	X	X	X	X	X	X	X
BioImage Suite (3.01)	(X)	(X)	(X)	-	-	-	X	-	X	(X)	-	(X)
ImageJ (Fiji) (Madison, updated 02/2012)	X	X	X	(X)	-	X	X	(X)	(X)	(X)	-	X
Vaa3D (previously V3D) (2.707)	(X)	(X)	(X)	-	-	-	-	-	X	-	-	(X)
VisBio (3.40RC1)	-	-	-	-	-	-	-	-	(X)	-	-	-
MATLAB (7.9.0.529) + **IPT*** (6.4)	(X)	X	X	(X)	(X)	(X)	X	-	(X)	(X)	(X)	X
Imaris (7.4.0) (Bitplane)	(X)	X	(X)	X	(X)	X	(X)	X	X	(X)	-	(X)
FluoView (3.01.01.09) (Olympus)	X	(X)	-	-	-	(X)	(X)	-	(X)	-	-	-
Volocity (6.0.1) (PerkinElmer)	(X)	X	(X)	X	X	(X)	(X)	X	X	(X)	-	(X)
ZEN Lite (2011) (Zeiss)	X	-	-	-	-	-	-	-	-	-	-	(X)

*Image Processing Toolbox

Nature Methods: doi:10.1038/nmeth.2047

Supplementary Table 7. Proof of concept integrin analysis results obtained with other software.

In cases where it was possible to conduct at least somewhat similar analyses with other software, as those done with BioImageXD (see Supplementary Table 5), such analyses were attempted with sample datasets from the 5' and 45' timepoints and with a live cell 4D series. Parameters were adjusted to be the same as used with BioImageXD as well as possible, and numerous attempts were made with each program to achieve the best results. However, since corresponding features in different programs are sometimes very different, and the exact effect of different parameters is not always known, the results are an approximation only, and it should be noted that better results might still be obtainable with other software by expert users or users having self-made plugins or scripts. Selected results are shown below, representing all three types of analyses conducted: integrin clustering, internalization analysis and motion tracking. A missing value means that no method was found to make such a quantification with that particular software. As can be seen, many programs simply didn't have the necessary features to make the quantifications. Those that did, such as Imaris and Volocity, often had insufficient control over the algorithms. This made it difficult to adjust them to cope with both early and late timepoints of receptor clustering data, leading to some erratic results with for instance average cluster volume and percent of clusters internalized. With BioImage Suite a major problem was that it did not seem to be possible to read the sample data correctly. With ImageJ (Fiji) problems were caused by limited 3D support and problems in some features, such as object separation, leading to erranous or missing results. With MATLAB (Image Processing Toolbox) even average distance measurement had to be programmed separately for these tests, and for instance internalization analysis and motion tracking would have required too extensive amounts of programming to be practically feasible. All in all, the results support the same notion as Supplementary Table 6: the proof of concept integrin analyses would have been difficult to conduct with other software.

Software	Number of clusters		Average cluster volume (µm)		Average distance between clusters		Cell membrane area (µm^2)		% of clusters inside cell		Average speed of clusters (µm/s)	Average directional persistence of clusters
	5'	45'	5'	45'	5'	45'	5'	45'	5'	45'		
BioImageXD (1.0RC3)	**1402**	**558**	**0.087**	**0.163**	**23.0**	**14.7**	**2509**	**2168**	**76%**	**95%**	**0.0040**	**0.68**
BioImage Suite (3.01)	-	-	-	-	-	-	-	-	-	-	-	-
ImageJ (Fiji) (Madison, updated 02/2012)	**905**	**394**	**0.087**	**0.177**	-	-	**3729**	**2435**	-	-	-	-
Vaa3D (previously V3D) (2.707)	-	-	-	-	-	-	-	-	-	-	-	-
VisBio (3.40RC1)	-	-	-	-	-	-	-	-	-	-	-	-
MATLAB (7.9.0.529) + **IPT*** (6.4)	**784**	**376**	**0.103**	**0.192**	**22.81**	**14.89**	-	-	-	-	-	-
Imaris (7.4.0) (Bitplane)	**1414**	**637**	**0.178**	**0.342**	-	-	**3060**	**3544**	**76%**	**102%**	**0.0080**	**0.62**
FluoView (3.01.01.09) (Olympus)	-	-	-	-	-	-	-	-	-	-	-	-
Volocity (6.0.1) (PerkinElmer)	**1437**	**183**	**0.164**	**0.083**	**24.9**	**14.6**	**3139**	**1194**	**73%**	**49%**	**0.0030**	**0.80**
ZEN Lite (2011) (Zeiss)	-	-	-	-	-	-	-	-	-	-	-	-

*Image Processing Toolbox

Nature Methods: doi:10.1038/nmeth.2047

Supplementary Methods. Materials and methods (page 1/8).

Programming

BioImageXD is written in Python, with C++ used for more demanding tasks in order to improve performance. The software utilizes and extends the mature open source libraries VTK (Visualization Toolkit) and ITK (Insight Segmentation and Registration Toolkit). The graphical user interface utilizes the wxPython library. BioImageXD consists of about 70,000 lines of Python code and 20,000 lines of C++ code. All code is written to be platform independent. The C++ code is written to extend the VTK and ITK libraries, while Python is used to define the core, the graphical user interface, and the modules of the software. The code is structured and commented, and its architecture supports straightforward addition of new Python modules that can use the C++ libraries. As an example of this, new modules are constantly being created and existing ones extended in some of the applications where BioImageXD is used.

Management and quality control

BioImageXD is designed by a multidisciplinary team of scientists from an applications point of view, and implemented by software engineering professionals. The development is headed by the main development team, consisting of five people from three different academic institutions, all having extensive and different types of bioimaging-related experience. Design decisions and feature plans are written into a web-based documentation system, where each team member has access to relevant material. The same system, specifically designed for BioImageXD, is also used to monitor progress, comment and discuss current issues, and it serves as a framework for production meetings and for documenting meeting decisions. Different software installation packages and the source code are maintained at SourceForge, which also contains a public bug tracking system. Links to these are available on the BioImageXD web site at www.bioimagexd.net. Each new software version is first subjected to a testing protocol covering critical functionality of the software, carried out by the main development team and volunteer users. Bug fixing sessions and repeated testing follow in several iterations, until the software version is stable and functional enough for wider distribution. The software contains a built-in bug reporting feature that the user can activate at any time if a problem is encountered. The system also automatically sends a bug report after a program crash, if the user so wishes. All such reports are evaluated by the development team, and users who have supplied an e-mail address are given a personal response.

Software comparisons

A testing protocol was devised to compare the features and performance of different bioimaging programs, based on the six criterion groups defined in this paper for scientific bioimaging software. Due to big differences between programs it was difficult to find procedures that could be similarly carried out with all programs and the performance speeds thus meaningfully compared. Several procedures were tested with all of the compared programs, including different methods of viewing datasets, intensity transfer function processing, creating animations, performing segmentation and tracking, analyzing colocalization and merging data channels. In the end four simple and sufficiently repeatable procedures were chosen for the performance speed comparisons, representing different key aspects of the programs, from file handling to image processing to 3D visualization. The entire testing protocol, for both feature and performance comparisons, is described in Supplementary Table 2.

Supplementary Methods. Materials and methods (page 2/8).

A selection of typical programs, both open source and proprietary, was made for the comparisons. Comparing BioImageXD to proprietary software with closed source code was somewhat complicated, as these programs do not generally fulfill the first and perhaps most important criterion of openness. Their suitability for scientific work is therefore debatable, despite their additional benefits, such as better stability due to the resources of large commercial companies. BioImageXD was nonetheless compared to proprietary programs as well, as they are relatively widely used.

All tests were made as pairwise comparisons between BioImageXD and one other program, both running on the same computer. The same set of two confocal microscopy files, one 3D stack and the other a 4D (3D timelapse) dataset, was used for all tests. The comparisons were made as objectively as possible, considering that different bioimaging programs can be very different even on a fundamental level, and that new software versions become constantly available with new features and other changes. The results of the comparisons (Supplementary Tables 3 and 4) are therefore only approximate, indicating general trends rather than detailed differences.

Different software packages were also compared from the point of view of whether it would be possible to conduct the proof of concept integrin analyses with them, and where possible, calculations with sample datasets were made (Supplementary Tables 6 and 7).

Cells and reagents

A human Saos-2 cell line stably expressing integrin α2 was used for all experiments. The cells were grown at 37 °C on high quality cover glasses for two days in Dulbecco's modified eagle's medium with 10 % fetal calf serum.

Antibodies and chemical reagents/inhibitors used:

• Mouse antibody against integrin α2 (16b4)	dilution 1:500	Serotec
• Rabbit antibody against caveolin-1	dilution 1:150	Santa Cruz
• Secondary antibody anti-mouse	dilution 1:400	Molecular Probes
• Secondary antibody anti-mouse-Alexa-633	dilution 1:400	Molecular Probes
• Secondary antibody anti-mouse-Alexa-555	dilution 1:400	Molecular Probes
• Secondary antibody anti-rabbit-Alexa-555	dilution 1:400	Molecular Probes
• CellTracker Green	concentration 0.5 μM	Molecular Probes
• Bisindolylmaleimide I (inhibitor of proteinkinase Cs)	concentration 5 μM	Sigma
• Filipin III (inhibitor of caveolae formation)	concentration 1 μg / ml	Sigma
• Genistein (inhibitor of tyrosine kinases)	concentration 100 μM	Sigma
• L-NAME (inhibitor of nitric oxide synthase/caveolae)	concentration 50 μM	Sigma
• Safingol (inhibitor of protein kinase C α)	concentration 10 μM	Sigma
• Staurosporine (inhibitor of protein kinases)	concentration 1 nM	Calbiochem

Supplementary Methods. Materials and methods (page 3/8).

Fixed samples. Cells were first incubated at 37 °C for 1h with an inhibitor (or PBS for control samples), and then the medium was changed to a new solution, containing the inhibitor, primary antibody (anti-α2 integrin) and CellTracker, and the incubation was continued for another 1h. Unbound reagents were then washed away with warm medium, and medium containing inhibitor and secondary antibody (anti-mouse-Alexa-633) was added. The fluorophore-conjugated secondary antibody both induced integrin clustering and labeled the clusters. Samples were incubated for 5, 10, 15, 30 or 45 minutes, fixed with 4% paraformaldehyde (20 min RT), permeabilized with triton X-100 (0.2% in PBS, 5 min RT) and washed with PBS. The samples were then incubated for 1 h at RT with primary antibody (anti-caveolin-1), washed with PBS, incubated for 30 min at RT with secondary antibody (anti-rabbit-Alexa-555), washed with PBS, and finally mounted to objective glass slides with Mowiol-Dabco. All steps involving fluorescent probes were carried out in the dark as much as possible.

Live cell samples. Cells were first grown for 2 days on chambered coverglasses (Lab-Tek; 8 chambers per glass, chamber volume 0.4 ml), and before imaging the medium was changed to CO_2 independent medium (Gibco). The same antibodies, inhibitors and concentrations were used as for fixed samples.

Samples for validation of the internalization analysis. Fixed samples were prepared as described above, but a non-conjugated secondary antibody (anti-mouse) was used to induce integrin clustering. After fixation the samples were incubated with an Alexa-633-conjugated secondary antibody (30 min RT), and again with an Alexa-555-conjugated secondary antibody after permeabilization. Thus integrins inside the cell were labeled with Alexa-555 only, whereas integrins on the surface were labeled with both Alexa-633 and Alexa-555. (The antibody concentrations used were non-saturating, as determined with trial experiments.) The samples were prepared for 5 and 45 min timepoints, and no inhibitors were used.

Confocal microscopy

All imaging was conducted with a Carl Zeiss Axio Observer.Z1 equipped with an LSM 510 confocal module with 3 PMT detectors. A plan-apochromat 63x 1.4 oil immersion objective was used. Lateral pixel density and axial optical slice thickness were adjusted for optimal sampling (as calculated by the LSM software). Pinholes were adjusted so that all channels had the same optical slice thickness of 0.38 µm and image acquisition speed was approximately 1.8 µs per voxel (averaging 2). Detector gains were kept constant, with values based on trial experiments and chosen so that no notable voxel saturation occurred with any sample or at the end of a time series.

Fixed sample imaging. Three channels, scanned separately line-by-line (multitracking) were used, with a pinhole-calibrated dichroic and filter setup optimized to minimize bleed-through: main dichroic HFT UV/488/543/633, secondary dichroics NFT 635 VIS and NFT 545, emission filters LP 650, BP 505-530 and BP560-615. Excitation wavelengths were 633, 488 and 543 nm, respectively. Three-dimensional image stacks were acquired from 10-15 randomly selected cells per sample, from 3 repeats of the experiment.

Live cell imaging. The chambered coverglasses were transferred to the preheated sample holder of the microscope. The cells were then incubated with the inhibitors for 1h, after which 2 µl of primary antibody (anti-α2 integrin) was added and the incubation continued for another 1h. Unbound antibody was washed away with medium, after which the inhibitor was added again, together with 1 µl of secondary antibody (anti-mouse-Alexa-555). Imaging was immediately started for a randomly selected cell, with 21 three-dimensional image stacks acquired with a 3 min time interval. One channel with an excitation wavelength of 543 nm was used (same dichroics and filters as in fixed cell imaging). Imaging was constantly monitored, and if excessive focal drift or other changes occurred the framing of the cell was re-done in all 3-dimensions and imaging continued immediately thereafter. The experiments were repeated at least three times for every inhibitor and control. Importantly, cells were constantly kept at 37 °C and no ice or room temperature incubations were used, so as to avoid any effects caused by temperature changes (that are typically present in internalization studies involving incubations on ice). All washes and medium changes were carried out with solutions preheated to 37 °C, and a precision thermometer was used to measure the actual temperature in the chamber being imaged, and the microscope heating system adjusted to maintain this temperature exactly at 37 °C. Control experiments were performed to confirm that the primary antibody alone did not cause integrin clustering or internalization, even when incubated with the cells for several hours.

Simulated image data

Simulated data was created, using BioImageXD's simulation tool, to validate the developed analysis protocols. In total, 18 simulated datasets were created, nine modeling fixed samples and nine modeling live cell samples. For both fixed and live cell models, three different noise levels and three different settings of clustering were used to give a total of nine different combinations. Medium levels of noise and clustering were set to be as close to real data averages as possible.

Results of real data analyses were used to initialize the number and volume of integrin clusters in simulated data of fixed samples. The number of objects was set to 1186 and average volume to 0.055 µm³ for the first time point. The initial locations of the objects were defined using cell surface models from real data analyses. The initial distance from each object to the cell surface was set using normal distribution with a sigma value of 0.3 µm. Clustering was simulated by making a specific percentage of the objects move towards a few fixed locations while the rest of the objects moved randomly. Object displacement per minute was set to vary between 38 and 152 nm. Each of the nine simulations had time points modeling real data from the 5 min to 45 min situations.

The number of objects for live cell simulated data was initialized as a random number between 100 and 200. The initial volumes of the objects ranged from 0.035 to 0.14 µm³. The speed of the moving objects was defined to be between 132 nm/min and 360 nm/min. Each of the nine simulations had 21 time points, modeling real data from 0 to 60 min with 3 min intervals.

Quantitative analyses

After extensive testing, protocols were devised for the analyses of both fixed and live cell images as follows. Analyses were conducted with the

Supplementary Methods. Materials and methods (page 5/8).

BioImageXD Batch Processor (and Procedure list task). Each numbered section below corresponds to one run. See “Algorithms” below for more information about the algorithms used.

Fixed cell images

1. Preprocessing (same for all channels)
- Automatic background subtraction (Most common value)
- Extract subset (used only for datasets containing more than one cell, to separate the cells into different datasets, based on specified regions of interest)

2. Segmentation and analysis I
Integrin channel:
- Dynamic threshold (Mean; Radius 5,5,5; Threshold 4, Use image spacing)
- Object separation (Level 0.5, Use image spacing, Remove objects smaller than 3)
- Analyze segmented objects (Results parameters: ObjAvgIntensity, ObjAvgVolInUm, AverageDistance, NumberOfObjects)

CellTracker channel:
- Gaussian smooth (Radius factor 4,4,1; Dimensionality 3)
- Threshold (Lower threshold 1; repeated also with thresholds 4 and 7 for Supplementary Fig. 6)
- Dilate (Kernel 4,4,1)
- Erode (Kernel 4,4,1)
- Convert to polygonal data (Simplify 65, Preserve topology, Iso-surface value 255)

3. Colocalization
- Calculate thresholds for colocalization (Source datasets: Integrin channel, Caveolin-1 channel)
- Analyze colocalization (Source datasets as above; result parameter: M1; for Supplementary Fig. 6 the two differentially stained Integrin channels were analyzed for the result parameter DiffStainIntCh1)

4. Analysis II
- Analyze polydata (Polydata image: Cell Tracker result from previous; Segmented objects: Integrin result from previous)
- Connected component labeling
- Analyze segmented objects (Results parameters: ObjAvgAreaInUm, ObjAvgVolumeInUm, PercentageVoxelsInside, AvgDistanceCOMtoCellCOM)

Supplementary Methods. Materials and methods (page 6/8).

Live cell images

1. Concatenation
- Combine time series (used only for time series consisting of several datasets)

2. Registration
- Multi resolution translation registration (Use previous timepoint as fixed image; Level 3: 2, 2, 1, 30, 5, 0.025; Level 2: 4, 4, 2, 30, 25, 0.25)

3. Segmentation
- Automatic background subtraction (Most common value)
- Hybrid median 2D
- Threshold (Lower threshold 78; repeated also with threshold 58 for validation of tracking result)
- Connected component labeling (Remove objects smaller than 3)
- Analyze segmented objects

4. Tracking
- Create motion tracks (Max speed 1.0, Min speed 0.0, Weight 40, Deviation from max/min 100, Max size change 100, Weight 25, Max change of intensity 100, Weight 15, Direction 180, Weight 20, Min length of track 3, Min size of tracked objects 6)

5. Analysis
- Analyze motion tracks

Algorithms

Many of the algorithms mentioned above are self-explanatory. The rest are explained in more detail here. Full algorithm and other details are available in the BioImageXD source code.

