10

Image Processing and Image Analysis in Microscopy

Daniel SAGE[1] and Anaïs BADOUAL[2]

[1] Biomedical Imaging Group and Center for Imaging,
École Polytechnique Fédérale de Lausanne (EPFL), Switzerland
[2] SAIRPICO, Centre INRIA de l'Université de Rennes, France

10.1. Introduction

In recent years, microscopy images have played a crucial part in the life sciences. They are the privileged witnesses for the observation at the molecular level of the great many and very complex intra-cellular interactions. As a corollary, the quantitative analysis of these images has become essential to improve the understanding of this cellular machinery.

Throughout this book, the reader has discovered a very wide range of microscopy techniques, thus demonstrating the ingenuity of scientists striving to challenge the knowledge of the living world ever further. While the acquisition of microscopic information varies greatly from one microscopy modality to another, they all now share a common denominator: the digital world. Indeed, since the 2000s, behind every microscope there are now automated systems, digital sensors, digital cameras and of course one or more computers. The latter are present throughout the entire process from image acquisition to the interpretation of their content. The nature of experiments has therefore evolved from purely qualitative observations to quantitative analyses on computers.

For a color version of all figures in this chapter, see www.iste.co.uk/sibarita/photonic.zip.

Photonic Imaging for Biology,
coordinated by Jean-Baptiste SIBARITA. © ISTE Ltd 2025.

The automatic quantification of images and the extraction of relevant information have become a major challenge for the biosciences. Discoveries in biology, progress in the knowledge of the cell, are often significantly linked to the analysis of microscopy images. This is why the demand for dedicated computer tools is increasingly important. However, image analysis solutions that are versatile enough and capable of absorbing the huge flow of data (Ouyang and Zimmer 2017) still remain very uncommon, especially with the new modalities such as super-resolution microscopy or large field of view imaging.

The challenges that must be addressed are at the same time technological, logistical, as well as methodological with the consequent developments that have been gathered and centered around a new discipline called *computational bioimaging* or *bioimage informatics* (Myers 2012; Meijering et al. 2016). This discipline is boosted by the synergies and transversal exchanges between the fields of biology, microscopy, signal processing and computer vision, combining approaches known as *model-based* and, more recently, approaches referred to as *data-driven*[1] (Meijering 2020).

In this chapter, we address the main aspects of computational bioimaging applied to the microscopy of living organisms. After setting the current context of microscopy imaging (section 10.2), we will present the most notable operations of image processing (section 10.3) and image analysis (section 10.4) of microscopy under the dual umbrella of model-based and data-driven approaches. The goal is to give the reader the general principles underlying algorithms and the terms used in the field allowing for more in-depth research. We will conclude with a perspective on the challenges of microscopy image processing and analysis (section 10.5).

10.2. Context

10.2.1. *Multidimensional images*

Microscopy images have nothing in common with images or photos that we usually come across in our daily lives, both in terms of size and content. They are rather regular groupings of measurements that reflect physical or biological phenomena, and which are intended to be quantified by a computer. In bioimaging, an image is an object that can have up to five dimensions (5D). The data originating from the lateral plane form a 2D image with two XY axes, to which is added the spatial dimension Z, obtained by scanning in the optical axis. The whole constitutes a volume

[1] In this chapter, we use the term *model-based* for approaches designed by engineers based on physical, mathematical or biological modeling. These *model-based* methods are commonly referred to as conventional algorithmic methods as opposed to the newer machine learning techniques that we call *data-driven* methods. These machine learning techniques, including deep learning, are exclusively built around data without an explicit model.

of 3D data that enables the three-dimensional nature of the specimens to be captured. It should be noted that the axial resolution (Z) is often worse than the lateral resolution (XY) due to the technical limitations of microscopes. The quantification of dynamic processes requires the acquisition of time-lapse volumes at regular time intervals, thus forming a 4D space–time object. Finally, since the 2000s, fluorescence has become widely used in biology because it enables specific cellular structures to be isolated in different color channels of the image using markers of different wavelengths. This is the fifth dimension. This dimension is sometimes also derived from spectral imaging techniques (Zimmermann et al. 2003). For a specific experiment, it is therefore not uncommon to be dealing with 5D images (see Figure 10.1).

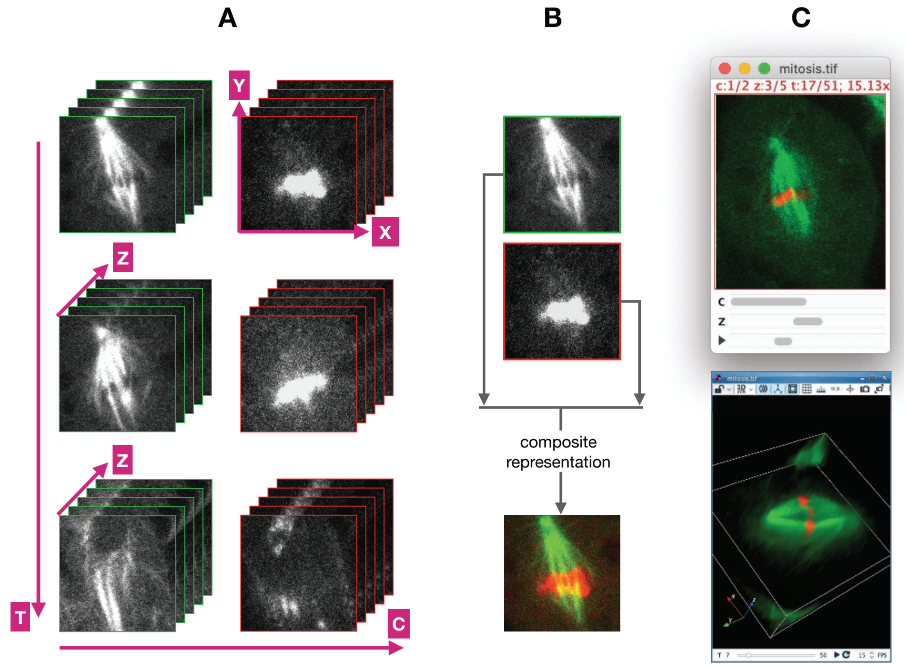

Figure 10.1. *Views of a 5D image. A. Illustration of a 5D image with five focal planes (Z), three time frames (T) and two fluorescent channels (C). B. Two 2D fluorescent channels (XY) of the 5D image are combined into a "composite" image with red and green color mapping. C. Top: ImageJ viewer of a 5D image with three sliders (C, Z, T) (Schindelin et al. 2015). Bottom: Icy viewer of a 5D image with volumetric rendering and a slider (T) (de Chaumont et al. 2012). Source: "mitosis.tif" 5D image provided as an example with the ImageJ software*

An image f is mathematically modeled by a function of several variables x, y, z, t, c which, in the numerical world, take their values in a finite space (see equation [10.1]). In the rest of this chapter, we use vector notation where $\mathbf{x} =$

$\{x, y, z, t, c\}$. The intensity $f[\mathbf{x}]$ at a given position $\mathbf{x}$ is discretized over a number of gray levels (8 bits in general, 16 bits in microscopy (see Figure 10.3(A), quantization)):

$$\begin{aligned} f : \Omega^5 &\to \mathcal{D} \\ \mathbf{x} &\to f[\mathbf{x}] \end{aligned} \qquad [10.1]$$

This mathematical formalism is used for the design of more efficient and generalizable algorithms, such as for the computation of the gradient vector (multidimensional derivative) which is the basis of edge detection algorithms.

10.2.2. *Major challenges in bioimaging*

Scientific challenges

Quantitative image analysis in biology has a wide range of applications, from simple counting of cell nuclei, to measuring protein–protein interactions, not forgetting the construction of the cell line of a *C elegans* embryo. The scientific challenge therefore consists of extracting information in an unbiased way that can be reproduced, such as, for example, for detecting a very dense set of particles (such as cells, organelles or proteins) and also for characterizing their morphology, dynamics, divisions, response to gene expression, or still the localization and tracking of single molecules, all from images with almost indistinguishable or partially degraded signals. An additional issue inherent to the study of the living is the great variability of the objects of study and of the acquisition methods (transmitted light, staining of histological sections, etc.).

By accounting for all dimensions of the image (5D), image analysis presents new methodological challenges. For example, for 3D segmentation, it is more reliable to extract a 3D-modeled object directly from the entire volume, rather than using traditional 2D segmentation methods that proceed in a plane-wise manner before the final assembly to obtain a 3D object. Naturally, the analysis with all dimensions is more demanding in computational terms.

Technical challenges

Modern microscopes have the ability to generate multidimensional images with high spatiotemporal resolution. The resulting multidimensional images are very large, sometimes of several terabytes (TB). As an example, light-sheet microscopes can produce several TB/h of data. In addition to the scientific challenges mentioned above, this massive data, especially 4D and 5D, complicates all operations, even the simplest ones such as archiving or transfer, especially for image processing and analysis operations.