Automatic background subtraction (Most common value). Computes the most common voxel scalar value in an image stack, and subtracts that value from every voxel in the image stack, making 0 the most common value. Underflow values are clamped to the minimum value of the voxel type.

Extract subset. Crops an image stack as specified by a region of interest. For non-rectangular ROIs, the image stack is cropped to the smallest possible rectangular region, and 0 is set as the scalar value of the voxels within this region that are outside the ROI.

Supplementary Methods. Materials and methods (page 7/8).

Dynamic threshold. First convolves an image with a mean or median operator, and then subtracts the result from the original image. Then thresholds the resulting image with a user-specified value.

Object separation. Operates on binary image data. First computes euclidean distance to the closest background voxel, and inverts this result. Then calculates a watershed transform of the result of the first step, using local minima voxels as seed points. Finally, the result of the watershed transform is masked using the original binary image data. The result is a label image.

Analyze segmented objects. Measures statistics of individual objects, using a labeled image stack and the original image stack as inputs. Produces statistics such as average object volume, average intensity, and the number of objects.

Convert to polygonal data. Creates polygonal data from binary image data. First extracts a polygonal surface mesh using a marching cubes algorithm (Lorensen, W. & Cline, H. Marching Cubes: A high resolution 3D surface construction algorithm. *Computer Graphics* **21**, 4 (1987).) Then smooths the polygonal data by reducing the number of polygons with a decimation algorithm (Schroeder, W., Zarge, J., & Lorensen, W. Decimation of triangle meshes. *Proceedings of SIGGRAPH*, 65-70 (1992). http://www.vtk.org/doc/release/5.6/html/a00456.html)

Analyze polydata. Measures object location in proportion to constructed closed polygonal surface data. Computes whether object voxels and object centers of mass are inside or outside of the surface and their distance to the surface.

Calculate thresholds for colocalization. Automatically calculates thresholds for colocalization using the Costes method (Costes, S.V. *et al.* Automatic and quantitative measurement of protein-protein colocalization in live cells. *Biophys. J.* **86**, 3993-4003 (2004).)

Analyze colocalization. Measures colocalization statistics, such as Manders' colocalization coefficients and Pearson's correlation, using either automatically or manually determined thresholds.

Multi resolution translation registration. Uses a multi-resolution Gaussian image pyramid to solve a registration problem between two distinct 3D image stacks. For each level of the pyramid, tries to iteratively find an optimal 3D translation to maximize similarity between the image stacks using a gradient descent optimizer. Similarity of the image stacks is defined by a mean squares metric. The result is a 3D translation matrix.

Hybrid median 2D. First computes two values: the median of the + neighbors and the median of the x neighbors. Then computes the median of these two values and the center pixel. The result of this second median is the output pixel value.

Create motion tracks. A 3D single-particle tracking method for clustering particles. Not yet published, but the source code is available in the BioImageXD repository.

Analyze motion tracks. Measures statistics from motion tracks, such as average particle speed and directional persistence.

Supplementary Methods. Materials and methods (page 8/8).

Statistical analyses

A t-test for unequal sample sizes and unequal variances was used. Datasets were confirmed to follow normal distribution with normal probability plots. A two-tailed test was used, because it was not in all cases possible to predict in which way the tested inhibitors might affect. Normally the authors use an alpha level of 0.05 for these types of analyses, but because several parameters were compared from the same datasets in this case, the alpha level was set to 0.02 for fixed datasets and 0.03 for live cell datasets, roughly following the Bonferroni correction. (The number of parameters would have suggested a more substantial correction, but because some of the parameters are likely to be dependent, compromise values were set.)

Preparation of figures, tables and videos

“Linear” intensity transfer function modifications were applied to microscopy images as necessary to improve their clarity for visualization purposes. The modifications were identical for all images in panels containing several comparable images. All images were recorded with the *Save snapshot image* feature of BioImageXD. The look-up-tables (palettes) used are shown next to the images where appropriate.

Figure 2. Letters and numbers in panels (c), (e) and (i) were re-drawn with Corel Photo-Paint X4 for better visibility.

Supplementary Table 5. A formula was devised for showing the effects of the inhibitors in intensities of green or red color: 255(x/c-1)(1-20p), where x = value with inhibitor, c = corresponding value in control experiment, p = p-value for the difference between x and c. This formula gives a value between -255 and 255, which was used (as an integer) as the intensity of either the green component (positive values) or red component (negative values) in drawing the RGB color value for the table. The formula considers both the relative difference in averages and the p-value, but if the p-value was greater than 0.05, the formula was not used, and white color (signifying no effect) was drawn instead. The plusses and minuses were drawn based on the green and red intensities, respectively, in such a way that an intensity between 1 and 89 gave one symbol, an intensity between 90 and 150 two symbols and an intensity higher than 150 three symbols. Thus, the symbols do not directly indicate statistical significance (as it is already considered in the color intensity), but clarify differences in the color intensities.

Animator sample videos. Both videos were created with the BioImageXD animator (720x576, 25 f/s) and converted to publication format with Avidemux 2.5.4. For the tracking video the dataset was surface-rendered (transparency 35), with a magenta look-up-table, together with *Visualize motion tracks* set to display one sample track. Keyframe animation was used. For the animation of going inside a cell both the integrin channel (red look-up-table) and the cell tracker channel (white look-up-table) were surface-rendered, and view angle was set to 75. A 5-node camera path was used.

Screen capture videos. All three videos were recorded with BB Flashback express 2.8.2, during actual use of BioImageXD, and converted to publication format with Avidemux 2.5.4. Acquisition frame rate was reduced in order to give the computer more performance power for BioImageXD, and was 8 f/s for the animator and segmentation videos and 16 f/s for the rendering video.

Nature Methods: doi:10.1038/nmeth.2047

PII

APPLICATION INDEPENDENT GREEDY PARTICLE TRACKING METHOD FOR 3D FLUORESCENCE MICROSCOPY IMAGE SERIES

by

Lassi Paavolainen, Pasi Kankaanpää, Pekka Ruusuvuori, Gregory McNerney,
Mikko Karjalainen & Varpu Marjomäki 2012

9th IEEE International Symposium on Biomedical Imaging (ISBI)

APPLICATION INDEPENDENT GREEDY PARTICLE TRACKING METHOD FOR 3D FLUORESCENCE MICROSCOPY IMAGE SERIES

L. Paavolainen[1]*, P. Kankaanpää*[2]*, P. Ruusuvuori*[3,1]*, G. McNerney*[4]*, M. Karjalainen*[1] *and V. Marjomäki*[1]

[1]Department of Biological and Environmental Science, University of Jyväskylä, Jyväskylä, Finland
[2]Department of Biochemistry and Food Chemistry, University of Turku, Turku, Finland
[3]Department of Signal Processing, Tampere University of Technology, Tampere, Finland
[4]Center for Biophotonics Science and Technology, University of California Davis Medical Center, Sacramento, CA, USA

ABSTRACT

Single-particle tracking is computationally a challenging problem, and usually solved with local methods. Local methods suffer from defects in the image data or in the detection of particles, such as temporal disappearing of particles. A particle tracking method has to provide a solution also to real disappearing and appearing of particles as a result of merging and splitting. Here, we present an efficient, greedy algorithm as a solution to the particle tracking problem. This improved local method is application independent, as it has high configurability of the function used to solve particle correspondence. To demonstrate the accuracy of the method, we apply it to real microscopy image data with the BioImageXD software, validate it using simulated image data, and compare it to a well-known existing method.

Index Terms— Particle tracking, Biomedical image processing, Image motion analysis, Optical microscopy, Greedy algorithms

1. INTRODUCTION

In order to gain in-depth understanding of the cellular physiological processes during cell division, migration and pathological processes such as cancer and inflammation, it is imperative to study these processes in live cells. Recent advances in biomedical imaging have facilitated the detection of fast cellular processes in 3D.

The dynamic behaviour of particles ranging from molecular structures to whole cells, can be analysed using single-particle tracking methods. In single-particle tracking, the three-dimensional spatial locations of particles are traced through a time-lapse sequence of image volumes, producing particle trajectories that can be analysed for instance for speed or direction of movement.

However, single-particle tracking in fluorescence microscopy is a challenging task. First, the size, shape and intensity of fluorescent particles often vary through the recorded time series. This complicates the process of linking the detected particles, which there may be hundreds of, to the corresponding ones in the following time point. Second, estimating the particle location in the next time point may be problematic. Microscopic particles in cells often exhibit Brownian motion, causing their movements to be somewhat erratic, even if they are being actively transported. Larger biological particles, such as cells, also tend to constantly change their direction of movement and have a complex, labile morphology. For these reasons, many particle tracking methods search for the corresponding particle using a specified maximum distance from the location of the particle in the previous time point, instead of trying to estimate the direction of movement. Third, there are many events that can occur during the image series that complicate tracking [1]. In a *merge* event, two or more particles merge into one. This can happen when particles are closer to each other than what the resolution limit of the imaging device is, or when a real cluster is formed. A *split* event occurs when a single particle is split into two or more distinct particles. Additional events to take into account include *appearing* and *disappearing* of particles in the middle of a time series. Appearing of a completely new particle occurs when a particle has moved into focus or into the field of view, or when a particle has gained enough size and intensity as a result of the clustering of fluorophores. *Disappearing* is defined as an event where an individual particle disappears without merging with other particles.

General single-particle tracking problem is NP-hard [2], which makes finding a global solution, without approximations, computationally unfeasible even for tens of particles [1, 3]. To circumvent this constraint, many tracking algorithms solve particle correspondence locally. Greedy optimization makes particle tracking efficient, but it usually cannot give an optimal global solution to the particle tracking problem [4]. Also, some local algorithms do not handle merge and split events [5]. To improve local results, some algorithms use an additional global step to recover the temporal disappearing of particles [2, 3]. The downside is that many of these methods treat merge and split events as temporal disappearing and appearing of particles [2, 3].

In this paper, we present a new, greedy, local single-particle tracking method for 3D fluorescence microscopy image data, designed to overcome the challenges mentioned. Our algorithm handles merge, appear and disappear events using only one-pass through the particles in the time series. A split event is considered as the appearing of one new particle. We focus on finding trajectories of particles rather than methods to detect them. There are several methods for detecting particles, for instance spot detection methods for finding small point-like structures [6], or active contour methods for detecting large cells [7, 8]. The single-particle tracking method presented here is not application specific, and can thus be used to track the movement of virtually any type of particles.

2. METHODS

2.1. Particle tracking method

Proposed particle tracking method makes decisions using information of the existing trajectories and of the particles of the time point under evaluation. Optimization of the particle selection for each tra-

jectory is based on the calculation of a goodness value. The goodness value is calculated using particle location, size, average intensity and direction of movement, and information of previous particles in the trajectory. The user can control the method by setting weights to these four parameters. Additionally, the user can define terms of maximum deviation for the parameters that each particle has to fulfill before being considered for the trajectory. If required, the weights and terms can also be used to limit the tracking to specific trajectories, such as those of fairly straight moving particles.

The terms for size, average intensity and direction angle are defined using the specified maximum deviations, as compared to the previous particle in the trajectory. For the location term, the user can define a minimum (D_l) and a maximum (D_h) distance in addition to the maximum deviation. Let $\sigma_d, \sigma_s, \sigma_i, \sigma_a \geq 0$ be the user-defined deviations of distance, size, average intensity and direction angle, respectively. Also, let p_t be the time point, p_l the location, p_s the size, and p_i the average intensity of particle p, and let p_a be the angle of particle direction, and t_k the particle of k:th time point in trajectory t. Using these definitions, all particles p that fulfill the terms presented in equation 1 are considered to be included in the trajectory t.

$$\begin{aligned} D_l * (1 - \sigma_d) \leq d \leq D_h * (1 + \sigma_d) \\ \frac{|p_s - (t_{p_t-1})_s|}{(t_{p_t-1})_s} \leq \sigma_s \\ \frac{|p_i - (t_{p_t-1})_i|}{(t_{p_t-1})_i} \leq \sigma_i \\ \frac{|p_a - (t_{p_t-1})_a|}{(t_{p_t-1})_a} \leq \sigma_a \end{aligned} \tag{1}$$

where $d = \|p_l - (t_{p_t-1})_l\|$. The direction angle of particle p_a is calculated using the location of the particle p_l and the locations of two previous particles in the trajectory $(t_{p_t-1})_l$ and $(t_{p_t-2})_l$. Let $\vec{u}$ and $\vec{v}$ be vectors defined as $\vec{u} = p_l - (t_{p_t-1})_l$ and $\vec{v} = (t_{p_t-1})_l - (t_{p_t-2})_l$. The direction angle of the particle is then

$$p_a = \frac{\vec{u}}{||\vec{u}||} \cdot \frac{\vec{v}}{||\vec{v}||}.$$

To determine the goodness of fit for particle p in trajectory t, we define function Ψ separately for distance, size, average intensity and direction angle as defined in equation 2.

$$\begin{aligned} \Psi_d(t,p) &= \begin{cases} 1 & \text{if } D_l \leq d \leq D_h \\ \left| \frac{D_l - d}{D_l * \sigma_d} - 1 \right| & \text{if } d < D_l \\ \left| \frac{d - D_h}{D_h * \sigma_d} - 1 \right| & \text{if } d > D_h \end{cases} \\ \Psi_s(t,p) &= \left| \frac{|p_s - (t_{p_t-1})_s|}{(t_{p_t-1})_s * \sigma_s} - 1 \right| \\ \Psi_i(t,p) &= \left| \frac{|p_i - (t_{p_t-1})_i|}{(t_{p_t-1})_i * \sigma_i} - 1 \right| \\ \Psi_a(t,p) &= \left| \frac{|p_a - (t_{p_t-1})_a|}{(t_{p_t-1})_a * \sigma_a} - 1 \right| \end{aligned} \tag{2}$$

Finally, goodness of fit for particle p in trajectory t is computed using equation 3.

$$\Omega(t,p) = \sum_k w_k * \Psi_k(t,p) \tag{3}$$

Step 1. If first time point, create new trajectory for every particle in the time point and finish iteration.

Step 2. Calculate goodness value using equation 3 for every particle and track combination that fulfills definitions of equation 1.

Step 3. Go through each goodness value from highest to lowest. Add particle p to trajectory t, if t and p used to calculate goodness value are not used yet, t is not terminated, and $\Omega(t,p) > \alpha * h_t$, where h_t is the highest goodness value calculated for t. Mark every trajectory, that had same particle as trajectory t in previous time point, as terminated.

Step 4. Add particle with highest goodness of fit for every track that is not used or terminated. If there are no particles that fulfill definitions of equation 1, then mark trajectory as terminated.

Step 5. For every particle that is not used yet, create new track and add particle to that track.

Fig. 1. Description of the steps of our particle tracking algorithm, iterated for every time point. $\alpha \in [0,1]$ is a user-defined clustering factor. Higher α value results in more merging of trajectories.

where w_k, $k \in \{d, s, i, a\}$ are the user-defined weights for the four parameters. For the weights it holds that $w_k \in [0,1]$ and $\sum_k w_k = 1$.

The proposed particle tracking method creates trajectories using a breadth-first search of particles. Unlike depth-first search, this makes it possible to optimize the selection of the particles for all trajectories at every time point. The key steps of the particle tracking method, iterated for every time point, are presented in Figure 1. Step 1 is initialization, done only for the first time point. Step 2 calculates the goodness of fit of the particles to existing trajectories. In step 3, unused particles are assigned to unused trajectories. A trajectory does not necessarily get the particle that has the highest goodness value calculated in step 2, since that particle might already be assigned to another trajectory for which it had a higher goodness of fit. Clustering of the particles is handled in step 4, where also trajectories without suitable particles are terminated. This provides a solution to the *merge* and *disappear* events. Finally, in step 5, new trajectories are created for all unused particles. This step solves the *appear* and *split* events. An open source implementation of the proposed particle tracking method is available in the BioImageXD software [9].

2.2. Real data

We tested our particle tracking method with the known phenomenon of echovirus 1 induced clustering of its receptor, $\alpha 2 \beta 1$ integrin [10]. The integrins were labeled with a fluorophore-conjugated antibody, and five consecutive image series were acquired, each consisting of 40 three-dimensional image stacks taken with 30 second intervals with a confocal microscope (Olympus FV1000; slice dimensions 512 x 512, number of slices per stack 19–22, voxel size 0.207 x 0.207 x 0.500 μm^3).

Before tracking, particles were detected from all 200 image stacks with an automatic spot detection method as follows: 1) difference of Gaussian filtering with 3D kernel, 2) voxel classification in three groups with Otsu's automatic thresholding [11], 3) voxels in

the brightest group labeled as particles with connected component labeling (removing particles smaller than seven voxels), and 4) particles quantified for centroid, volume and average intensity (values used in subsequent tracking). The number of detected particles per image stack ranged from 90 to 265. All processing was done with the BioImageXD software [9].

Trajectories were created using the particle tracking method with following parameters: $D_l = 0.0\mu$m, $D_h = 1.0\mu$m, $\sigma_d = 1.0$, $\sigma_s = 1.5$, $\sigma_i = 1.0$, $\sigma_a = \pi$, $w_d = 0.4$, $w_i = 0.25$, $w_s = 0.15$, $w_a = 0.2$, and $\alpha = 0.75$.

2.3. Simulated data

We created three sequences of simulated image stacks to validate our particle tracking method. Every simulated series had 20 time points (dimensions 512 x 512 x 20, voxel size 0.207 x 0.207 x 0.500 μm^3, temporal resolution 30 seconds). For the first time point, particles were set to discrete locations with a random average intensity value between 96 and 128. The number and sizes of the particles were initialized using results of the real data analysis. The speed and direction of movement of each particle was set at every time point. Clustering of the particles was defined to increase their size and average intensity.

The first series was initialized with 103 particles, set to move in random directions at random speeds less than 34.5 nm/sec, without clustering. The second series was initialized with 154 particles, set to move at random speeds less than 69 nm/sec, with 15% of the particles closer than 1.4 μm to each other clustering at every time point. To facilitate clustering, 15% of the particles at every time point were set to move towards one of six randomly initialized target points. Other particles were set to move in random directions. The third series was set to have more merging of the particles than the second series. The speeds of the particles were set to be less than 138 nm/sec. 30% of the particles closer than 1.8 μm were set to cluster and 30% of the particles set to move towards three target points. All simulations were created with the BioImageXD software [9].

The particle tracking method was used to create trajectories for all three simulated series. Parameters were the same for all series, except for D_h: $D_l = 0.0\mu$m, $\sigma_d = 1.0$, $\sigma_s = 0.5$, $\sigma_i = 0.5$, $\sigma_a = \pi$, $w_d = 0.4$, $w_i = 0.25$, $w_s = 0.25$, $w_a = 0.1$, and $\alpha = 0.75$. D_h was set to 1.0μm, 1.5μm, or 2.5μm for the first, second, and third series respectively, to create trajectories with varying particle speeds.

3. RESULTS

3.1. Real data results

We analysed the movement of the particles in real data using four parameters: average number of time points in the tracks, average length of the tracks, average speed of the particles and average directional persistence. The average number of time points is defined as the duration of a track in frames. Directional persistence of track t is defined as

$$\frac{||(t_n)_l - (t_1)_l||}{\sum_{i=2}^{n} ||(t_i)_l - (t_{i-1})_l||}$$

where n is the number of time points in trajectory t.

The results (Table 1) show that average speed of the particles tends to slow down with time. Directional persistence seemed greatest in the first series, meaning less Brownian motion, although the difference is small. The average number of time points per track was smallest in the first series, possibly caused by more active merging of the particles. On the whole, the results of the five series were consistent, and in agreement with the biology of the integrin clustering phenomenon [10], indicating that our particle tracking method was properly measuring the phenomenon. An example of the trajectories created is shown in Figure 2.

Table 1. Results of the analysis of real microscopy data. Four parameters are shown for each series: average number of time points in tracks, average length of tracks, average speed of particles and average directional persistence.

Time point	Avg. tpts	Avg. length (μm)	Avg. speed (nm/sec)	Avg. dp
33 min	8.42	4.15 ± 0.16	21.8 ± 0.5	0.53 ± 0.02
62 min	10.37	3.75 ± 0.16	16.0 ± 0.5	0.44 ± 0.02
92 min	10.10	4.05 ± 0.12	19.6 ± 0.4	0.48 ± 0.01
118 min	9.57	3.76 ± 0.13	18.2 ± 0.5	0.48 ± 0.02
141 min	10.77	3.74 ± 0.18	15.4 ± 0.5	0.46 ± 0.02

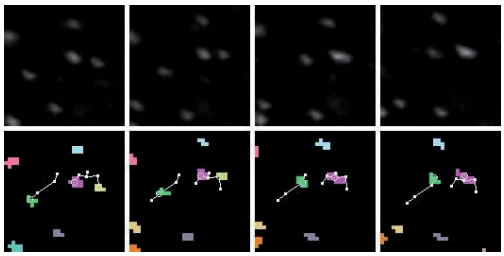


Fig. 2. Particle tracking results of real data of four consecutive time points. Upper row: small regions cut from a single plane of the original image data. Lower row: detected particles from the same regions, with three sample trajectories. Particles belonging to the same trajectory have the same color. Two of the trajectories merge in the third time point.

3.2. Simulated data results

We analysed the movement of the particles in simulated data using four parameters: number of tracks, average number of time points in tracks, average speed of particles and average directional persistence. The number of tracks was used to measure whether the particle tracking method correctly estimated merging of the particles.

We compared the ground truth of the simulated data (Table 2) to the particle tracking results of the same data (Table 3). The average speed and average directional persistence matched within error limits for all simulations. The number of tracks and the average number of time points in tracks also matched for simulation 1, but small discrepancies emerged with the other simulations, as expected with increasing particle speed. Still even the result for simulation 3 was reasonably good, considering that it is an extreme case, where the average speed of particles is over three times higher than the fastest movement detected from real data.

We compared our method to the previously published and validated method by Jaqaman et al [1]. We configured the Jaqaman method using the same values of parameters as used in our method, but since the methods have somewhat different sets of parameters, the results acquired with the Jaqaman method are only approxi-

Table 2. Ground truth of simulated data for method validation. For each simulation, four parameters are shown: number of tracks, average number of time points in tracks, average speed of particles in tracks and average directional persistence.

Series	# of tracks	Avg. tpts	Avg. speed (nm/sec)	Avg. dp
Simulation 1	103	20.0	17.4 ± 0.3	0.25 ± 0.02
Simulation 2	154	18.1	34.5 ± 0.5	0.30 ± 0.02
Simulation 3	146	16.2	69.3 ± 1.1	0.36 ± 0.02

Table 3. Results of the analysis of simulated data. For each analysed series, the same four parameters are shown as in Table 2.

Proposed method	# of tracks	Avg. tpts	Avg. speed (nm/sec)	Avg. dp
Simulation 1	104	19.8	17.9 ± 0.3	0.25 ± 0.02
Simulation 2	164	16.9	35.5 ± 0.6	0.32 ± 0.02
Simulation 3	177	13.3	68.5 ± 1.1	0.39 ± 0.02
Jaqaman method [1]	# of tracks	Avg. tpts	Avg. speed (nm/sec)	Avg. dp
Simulation 1	115	17.7	18.2 ± 0.5	0.28 ± 0.02
Simulation 2	173	15.7	33.8 ± 0.7	0.35 ± 0.02
Simulation 3	147	15.1	70.6 ± 1.7	0.35 ± 0.02

mate. Both methods gave fairly similar results for the average speed and the average directional persistence for all simulations, but our method generally produced better results for the number of tracks and the duration of tracks. Here the Jaqaman method performed better only with the extreme case of the fast moving and highly merging particles of simulation 3, most likely because of an additional linking step done to the tracks.

We also analysed the number of perfect tracks (tracks in the analysis results that have a complete match in the ground truth) and successful linking of individual particles between time points. The results show (Table 4) that our particle tracking method worked almost perfectly with simulation 1, which did not include any merging. Only one incorrect link was made due to a single merging of two particles, and as a consequence, two tracks were set as non-perfect and one new track was created. This illustrates the sensitivity of the perfect tracks metric. Simulation 2 gave a 98.9% accuracy for the links and over 50% of perfect tracks. The fast moving and highly clustering simulation 3 gave 26.0% of perfect tracks, but the linking accuracy was still high (94.7%), proving that the quantitative analysis of the movement of the particles can be reliably derived as presented in Table 3.

Our method resulted in a higher number of perfect tracks and correct links than the Jaqaman method, for all simulations. The results differed from the quantitative measurements in the fast moving and highly merging simulation 3. It seemed that longer tracks generated by the Jaqaman method often had at least one incorrect link assigned.

4. CONCLUSIONS

We have presented a new, greedy single-particle tracking method. This method can be used to efficiently track hundreds or thousands of particles, as it only needs one-pass through the particles in a time series. The method provides a solution to the particle correspondence problem, including handling of merging, splitting, appearing and disappearing events. Options to configure the method for numerous different applications were presented. We tested the method by quantifying particle motion in real, relevant microscopy data, with expected results. We then validated the accuracy of the method using simulated image series. Finally, we compared the accuracy of the method to a previously published, well known, single-particle tracking method. The results showed our tracking method to be reliable and functional.

Table 4. Results of the track analysis of simulated data. The number of perfect tracks compared to the total number of tracks, and the number of correct particle links compared to total particle links, are shown for each analysed series.

Proposed method	Tracks (perfect / total)	Links (correct / total)
Simulation 1	101 / 104 = 97.1%	1956 / 1957 = 99.9%
Simulation 2	94 / 164 = 57.3%	2579 / 2609 = 98.9%
Simulation 3	46 / 177 = 26.0%	2067 / 2183 = 94.7%
Jaqaman method [1]	Tracks (perfect / total)	Links (correct / total)
Simulation 1	79 / 115 = 68.7%	1883 / 1917 = 98.2%
Simulation 2	59 / 173 = 34.1%	2370 / 2535 = 93.5%
Simulation 3	4 / 147 = 2.7%	1594 / 2075 = 76.8%

5. REFERENCES

[1] K. Jaqaman et al., "Robust single-particle tracking in live-cell time-lapse sequences," *Nat. Methods*, vol. 5, no. 8, pp. 695–702, 2008.

[2] K. Shafique and M. Shah, "A noniterative greedy algorithm for multiframe point correspondence," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 27, no. 1, pp. 51–65, 2005.

[3] C.J. Veenman, M.J.T. Reinders, and E. Backer, "Resolving motion correspondence for densely moving points," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 23, no. 1, pp. 54–72, 2001.

[4] E. Meijering, I. Smal, and G. Danuser, "Tracking in molecular bioimaging," *IEEE Signal Proc. Mag.*, vol. 23, no. 3, pp. 46–53, 2006.

[5] W. Tvarusko et al., "Time-resolved analysis and visualization of dynamic processes in living cells," *Proc. Natl. Acad. Sci. USA*, vol. 96, no. 14, pp. 7950–7955, 1999.

[6] P. Ruusuvuori et al., "Evaluation of methods for detection of fluorescence labeled subcellular objects in microscope images," *BMC Bioinformatics*, vol. 11, pp. 248, 2010.

[7] A. Dufour et al., "Segmenting and tracking fluorescent cells in dynamic 3-D microscopy with coupled active surfaces," *IEEE Trans. Image Proc.*, vol. 14, no. 9, pp. 1396–1410, 2005.

[8] C. Zimmer et al., "On the digital trail of mobile cells," *IEEE Signal Proc. Mag.*, vol. 23, no. 3, pp. 54–62, 2006.

[9] P. Kankaanpää, L. Paavolainen, V. Marjomäki, J. Heino, and D. White, "BioImageXD," `http://www.bioimagexd.net/`, 2011.

[10] M. Karjalainen et al., "Echovirus 1 infection depends on biogenesis of novel multivesicular bodies," *Cell. Microbiol.*, vol. 13, no. 12, pp. 1975–1995, 2011.

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

PIII

QUANTITATIVE ANALYSIS OF DYNAMIC ASSOCIATION IN LIVE BIOLOGICAL FLUORESCENT SAMPLES

by

Pekka Ruusuvuori*, Lassi Paavolainen*, Kalle Rutanen*, Anita Mäki, Heikki Huttunen & Varpu Marjomäki

Submitted manuscript

Quantitative analysis of dynamic association in live biological fluorescent samples

Pekka Ruusuvuori[1,2,**], Lassi Paavolainen[2,3,⋆], Kalle Rutanen[1,4,⋆], Anita Mäki[2], Heikki Huttunen[1], Varpu Marjomäki[2]

1 Department of Signal Processing, Tampere University of Technology, Tampere, Finland
2 Department of Biological and Environmental Science/Nanoscience Center, University of Jyväskylä, Jyväskylä, Finland
3 Department of Mathematical Information Technology, University of Jyväskylä, Jyväskylä, Finland
4 Department of Mathematics, Tampere University of Technology, Tampere, Finland
⋆ These authors contributed equally.
∗ E-mail: Corresponding pekka.ruusuvuori@tut.fi

Abstract

Determining vesicle localization and association in live microscopy may be challenging due to non-simultaneous imaging of rapidly moving objects with two excitation channels. Besides errors due to movement of objects, imaging may also introduce shifting between the image channels, and traditional colocalization methods cannot handle such situations. Our approach to quantifying the association between tagged proteins is to use an object-based method where the exact match of object locations is not assumed. Point-pattern matching provides a measure of correspondence between two point-sets under various changes between point-sets. Thus it can be used for robust quantitative analysis of vesicle association between image channels. Results for a large set of images shows that the point-pattern-based method performs comparably in colocalization studies for fixed cells and demonstrates improved capability to detect association of closely located vesicles in live cell-microscopy.

Introduction

Live cell-imaging in subcellular scale has revolutionized the way cells are studied in molecular cell biology. As microscopy and imaging devices have enabled efficient and accurate live cell-imaging in high resolution, the demand for automated image analysis and interpretation has become obvious. For example, tagging proteins with specific fluorescent stains enables studying various cell functions through detection of protein-specific cell organelles, provided that the fluorescence-signal captured in digital images can be accurately analyzed. The spatial pattern and location [1, 2] of the detected signal may reveal the cell function or role of proteins, and colocalization of tagged proteins is in particular of interest [3]. In cell biology, close association of cellular structures, such as vesicles, occurs, e.g., in situations when vesicle pathways follow similar tracks or when close association is meaningful and leads to possible fusion events. There are very few tools available to study association. Instead, there are several tools to study colocalization, represented by different-colored voxels occupying the same spatial location. Association may be defined by a chosen distance between the objects. If differently colored objects are frequently associated they may be considered to keep near each other over time and follow each other in the cell cytoplasm. Analysis of closely associated objects in fixed cells allows accurate analysis, without errors caused by the movement of the objects between subsequent imaging of different channels – provided that the channels are aligned. However, in a live-imaging setup the quantification of sudden and transient events is challenging [4], and live imaging is prone to such errors that depend on the speed of the imaging setup.

Live imaging of cytoplasmic vesicles that are elicited from the plasma membrane after e.g. growth-factor stimulation reveals important aspects of the fate of these crucial cellular regulators. Sorting of

the growth-factor receptor to cellular compartments that mediate degradation of the receptor causes attenuation of the signaling whereas recycling of the receptor back to the plasma membrane allows further signaling. Recently, it has been acknowledged that several growth-factor receptors (e.g. epidermal, platelet derived and vascular endothelial growth factors) reside in close proximity to several cell surface integrin receptors (e.g. $\alpha 2\beta 1$, $\alpha 5\beta 1$) and some of them even have direct effects on each other's function [5, 6]. Like growth-factor receptors, integrins are important signaling receptors regulating vital cellular processes, like cell survival, differentiation and proliferation. As the growth-factor receptors and integrins use similar signaling pathways and show mutual regulation of important cellular processes it has become important to follow their movement in live cells. The dynamic nature of these endocytic and recycling pathways and the fact that these pathways may interact with each other provides complexity and makes reliable interpretation of the imaging results a very challenging task.

Traditionally colocalization analysis has been a subjective process, performed as a visual comparison of overlapping signal in two channels. Recent increase of the amount of image data and need for statistical analysis have shifted colocalization towards a more quantitative analysis. In many cases, colocalization is only partial leaving some voxels close by suggesting that fusion between the two colors has been meaningful but not complete. The partial colocalization may also indicate compartmentalization inside the structures. Therefore, instead of just colocalization, one could measure association with both voxels in close proximity and, in live images, voxels moving as one unit even if not entirely colocalizing. One example case where determining true colocalization may be ambiguous is given in Fig. 1 where a single structure or a pair of vesicle-like structures in close association is shown, and in Fig. 2 where the same vesicle is shown in a larger context with several similar structure pairs in two fluorescence channels.

One of the most straightforward quantitative methods include the pixel or voxelwise analysis where intensity of each element in an image channel is plotted against its counterpart in the other channel. This method, accompanied with statistical analysis of the significance of colocalization [3, 7] remains as one of the most common ways to estimate colocalization. Typically the studied biological phenomenon is related to specific subcellular organelles that have been labeled. Thus, instead of observing all the pixels or voxels containing signal it may be of interest to concentrate on the detected spots. Since the imaging resolution enables detection of individual organelles, it is possible to determine a quantitative estimate of colocalization objectwise [8–11]. Such approach makes it possible to take into account small changes in image due to imaging lag or other errors in the imaging phase. This can be advantageous in live cell-microscopy, where particle trafficking may be fast, compromising the accuracy of pixelwise colocalization analysis.

The essential idea of the proposed method is to determine the mapping between images such that objects found in an image are paired with objects found in the compared image. Such problem is commonly addressed in dynamic monitoring of subcellular objects using live imaging where fluorescence-tagged organelles are followed throughout the imaging sequence [12–14]. However, to the best of our knowledge, matching of point sets using two channels of only one timepoint has not been proposed before for studying protein localization in fluorescence microscopy. Though both analyses rely on the property that the same target is imaged, colocalization or association analysis differs from live cell particle tracking in the assumption that the same objects may not be visible in the compared image. Also limiting to still images at sparse time resolution hinders the use of tracking methods to determine the mapping between frames.

Matching unpaired point-sets under a given class of transformations is a problem which can be divided into two main classes. In what is often called registration, a transformation close to an optimum matching transformation is known beforehand, and the problem is to refine that transformation to a nearby local optimum. By the prior information this local optimum is then also hoped to be close to a global optimum. In contrast, in point-pattern matching nothing is known about the position of the global optima. Registration is local non-linear optimization, while point-pattern matching is global non-linear optimization.

In general, any point-set matching algorithm could be used as a basis of the proposed association analysis. Here we design an algorithm, based on point-pattern matching, particularly suited for the analysis of fluorescence microscopy images. PPM algorithms can be designed to be robust against changes in image geometry, e.g., changes in the number of detected particles, scale or shifts in image. In our application, the number of detected particles can typically vary significantly, whereas large systematic changes such as rotation or scaling between corresponding point locations are not expected. Thus, we may limit to translations and allow the method to only accept the transformation when the point sets are close to each other. Such small changes can be compensated by using PPM, however, they may cause the objects to be a miss in traditional co-localization analysis where co-localization is determined by pixelwise comparison of channels.

As an alternative way of finding point-pair correspondense, we use the Iterated Closest Points (ICP) [15] [16], or perhaps better described by Iterated Corresponding Points, which is a popular class of algorithms for solving the registration problem. There are many algorithms in this class, many of which are reviewed in [17]. An important trend in ICP-based algorithms has been to make them robust to missing/extraneous points (subset matching) and noise. As practical variants of this type, we mention the Biunique ICP [18] and the Trimmed ICP [19]. We provide a comparison to the Biunique ICP algorithm by making its initial transformation match the centroids of the point-sets; such an assumption may or may not hold for given microscopic images.