A significant aspect in bioimaging is the visualization of images and data obtained by the analysis. Nonetheless, the visualization of 3D, 4D and 5D data is not easy on the usual 2D media, such as screens or paper. Even 3D volumetric rendering techniques only partially reflect the complex three-dimensional reality of the biological objects observed.

Having reliable annotated data is crucial to quantitatively evaluate the image analysis methods developed by the community and to train those based on deep learning (i.e. data-driven). The major issue is that manually annotating 3D, 4D and 5D images is an extremely tedious or even impossible task, mainly due to the obstacles to their visualization. Consequently, it cannot be reliably performed and exhibits significant intra- and inter-experimentalist variabilities. For a vast majority of applications on living cells, there is a recurring lack of verified annotations that could be used as a basis for learning. To overcome this challenge, ingenious image processing solutions are needed, such as the use of techniques to reduce the dimensionality of data (principal component analyses, kymographs or auto-encoders), the use of new visualization methods such as virtual reality, or even the development of large annotated synthetic databases.

The analysis of cellular processes therefore requires advanced methods for the manipulation of terabytes of data and solving these scientific challenges. Furthermore, as technology evolves in microscopy (microscopes, fluorescence probes (Berlin et al. 2015)), images are being enriched with new information that requires continuous developments in image processing and analysis.

10.2.3. *Software for bioimaging*

A lot of software is open source in bioimaging (Lucas et al. 2021). This has been produced by and for the bioimaging research community that has gathered around a very active forum image.sc[2]. This collaborative work makes it possible to follow technological advances in microscopy as closely as possible and to meet the precise needs of biologists with the development of ready-to-use tools.

In this context, ImageJ is a software program with an exclusive story[3]. Its creator, Wayne Rasband of the National Health Institute, developed it over a period of 20 years. ImageJ was quickly adopted to become almost universal in bioimaging. Due to its open architecture and code written in a very popular language (Java), hundreds of developers from all over the world have been able to add scripts, macros and plugins to satisfy most of the needs of microscopy image analysis. ImageJ is the source of a

[2] image.sc: https://forum.image.sc.

[3] ImageJ: https://imagej.nih.gov/ij/.

software ecosystem (Schindelin et al. 2015) centered around the new ImageJ2[4]. There are several distributions of ImageJ, including the very popular Fiji[5] which is today installed on most biologist computers.

Other open-source projects for a generalist platform for bioimaging have been initiated:

– Icy[6] from the Institut Pasteur, coded in Java, is capable of visualizations and native operations in 5D.

– QuPath[7] by Pete Blankhead is very popular for examining digital pathology slides.

– CellProfiler[8] from the Broad Institute provides image analysis pipelines for biologists.

– Ilastik[9], mainly developed at the University of Heidelberg, includes a comprehensive set of machine learning algorithms.

– Napari[10], initiated by the Chan-Zuckerberg Biohub, is a multidimensional image analysis software written in Python.

While Java has long been the language of choice for bioimaging, Python is now often preferred mainly because of the attractiveness of Jupyter Notebooks, as well as because of the native access to powerful deep learning frameworks such as TensorFlow[11] and PyTorch[12]. The new bioimaging platform, Napari, is a driving force of this evolution towards Python, by combining an efficient interactive multi-dimensional viewer and a large number of plugins for image analysis.

Although there is a significant offer of software programs, it only partially covers the specific needs of image analysis and it is often necessary to use specific software developments for a given type of image and for a given application. Within this context, it is essential that scientists adopt the proper practices of open science, by sharing data (Ellenberg et al. 2018), analysis protocols, software source codes

[4] ImageJ2: https://imagej.net/software/imagej2/.

[5] Fiji: https://imagej.net/software/fiji/.

[6] Icy: http://icy.bioimageanalysis.org/.

[7] QuPath: https://qupath.github.io/.

[8] CellProfiler: https://cellprofiler.org/.

[9] Ilastik: https://www.ilastik.org/.

[10] Napari: https://napari.org/.

[11] TensorFlow: https://www.tensorflow.org/.

[12] PyTorch: https://pytorch.org/.

(Carpenter et al. 2012), deep learning models[13] and validation methods in the spirit of collaborative research (Cohen 2014).

10.2.4. *Image analysis pipeline*

In computer vision, an image analysis pipeline generally includes two phases (see Figure 10.2): image processing operations (section 10.3) and image analysis operations (section 10.4). These two phases are found in microscopy, to which are sometimes added an image reconstruction step, which consists of numerically combining a set of measurements or images.

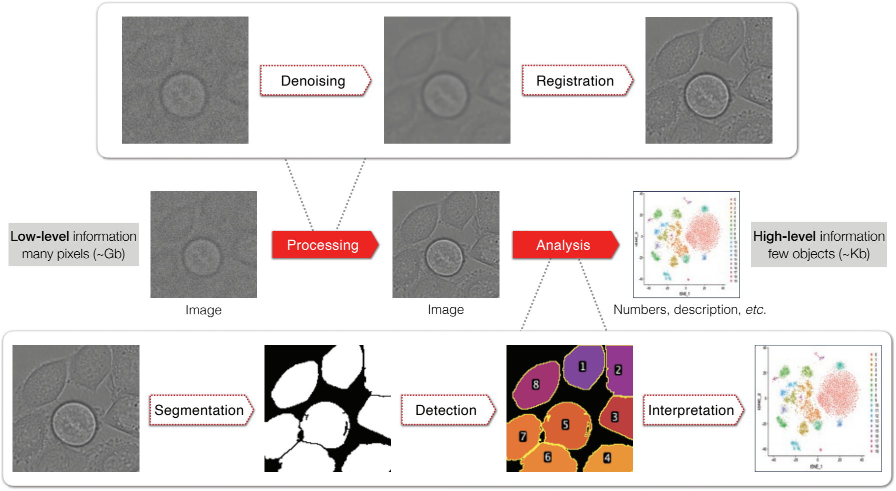

Figure 10.2. *An example of an image analysis pipeline. A pipeline is often made up of different steps that allow shifting from very low-level information (with a very large amount of pixel values) to very high-level information (with a very limited number of results). Only the key steps that are generally common to all image analysis pipelines are depicted in this figure. Depending on the application, other steps may be required such as image reconstruction, image registration, object classification or object tracking. Source: image from the Cell Tracking Challenge, http://celltrackingchallenge.net*

In order to maximize the performance of the image analysis pipeline for a given application, image acquisition must be integrated from the design stage of the project by seeking the best compromise of the acquisition parameters. This avoids, for example, having too high a spatial resolution and insufficient time resolution if the goal is to follow fast movements.

[13] Biomage Model Zoo: https://bioimage.io.

In general, an image analysis pipeline is broken down into basic steps where each step can be performed with tools ready for use. On the other hand, it is therefore imperative to validate the pipeline in its entirety to limit the inexorable propagation of errors in the pipeline and reduce the dependency of the parameters of each step.

Image processing and image analysis

Image processing operations are digital transformations of images that take an image on input and output an image, generally of the same dimension and size, and whose purpose is to facilitate the extraction of information. These are thus often pre-processing operations whose objective is to improve the structures of interest in the images, for example, by removing blur or reducing noise, while improving the structures of interest. Strictly speaking, image analysis operations are those that have an input image and non-image output results, such as a number of detected objects, statistical information about the size of the objects or their dynamics, etc.

Image reconstruction

For a number of recent microscopy modalities, access to the image is not direct. The optical detectors of the microscope record a set of measurements that must be collectively analyzed to reconstruct a computed image. This is the case, for example, of the SIM (Structured Illumination Microscopy) and SMLM (Single-Molecule Localization Microscopy) super-resolution modalities (Valli et al. 2021), which have a computer reconstruction phase that leads to overpassing the physical limits of light diffraction. In SIM (Gustafsson 2000), nine spatially modulated images are generally acquired and then merged in the Fourier domain to obtain a single image with twice the resolution (see Chapter 7). In SMLM, thousands of images of molecules made fluorescent in a stochastic manner are recorded. Algorithms detect and locate the positions of the sources very accurately to reconstruct a map of the centers that forms an image about 10 times better resolved (see Chapter 9). Image reconstruction algorithms play a key role in the quality of the reconstructed image (Sage et al. 2019).

The role of algorithms is concomitant with any reconstruction process. Once the physical model is properly characterized, the reconstruction is mathematically formalized as the solution of a matrix equation called *inverse problem*, in which the unknowns are the values of the pixels to be reconstructed. This inverse problem is often ill-posed because the information is partial, the model is approximate and the level of noise is often high.

Traditional model-based approaches versus recent data-driven approaches

The different steps of image reconstruction, processing and analysis can be performed separately or all together using *model-based* or *data-driven* approaches.