In this study we propose a novel method for association analysis. We show that the method is robust against moderate translations and object movement between image channels. The applicability of the method was demonstrated by following the entry of $\alpha 2\beta 1$ integrin and epidermal growth-factor receptor (EGFR) after triggering their internalization from the plasma membrane using fluorescent antibodies and fluorescent growth-factor, respectively. The results indicate that the vesicles containing $\alpha 2\beta 1$ integrin and EGF are very close to each other along their internalization pathway to the center of the cell. Quantitative comparison between the proposed PPM-based association algorithm and the traditional colocalization estimates suggests that the results are in line for fixed cell experiments, with the new method providing improved detection of association by closely located objects in live cell experiments.

Materials and Methods

Microscopy data and live cell-imaging

The A549 (human lung carcinoma) cell line (ATCC) was used in all experiments. The cells were grown in Dulbeccos modified Eagle Medium (DMEM; Gibco) supplemented with 10% inactivated fetal calf serum (FCS), L-glutamine and penicillin-streptomycin (Gibco BRL, Paisley, UK) at 37°C, in 5% CO2. Cells were plated on cell culture chambers (Ibidi 15 μ-slide 8-well) two days before and starved (serum-free DMEM) for the last 16 hours before the experiment.

EGFR was stimulated and followed by microscopy by adding first biotinylated EGF (0,5 μg/ml) on ice for 45 min and washed extensively. Then streptavidin Alexa 488 (5 μg/ml) was incubated on ice for 45 min and washed. $\alpha 2\beta 1$ integrin clustering was done as described previously [20]. Briefly, specific antibody (A211E10; a kind gift from Dr. Fedor Berditchevski, Institute of Cancer studies, Birmingham, United Kingdom) against integrin was bound for 45 min on ice. Cells were washed extensively and incubated with a clustering secondary antibody (goat anti-mouse Alexa 594; Invitrogen). After washing, the cells were incubated in serum-free DMEM at 37°C to allow internalization. In practice, biotinylated EGF and integrin antibody was added simultaneously on cells and washed, and subsequently the fluorescent conjugates were added together on ice. The Ibidi slides were transferred to Zeiss Cell Observer HS (37°C, 5% CO2). The live imaging was performed using Colibri LED light source at 470 (41%) and 590 (46%) wavelengths for Alexa 488 and 594, respectively. Videos with 5.77 seconds (Live I), and 5.16 seconds (Live II) intervals were taken. In Figs. 1 and 2, close-ups of vesicles imaged using the maximum speed

are illustrated.

For control experiments imaging perfectly colocalizing intensities, $\alpha 2\beta 1$ integrin clustering was induced using similar amounts of two fluorescent conjugates (goat anti-mouse Alexa 488 and 594). In addition, control videos of stable, unmoving vesicles were imaged after cells were fixed with 4% PFA for 20 min. For comparison, quantification of colocalization was determined using colocalization algorithms embedded in the free, open source software package, BioImageXD (`http://www.bioimagexd.net`, [21]). Only high intensity integrin clusters were selected for the colocalization analysis by first removing the background by subtracting the most common intensity value from the images. Next, masks for colocalization analysis were defined by filtering images with Difference of Gaussian filter and subsequent thresholding. Small particles (less than 3 pixels for fixed and 8 pixels for live cells) were removed from the masks in order to limit the detection of noise or small debris as spots. Finally, the masks were used for excluding background from the colocalization analysis.

Simulated data

One of the motivations for using a point-set-based method for determining association stems from the fact that the image channels may be shifted or aligned non-ideally during the measurement process. The measurement consists of two separate image acquisitions with different filter applied in order to capture the desired wavelengths, corresponding to the specific fluorescent protein markers. To illustrate the robustness against such misalignments, we generated image sets with varying level of global displacement in (x, y) space, and an additional random movement term for individual spots. The experiment can be considered as a simulated live cell experiment where the effect of potential displacement and movement due to imaging delay can be studied in a controlled manner. Importantly, the simulated experiments allow us to study the matching accuracy directly through examining the correct matches, mismatches and missing pairings. Thus, the simulations can be used both for evaluating the usefulness of the point-set based association analysis and for quantitatively comparing our PPM algorithm with state-of-the-art registration algorithm.

Simulated experiments were generated as images with additive background noise and spots with varying intensity as foreground objects similarly as in [14]. The simulation parameters, *i.e.*, number of objects, object size and intensity, were inferred from real data in order to generate data that resembles realistic experimental image data. The key parameters of the simulation process were varied as follows. A global translation in random direction between the image channels was added using parameter space $\{0, \ldots, 3\}$ (in pixels), where 0 corresponds to no global shift, and 3 corresponds to maximum allowed magnitude of global translation. The other key parameter controls the random movement of individual objects. The movement was implemented as an additive random term drawn from zero-mean normal distribution $N(0, \sigma_{mov})$, where $\sigma_{mov} \in \{0, \ldots, 5\}$ defines the magnitude of movement as deviation (in pixels) around the coordinate point. In the simulated images, the pixel size was set to correspond to $198 \times 198nm$. Furthermore, we generated three scenarios corresponding to low, intermediate, and high levels of association between channels 1 and 2, where the association levels with respect to channel 1 were set as 0.7477 in high, 0.5225 in intermediate, and 0.2072 in low association scenario. For channel 2, the association levels were set to 0.8557 in high, 0.4715 in intermediate, and 0.2805 in low association scenario. Number of objects was set to vary around a fixed number of 111 objects per channel, and the association levels between channels were controlled by adding and removing objects from random locations such that exactly the pre-set association levels were obtained. Finally, the simulations were replicated 10 times for each parameter settings. To summarize, the simulation study consisted of 4×6 parameter combinations repeated in three association scenarios, each replicated 10-fold, resulting to 720 images with two channels. Object locations were extracted from the images using the DoG spot detection as described earlier.

Determining vesicle association using colocalization analysis

Analysis of vesicle localization between two fluorescence-labeled image channels can be done in 2D or 3D. The true geometry of the samples means the 3D imaging with confocal microscope and subsequent processing in 3D enables more accurate analysis in theory. However, in live imaging such setting is not always applicable, since 3D imaging is time consuming whereas intracellular trafficking and object movement may be fast. In addition, use of 2D imaging enables higher throughput making 2D images a commonly used compromise in live colocalization studies. Thus, we will use 2D images taken in time-lapse live cell-imaging settings.

Some of the most widely used automated statistical colocalization estimates rely on correlation of the pixel intensities between the image channels. Here we use two pixelwise colocalization estimates, Pearson correlation and Manders' coefficient [22], as reference methods for comparison purposes. Pearson's correlation $r_p \in [-1 \ldots +1]$ between channels g_1 and g_2 is given as

$$r_p = \frac{\Sigma_x (g_1(x) - \overline{g}_1)(g_2(x) - \overline{g}_2)}{\sqrt{(\Sigma_x (g_1(x) - \overline{g}_1)^2)\,(\Sigma_x (g_2(x) - \overline{g}_2)^2)}} \tag{1}$$

where $\overline{g}$ denotes the channel mean intensity. Positive correlation indicates match between channel intensities and suggests there exists colocalization of some level, whereas values close to zero show no correlation and thus give no evidence of colocalization. Possible negative correlations would indicate a negative relation of pixel intensities between compared channels. Another well-known statistical measure of pixelwise colocalization is the Manders' colocalization coefficient M_i, which is defined for channel g_i as

$$M_i = \frac{\Sigma_x g_{i,\mathrm{coloc}}(x)}{\Sigma_x g_i(x)} \tag{2}$$

where the colocalized proportion of the signal $g_i(x)$ is given by

$$g_{i,\mathrm{coloc}}(x) = g_i(x), \quad \text{if } g_{ref}(x) > t_{ref} \tag{3}$$

where the selection of the threshold value t_{ref} for the reference channel is essential in determining the colocalizing signal. Threshold selection, however, is not trivial and despite automated methods are available [3, 23] the selection may sometimes need to be adjusted by user. Recently, a colocalization measure combining both r_p and M_i has been proposed in [24]. Here we have used the masks presented earlier to define the region of interest for the quantitative colocalization estimators. The results by both pixelwise estimators are obtained using the implementation available in BioImageXD.

Vesicle association analysis using point-pattern matching

Given two finite sets of points, say $P, Q \subset \mathbb{R}^d$, and a set $F \subset (\mathbb{R}^d \to \mathbb{R}^d)$ of allowed transformations, a point-pattern matching algorithm attempts to determine a transformation $f \in F$ such that at least some subset of the points in $f(P)$ would match some subset of Q. In this paper we will fix F as the set of translations, i.e. each $f \in F$ is of the form

$$f(p) = p + t, \tag{4}$$

for some $t \in \mathbb{R}^d$.

The term match must be defined carefully to obtain meaningful results from a point-pattern matching algorithm. For example, defining a match between $P' \subset P$ and $Q' \subset Q$ by the relation $f(P') = Q'$ means that a possible match is destroyed by an arbitrary small translation to any point in either P' or Q'. Since any real-world measurement contains noise, a practical point-pattern matching algorithm needs to be able to maintain a match under small deviations of the point-sets, i.e. to be robust under noise. In

addition, since in practice some measurements can be missing or extraneous in either P or Q, a practical point-pattern matching algorithm should be able to find matches between subsets $P' \subset P$ and $Q' \subset Q$ also.

Considering the application area, if we are to apply point-pattern matching to determine colocalization, then the need for both kinds of robustness is seen as follows. First, it is likely that the detected object-sets do not match perfectly even in the case of nearly perfect colocalization, since the measurements are from different objects. Second, differences originate from biological variation which leads to varying levels of colocalizing points. Third, even when the targeted objects are colocalized, the point-sets include "noise" from object movements.

The way the quality of a matching is measured affects the robustness of a point-pattern matching algorithm tremendously. For example, if there were an exact copy of the model point-set (what to find) in the scene point-set (from where to find), but there were an additional distant cluster of points, we would like the cluster to not affect the matching result. Many papers on point-pattern matching concentrate on minimizing the Hausdorff distance between point-sets [25] [26] [27], defined as

$$d_H(P,Q) = \max\{\sup_{p\in P}\inf_{q\in Q} d(p,q), \sup_{q\in Q}\inf_{p\in P} d(p,q)\}. \tag{5}$$

Unfortunately, this distance can be made arbitrarily large by introducing an additional distant point in either X or Y. For this reason, we reject the minimization of Hausdorff distance as a practical matching strategy. To improve on the robustness issue, several authors have proposed using partial Hausdorff distance instead, where the supremum is taken only over a given percentage of the smallest distance values. By doing this, it is hoped that the procedure correctly rejects any points that are too far away to be meaningful for the matching. Unfortunately, no percentage is small enough to guarantee that such outliers are correctly rejected; the number of additional distant points can always be increased so that the ratio of outliers exceeds the given percentage. For this reason, we also reject the minimization of the partial Hausdorff distance as a practical matching strategy.

Instead, we adopt the matching criterion from [28]. Intuitively, a point $p \in P$ matches a point $q \in Q$ under $f \in F$, if $f(p)$ is close to q. By extension, $P' \subset P$ matches $Q' \subset Q$, if each point in $f(P')$ has a unique match in Q'. This intuition is made exact in the next section where we formulate the matching algorithm. Our approach is to find a match under a given pointwise matching distance, and then determine the amount of association from this correspondence.

Van Wamelen et al. [28] presented a fast algorithm for point-pattern matching in $\mathbb{R}^2$ under conformal affine transformations, with a robust matching criterion which we adopt. We do not, however, select this algorithm, because the application we are dealing with requires a high degree of robustness. Moreover, the algorithm can fail on given parameters, for which there is no systematic way to set to suitable values beforehand. In fact, the requirement for finding a match if such exists sets a limit for possible algorithms, and we are not aware of existing studies that would fit our application. Instead, we will construct one in the next section.

Point-pattern matching under translations

In the following we present an algorithm for point-pattern matching between two finite point-sets in $P, Q \subset \mathbb{R}^d$, $|P| = n$, $|Q| = m$, when the class of transformations F is given by translations $f \in F$ which align P to Q:

$$\exists p \in P, q \in Q : f(p) = q. \tag{6}$$

The algorithm either reports that there is no match, or reports a bijection between subsets $P' \subset P$ and $Q' \subset Q$, such that P' and Q' match.

Matching criterion

Let $P = \{p_1, \ldots, p_n\} \subset \mathbb{R}^d$ and $Q = \{q_1, \ldots, q_m\} \subset \mathbb{R}^d$ be two sets of points. Let $\|\cdot\| : \mathbb{R}^d \to \mathbb{R}$ be a norm in $\mathbb{R}^d$. Given $\delta \in \mathbb{R}$, $\delta \geq 0$, called the matching distance, and $\alpha \in \mathbb{R}$, $0 < \alpha \leq 1$, called the matching ratio, the point-set P is said to match the point-set Q if there is a set $M \subset P \times Q = \{(p_{i_1}, q_{j_1}), \ldots, (p_{i_{|M|}}, p_{j_{|M|}})\}$, called a matching, where each point in P and Q is part of at most one pair,

- $\forall s \in [1, |M|] : \|p_{i_s} - q_{j_s}\| \leq \delta$, and
- $|M| \geq \alpha n = M_\alpha$.

In addition to the matching criteria above, we also set a limit for the bias of a match, which will be discussed next.

Bias of a matching

Even if we find a matching M, it might be that the matching is of poor quality. Assume that for a translation $f \in F$ it holds that $Q = f(P) \cup \{q'\}$, where $q' \in \mathbb{R}^d$, and it holds that $\|q' - q\| = \delta$ for some $q \in Q$. That is, $f(P)$ matches Q perfectly, but there is an extraneous point q' nearby q. Let $f' \in F : f'(x) = f(x) + (q' - q)$. Then also $f'(P)$ matches Q. However, the matching given by f' is of poor quality because the difference vectors between the pairs in the matching are all (except one) in the same direction. We would rather want the error to be distributed uniformly in all directions. To avoid these systematically poor matchings, we define the bias of a matching M by

$$\text{bias}(M) = \frac{\|\sum_{s=1}^{|M|} [p_{i_s} - q_{j_s}]\|}{\delta |M|}. \tag{7}$$

We will then define a maximum allowed bias $\beta \in [0, 1] \subset \mathbb{R}$, and require from a matching that $\text{bias}(M) \leq \beta$. The matching provided by f' in the example can then be avoided, by choosing β properly, since the errors average near to zero with f and near to one with f'.

Nearest neighbors searching

In nearest neighbors searching, the problem is to report those k points in Q which are closest to a given point $p \in \mathbb{R}^d$. To search for the k nearest neighbors in Q, we shall use the kd-tree data structure with the sliding midpoint splitting rule [29]. This data structure can be constructed for Q in $O(dm \log m)$ and $O(dm)$ space.

In the following we shall assume that for each nearest neighbors search there will not be two points in Q with the same distance to p. This simplification is without loss of generality; in an actual implementation one gives a secondary order to equidistant points, for example, by giving them an ordered labeling. Let $B_r(x) = \{y \in \mathbb{R}^d : \|y - x\| \leq r\}$, where $r \in \mathbb{R}$ and $x \in \mathbb{R}^d$.

Maximum bipartite matching

Let $G = (V, E)$ be a graph (all graphs in this paper are directed and simple), where V is a finite set of vertices, and $E \subset V^2$ is a finite set of edges. A graph is called bipartite, if it can be decomposed as $V = L \cup R$ with $L \cap R = \emptyset$, and $E \subset L \times R$. A matching of G is a subset $M \subset E$ without common vertices (this definition is consistent with the matching of point-sets after we define V and E in the algorithm). A maximum matching of G is a matching of G with the largest possible number of edges $|M|$. We use the Hopkroft-Karp algorithm [30] to compute a maximum matching in a bipartite graph in $O(E\sqrt{V})$ worst case time, or in $O(E \log(V))$ average time for random graphs.

PPM Algorithm

Assume a matching distance δ, a matching ratio α, a maximum allowed bias β, and a maximum number of nearest neighbors k. The point-pattern matching algorithm consists of repeating the following steps for each pair $(p, q) \in P \times Q$.

1. Let $t = q - p$.
2. For each $j \in [1, n]$, find k (or as many as possible) nearest points $N_j \subset Q$ to $p_j + t$ in $Q \cap B_\delta(p_j + t)$.
3. Let $G = (V, E)$ be a (bipartite) graph, where $V = P \cup Q$, and $E = \cup_{j=1}^{n}(\{p_j\} \times N_j) \subset P \times Q$.
4. Find a maximum bipartite matching M in G.
5. If $|M| < M_\alpha$, start with a new pair (p, q).
6. If bias$(M) > \beta$, start with a new pair (p, q).
7. Return M as a matching between P and Q.

Object-based association analysis using point-pattern matching

In this section we apply the above point-pattern matching algorithm for estimating protein association between image channels. First, since we are using point-pattern matching, the fundamental requirement is to obtain point-sets where the points denote fluorescence-labeled objects detected from the image. These objects, appearing as resolution-limited, low contrast blurry spots are challenging to extract, but methods for detection have been presented and evaluated in the literature [31, 32]. Here we leave the discussion about the selection of detection algorithm out of the scope and note that we made the method selection based on experimenting and used the method described earlier for detecting the fluorescent objects, and the centroids of detected objects from two channels form the point-sets P and Q.

Given the point-pattern matching algorithm under translations described above, it is now possible to define the similarity between point-sets by finding a transformation between P and Q. This transformation gives us the following information. First, if a match cannot be found, there is no association at the specified level of correspondence, which is given as the matching ratio α. Second, if a match has been found, we can check the transformation in order to find out how much the point coordinates had to be altered in order to find a match. A moderate transform suggests that true association exists at the matching ratio α, whereas a drastic transformation tells about a correspondence found by chance which should not be counted as association. Using these observations, a rule for estimating association under the restrictions can be defined by

$$C = \max\{\alpha \in [0, 1] : P \text{ matches } Q\}, \tag{8}$$

where α is the matching ratio, and the matching criterion is as defined previously. While α is a real number, the matching algorithms only differ on finitely many values of α, corresponding to the different number of required points $M_\alpha \in [1, n]$ in a matching. The values of C in Eq. 8 can be interpreted similarly as, for example, direct pixel or objectwise overlap values – values close to the maximum value of 1 correspond to high level of association and values close to the minimum value 0 mean there is very little association – with the only difference being that also closely located objects are allowed and direct overlap is not required in the case of the point-pattern-matching-based association. Fig. 3 illustrates an example matching using simulated data with 300 points drawn from normal distribution and 0.2 ratio of missing points between channels, and with translation and noise introduced for the point-set. The matches are marked with lines in the close-up, and circles illustrate the search range defined with parameter δ.