Model-based approaches (deconvolution, active contours, Markov Random Fields), recognized as traditional approaches, are based on physical or biological models. Their main advantage is to easily integrate prior knowledge of the process under study (shape, trajectory, possible distortion, kinetics), and due to their mathematical formalism, they allow for error analysis and an interpretation of the results obtained. The challenge with these approaches is to succeed in correctly modeling the underlying processes with appropriate levels of detail and realism. This requires some *ingenuity* from the designer.

The more recent data-driven approaches (such as for instance machine learning and deep learning) for their part are based on *learning* statistical models by accumulating information from a very large number of example data. These methods adapt better to the complexity of data and therefore generally perform better than model-based approaches. Once trained, they are also more computationally efficient. On the other hand, they pose a challenge with regard to the aspects of generalizability, explainability and interpretability (Meijering 2020) and the very high computational cost of their learning phase.

The choice of one or the other of these two approaches depends on the complexity of the task to be accomplished, the degree of variability of data, the prior knowledge of the underlying physical models, data availability and the expected interpretability of the results. One idea that is currently gaining momentum is to combine these two paradigms to obtain the best of both worlds (*hybrid* methods) (section 10.5).

10.3. Biomage processing

Digital image processing is a well-documented field (Russ 2002), and many image processing software programs include a wide range of tools. Many of the tools available are limited to 2D images, although many processings would benefit from being applied to other dimensions. In this section, we first present the fundamental operations of image processing, and we then develop three common applications for microscopy.

10.3.1. *Core processings*

Pixel-wise image processing operations are the simplest. They proceed by transforming the image at the level of the individual pixel by changing their value independently of the neighboring pixels. These operations are primarily useful for human image observation, for example, to correct photobleaching or to improve contrast by adjusting the histogram or by pseudo-color (see Figure 10.3A). More complex, filtering transformations are widely used in image processing and analysis because they make it possible to determine the value of each pixel of the output image based on a block of pixels in the input image. The neighbors of the pixel in question are generally the ones that will inform about the local context.

The common image processing filters include the family of linear filters that are entirely characterized by a mathematical operation: the convolution of two functions, f being the input image and h being the kernel of the filter, called the convolution kernel. The result of a convolution is a function g which is an image of the same size as f. The support Ω of the filter kernel h is critical to take more or less neighborhood context. The smallest 2D-centered support is 3 x 3 (see Figure 10.3B). The combination of convolutional operators into very large networks with multiple layers form convolutional neural networks (CNNs) that are at the heart of deep learning methods (Goodfellow et al. 2016). The discrete convolution is written as:

$$\begin{aligned} g[\mathbf{x}] &= f[\mathbf{x}] * h[\mathbf{x}] \\ &= \sum_{\mathbf{k}\in\Omega} f[\mathbf{k}]\, h[\mathbf{x}-\mathbf{k}]. \end{aligned} \qquad [10.2]$$

The values or coefficients of the filter kernel h determine the effect of the filter. For instance, with given coefficients, we can make the image sharper with better defined edges, which will consequently amplify the noise. By choosing other coefficients for the filter kernel h, an integrator/averaging effect can be obtained such that it leads to a reduction in noise but will also make the image more blurry (see Figure 10.3B). These filters include the Gaussian filter which is the most widely used because it allows doing just about anything by the setting of a single parameter σ (standard deviation of the Gaussian) that gives access to the different scales and structures of different sizes in an image (see Figure 10.3C).

Linear filters (convolutions) with large supports can be efficiently performed in a dual domain to the spatial domain of the images: this is the frequency domain called the Fourier domain. It is common practice in signal processing to switch to a frequency representation of the signal using the Fourier transform, even when addressing multidimensional signals such as images (Broughton and Bryan 2018). The fast Fourier transform (FFT) algorithm ensures the fast and loss-free conversion of a domain. As an example, all convolution operations in deconvolution (section 10.3.3) are performed in the frequency domain to be faster.

Mathematical morphology filters, called rank filters, are another class of (nonlinear) filters that modify the shape of objects in the image with a neighborhood $\mathbf{W}$ known as a structuring element (Soille 2013). In this context, the two basic operations are the minimum filter and the maximum filter, also called erosion and dilation, which can be chained together to simplify images. Rank filters are used a lot as segmentation pre-processing, for example, to fill holes in an object or to remove a non-uniform background.

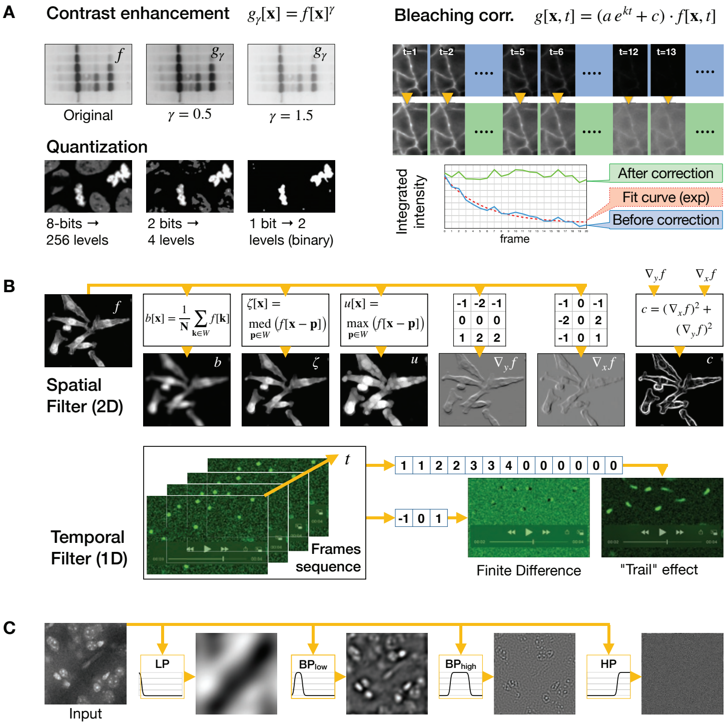

Figure 10.3. *Core operations in image processing. A. Pixel-wise operations. Left: contrast improvement in an image by using a Gamma correction function and simplifying an 8-bit image by quantization on a reduced number of bits. Right: correction of the loss of the fluorescent signal due to photobleaching with a decreasing exponential function parameterized by a, c and k (red dotted line); time series of images before correction in blue and after correction in green. B. Top: 2D spatial filters. From left to right. Blurred image b resulting from a mean filter (box filter) of size $N \times N$ ($N = 7$). Images ζ and u obtained by applying the median and maximum nonlinear filters, respectively, to a neighborhood $\mathbf{W}$. The median filter is very helpful for denoising and for reducing details in the image without losing edges. Edge detection using the Sobel filter c by summing images ∇_y and ∇_x that were obtained by 3x3 finite-difference linear filters. Bottom: example of 1D linear filters in time, applied to a sequence of images, a filter with a “trail” effect, and a finite difference filter. C. Image decomposition with, from left to right, a lowpass filter (LP) to find a non-flat background; a bandpass filter set to low (BP_{low}) to have the nuclei; a bandpass filter set to high (BP_{high}) to have the small particles; a highpass filter (HP) to show the noise*

10.3.2. *Denoising of microscopy images*

Due to physical and technical limitations, the acquisition of a microscopy image is always the result of a compromise between spatial and/or temporal resolution, SNR and reduction of phototoxicity. Thereby, the low illumination to limit photobleaching and protect the living sample induces a high noise level with respect to the signal. Therefore, numerical denoising algorithms will be applied to reduce noise.

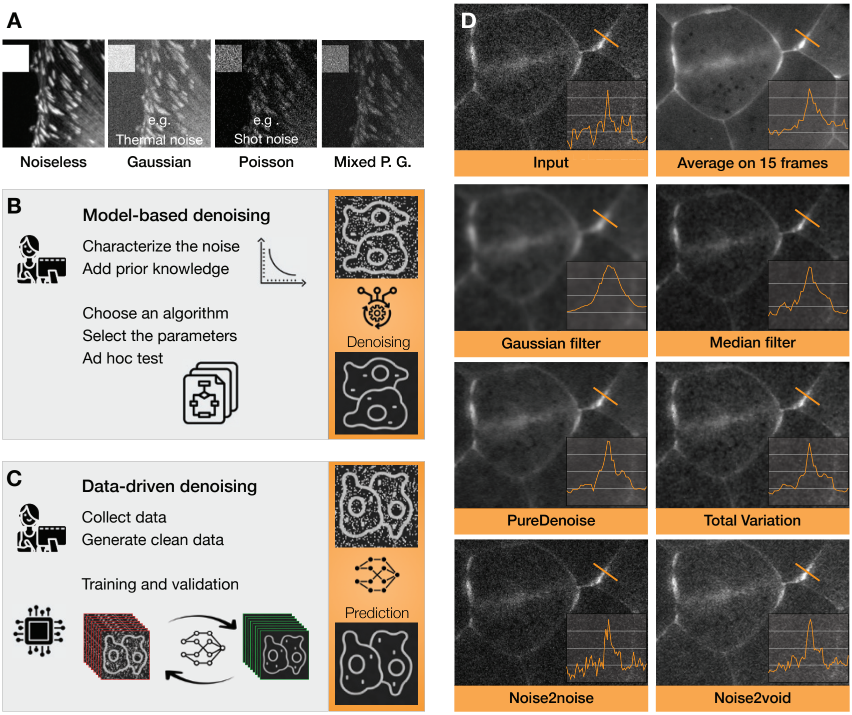

Figure 10.4. *Denoising of microscopy images. A. Examples of different types of noise in fluorescence microscopy: Gaussian, Poisson, a mixture of Gaussian and Poisson. Source: Cell Image Library CIL 38985. B. Design of a model-based denoising algorithm where the developer's involvement is key to best characterizing the noise model. C. Design of a data-driven algorithm where the prediction is based on a neural network trained with a large number of examples. D. Examples of denoising results with several algorithms applied to the "input" image. The denoising result at the top right was obtained by acquiring 15 noisy images of the same (fixed) structure and averaging them*

For practical reasons, we present here some denoising techniques by limiting ourselves to the 2D case. Nevertheless, the algorithms should be applied together with the other dimensions in order to be effective at better exploiting contextual information.

Source of noise

In microscopy, there are many sources of intrinsic noise during acquisition. A dominant source of noise in fluorescence microscopy is shot noise, which originates from the process of counting photons. It is stochastically modeled by a Poisson distribution. There are other sources of noise created by the electronics for reading the signal, such as dark current or readout noise which follow a Gaussian distribution (see Figure 10.4A). In real conditions, the Poisson/Gaussian distribution of noise is poorly known, which makes it difficult to choose and parameterize denoising algorithms.

Model-based methods

Denoising is based on two assumptions: 1) the noise is random and 2) the noise is independent from one pixel to another.

Since noise mostly occurs in high frequencies, a lowpass filter such as the *Gaussian* filter can be applied as a first approach. This linear filter reduces the high signal frequencies, but at the same time, it smooths out the details and contours of objects. It acts a bit as if the resolution of the image were reduced, which is sometimes very dearly earned during the acquisition phase. This major disadvantage of the Gaussian filter explains the development of more selective filtering methods that tend to better preserve the edges, such as the very basic *median* filter (see Figure 10.3B), the iterative anisotropic diffusion filter, the *mean-shift* aggregation filter, the bilateral filter and the inverse filter with total variation regularization (TV) (Meiniel et al. 2018). Filtering in a domain suitable for images is a relevant alternative, for example, in the *wavelet* domain which has excellent signal decorrelation properties with respect to noise (Luisier et al. 2010). Pyramidal image decomposition in wavelet bases is widely used in biomedical image processing because it provides efficient access to multi-scale analysis. In bioimaging, the *non-local mean* algorithm (Boulanger et al. 2009) is another approach that uses the redundancies of local structures in the data (self-similarities), which are very common in biology where images are often composed of a juxtaposition of fairly similar patches.

With all these so-called model-based methods (see Figure 10.4B), parameterization can prove difficult without an explicit characterization of the noise sources and structures of interest. This is why machine learning approaches solely based on data themselves have become very popular because they require no mathematical modeling of noise (Laine et al. 2021).

Data-driven methods

The machine learning approach is based on a training phase (see Figure 10.4C) which is an optimization step of the parameters of the function transforming the noisy image into a denoised image. In deep learning, this transformation function is built with an artificial neural network comprising millions of parameters to be optimized. Once the network has been trained, a prediction has simply to be applied to denoise a new image acquired under the same experimental conditions. Deep learning methods are currently the most efficient of the denoising methods (Kefer et al. 2021) (see Figure 10.4D).

In the supervised mode, training requires a large number of pairs of images: a noisy image and a noise-free image. Since noise-free images are not available in real conditions, other means must be found to obtain these pairs of images, either by simulation by adding noise to low-noise images, or by acquiring two images simultaneously, one with low illumination and one with high illumination, as in the CARE method (Weigert et al. 2018).

More recently, several self-supervised methods have been proposed for efficient denoising without clean noise-free images. Among these methods, the pioneer is Noise2Noise, developed by Lehtinen et al. (2018), which uses pairs of images for training, each consisting of two independent realizations. Noise2Void (Krull et al. 2019) and Noise2Self (Batson and Royer 2019) operate with another strategy to form pairs of images with the noisy image and the noisy image altered by randomly deleting one pixel. These methods can therefore very efficiently predict the erased pixel from the neighborhood. They have given rise to many variations: Noise2Same, Noise2Score, Noise2inverse, StructN2V, etc.

10.3.3. *Deconvolution of microscopy images*

Image formation model and PSF

Image formation in fluorescence microscopy follows the physical laws of light propagation that can be mathematically modeled by a convolution between the emitted light signal and the point-spread function (PSF) to which noise is added (see Figure 10.5A). The PSF represents the response of a microscope to a point light source.

Therefore, in 3D, the PSF integrates the deformations of each focal plane which gives it this typical double cone (hourglass) shape (see Figure 10.5A). The size of the PSF determines the spatial resolution of at best 250 nm laterally and 900 nm axially, and it determines the ability of an optical system to separate (resolve) two point light sources. Accurate knowledge of the PSF is essential for image reconstruction. A PSF can be either obtained experimentally by the acquisition of a z-stack of fluorescent microbeads, or analytically by numerical simulation from the optical parameters of the microscope (Kirshner et al. 2013).

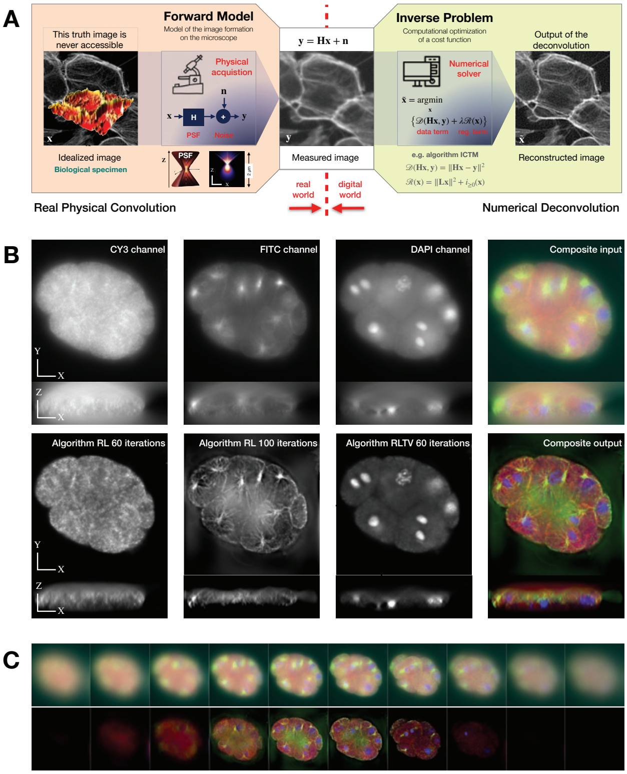
A
Forward Model
Model of the image formation on the microscope
This truth image is never accessible
Physical acquistion
n
x
H
+
y
PSF
Noise
Idealized image
Biological specimen
z
PSF
2 µm
$y = Hx + n$
Measured image
real world
digital world
Inverse Problem
Computational optimization of a cost function
Output of the deconvolution
Numerical solver
$\tilde{x} = \underset{x}{\operatorname{argmin}} \{\mathcal{D}(Hx, y) + \lambda\mathcal{R}(x)\}$
data term
reg. term
e.g. algorithm ICTM
$\mathcal{D}(Hx, y) = \|Hx - y\|^2$
$\mathcal{R}(x) = \|Lx\|^2 + i_{\geq 0}(x)$
$\tilde{x}$
Reconstructed image
Real Physical Convolution
Numerical Deconvolution
B
CY3 channel
FITC channel
DAPI channel
Composite input
Y
X
Z
Algorithm RL 60 iterations
Algorithm RL 100 iterations
Algorithm RLTV 60 iterations
Composite output
C

Figure 10.5. *Deconvolution of microscopy images. A. Formalization of deconvolution as an inverse problem. The forward model models the physical acquisition of the image by the microscope. The problem thus defined is solved by a numerical optimization method. The ICTM algorithm given as an example optimizes a criterion in the sense of the least squares with a Tikhonov regularization and a non-negativity constraint. B. 3D deconvolution of three fluorescent channels, CY3, FITC, DAPI. The sample is a C elegans embryo in which the first eight cell nuclei can be identified in DAPI (blue). The color results are a composition of the three fluorescent channels in a color composite image in red, green and blue. The images are maximum intensity projections along XY and XZ. Top: source image; bottom: image after deconvolution with an RL or RLTV algorithm. C. 3D deconvolution of the same sample. Presentation of a series of images in Z in the composite color mode showing the effect of reducing out-of-focus blurring. Top: images before deconvolution; bottom: images after deconvolution. All deconvolution images were generated with the open-source software DeconvolutionLab2 (Sage et al. 2017)*

Outcome of the deconvolution

The shape of the PSF generates blurring effects that are much more pronounced in the axial direction (see Figure 10.5A); the further away from the focal plane, the more the structures are drowned in an out-of-focus blur. Deconvolution results in reversing the effect of physical convolution by attempting to reassign the signal dispersed by the PSF to its original location; this greatly increases the SNR and signal dynamics, without losing photons. Other scientific disciplines, such as astronomy, use deconvolution routinely (Sarder and Nehorai 2006), but because of the 3D processing, this is the application in microscopy that gives the most remarkable deconvolution results (see Figure 10.5C).