Notes on implementation and 3D

Using two-dimensional projections as a basis of association analysis is a choice made for practical reasons. However, cells are naturally 3D objects, and thus confocal microscopy suits well for true 3D colocalization studies. In such cases, the objects can also be extracted in 3D, yielding an additional location coordinate. Although 2D imaging is used in the experiments of this article, estimating association with the new point-pattern-based method is also possible in 3D. Majority of the computational workload comes from the matching process, thus we implemented the PPM matching algorithm in C++. The implementation is available from supplemental site `http://www.cs.tut.fi/sgn/csb/VesicleAssociation`.

Results

We present quantitative results for both simulated and real data. Simulated data, for which ground truth is available, is designed for demonstrating the properties and validating the performance of the point pattern matching based association algorithm. Importantly, simulation serves as a powerful tool for validating the novel approach based on point-set matching, and it enables comparing the proposed matching algorithm to state-of-the-art algorithm directly, using matching accuracy as the criterion. Real data, collected using experimental setup described in Materials and Methods section, allows comparison with traditional colocalization measures in true use cases. Due to the lack of ground truth only indirect measures of accuracy can be used, as is typically the case for real data. The real experiments, however, can be validated through the biological setup. We have used two different scenarios; fixed cells with very high level of colocalization between the labeled structures and live cell experiments with known association of labeled structures without a perfect overlap.

The results section starts by an extensive simulation study where we demonstrate the properties of the PPM-based association algorithm and compare it with ICP matching. Furthermore, we show how the proposed method is able to detect associations under circumstances where traditional colocalization estimates fail to produce accurate results. Second, we present a comparison with fixed cells where traditional colocalization measures are known to perform well. Third, we show how the association algorithm performs in live cell experiments and again compare against traditional colocalization methods. Finally, we demonstrate robustness to artificially generated imaging delay between frames in live cell experiments.

Robustness to channel displacement and object movement with simulated data

One of the key advantages in simulation is that the ground truth, that is, the correspondence between objects in the two channels is known. This enables the use of quantitative measures of matching accuracy instead of evaluating the results only based on the association estimate. Here we use the ground truth information for determining true positive matches (object paired with a correct counterpart), false positive matches (mismatch, object paired with a wrong counterpart) and false negative matches (object not paired though counterpart exists), from which the precision, recall, and subsequently, the F-measure are determined as explained in [33].

In Fig. 4, the F-measures by PPM (triangle) and ICP (circle) are shown as summaries across all replicates; in (a) the results are presented for different values of object movement σ, and in (b) as a function of global translation. All three simulation scenarios are shown in the same figure; high association level with solid line and light grey, intermediate association level with mid grey dash line, and low association level with dotted dark grey line. The results are averaged over the 10 replicates, and parameter combinations leading to undefined F-measure (due to failed matching where none on the pairings were true positives) were left out of the graph. The results for both algorithms are obtained with matching distance $\delta = 6$, which corresponds to roughly $1200nm$ distance with the simulated data. Further, we used the simulation experiment for studying the effect of the matching distance parameter

in both point-set based algorithms (Supplemental results) without significant changes in the relative performances.

The results in Fig. 4 confirm that the point-set based approach for determining object association is able to handle moderate object movements (Fig. 4 (a)) and that global translation between image channels can be eliminated very efficiently (Fig. 4 (b)). When examining the results given by the two matching algorithms, it can be seen that point-pattern matching slightly but consistently outperforms the iterated closes point method for high and intermediate association values. Moreover, the results confirm that, as can be expected, both of the methods relying on point-set matching generally perform more accurately when the level of association is higher, but the drop in matching accuracy due to lowering association level from approximately 0.80 to 0.25 was not dramatic for the point-pattern matching, whereas the iterated closest points based method resulted to several failed pairings with low association values (leading to missing values in the graphs). Given that it is common in this application that the association level may be low either due to the lack of association in the studied biological phenomenon, or due to severe imbalance in the number of objects in the channels, we will conclude based on the presented results that the proposed PPM-based method should be preferred for matching the point sets. In the remaining experiments, we will only use PPM for representing the proposed point-set based approach.

Comparison with traditional colocalization estimates with fixed cells

Second, we estimate colocalization for fixed samples from human lung carcinoma cells. This data can be considered as a reference set since the cells are fixed and $\alpha 2\beta 1$ integrin was labeled using similar amounts of two fluorescent conjugates inducing almost perfect colocalization. The quantitative results for four sets of fixed cells (denoted by Fix I - IV) comprising 277 images with two fluorescence channels are show in Table 1, where $\mu\#_i$ is the average number of objects and $C_{PPM}i$ is the association estimate by the new PPM-based method in channel i. For PPM, we used matching distance $\delta = 4$, and maximum bias $\beta = 0.1$. The matching distance was determined through setting a limit for the allowed area of determining association using expert knowledge and information about the pixel dimensions; here $\delta = 4$ corresponds to $< 800nm$. As expected, the association results given by the PPM-based method are rather high which is well in accordance with the experimental setup.

For comparison, we estimated the colocalization by the commonly used Pearson correlation (r_p) as well as with the Manders' colocalization coefficient (M_i) where the DoG method was used for masking as explained earlier, and i refers to the image channel. Also the direct overlap percentage of pixels ($O_{pix}i$) was calculated for both channels using the masked images. The results suggest that the PPM-based method yields colocalization estimates which in general behave similarly as the Manders' colocalization coefficient as well as the traditional pixelwise colocalization estimate. Importantly, allowing the point-set based method to determine association within the matching distance instead of limiting to direct colocalization does not seem to result to overestimated values when compared to the traditional colocalization measures.

Association and colocalization in live cell-imaging

Next, we assess the PPM-based method in live microscopy where movement of vesicles potentially affects to traditional colocalization estimates. We use two live microscopy datasets with integrin labeled cells from the A549 cell line imaged at over 150 time-points. The labeled structures are now different, thus direct colocalization is expected to be low but the structures are known to be closely associating. An example image can be seen in Fig. 5 (a) and the objects detected from both channels are shown in (b). In total, there are 330 images with two fluorescence channels, and the average particle counts (size limited to be 8 pixels minimum) are given in Table 2. The PPM maximum bias was again set to $\beta = 0.1$, and matching distance was set to $\delta = 4$ corresponding to $< 800nm$ distance, which defines the allowed area for determining association. The matching process is visualized in Fig. 5 (c) where the matching area (defined by the matching distance) is shown with white circle around the transformed point locations, and

paired objects are shown with blue lines connecting the object centers after applying the transformation by PPM algorithm.

In Table 2, the results for a live cell experiments are given. Based on the results, the traditional methods do not give much colocalization whereas the association estimate by PPM suggests that, even though the structures are not directly overlapping, there exists a certain level of association which is only revealed by the new point-set based method. The close-up area shown in Fig. 5 (c) shows examples of points located very closely in the two channels having very little or no direct overlap, but which are paired by the PPM algorithm, leading to detected association. This is in line with control colocalization measurements of 3D data in confocal microscopy [34] and shows a high amount of association but very limited direct colocalization. Results for the two datasets (denoted as Live I & II in Table 2) are almost identical. The data thus suggest that EGFR and integrin-positive structures stay close but separate after their triggered internalization. This has also been recently shown by us using confocal 3D colocalization analysis [34].

Robustness to imaging delay with real data

The robustness of the algorithm against delay in imaging process is studied next. Different levels of delay are considered by using frames $I_0(n)$ and $I_1(n+t)$ for determining colocalization, where I_0 and I_1 are the two image channels, n is the current frame and $t \in \{0, \ldots, 3\}$ is the delay in frames, where the length of delay is defined by the imaging frequency (here the delay is multiplicates of 5.7s). The results are shown in Fig. 6 both as numerical estimates and as relative values normalized by the first, non-delayed datapoint. For PPM we evaluated the effect of search range parameter δ by giving values 4, 6, 8, 10. Given the rather long delay, this time it is justified to use larger values for δ. The objectwise overlap was calculated for comparison purposes using the same segmentation masks which were also used as a basis of PPM matching. The results represented as relative to the non-delayed case reveal how the direct objectwise colocalization estimate and the PPM method with too small search range are sensitive to imaging delay, whereas association values obtained with larger search ranges (here, 8 and 10) are less affected by the artificially introduced delay. The results also indicate that PPM-based association estimates are larger than direct overlap-based colocalization, as was expected due to the lack of direct colocalization of the labeled structures.

Discussion

In this article, we have presented a computational method for estimating protein association between two image channels using an object-based point-pattern matching approach. The method searches a mapping between point sets detected from the image channels through pairing individual objects detected in both channels. The association is estimated as the fraction of paired objects, creating a measure that is directly comparable to the colocalization percentages between overlapping objects or masked image pixels/voxels. The advantage of the proposed method stems from the inherited robustness of point-pattern matching against directed movement between the image channels, which could be potentially caused by misaligned image channels. Also other moderate transformations, such as random object movement during the lag in the imaging of the fluorescence channels, can be compensated within the search area. Any movement of the fluorescence-labeled subcellular structures will potentially lead to missed colocalization unless the movement is compensated, whereas the proposed point-pattern-matching-based method is able to resolve association in cases where moderate object movement exists.

The proposed PPM-based method was experimentally validated with several time-lapse image sequences with hundreds of image frames, each image typically containing in the order of hundreds of fluorescence-labeled subcellular objects. The results obtained for simulations demonstrate the benefits of the proposed method. Misalignments and random movement of individual objects cause significant drop

in the performance of traditional methods, whereas the matching based PPM and ICP methods performed well when the random movement was moderate. Moreover, the results obtained for fixed human lung carcinoma cells with known colocalization of two $\alpha 2\beta 1$ integrin fluorescent conjugates show that the estimates by the point-pattern-matching-based method are well in accordance with the traditionally applied pixelwise colocalization estimates, such as the Manders' colocalization coefficient and Pearson correlation, as well as with colocalization estimated with direct object-based overlap. Finally, we studied how association estimate performs in live cell experiments using live microscopy of $\alpha 2\beta 1$ integrin and EGF labeled cells from the A549 cell line. It was shown that dynamic association of two structures which do not perfectly overlap could be detected by the proposed method. Further, the effect of delay on the colocalization in live imaging situations was demonstrated, and the results under artificially created heavy delay further confirm the robustness of our method compared to estimate without movement compensation.

Acknowledgments

The authors would like to thank Academy of Finland projects #140052 (PR), #114727 (VM) for funding.

References

1. Weston DJ, Adams NM, Russell RA, Stephens DA, Freemont PS (2012) Analysis of spatial point patterns in nuclear biology. PLoS One 7: e36841.

2. Dodgson J, Chessel A, Yamamoto M, Vaggi F, Cox S, et al. (2013) Spatial segregation of polarity factors into distinct cortical clusters is required for cell polarity control. Nat Commun 4: 1834.

3. Bolte S, Cordelires FP (2006) A guided tour into subcellular colocalization analysis in light microscopy. J Microsc 224: 213–232.

4. Boulanger J, Gidon A, Kervran C, Salamero J (2010) A patch-based method for repetitive and transient event detection in fluorescence imaging. PLoS One 5: e13190.

5. Yu X, Miyamoto S, Mekada E (2000) Integrin alpha 2 beta 1-dependent egf receptor activation at cell-cell contact sites. J Cell Sci 113 (Pt 12): 2139–2147.

6. Ivaska J, Heino J (2010) Interplay between cell adhesion and growth factor receptors: from the plasma membrane to the endosomes. Cell Tissue Res 339: 111–120.

7. Comeau JWD, Costantino S, Wiseman PW (2006) A guide to accurate fluorescence microscopy colocalization measurements. Biophys J 91: 4611–4622.

8. Zhang B, Chenouard N, Olivo-Marin JC, Meas-Yedid V (2008) Statistical colocalization in biological imaging with false discovery control. In: Proc. 5th IEEE Int. Symp. Biomedical Imaging: From Nano to Macro ISBI 2008. pp. 1327–1330.

9. Wörz S, Sander P, Pfannmöller M, Rieker RJ, Joos S, et al. (2010) 3d geometry-based quantification of colocalizations in multichannel 3d microscopy images of human soft tissue tumors. IEEE Trans Med Imaging 29: 1474–1484.

10. Chessel A, Dodgson J, Carazo-Salas R (2012) Spherical spatial statistics for 3D fluorescence videomicroscopy. In: Biomedical Imaging (ISBI), 2012 9th IEEE International Symposium on. pp. 1747–1750. doi:10.1109/ISBI.2012.6235918.

11. Lagache T, Meas-Yedid V, Olivo-Marin J (2013) A statistical analysis of spatial colocalization using ripley's k function. In: Biomedical Imaging (ISBI), 2013 IEEE 10th International Symposium on. pp. 896–901. doi:10.1109/ISBI.2013.6556620.

12. Sbalzarini IF, Koumoutsakos P (2005) Feature point tracking and trajectory analysis for video imaging in cell biology. J Struct Biol 151: 182–195.

13. Jaqaman K, Loerke D, Mettlen M, Kuwata H, Grinstein S, et al. (2008) Robust single-particle tracking in live-cell time-lapse sequences. Nat Methods 5: 695–702.

14. Paavolainen L, Kankaanpää P, Ruusuvuori P, McNerney G, Karjalainen M, et al. (2012) Application independent greedy particle tracking method for 3D fluorescence microscopy image series. In: Biomedical Imaging (ISBI), 2012 9th IEEE International Symposium on. pp. 672–675. doi: 10.1109/ISBI.2012.6235637.

15. Besl PJ, McKay ND (1992) A method for registration of 3-d shapes. IEEE Trans Pattern Anal Mach Intell 14: 239–256.

16. Chen Y, Medioni G (1992) Object modelling by registration of multiple range images. Image Vision Comput 10: 145–155.

17. Rusinkiewicz S, Levoy M (2001) Efficient variants of the icp algorithm. In: Proc of the Third Intl. Conf. on 3D Digital Imaging and Modeling. pp. 145–152.

18. Zhang L, Choi SI, Park SY (2011) Robust icp registration using biunique correspondence. In: Intl Conf on 3D Imaging, Modeling, Processing, Visualization and Transmission, 3DIMPVT 2011, Hangzhou, China, 16-19 May 2011. pp. 80-85.

19. Chetverikov D, Stepanov D, Krsek P (2005) Robust euclidean alignment of 3d point sets: the trimmed iterative closest point algorithm. Image Vision Comput 23: 299-309.

20. Upla P, Marjomäki V, Kankaanpää P, Ivaska J, Hyypiä T, et al. (2004) Clustering induces a lateral redistribution of alpha 2 beta 1 integrin from membrane rafts to caveolae and subsequent protein kinase c-dependent internalization. Mol Biol Cell 15: 625–636.

21. Kankaanpää P, Paavolainen L, Tiitta S, Karjalainen M, Päivärinne J, et al. (2012) BioImageXD: an open, general-purpose and high-throughput image-processing platform. Nat Methods 9: 683-689.

22. Manders E, Verbeek F, Aten J (1993) Measurement of co-localization of object in dual-colour confocal images. J Microsc 169: 375–382.

23. Costes SV, Daelemans D, Cho EH, Dobbin Z, Pavlakis G, et al. (2004) Automatic and quantitative measurement of protein-protein colocalization in live cells. Biophys J 86: 3993–4003.

24. Villalta JI, Galli S, Iacaruso MF, Arciuch VGA, Poderoso JJ, et al. (2011) New algorithm to determine true colocalization in combination with image restoration and time-lapse confocal microscopy to map kinases in mitochondria. PLoS One 6: e19031.

25. Goodrich MT, Mitchell JSB, Orletsky MW (1999) Approximate geometric pattern matching under rigid motions. IEEE Trans Pattern Anal Mach Intell 21: 371–379.

26. Aiger D, Kedem K (2010) Approximate input sensitive algorithms for point pattern matching. Pattern Recogn 43: 153–159.

27. Cho M, Mount D (2005) Improved approximation bounds for planar point pattern matching. In: Dehne F, López-Ortiz A, Sack JR, editors, Algorithms and Data Structures, Springer Berlin Heidelberg, volume 3608 of Lecture Notes in Computer Science. pp. 432-443.

28. van Wamelen PB, Li Z, Iyengar SS (2004) A fast expected time algorithm for the 2-d point pattern matching problem. Pattern Recogn 37: 1699–1711.

29. Maneewongvatana S, Mount DM (1999) It's okay to be skinny, if your friends are fat. In: Center for Geometric Computing 4th Annual Workshop on Computational Geometry.

30. Hopcroft JE, Karp RM (1973) An n5/2 algorithm for maximum matchings in bipartite graphs. SIAM J Comput 2: 225-231.

31. Smal I, Loog M, Niessen W, Meijering E (2010) Quantitative comparison of spot detection methods in fluorescence microscopy. IEEE Trans Med Imaging 29: 282–301.

32. Ruusuvuori P, Äijö T, Chowdhury S, Garmendia-Torres C, Selinummi J, et al. (2010) Evaluation of methods for detection of fluorescence labeled subcellular objects in microscope images. BMC Bioinformatics 11: 248.

33. Fawcett T (2006) An introduction to roc analysis. Pattern recognition letters 27: 861–874.

34. Karjalainen M, Rintanen N, Lehkonen M, Kallio K, Mäki A, et al. (2011) Echovirus 1 infection depends on biogenesis of novel multivesicular bodies. Cell Microbiol 13: 1975–1995.