At the same time, deconvolution acts as a denoising and contrast enhancement method, and primarily it considerably reduces out-of-focus blurring (see Figure 10.5B) while preserving the quantitative character of the signal. This thus slightly improves the axial resolution.

Deconvolution as an inverse problem

Deconvolution can be formalized as an inverse problem in which the forward model is a convolution of the observation $\mathbf{x}$ with a filter $\mathbf{H}$ (PSF) to which a noise $\mathbf{n}$ is added (see Figure 10.5A). Since there are only partial and noisy measurements, the problem is ill-posed and can lead to a non-unique solution. Regularizations should be added to restrict the solution to a space that is a priori acceptable. Two regularizations are common, Tikhonov regularization for rather smooth signals, and total variation (TV) regularization for signals composed of piecewise functions. For model-based methods, there are two existing classes of algorithms: direct methods (regularized inverse filter) which are fast but allow only little control over regularization, and iterative methods that gradually add content while adapting the regularization effect, iteration after iteration. In the iterative algorithms, the reference algorithm is the Richardson–Lucy (RL) deconvolution method (Richardson 1972). Many other algorithms can be found such as the Richardson–Lucy total variation (RLTV) (Dey et al. 2006) or iterative constraint Tikhonov–Miller (ICTM) (Van Kempen (2000)) used for the figure (see Figure 10.5C).

Deconvolution is a great example of how the computer can work jointly with the microscope to produce higher-resolution images with a better SNR by means of a process of digital restoration. Deconvolution has limitations that can generate reconstruction artifacts when the noise is too high or when the linear model of the PSF poorly approximates the nonlinear optical system. Deconvolution remains a rather expensive process in terms of computational time. Deconvolution is still in development to develop faster algorithms that can work with nonlinear models. The other direction of development consists of using deep learning, which for the moment faces a lack of properly resolved ground truth data.

10.3.4. *Image registration*

Image registration algorithms alter the coordinates of one or more destination images to align with a source image, either in 2D or 3D. In microscopy, the source and destination data can come from different detectors, different modalities, different time points or sometimes even from alignment with atlases.

Image registration is very commonly needed in bioimaging to be able to geometrically align inconsistent data. CLEM correlative imaging is a good example of this, by matching high-resolution electron microscopy images to light microscopy images (Anderson et al. 2019). This is a task that can be particularly complex when a transformation has to be established between two modalities whose resolution and content are different.

The prior knowledge of a transformation model is an essential element for registration. A distinction is made between rigid models, which correspond to a global affine transformation, and more complex elastic models, which correspond to local deformations.

There are two families of algorithms in image processing, namely intensity-based and feature-based algorithms. Intensity-based algorithms examine the pixel-to-pixel relationships of images to compute the affine transformation matrix or to estimate local deformations. To speed up the computation, a multi-resolution strategy is often used, where the first solution is estimated at a coarse scale (less pixels to go faster). This first estimate is propagated to the higher scale to be refined and so on to the finest scale. This family of methods include in particular the optical flow, which is a very powerful method for estimating the apparent movement and local displacements of pixels, without prior object detection.

Feature-based algorithms proceed in two phases: 1) a detection of elementary visual features (section 10.4.1), such as key points or contours, and 2) a mapping of the features. This is often the approach used for registering unstabilized images or for tracking objects (section 10.4.4).

10.4. Bioimage analysis

In order to guarantee reliable statistics in life sciences, it is essential to acquire a large number of images, which requires automating image analysis. This also makes it possible to ensure unbiased quantification by a human operator. Many books and much software are dedicated to image analysis in computer vision; however, few are yet focusing on the specificities of bioimages (Miura and Sladoje 2020). Below, we present the main tasks of image analysis, visual feature and object detection, image segmentation and object tracking within the context of bioimaging.

10.4.1. *Feature and object detection*

The detection of elementary visual structures such as points of interest, contours or objects of study (nuclei, cells) in microscopy images is an important step in an image analysis system because it is at the basis of numerous methods for object counting, segmentation and tracking.

The methods mainly rely on conventional computer vision techniques, and more specifically the detection of the following structures:

– Edge detection, based on the computation of the gradient. Edges are useful for characterizing the morphology of objects. It can also be applied in 3D to detect membranes (see Figure 10.6A).

– Ridge detection, based on the computation of the eigenvalues of the Hessian matrix (multidimensional partial second derivative) (Frangi et al. 1998). The ridges correspond to the filaments or fibers often present in images in biology (see Figure 10.6B).

– Keypoint detection, based on the evaluation of the gradient at multiple scales, for instance, with the very popular SIFT algorithm (Lowe 1999). Keypoints are local structures, such as filament ends or junctions. Keypoints are very useful for matching images (sections 10.3.4 and 10.4.4) (see Figure 10.6D).

– Directionality detection, based on the analysis of local structures extracted from the gradient structure tensor matrix (Püspöki et al. 2016). This information is very appropriate for evaluating diffusions, for example (see Figure 10.6E).

– Detection of bright spots, a common operation in bioimaging. A very simple approach consists of using bandpass filters (such as the Difference of Gaussian (DoG) and Laplacian of Gaussian (LoG) (Sage et al. 2005) filters), and then finding the local maxima. If there is a model of the shape of the spots, one detection method is to fit the model to the data, which can have the advantage of accurate, sub-pixel localization. The latter approach is the basis for many SMLM algorithms (see Figure 10.6C).

Because they can be easily controlled, model-based methods are preferable for the detection of these elementary local features that have a very explicit model. For other features, without explicit description, data-driven approaches are more adequate; this is in particular the case for texture analysis (Depeursinge et al. 2017) (see Figure 10.6F).

Deep learning methods are well suited for complex object detection tasks. They strive to find a bounding box around each object of interest in an image (see Figure 10.6G). The most well-known methods for object detection in computer vision include R-CNN, which operates in two steps (Girshick et al. 2014; Girshick 2015). The first step extracts region-proposals, and the second classifies them by refining the positions of the bounding boxes. For its part, the YOLO (You Only Look Once) method makes use of a single convolutional lattice to predict both the bounding boxes and the class probabilities for these boxes (Redmon et al. 2016). While these methods are widely used in computer vision, they have only recently been applied to microscopy images, and mainly for the detection of particles and nuclei (Xiao and Yang 2017; Yuan et al. 2019).

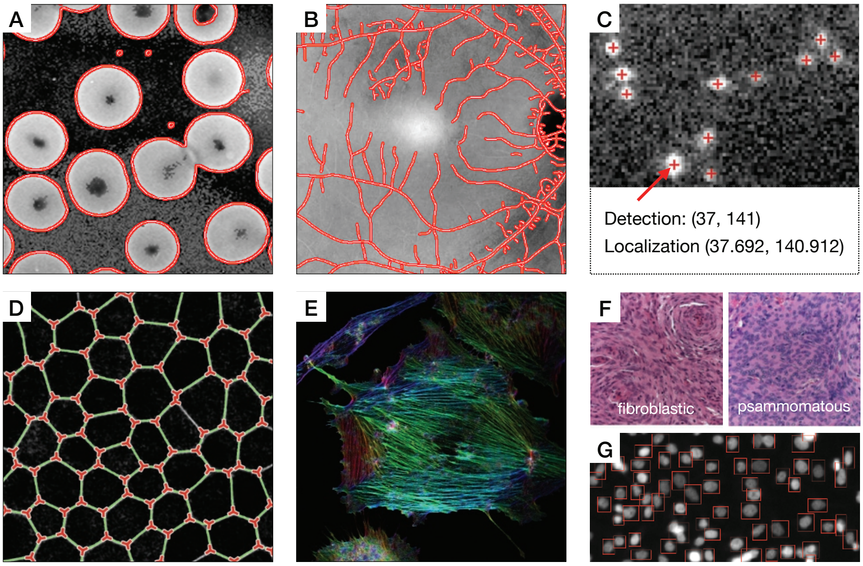

Figure 10.6. *Illustrations of structure and object detection. A. Cell edge detection. B. Ridge detection with the vascular structure of the retina. C. Detection of bright spots (single molecular emitters) using the local maxima of the response of a bandpass filter (DoG). The detected positions are approximated on the pixel grid (in integer value); a localization by fitting a 2D Gaussian function results in a more accurate position (subpixel). D. Three-branch junction detection. E. Detection (color-encoded) of filament directions. F. Examples of two different textures in histopathology images. G. Principle of object detection by bounding box*

10.4.2. *Segmentation*

Segmentation is an often mandatory step in image analysis whose purpose is to partition an image into different regions with similar properties. In other words, segmentation is able to extract objects of interest from the image. Conceptually, its interest lies in obtaining a high-level description of the objects in an image (such as their shape, for example). Segmentation is thus responsible for changing the pixel representation of an image into a representation of its content, such as a series of more relevant objects for the quantification step. This is a challenging task for which a large number of methods have been developed.