Figure Legends

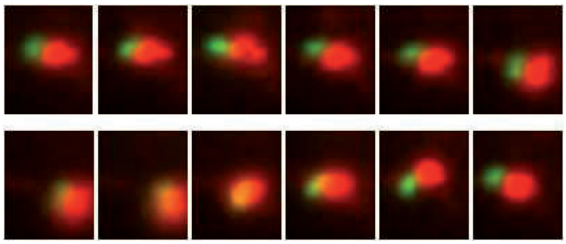


Figure 1. Close-up of a vesicle pair (red: integrin, green: EGF) in widefield microscopy images over a time-lapse. Time between successive frames is 5.7s. Movement of vesicles in living cells causes situations where determining true colocalization may be ambiguous – colocalization determined as a direct overlap potentially misses close association of rapidly moving vesicles, and on the contrary, association determined using any method using objects in the proximity may give false detections due to closely located vesicles.

Tables

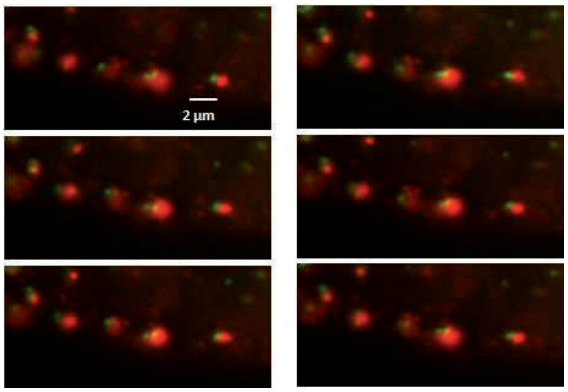


Figure 2. Close-up of a set of vesicles in live cell widefield microscopy experiment imaged over time (time between successive frames is 5.7s). Low contrast and blur makes object detection challenging.

Table 1. Colocalization and association estimates for fixed cell image sets. Average number of objects per channel for each set are given as $\mu\#_i$, association estimate by the PPM-based methods in $C_{PPM}i$, Manders' colocalization coefficient is M_i, pixelwise overlap is $O_{pix}i$, and r_p is the Pearson correlation.

Image set	#images	$\mu\#_1$	$\mu\#_2$	$C_{PPM}1$	$C_{PPM}2$	M_1	M_2	$O_{pix}1$	$O_{pix}2$	r_p
Fix I	71	154.41	126.46	0.6719	0.8123	0.7198	0.8111	0.6407	0.7300	0.7008
Fix II	70	154.40	67.37	0.4168	0.9493	0.4938	0.8946	0.3970	0.8469	0.7166
Fix III	69	201.32	82.06	0.3844	0.9345	0.4994	0.8705	0.3419	0.8266	0.7882
Fix IV	67	92.45	71.73	0.5645	0.7079	0.5441	0.7525	0.4784	0.6310	0.8116

Table 2. Colocalization and association estimates for live cell-imaging experiments. Average number of objects per channel for each set are given as $\mu\#_i$, association estimate by the PPM-based methods in $C_{PPM}i$, Manders' colocalization coefficient is M_i, pixelwise overlap is $O_{pix}i$, and r_p is the Pearson correlation.

Image set	#images	$\mu\#_1$	$\mu\#_2$	$C_{PPM}1$	$C_{PPM}2$	M_1	M_2	$O_{pix}1$	$O_{pix}2$	r_p
Live I	156	103.60	260.69	0.3813	0.1528	0.1618	0.0850	0.1595	0.0615	-0.0294
Live II	174	142.37	410.02	0.4682	0.1650	0.1574	0.0831	0.1690	0.0561	-0.0590

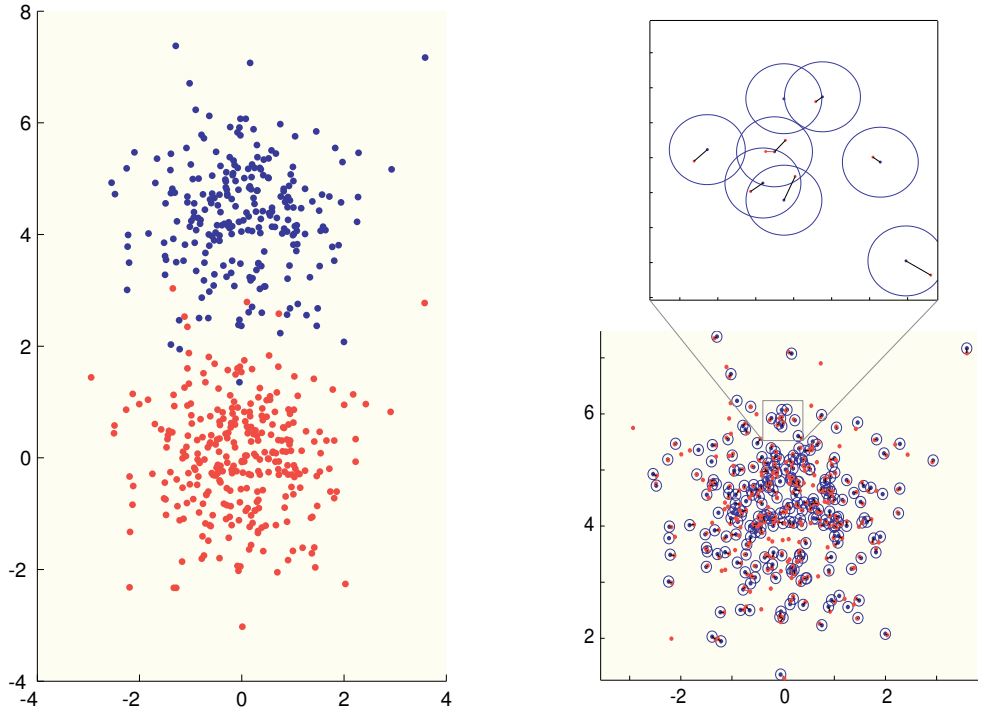


Figure 3. Example of matching point-sets. Left: Point-set (red) and an altered set (blue) under noise, transformation, and with missing points with probability of 0.2. Right: The same point-sets after matching. Each matched point has been marked with a line to the corresponding point in the other set. The search area has been shown with circles. Up: Close up where matches and search areas can be seen.

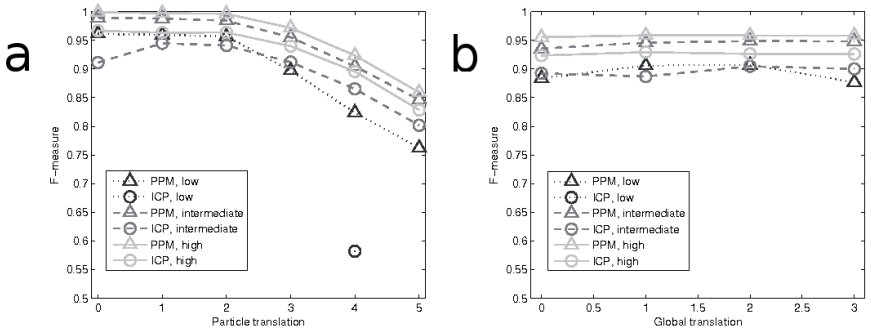


Figure 4. Result summary for simulated experiments. Results for PPM (triangle) and ICP (circle) are given as average F-measure for ten replicate simulations for each parameter combination. All three simulation scenarios are shown in the same figure; high association level with solid line and light grey, intermediate association level with mid grey dash line, and low association level with dotted dark grey line. (a) Results illustrated with standard deviation of the random particle movement (σ_{mov}) as a parameter. (b) Results illustrated with length of the global transformation (in pixels) as a parameter. F-measure is calculated through quantifying true and false matches, and results are not shown for parameter combinations leading to undefined F-measures.

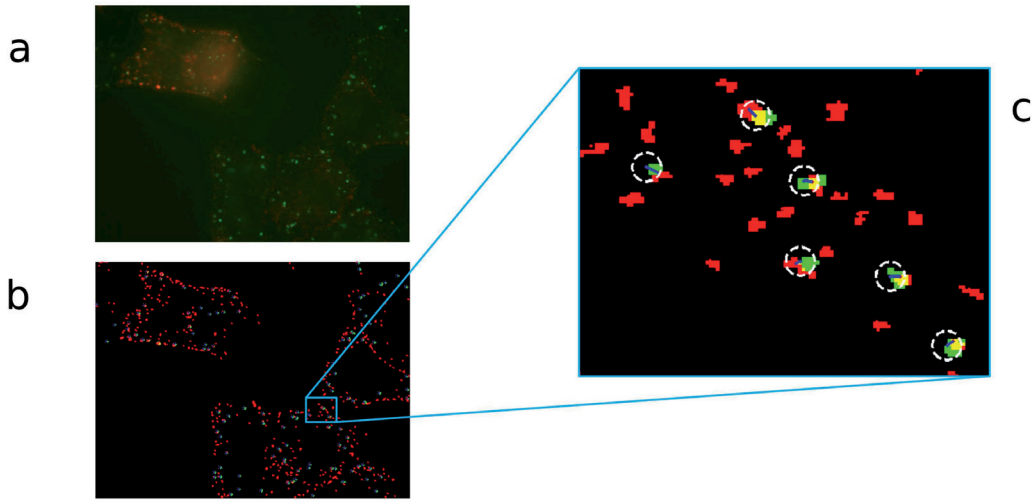


Figure 5. Frame from a live-cell imaging experiment with; a) overlay of original image channels, b) detected objects, colors correspond to image channels and direct pixelwise overlap (colocalization) is visible as yellow color, c) close-up showing transformed point set and search area with white dashed circles and found matches (association) with blue lines.

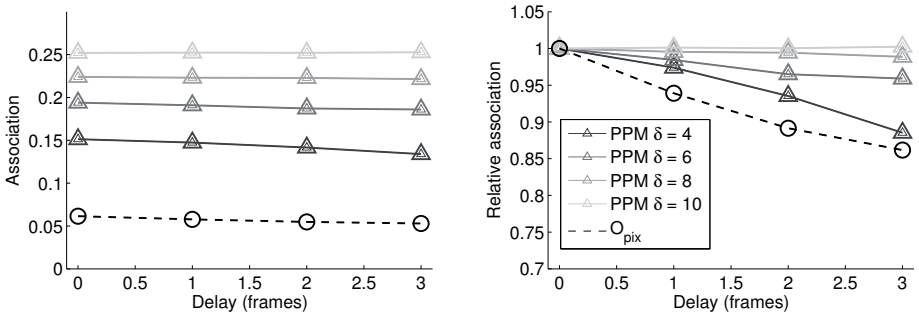


Figure 6. Effect of delay to the estimate. Delay is given in frames (0 . . . 3). The left graph shows the numerical estimates for association by PPM using different parameter values for search range (grey graphs with triangle markers) and direct pixelwise colocalization (black graph with circle markers). On the right the results are presented relative to the first datapoint, *i.e.* the values are divided by the estimates obtained for the first, non-delayed values.

PIV

SEQUENTIAL STATISTICAL RECONSTRUCTION FOR ELECTRON TOMOGRAPHY WITH MISSING WEDGE

by

Lassi Paavolainen*, Erman Acar*, Uygar Tuna, Sari Peltonen, Pan Soonsawad, Varpu Marjomäki, R. Holland Cheng & Ulla Ruotsalainen

Manuscript

Sequential Statistical Reconstruction for Electron Tomography with Missing Wedge

Lassi Paavolainen[1,⋆], Erman Acar[2,3,⋆], Uygar Tuna[2,], Sari Peltonen[2,3], Pan Soonsawad[4], Varpu Marjomäki[1], R. Holland Cheng[4], Ulla Ruotsalainen[2,3,∗]

1 Department of Biological and Environmental Science / Nanoscience Center, University of Jyväskylä, Jyväskylä, Finland

2 Department of Signal Processing, Tampere University of Technology, Tampere, Finland

3 BioMediTech, Tampere University of Technology, Tampere, Finland

4 Department of Molecular and Cellular Biology, University of California, Davis, Ca, USA

⋆ **Equal contribution**

∗ **E-mail: ulla.ruotsalainen@tut.fi**

Abstract

Electron tomography of biological samples is used to study the organization and the structure of subcellular complexes in great detail. However, electron tomography is an ill-posed problem, as projections cannot be imaged over full tilt angle range with conventional techniques. Missing information results in artifacts and in loss of resolution in the reconstruction when no prior knowledge of the sample is available. The aim of this study was to develop a modified statistical sequential reconstruction method (sMAP-EM) to electron tomography of biological samples without using prior knowledge. The effects of missing information due to limited angular range is corrected with this statistical reconstruction method. The reconstruction method was tested with simulated and experimental subcellular data, and compared to widely used weighted backprojection (WBP) and simultaneous iterative reconstruction technique (SIRT) methods. An ellipsoid fitting method was developed to statistically quantify gold particles in the reconstructions. The ellipsoids were used to analyze how well the measured particles can be detected. The results show that the presented method corrected the effects of the missing information seen as elongation very well. The applicability of automatic analysis methods was enhanced with the improved contrast ratio. The presented method enables analysis of subcellular structures with better three-dimensional resolution than the traditional methods.

Introduction

Electron tomography (ET) of cellular samples is widely used technique for three-dimensional (3D) reconstruction of complex subcellular structures with the resolution enabling detection of organization of macromolecular complexes [1]. In ET, transmission electron microscope (TEM) is used to image typically 200–500 nm thick samples [2]. ET eliminates the need for finer sectioning of the sample for 3D imaging. ET fills the gap between structural methods, such as single-particle reconstruction, and optical microscopy. Using modern sample preparation techniques, such as cryo methods, ET enables studies of molecular mechanisms in subcellular structures in their native context [1, 2].

Tomography is a method to reconstruct a 3D model of a sample from a collection of two-dimensional (2D) transillumination images of the sample taken in multiple orientations. In its simplest form, either the sample or the illumination source and detector are rotated around a single axis for full 180 or 360 degrees while projection images are taken in fixed, typically 1–2°, intervals [1].

ET of biological samples holds two major issues in acquisition of tomography images. First, in general case, sample cannot be imaged in full 180° range since the structure of the sample holder and limited space between the pole pieces of the objective prevent acquiring images with the highest tilt angles [3]. In addition, as the sample has slab shape, already in 60° tilt angle electron beam has to pass approximately twice as much material as in zero degree tilt angle [4] making the quality of high tilt angle images worse.

Typically, ±60–70° range is used in ET. Missing angular range, known as *missing wedge*, together with the increase in depth of the material the electron beam has to pass in high tilt angles, causes anisotropic resolution as elongation and blurring of the objects in axial direction. Second, the excess electron dose damages the sample by destroying the smallest details, and can induce deformation such as shrinkage in beam direction [4]. To avoid the damage to the sample, it is important to keep cumulative electron dosage as low as possible. However, low electron dosage decreases signal-to-noise ratio, and as such, also resolution. A compromise between the number of projections and the used dosage on each projection needs to be found to maximize the resolution of the imaged sample.

The most used reconstruction methods in ET are weighted backprojection (WBP) [5, 6] and simultaneous iterative reconstruction technique (SIRT) [7] due to their simplicity and wide availability in tomographic reconstruction software packages. However, backprojection methods are sensitive to the missing wedge, and can result in severe artifacts in limited angle tomography such as ET [8]. Algebraic reconstruction technique (ART) [9] is another widely used method solving the reconstruction problem using a set of linear equations. Both SIRT and ART are iterative methods that approximate 3D volume by minimizing the difference between original projections and projections of reconstructed volume. The methods were developed to be usable also in ET with large missing wedge using sparse set of projections. However, neither SIRT nor ART fills the missing wedge or is able to handle varying sample depth between low and high tilt angles [10]. Recently, SIRT method using WBP in initialization and backprojection steps was presented [11] to improve reconstruction compared to either of the methods separately.

Regularization and *a priori* knowledge are used to reduce artifacts common in traditional reconstruction methods in ET. Angle dependent non-linear anisotropic diffusion filtering [12] has been applied to projection images to compensate varying depth of the sample. To compensate the missing wedge, total variation and directional smoothing [13] have been applied to projection images. Shape-based regularization using prior models of particles included in the sample [14], and segmented mask as a prior in modified SIRT [15] have been presented recently. However, in general ET reconstruction problem, no *a priori* knowledge of the sample is available, unless using information of known objects, such as colloidal gold particles, as prior. Recently, compressed sensing [16, 17] has been applied to ET reconstruction to solve artifacts caused by the missing wedge. Compressed sensing reconstruction methods have a requirement of prior knowledge of sparse representation of the objects in the sample which was solved in [16, 17] using image gradient improving especially boundary regions in the images.

The aim of this study was to introduce a new statistical image reconstruction method capable to correct the effects of missing wedge even when prior knowledge of the objects is not available. The chosen modified maximum a posteriori expectation maximization (sMAP-EM) method [18] searches the most likely cross-sectional images given the measured projections from TEM. We regularize the iterative reconstruction with a median filtered image of the previous iteration in a weighted one-step late algorithm to control the noise in the reconstructed images [19]. The reconstruction estimates values for the missing wedge during the iterations and forms a three-dimensional image from the measurements. Instead of running the iterative reconstruction once to convergence with a chosen weight for the regularization we apply the method sequentially. In the sequences, the previous result is initializing the next iteration and by more accurate initialization of the iterative reconstruction we can gradually decrease the weight of the regularization process thus improving the image quality step by step during the sequences.

We tested the sMAP-EM method with two different noise level simulated datasets and an experimental dataset, and compared to the WBP and SIRT methods. Gold particles in the reconstructions were measured with automatic ellipsoid fitting method. The sMAP-EM reconstruction produced better resolution according to ellipsoid fitting in the axial direction showing better compensation of the missing wedge than the compared methods SIRT and WBP. Also the contrast ratio (CR) in the sMAP-EM reconstructions was better than in the WBP or SIRT reconstructions. We can conclude that the sequential statistical MAP-EM image reconstruction method succeeds to correct the effects of the missing wedge very well.

Materials and Methods

Datasets

1) Cell phantom: The cell phantom (Figure 1a) was created in order to test the developed method with a realistic numerical phantom of an intra-cellular structure. The constructed phantom is of the size 512x512x128 in the (x, y, z) directions, with isotropic voxel size of 1 nm. It contains 3 fairly large spherical objects mimicking virus particles. Each virus particle has a diameter of 80 nm. A large ellipsoidal shaped object representing a cell vesicle was also included. This vesicle was cut in half from the middle z-section emulating the microtomy process. The lengths of the axes of this vesicle are 102, 85 and 51 nm. Both virus particles and vesicle were created using textures from the reconstruction of experimental data. Additionally, 11 small and dense spherical objects presenting gold particles were added. These gold particles, having diameter 7, 9 and 11 nm, had high density to simulate TEM imaging. Non-uniform background density was set to smoothly vary 2.4% between the minimum and maximum density. Projections were taken in tilt range [-60°, +60°] with 1° increment. With the addition of noise to the projection data, two main components of noise in experimental TEM projections, Poisson noise due to electron count, and Gaussian noise due to the CCD sensor, were simulated. The obtained projections were contaminated with two different levels corresponding to 16.1% (noise level 1, NL 1, Fig. 1b) and 18.0% (noise level 2, NL 2) noise contamination according to the coefficient-of-variation test.

2) Experimental vesicle data: An experimental set of TEM projections of an intra-cellular vesicle was used as in [20], see Fig. 1c, formed by a series of TEM images obtained with the JEM-2100F Field Emission Electron Microscope (JEOL Ltd., Tokyo, Japan) at a voltage of 200 kV. It comprises 124 images of a cellular vesicle taken at an interval of approximately 1° in the range [-65°, +58°], with a magnification of 10k. Projection images were aligned by correlation of gold particles as in [20]. The dimensions of the projection images were 958x712 pixels and the pixel size was 1 nm.

Sequential MAP-EM Image Reconstruction Method

Statistical iterative methods are widely used in the field of tomographic image reconstruction [21–24]. A modified version of the well-known MAP-EM method [18] was used in this study for solving the missing wedge problem. The novel modication to the MAP-EM method is the sequential application of the spatial domain regularization. The block diagram of one sequence of the method is given in Figure 2.

The first sequence is initialized with an image of ones. First, the image is projected to the sinogram domain (1). Then the correction sinogram is calculated by dividing these projections to the measured projections (2). In the correction sinogram, the region which falls into the unknown projection bins is set to value 1. Then the correction sinogram is back-projected to the image domain (3) to obtain a correction image. This correction image is combined (4) with the regularization result (5) to update the current image estimate. The weight of the regularization result is controlled with the regularization parameter β. The value of β is varied throughout the sequences as shown in Figure 3.

First β value is set to 1 which corresponds to full regularization. Then it is reduced to 0.1 linearly in 10 steps and a final step is performed with β=0.01. By decreasing the β value, the blurring effect of the regularization filter is reduced. Therefore the resolution and the intensity contrast are enhanced. However, β is never set to 0 in order to utilize the low pass filter characteristic of the regularization filter and to enhance the visual quality of the final reconstructed image.

To analyze the change in the reconstructed image during sequences quantitatively, the mean square error (MSE) is calculated as:

$$MSE = \frac{1}{N}\sum_{j}(f'_j - f_j)^2,$$

where f_j is the reconstruction result and f'_j is the ground truth of the j^{th} pixel. The number of iterations

at each sequence was determined experimentally for the noisy data according to MSE. The change in the MSE throughout the sequences for one slice is seen in Figure 4.

The MSE value decreases during the iterations as it is seen from the figure at the first 10 sequences. At the final sequence the decrease of β value from 0.1 to 0.01 introduces some noise to the image which yields an increase in the MSE curve. However, decreasing weight of that low pass regularization filter, the visual quality of the image is enhanced at the final sequence in terms of the resolution and the intensity contrast as it is seen in the images and the image profiles in Figure 5.

The reconstruction method is implemented in MATLAB® (MathWorks Inc., MA, USA) and the code is run on the computer grid Techila (Techila Technologies Ltd., Tampere, Finland). It took about 960 hours for the cell phantom dataset and 4230 hours for the experimental vesicle dataset in CPU time to get the reconstruction results. Approximately 120 AMD 64 bit processor worker computers participated in the experiments. Therefore the reconstruction of the cell phantom dataset took about 8 hours and the experimental vesicle dataset about 35 hours.

Compared reconstruction methods

The sequential MAP-EM method is compared with two common reconstruction methods, namely the WBP [5] and the SIRT [7]. Implementations available in Tomo3D software [25] were used to test the methods. SIRT was run with 30 iterations in cell phantom data and 50 iterations in experimental vesicle data.

The WBP method simply distributes the projection data over the image plane proportionally to the contribution of each image element (pixel). This backprojection operation has a transfer function with a low pass characteristic. In order to compensate that low pass effect, the projection data is weighted with high pass filter coefficients in real space. The WBP method can be mathematically expressed as

$$f_j = \sum_i a_{ij} W(p_i),$$

where f_j is the reconstruction result of the j^{th} voxel, a_{ij} is the system matrix element defining the contribution of the j^{th} voxel to the i^{th} projection, p_i is the i^{th} projection and $W(\cdot)$ represents the high pass filtering operation [5].

SIRT is another well-known reconstruction method which enhances the image iteratively using forward and backward projections. The method is initialized with an arbitrary reconstruction estimate. First, the image is projected to the sinogram domain. Next, the difference between the measured projections and the projections from the current reconstruction is calculated. Finally, the difference is back-projected to the image domain to update the current image estimation. The method can be formulated as [7]:

$$f_j^{(k+1)} = f_j^{(k)} + \sum_i a_{ij} \frac{p_i - \sum_j a_{ij} f_j^{(k)}}{\sum_j a_{ij}^2},$$

where $f_j^{(k)}$ is the image reconstructed at the k^{th} iteration.

Evaluation methods

Both visual and quantitative evaluations were made from all reconstructions. Visual evaluations were made with BioImageXD software [26]. Elongations in all axes were analyzed from the reconstructions to measure the effect of the missing wedge to the resolution. To assess the 3D resolution, a new method to measure the resolution with ground-truth gold particles was developed. With isotropic resolution, elongation would be 1.0 in all directions, as gold particles are spherical in nanometer scale. However, as a result of the missing wedge, gold particles tend to be elongated especially in axial direction. Elongation

and resolution measurements were made by fitting an ellipsoid to each gold particle analyzed in the reconstructions.

Ellipsoid fitting was done by creating a set of ellipsoids $x^2/a^2 + y^2/b^2 + z^2/c^2 = 1$ with varying ellipsoidal parameters $a, b, c \in \mathbb{N}$. Voxels inside of an ellipsoid were set as mask. Normalized cross-correlation was used to search for the location of gold particle in the reconstruction from a small volume around the initial location. Initial 11 gold particle locations were known in the cell phantom volume. From the experimental vesicle data, locations of 7 isolated gold particles were initialized manually. Parameters of the best fitting ellipsoid for each gold particle were extracted from correlations by selecting the highest correlation coefficient.

Shape of the fitted ellipsoid was used to analyze gold particle elongation, which was used as a measure of resolution in the axial direction. Elongations were calculated directly from the ellipsoid parameters in all directions as $e_{yx} = b/a$, $e_{zx} = c/a$ and $e_{zy} = c/b$. The resolution of cell phantom reconstructions were analyzed by comparing the parameters of fitted ellipsoid to the radius (r) of ground-truth gold particle in the original volume. The resolution in all directions were calculated as $r_x = a/r$, $r_y = b/r$ and $r_z = c/r$.

CRs were evaluated from the reconstructions to quantify how well gold particles can be visualized and analyzed. The higher CR values mean higher contrast, and better visibility, in the reconstructed volumes. CR was calculated as

$$CR = \frac{\sum_{x \in I} f(x)/|I|}{\sum_{x \in O} f(x)/|O|},$$

where f is the reconstructed volume, $x \in \mathbb{N}^3$ is the voxel index, and the sets I and O are the voxels inside the volume of interest and in the volume outside the volume of interest, respectively. The volume of interest was defined by the fitted ellipsoid. The outside volume of interest was defined as the volume around the volume of interest with the same shape but twice the volume of the volume of interest. Double volume was selected to confirm that the outside volume of interest was covering the volume of interest in all directions even with the smallest gold particles.

Results

Numerical phantom results

Orthogonal slices of the reconstructions are presented in Figure 6. The quantitative measurements for the elongation, the resolution and the contrast ratio measurements of 11 gold particles in the cell phantom reconstructions of both noise levels are presented in Table 1. The quantitative measurements are based on the ellipsoid fitting (Figure 7). The sMAP-EM reconstruction results were consistently better with extremely good resolution. The method produced reconstructions with low elongation in the lateral direction. The 3D resolution in all directions are close to the ground-truth values. The WBP and the SIRT reconstructions show decrease in resolution and increase in elongation in the axial direction, as measured by the fitted ellipsoids. The resolution in y-direction is also extremely good in the WBP and the SIRT reconstructions. However, the WBP reconstruction also results in unsymmetric gold particles in the lateral direction.

The CR results in Table 1 show that the sMAP-EM reconstruction produces much better contrast ratio than the WBP and the SIRT reconstructions. This is visually presented in Figures 6 and 7. Artifacts are not visible in the sMAP-EM reconstruction in Figure 6. The WBP and the SIRT reconstructions show shadows near objects in the lateral direction. Figure 6 shows different amount of noise in the reconstructions of different noise levels. Higher noise seems to reduce visible artifacts in the WBP and the SIRT reconstructions.

Experimental data results

Orthogonal slices through an isolated gold particle marked with white arrow are presented for all of the reconstructions in Figure 8. The gold particle is much more distinguishable in the sMAP-EM reconstruction compared to the WBP and the SIRT reconstructions. The same can be seen in the ellipsoid fitting on a gold particle presented in Figure 9.

The quantitative measurements for the elongation and the contrast ratio are based on the ellipsoid fitting. The measurements of 7 manually selected isolated gold particles in the reconstructions of the experimental vesicle dataset are presented in Table 2. The sMAP-EM reconstruction has the lowest axial elongation and the best axial resolution. The contrast ratio in the sMAP-EM reconstruction is almost three times higher than in the WBP and the SIRT reconstructions. There is no elongation in the gold particles in lateral direction in the SIRT reconstruction. However, the axial elongation in the SIRT reconstruction is higher than in the sMAP-EM and the WBP reconstructions. The WBP reconstruction shows elongation in y-direction, as in cell phantom reconstructions.

Discussion

In this work, a sequential statistical reconstruction method, sMAP-EM, is applied in ET and compared to widely used and available WBP and SIRT methods. The methods are used to reconstruct a simulated cell phantom and experimental data of cell vesicle. The phantom is used to validate the results against known ground-truth, and also used to test how different noise levels affect the results. The experimental data is used to test the methods with a sample of complex biological structures. Quantitative and qualitative evaluations are made of the reconstructions. Quantitative measurements of the gold particles are made by ellipsoid fitting. Statistical evaluation of the elongation and the contrast ratio is done from 11 gold particles in the cell phantom data and from 7 gold particles in the experimental vesicle data. In addition, the resolution is analysed from the cell phantom data by comparing the fitted ellipsoids to the ground-truth gold particles.

Threshold-free ellipsoid fitting method was developed to analyze resolution of the reconstructions. Diameter of each gold particle is known in the cell phantom data that was used as comparison fit to the ellipsoid fitting result used to deduce the resolution measurement. The exact diameter of the gold particles in the experimental vesicle data are not known. However, the resolution of the reconstructions in axial direction are compared by the length of the fitted ellipsoids. The ellipsoid fitting was working in all sMAP-EM reconstructions. As the WBP and SIRT reconstructions have such a low contrast, the ellipsoid fitting showed more variation especially in noise level 2 cell phantom and experimental vesicle data.

The quantitative results show the sMAP-EM reconstructions to be much better than the WBP and the SIRT reconstructions. The same can be concluded with qualitative analysis of the reconstructions. The sMAP-EM reconstructions have much higher contrast than the WBP and the SIRT reconstructions. Improved contrast ratio enables better visual interpretation and automatic analysis currently still difficult in the ET. This is shown in the WBP and SIRT reconstructions of the noise level 2 cell phantom dataset. Both WBP and SIRT reconstructions include artifacts due to the missing wedge. These artifacts do not exist in the sMAP-EM reconstructions.

WBP is a fast and easy to implement image reconstruction method. However, it suffers in reconstructing the projection datasets with a missing wedge like in our ET case. SIRT and sMAP-EM, as iterative methods, can handle such datasets better. SIRT uses a linear regression model for the data and solves the problem in the least squares sense. However, sMAP-EM uses a Poisson distribution model for the data and solves the problem in the maximum likelihood sense. One reason that the contrast values for the sMAP-EM were better than SIRT is the fact that Poisson model represents the projection data better than the linear regression model. Another reason is the regularizing median filter which imposes

the solution a priori information that the intensity values are similar in the small neighborhood of each pixel of the reconstructed image. Both SIRT and sMAP-EM are computationally expensive. sMAP-EM is slower than SIRT due to the regularization process and the required number of iterations. Although the ET image reconstruction time is not as critical as reconstruction time of the other tomographic imaging modalities, it can be reduced by using an optimized and adaptive strategy to change the predefined sequence dependent regularization coefficient β. Moreover, the computing time for the method can be reduced with a better computing system (CPU, GPU, grid technologies, etc.) and optimized software in the near future.

The results of the sMAP-EM reconstruction method are very good with simulated and experimental data. Quantitative measurements show the sMAP-EM reconstruction to have better resolution than the SIRT and the WBP reconstruction. Visual inspection supports this and shows that sMAP-EM reconstruction reduces missing wedge artifacts. We believe that when accurate 3D analysis of the structures in ET is needed, the sequential MAP-EM would be the method to choose.

Acknowledgments

The project was partly supported by NIH (AI095382), FiDiPro (1913/31/2012), and Discovery Grant (UCDG178969). The authors would like to thank Toshio Moriya (Tampere University of Technology, BioMediTech, Finland) for giving constructive comments for improving the manuscript.

References

1. McEwen BF, Marko M (2001) The emergence of electron tomography as an important tool for investigating cellular ultrastructure. Journal of Histochemistry & Cytochemistry 49: 553–563.

2. Lučić V, Förster F, Baumeister W (2005) Structural studies by electron tomography: From cells to molecules. Annual Review of Biochemistry 74: 833–865.

3. Midgley P, Weyland M (2003) 3D electron microscopy in the physical sciences: the development of Z-contrast and EFTEM tomography. Ultramicroscopy 96: 413–431.

4. Koster AJ, Grimm R, Typke D, Hegerl R, Stoschek A, et al. (1997) Perspectives of molecular and cellular electron tomography. Journal of Structural Biology 120: 276–308.

5. Radermacher M (2006) Weighted back-projection methods. In: Frank J, editor, Electron Tomography: Methods for Three-Dimensional Visualization of Structures in the Cell, New York: Springer. 2nd edition, pp. 245–273.

6. Orlov IM, Morgan DG, Cheng RH (2006) Efficient implementation of a filtered back-projection algorithm using a voxel-by-voxel approach. Journal of Structural Biology 154: 287–296.

7. Gilbert P (1972) Iterative methods for the three-dimensional reconstruction of an object from projections. Journal of Theoretical Biology 36: 105–117.

8. Delaney A, Bresler Y (1998) Globally convergent edge-preserving regularized reconstruction: an application to limited-angle tomography. IEEE Transactions on Image Processing 7: 204-221.

9. Gordon R, Bender R, Herman GT (1970) Algebraic reconstruction techniques (ART) for three-dimensional electron microscopy and X-ray photography. Journal of Theoretical Biology 29: 471–481.

10. Penczek PA (2010) Fundamentals of three-dimensional reconstruction from projections. Methods in Enzymology 482: 1–33.

11. Wolf D, Lubk A, Lichte H (2014) Weighted simultaneous iterative reconstruction technique for single-axis tomography. Ultramicroscopy 136: 15–25.

12. Maiorca M, Hanssen E, Kazmierczak E, Maco B, Kudryashev M, et al. (2012) Improving the quality of electron tomography image volumes using pre-reconstruction filtering. Journal of Structural Biology 180: 132–142.

13. Aganj I, Bartesaghi A, Borgnia M, Liao HY, Sapiro G, et al. (2007) Regularization for inverting the radon transform with wedge consideration. In: 4th IEEE International Symposium on Biomedical Imaging: From Nano to Macro. pp. 217–220.

14. Gopinath A, Xu G, Ress D, Oktem O, Subramaniam S, et al. (2012) Shape-based regularization of electron tomographic reconstruction. IEEE Transactions on Medical Imaging 31: 2241–2252.

15. Zürner A, Döblinger M, Cauda V, Wei R, Bein T (2012) Discrete tomography of demanding samples based on a modified SIRT algorithm. Ultramicroscopy 115: 41–49.

16. Goris B, Van den Broek W, Batenburg K, Heidari Mezerji H, Bals S (2012) Electron tomography based on a total variation minimization reconstruction technique. Ultramicroscopy 113: 120–130.

17. Leary R, Saghi Z, Midgley PA, Holland DJ (2013) Compressed sensing electron tomography. Ultramicroscopy 131: 70–91.

18. Tuna U, Sohlberg A, Ruotsalainen U (2013) Can we reduce SPECT acquisition time using MAP-EM reconstruction. Journal of Pattern Recognition and Intelligent Systems 1: 54–63.

19. Alenius S, Ruotsalainen U (2002) Generalization of median root prior reconstruction. IEEE Transactions on Medical Imaging 21: 1413–1420.

20. Soonsawad P, Xing L, Milla E, Espinoza JM, Kawano M, et al. (2010) Structural evidence of glycoprotein assembly in cellular membrane compartments prior to Alphavirus budding. Journal of Virology 84: 11145–11151.

21. Bouman C, Sauer K (1993) A generalized Gaussian image model for edge-preserving MAP estimation. IEEE Transactions on Image Processing 2: 296–310.

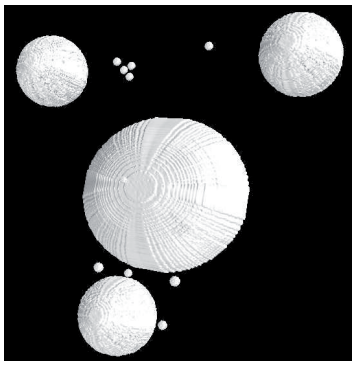
22. Fessler JA, Hero AO (1995) Penalized maximum-likelihood image reconstruction using space-alternating generalized EM algorithms. IEEE Transactions on Image Processing 4: 1417–1429.