Inevitably, no existing segmentation method can be considered generic enough for all applications. They differ according to the imaging modality, the type and shape of the objects of interest, the application and the level of automation.

Pixel-based methods

Basic segmentation methods often include thresholding, which classifies pixels according to their intensity without taking their context into account (see Figure 10.7A). Intensity thresholding is often combined with ad hoc processing solutions to correct unavoidable misclassifications, such as binary morphological operators (erosion, dilation, median) or the watershed that separates objects in contact. While these methods work well for objects that are homogeneous in size and light intensity (e.g. nuclei), they are generally not very robust in bioimaging when a great variability is present in the experimental data. They are indeed prone to experimental variations in intensity. When interaction is possible with the user, semi-automatic methods such as region growing can be used to help with segmentation, which agglomerates pixels of comparable intensity from a source point, called a seed, defined by the user.

Model-based methods

As an alternative to these segmentation methods based on pixel intensity, deformable models, also called *active contours* or *snakes* are widely used in bioimaging (see Figure 10.7B and 10.7C). They were first proposed in 2D by Kass et al. (1988) and generalized to 3D by Terzopoulos et al. (1988). Snakes are semi-automatic methods that offer an excellent trade-off among flexibility, efficiency and interactivity. They enable a manual correction of the segmentation result, which is a significant advantage because automatic segmentation methods rarely achieve an ideal accuracy.

Snakes extract objects of interest in images by deforming their outline towards the edges of the objects of interest by minimizing an energy function whose purpose is to find the optimal configuration of the snake outline. To construct a snake, two elements are needed: 1) a geometric representation of the snake that describes the nature of its outline and determines some geometric properties (e.g. smooth curves, topological properties). There are several families of representations: geodesics, meshes, parametric and subdivisions (Delgado-Gonzalo et al. 2014; Badoual et al. 2016). 2) The appropriate energy function that drives the adjustment of the snake contour to the image data. The choice of this term of energy is crucial because it affects the quality of the result of the segmentation and, as such, is application-dependent. Energy terms can be the attraction to the contour (*edge-based energy*) or the attraction to homogeneous regions (*region-based energy*) (Jacob et al. 2004).

While these methods have many advantages for segmentation in bioimaging, they also present some limitations. First, they require an initial prompt of the snake that can either be done manually or using an auxiliary detection method (see section 10.4.1). Second, the division of objects tends to be excessive when they are very close, which will require a collision detection algorithm. Finally, for images where the contrast between the object of interest and the background is weak, energies that rely simply on intensity information are not very effective. The spatial organization of the pixels (texture) within the object of interest must be incorporated into the energy function (Depeursinge et al. 2017). While some *texture-based energies* have been developed for snakes (Badoual et al. 2019b), data-driven segmentation methods, especially deep learning, excel in this kind of applications when a large amount of example data is available.

Data-driven methods

Data-driven methods usually turn segmentation into a pixel classification problem: each pixel is assigned a label or class, for example, class 0 for the image background, 1 for the nuclei and 2 for the cytoplasm. These supervised machine learning methods are trained from images whose pixels have been previously annotated by field experts.

Among the machine learning methods widely used in bioimaging, we find support vector machines (SVM) (Cortes and Vapnik 1995) or Random Forests (RF) (Ho 1995) particularly used in the Ilastik software (Berg et al. 2019). For these classifiers, the local characteristics of the pixels are calculated from many generic filters such as those presented in section 10.3.1, such as edge detectors, the multiscale Gaussian filter or texture-based filters (Depeursinge et al. 2017).

In deep learning, convolutional neural networks, such as U-Net (Ronneberger et al. 2015; Çiçek et al. 2016) or SegNet (Badrinarayanan et al. 2017), have become very popular because they outperform other segmentation methods of biological images (Ulman et al. 2017). In particular, their ability to automatically determine the right discriminating characteristics of pixels makes them particularly good at classifying pixels. They can be used to segment cells with complex textures, without the need to explicitly define an energy model or function, as shown in Figure 10.7D for the segmentation of HeLa cells. Most of these networks now work with 2D images and also in 3D at the cost of more expensive training. Within this context, these data-driven methods perform semantic segmentation, not instance segmentation. For each pixel, they estimate the probability of belonging to each predicted class (e.g. cell or background). To then achieve a segmentation of instances, the model-based methods mentioned above, such as a combination of thresholding and watershed or active contours, have to be applied to the result.

Instance segmentation has become very popular in computer vision and there are increasingly more pre-trained neural networks for bioimaging applications. This recent trend is evolving fast. At the moment, the best known can be mentioned:

– Cellpose (Schmidt et al. 2018) (see Figure 10.7E and 10.7F) is a model trained to detect cells of different shapes and in different modalities.

– StarDist (Stringer et al. 2021) (see Figure 10.7G and 10.7H) is capable of quickly segmenting thousands of objects with rounded shapes (such as nuclei), even when they can be found in very high density in images.

– YOLO (Redmon et al. 2016) (see Figure 10.7E and 10.7F) is a neural network that places a bounding box around image objects. This network must be re-trained with application-specific data.

– Segment anything for microscopy (Archit et al. 2023) follows the public availability of the segment-anything-model (SAM) which is a "foundation" model trained with 11 million natural (non-microscopy) images annotated with 1 billion masks. Segment anything for microscopy has extended SAM to microscopy data to provide a highly versatile annotation tool.

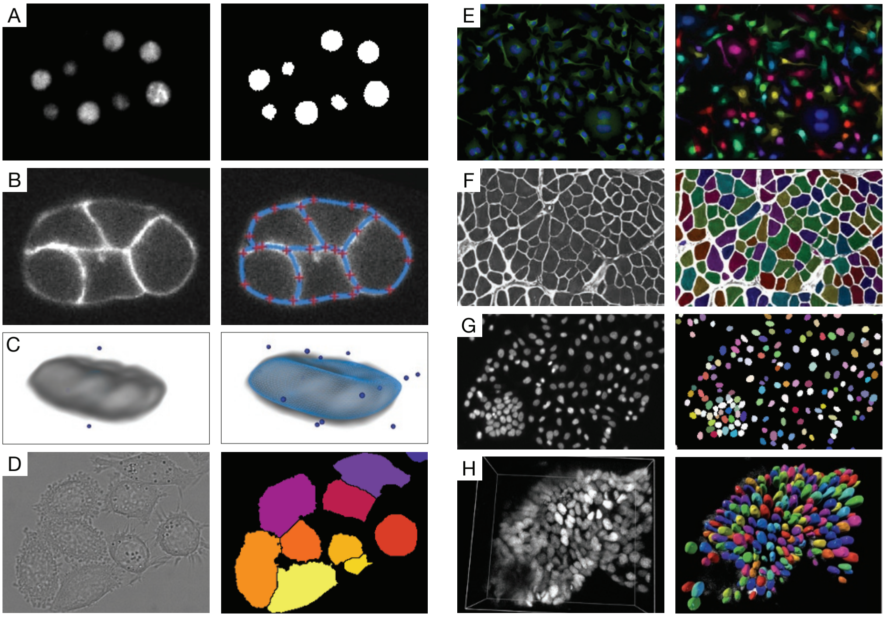

Figure 10.7. *Microscopy image segmentation. A. Nuclei segmentation by thresholding. B. Membrane segmentation of a C elegans embryo with a 2D (ridge-based energy) active contour (Badoual et al. 2019a). Source: R. Jankele and P. Gønczy, EPFL. C. Segmentation of a rat neuron nucleus with a 3D active contour (Badoual et al. 2021). D. HeLa cells acquired in phase contrast segmented with a U-Net-type CNN network. Image of the cell tracking challenge. E. and F. Cell segmentation with Cellpose. Source: http://www.cellpose.org/. G. and H. 2D and 3D dense nuclei segmentation, respectively, with StarDist. Sources: Broad Bioimage Benchmark Collection (Caicedo et al. 2019) and from Schmidt et al. (2018)*

10.4.3. *Image classification*

In bioimaging, the classification of objects of interest is an important step. It can be used for instance to differentiate between living and dead cells, cancer cells or cells during the division cycle, or ribosomes from mitochondria. To this end, there are methods known as image-level classification methods that assign a label to each input image, as opposed to pixel classification methods (seen in section 10.4.2 on segmentation).