23. Green PJ (1990) Bayesian reconstructions from emission tomography data using a modified EM algorithm. IEEE Transactions on Medical Imaging 9: 84–93.

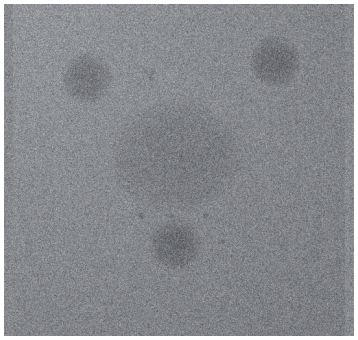
24. Hebert T, Leahy R (1989) A generalized EM algorithm for 3-D Bayesian reconstruction from Poisson data using Gibbs priors. IEEE Transactions on Medical Imaging 8: 194–202.

25. Agulleiro JI, Fernandez JJ (2011) Fast tomographic reconstruction on multicore computers. Bioinformatics 27: 582–583.

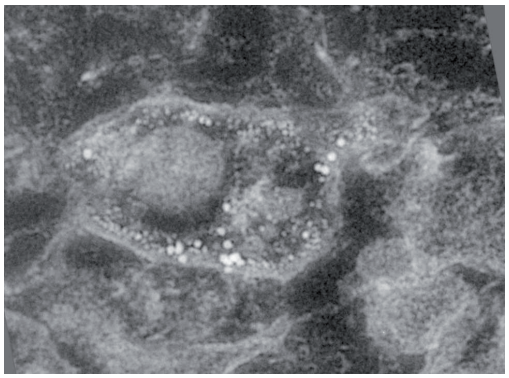
26. Kankaanpää P, Paavolainen L, Tiitta S, Karjalainen M, Päivärinne J, et al. (2012) BioImageXD: an open, general-purpose and high-throughput image-processing platform. Nature Methods 9: 683–689.



(a) Cell phantom ground-truth



(b) Cell phantom zero tilt projection image



(c) Experimental vesicle zero tilt projection image

Figure 1. Used datasets. a) Surface rendering of the cell phantom dataset. b) Zero tilt projection of the cell phantom with noise level 1. c) Zero tilt projection of the experimental vesicle data with inverted lookup table for improved visualization.

Figure Legends

Tables

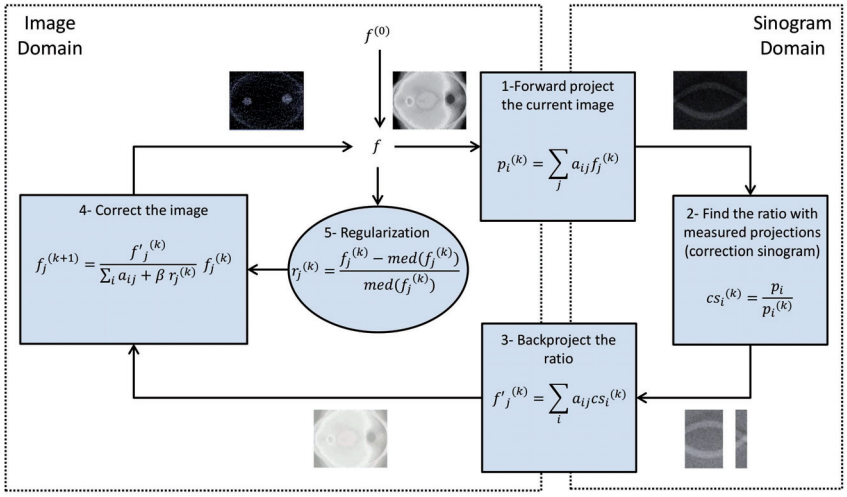


Figure 2. The block diagram of the Sequential MAP-EM Reconstruction Method. $f_j^{(k)}$ is the reconstruction result of the j^{th} voxel at k^{th} iteration, a_{ij} is the system matrix element defining the contribution of the j^{th} voxel to the i^{th} projection, p_i is the i^{th} projection, $med(\cdot)$ is the median filter with a 3x3 kernel size, β is the predefined sequence dependent coefficient defining the amount of regularization.

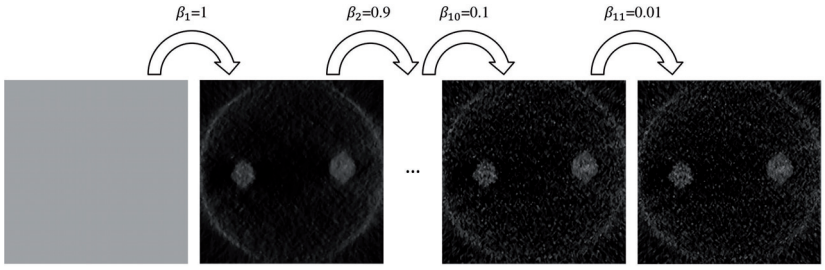


Figure 3. The sequential change of regularization weight, β. The sequences are initialized with an image of ones (the leftmost image). Then, for each β value, 91 iterations are performed. At the end of sequences with decreasing β values, the final reconstruction image is obtained (the rightmost image).

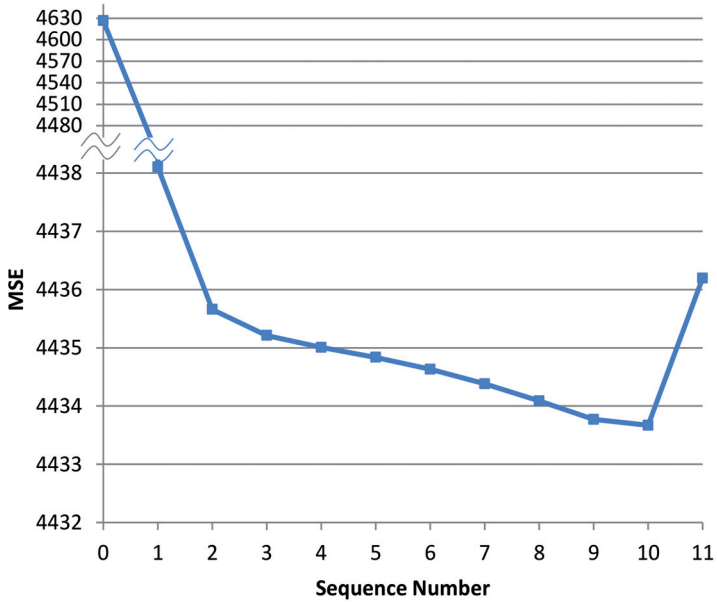


Figure 4. The change in the mean square error (MSE) throughout the sequences.

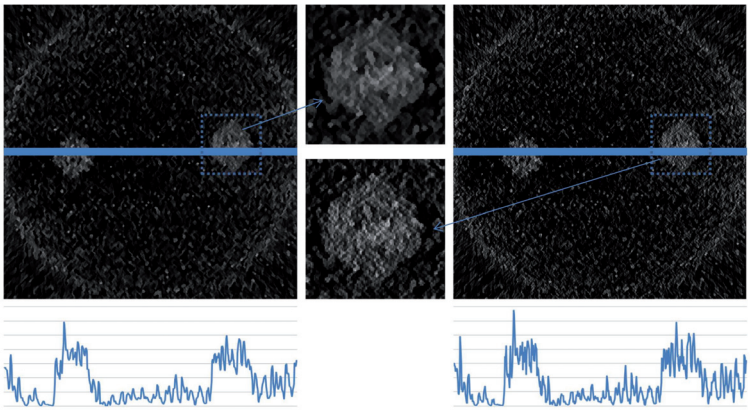
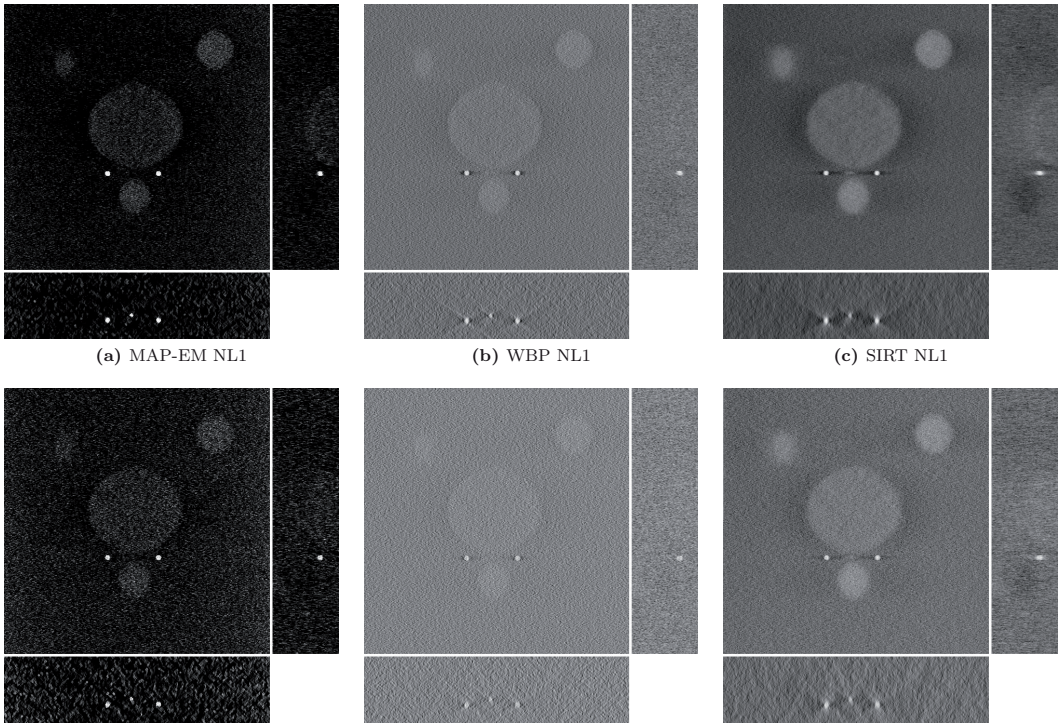


Figure 5. The reconstructed image at the end of the 10th (upper left) and the 11th (upper right) sequences and the profiles taken from the center of the images. The visual quality of the image is enhanced at the final sequence in terms of the resolution and the intensity contrast at the expense of an increase in the MSE.



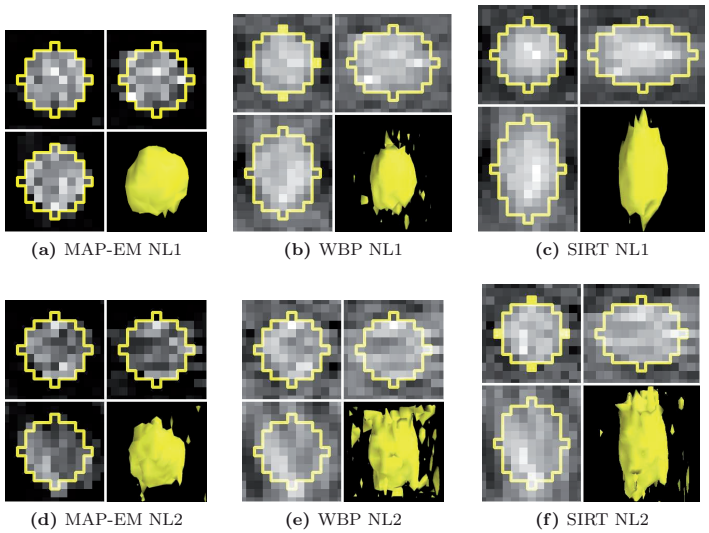
(a) MAP-EM NL1 **(b)** WBP NL1 **(c)** SIRT NL1

(d) MAP-EM NL2 **(e)** WBP NL2 **(f)** SIRT NL2

Figure 6. Orthogonal slices from the cell phantom reconstruction results. Orthogonal x-y, x-z and z-y slices through the center of the right gold particle. The intensities of images are enhanced linearly to improve visual comparability.

Table 1. Quantitative results for 11 gold particles in cell phantom dataset

Noise	Method	r_x	r_y	r_z	e_{yx}	e_{zx}	e_{zy}	CR
NL 1	MAP-EM	1.00 ± 0.00	1.00 ± 0.00	1.03 ± 0.10	1.00 ± 0.00	1.03 ± 0.10	1.03 ± 0.10	7.57 ± 1.12
	WBP	0.92 ± 0.11	1.00 ± 0.00	1.23 ± 0.17	1.11 ± 0.15	1.36 ± 0.25	1.23 ± 0.17	1.68 ± 0.06
	SIRT	0.98 ± 0.08	1.00 ± 0.00	1.46 ± 0.17	1.03 ± 0.10	1.50 ± 0.17	1.46 ± 0.17	1.69 ± 0.12
NL 2	MAP-EM	1.00 ± 0.00	1.00 ± 0.00	1.06 ± 0.13	1.00 ± 0.00	1.06 ± 0.13	1.06 ± 0.13	4.73 ± 0.75
	WBP	0.89 ± 0.13	1.00 ± 0.00	1.09 ± 0.13	1.15 ± 0.19	1.27 ± 0.35	1.09 ± 0.13	1.38 ± 0.04
	SIRT	0.98 ± 0.08	1.00 ± 0.00	1.29 ± 0.21	1.03 ± 0.10	1.32 ± 0.18	1.29 ± 0.21	1.44 ± 0.08



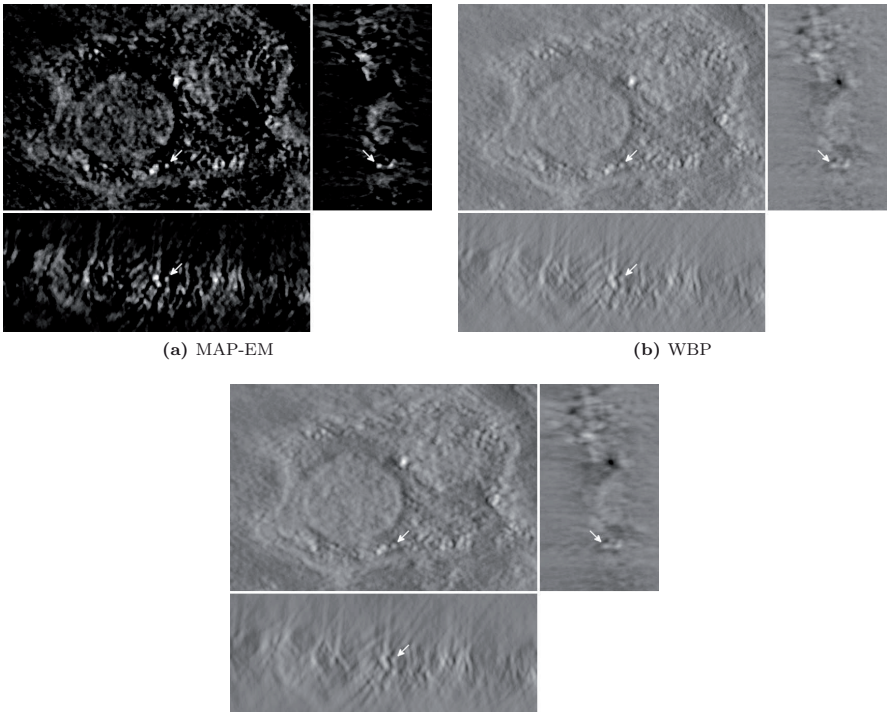
(a) MAP-EM NL1 **(b)** WBP NL1 **(c)** SIRT NL1

(d) MAP-EM NL2 **(e)** WBP NL2 **(f)** SIRT NL2

Figure 7. Fitted ellipsoids on an average gold particle in the cell phantom data. Orthogonal x-y (upper-left), z-y (upper-right), and x-z (lower-left) slices of the fitted ellipsoid drawn over an average gold particle. Surface rendering of the gold particle (lower-right). All images in the same row are in scale. Iso-value was selected as the mean of the average intensity value inside the volume of interest and the outside volume of interest. The surface rendering is suggestive of the overall shape of the gold particle and should not be directly compared between methods. Images are scaled to the maximum dynamic range.

Table 2. Quantitative results for 7 gold particles in experimental vesicle data

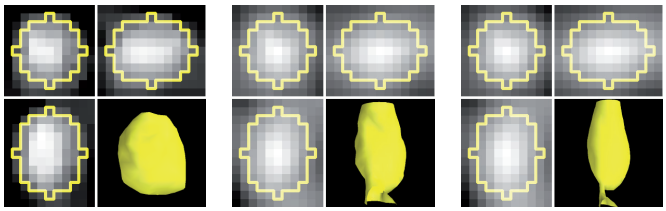
Method	e_{yx}	e_{zx}	e_{zy}	CR
MAP-EM	1.01 ± 0.17	1.27 ± 0.18	1.27 ± 0.18	3.01 ± 0.95
WBP	1.14 ± 0.13	1.57 ± 0.24	1.38 ± 0.18	1.18 ± 0.04
SIRT	1.00 ± 0.00	1.67 ± 0.37	1.67 ± 0.37	1.18 ± 0.03



(a) MAP-EM **(b)** WBP

(c) SIRT

Figure 8. Orthogonal slices from the experimental vesicle dataset reconstruction results. Orthogonal x-y, x-z and z-y slices through the center of the gold particle marked with white arrows. The intensities of images are enhanced linearly to improve visual comparability.



(a) MAP-EM **(b)** WBP **(c)** SIRT

Figure 9. Fitted ellipsoids on a gold particle in the experimental vesicle data. Orthogonal x-y (upper-left), z-y (upper-right), and x-z (lower-left) slices of the fitted ellipsoid drawn over the gold particle marked in Figure 8. The selected gold particle is good average representant of the fitted ellipsoids in the MAP-EM reconstruction. The gold particle is the one with the best resolution in the WBP and SIRT reconstructions. Surface rendering of the gold particle (lower-right). Iso-value selected as the mean of the average intensity value inside the volume of interest and the outside volume of interest. The surface rendering is suggestive of the overall shape of the gold particle and should not be directly compared between methods. Images are scaled to the maximum dynamic range.