It is seldom the case, however, to have to directly classify the entire image, which usually contains a multitude of objects of interest. Instead, sub-images are classified, namely patches of the original image, each comprising a unique object of interest. These patches are obtained by extracting objects in the form of a region of interest (ROI) during the segmentation and/or detection steps introduced earlier.

Most classification methods are based on machine learning. The most conventional approach is the kMeans method, which is an unsupervised method. This is instead referred to as clustering, i.e. the objective consists of separating objects of interest into different classes without attaching a label to them. The kMeans method is applied either directly to a set of images or after using a dimensionality reduction technique (e.g. PCA, T-SNE (Van der Maaten and Hinton 2008)). This method is quick and easy, but highly dependent on initialization and may fail when classes are complex and intertwined. Among the supervised methods widely used in bioimaging, we find the methods mentioned in segmentation for pixel classification, namely SVM and Random Forests, but here they are applied at the image scale and no longer at the pixel level. These classifiers require the extraction of structural features (*image features*) and are trained from labels of classes that must be known. More recently, deep learning methods, especially CNN-based (for instance, VGG16 (Simonyan and Zisserman 2014), ResNet (He et al. 2016)), automatically learn these image features from the data, and have demonstrated their effectiveness for microscopy image classification (Xing et al. 2017; Meijering 2020). The main limitation of these deep learning methods is the need for a large amount of annotated data for training (see section 10.5).

10.4.4. *Tracking of biological objects*

The tracking of cells, nuclei or any other bio-particles is of great importance in bioimaging. The analysis of trajectories in 2D or 3D underlines cell dynamics and interactions between objects of interest.

Tracking is a particularly complex task (Meijering et al. 2006) because there is a great diversity of objects of interest (targets) to be tracked, ranging from the entire animal (e.g. *C elegans*, mice) (Mathis et al. 2018) to the single molecule, including the often very numerous and dense cells in images (see Figure 10.8B). Moreover, their dynamics and cell divisions are all factors that complicate the design of algorithms. In bioimaging, tracking is rarely in real time, unlike computer-based vision applications; it is therefore possible to take the past and the future into account to make algorithms more robust.

The standard approach is to perform a "frame-to-frame" tracking with a spatial detection strategy for objects in each image followed by a time linking of the objects (see Figure 10.8A) (Meijering et al. 2012). The first phase consists of detecting the objects of interest in each of the images with segmentation, model-based or data-driven methods, as previously presented. Each detected object is then described by a vector of characteristics that integrates visual features such as position, size, color or morphology. In a second step, the objects of two consecutive images are linked by similarity and/or proximity by evaluating an energy function. Making the connections in pairs of images only in the sequence has a major disadvantage due to the risk of error propagation; a false detection in one image will have consequences for the rest of the image sequence. In order to make the algorithms more robust, it proves more efficient to account for time persistence, assuming short displacements to integrate past and future information. This is the case of the MHT (Multiple Hypothesis Test (Chenouard et al. 2013) tracking algorithm which is based on probabilistic associations with a few images, or the SpotTracker algorithm which computes the least energy-intensive path of a particle over the entire sequence (see Figure 10.8(D)) (Sage et al. 2005), or still the U-track algorithm which involves a multiple stage linking strategy, by starting by grouping the objects into small segments of trajectories and then combining them step by step (Jaqaman et al. 2008).

Deep learning methods only deal with the spatial segmentation part because the temporal linking step is always more efficient with model-based approaches.

Single molecule trajectography

These tracking methods are also used in the context of single-molecule imaging (SMLM, QDot, sptPALM, uPAINT, etc.) for determining the molecular interactions between the protein of interest and the partner molecules (Sibarita 2014). Using statistical approaches, it is also possible to compute diffusion maps from displacement vectors or from trajectories (Kusumi et al. 2014).

Trajectory analysis

The statistical analysis of trajectories provides relevant information for interpreting molecular interactions, or cell migration speeds. It is possible to calculate a given number of parameters, such as velocities, diffusion and directionality. Mean square displacement (MSD) makes it possible to classify trajectories according to their diffusive behavior, i.e. directional trajectories (superdiffusive), more confined trajectories (subdiffusive) or Brownian motions that are very common in biology, as shown in Figure 10.8C.

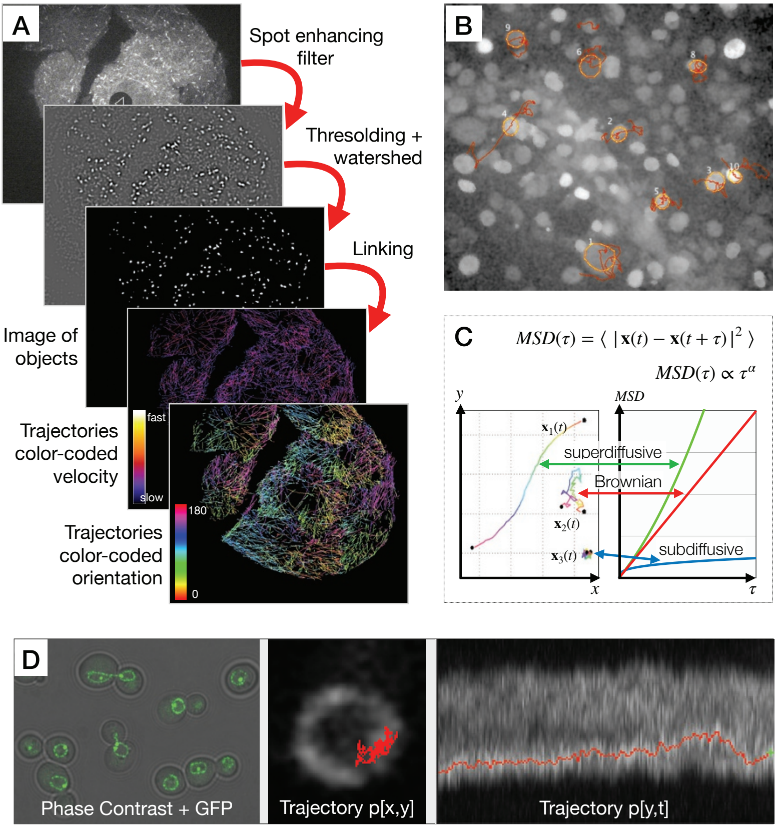

Figure 10.8. *Tracking of biological objects. A. Tracking of microtubule tips in a sequence of 400 images. From top to bottom: example of an image at a given time; image after bandpass filtering that eliminates background and noise; image after thresholding and watershed to isolate objects of interest; temporal linking of all detected objects and view of all trajectories with a color code that indicates the speed; view of all paths with color coding indicating orientation. B. "Image-to-image" segmentation and temporal connection of 10 cells chosen by the user in a very dense medium. Bioluminescence image for studies of the circadian cycle (Sage et al. 2010). C. Three examples of 2D trajectory:* $x_1(t)$ *with a superdiffusive displacement model,* $x_2(t)$ *with a diffusive normal displacement model also called the Brownian motion, and* $x_3(t)$ *with a subdiffusive model, confined to a small space. Left: representation of MSDs corresponding to the three types of trajectories. D. Tracking of a telomere in a cell nucleus. The cell membrane and telomere are labeled in GFP. The trajectory is illustrated by a red line on the planes (X,Y) and (Y,T)*

Trajectory analysis is also used to extract cell lineage including divisions and migrations, for example, for embryogenesis applications.

10.5. Discussion

In this chapter, we have presented the main guidelines for image processing and analysis applied to microscopy images. Parallels are often drawn between two methodological approaches: the conventional model-based approach and the data-driven approach which is based on technologies derived from artificial intelligence. While AI models have revolutionized the world of image analysis, and consequently the world of cellular imaging, fundamental questions are still the topic of debate about the balance between the benefits and risks of AI, the shift to a data-driven paradigm and the expected prospects of “smart microscopy”.

Advantages and risks of deep learning

Most image analysis system designers face a dilemma between conventional methods and DL approaches, often using selection criteria that remain largely unmastered or obscure, sometimes driven by biases or current hype.

Deep learning methods have clearly pushed the boundaries of bioimage analysis. They have enabled solving problems that were considered unsolvable with traditional approaches by being able to integrate complex information in a large quantity of images. Nevertheless, this results in significant costs: 1) the cost associated with building training databases. The generation of reference image banks is very labor-intensive in the field of life sciences. Big companies such as Google or Meta have also created large databases of natural images by exploiting the information available on the Internet without much regard for ethical issues and the intellectual property of this information. As mentioned in this chapter, Meta has made available a segment-anything foundation model that can segment microscopy images itself on high-performance machines. 2) The cost of training. Neural networks are indeed becoming larger and larger with millions of parameters to be optimized, which requires significant resources, both computational (e.g. GPU) and human (e.g. design, adjustment of hyper parameters).

DL systems have a “black-box” aspect that scientists hardly appreciate, as they have no proof of convergence and no traceability of errors. On the other hand, model-based systems are well mastered, with results that can be explained relatively to a physical model formulated by engineers; they can be generalized within the framework of established assumptions, and interpreted by humans.

Many applications have combined the two approaches in a hybrid design, taking advantage of both the versatility of DL and the control of physical models. Developers must understand both worlds perfectly to combine them rather than oppose them. They must have sufficient training to be able to distinguish a task that can be performed using a model and thus avoid the risk of using DL when it is not justified.

For a non-expert user, the use of deep learning methods may seem out of reach. For the training phase, not only must large amounts of ground truth data be collected, but a high-performance machine with large memory GPUs is also needed. In addition, it is most of the time necessary to have very good knowledge of programs in Python environments that are known to be difficult for neophytes. Efforts are being made to integrate training programs into more user-friendly environments, for example, with notebooks from ZeroCostDLMic (von Chamier et al. 2021). On the other hand, the use of pre-trained deep learning models is much more accessible, as long as there is a model that is exactly suitable for the intended application. The Bioimage Model Zoo (BMZ) (Ouyang et al. 2022) is a repository of standardized models for bioimaging that attempts to gather many pre-trained models. Deep learning predictions can usually be achieved on standard machines without GPUs. Therefore, ImageJ/Fiji includes a plugin for StarDist, a plugin for Noise2Void and also a deepImageJ plugin that make it possible to execute inferences with all BMZ models (Gómez-de Mariscal et al. 2021).

Shift to data design

DL methods, in supervised mode, require a large number of training images, in particular ground truth images or annotated images. These images must be representative of a given application; in other words, they must cover all the variability of situations. This is particularly problematic in bioimaging, due to the variety of phenotypes, dynamics and interactions of the living environment. In the absence of large reference databases for bioimaging, other strategies must be used:

– large databases of all origins designed for computer vision. The networks pre-trained with these databases can then be fine-tuned from a few application-specific bioimages;

– data annotated manually by experts or, alternatively, by non-experts;

– artificial data generated by a simulator. In fluorescence microscopy, for which image formation models are well mastered, the results are promising;

– microscopy tips for acquiring images in two modes, in high and low qualities (Weigert et al. 2018).

Some initiatives have been taken, for specific applications, to create open image databases (e.g. Kaggle, Grand Challenges), which can be enhanced in a collaborative way.

With the emergence of DL, all the challenges of bioimage analysis have shifted from the design of algorithms based on the development of physical or biological models, to a massive data collection process to build training databases, which represents a major effort by the bioimaging community.

Smart microscopy

A new era has very recently begun with the developments in “smart microscopy” (Carpenter et al. 2023). This approach aims to perform real-time image analysis during the image acquisition process in order to collect only the images necessary for the experiment. This real-time image analysis can be either model-based or data-driven. It therefore regularly provides information for controlling the microscope to adapt the images acquired to the phenotype observed. For example, we can imagine adapting the spatial and temporal resolutions, in real time according to the samples, in order to reduce toxicity by illuminating only what is necessary. Intelligent predictions can also guide acquisition to the right place and at the right time in order to focus only on the event of interest (Mahečić et al. 2022).

Smart microscopy and deep learning combined with model-based image analysis undoubtedly open up new perspectives in bioimaging that could provide intelligent solutions to data flows in an attempt to unravel the secrets of the fabulous cellular machinery.

10.6. References

Anderson, K., Nilsson, T., Fernandez-Rodriguez, J. (2019). Challenges for CLEM from a light microscopy perspective. In *Correlative Imaging: Focusing on the Future*, Verkade, P. and Collinson, L. (eds). Wiley, Hoboken, NJ.

Archit, A., Nair, S., Khalid, N., Hilt, P., Rajashekar, V., Freitag, M., Gupta, S., Dengel, A., Ahmed, S., Pape, C. (2023). Segment anything for microscopy. *bioRxiv*. doi: 10.1101/2023.08.21.554208.

Badoual, A., Schmitter, D., Uhlmann, V., Unser, M. (2016). Multiresolution subdivision snakes. *IEEE Transactions on Image Processing*, 26(3), 1188–1201.

Badoual, A., Galan, A., Sage, D., Unser, M. (2019a). Deforming tessellations for the segmentation of cell aggregates. In *IEEE 16th International Symposium on Biomedical Imaging (ISBI 2019)*, Cambridge, MA.

Badoual, A., Unser, M., Depeursinge, A. (2019b). Texture-driven parametric snakes for semi-automatic image segmentation. *Computer Vision and Image Understanding*, 188, 102793.

Badoual, A., Romani, L., Unser, M. (2021). Active subdivision surfaces for the semiautomatic segmentation of biomedical volumes. *IEEE Transactions on Image Processing*, 30, 5739–5753. doi: 10.1109/TIP.2021.3087947.

Badrinarayanan, V., Kendall, A., Cipolla, R. (2017). Segnet: A deep convolutional encoder-decoder architecture for image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(12), 2481–2495.

Batson, J. and Royer, L. (2019). Noise2self: Blind denoising by self-supervision. In *International Conference on Machine Learning*. PMLR, London.

Berg, S., Kutra, D., Kroeger, T., Straehle, C.N., Kausler, B.X., Haubold, C., Schiegg, M., Ales, J., Beier, T., Rudy, M. et al. (2019). Ilastik: Interactive machine learning for (bio) image analysis. *Nature Methods*, 16(12), 1226–1232.

Berlin, S., Carroll, E.C., Newman, Z.L., Okada, H.O., Quinn, C.M., Kallman, B., Rockwell, N.C., Martin, S.S., Lagarias, J.C., Isacoff, E.Y. (2015). Photoactivatable genetically encoded calcium indicators for targeted neuronal imaging. *Nature Methods*, 12(9), 852–858.

Boulanger, J., Kervrann, C., Bouthemy, P., Elbau, P., Sibarita, J.-B., Salamero, J. (2009). Patch-based nonlocal functional for denoising fluorescence microscopy image sequences. *IEEE Transactions on Medical Imaging*, 29(2), 442–454.

Broughton, S.A. and Bryan, K. (2018). *Discrete Fourier Analysis and Wavelets: Applications to Signal and Image Processing*. John Wiley & Sons, New York.

Caicedo, J.C., Goodman, A., Karhohs, K.W., Cimini, B.A., Ackerman, J., Haghighi, M., Heng, C., Becker, T., Doan, M., McQuin, C. et al. (2019). Nucleus segmentation across imaging experiments: The 2018 Data Science Bowl. *Nature Methods*, 16(12), 1247–1253.

Carpenter, A.E., Kamentsky, L., Eliceiri, K.W. (2012). A call for bioimaging software usability. *Nature Methods*, 9(7), 666–670.

Carpenter, A., Cimini, B., Eliceiri, K. (2023). Smart microscopes of the future. *Nature Methods*, 20, 962–964.

von Chamier, L., Laine, R.F., Laine, R.F., Jukkala, J., Spahn, C., Krentzel, D., Krentzel, D., Nehme, E., Lerche, M., Hernández-Pérez, S. et al. (2021). Democratising deep learning for microscopy with ZeroCostDL4Mic. *Nature Communications*, 12(1), 2276–2276.

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.

Chenouard, N., Bloch, I., Olivo-Marin, J.-C. (2013). Multiple hypothesis tracking for cluttered biological image sequences. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(11), 2736–3750.

Çiçek, Ö., Abdulkadir, A., Lienkamp, S.S., Brox, T., Ronneberger, O. (2016). 3D U-Net: Learning dense volumetric segmentation from sparse annotation. In *International Conference on Medical Image Computing and Computer Assisted Intervention*, Springer, New York.

Cohen, A.R. (2014). Extracting meaning from biological imaging data. *Molecular Biology of the Cell*, 25(22), 3470–3473.

Cortes, C. and Vapnik, V. (1995). Support-vector networks. *Machine Learning*, 20(3), 273–297.

Delgado-Gonzalo, R., Uhlmann, V., Schmitter, D., Unser, M. (2014). Snakes on a plane: A perfect snap for bioimage analysis. *IEEE Signal Processing Magazine*, 32(1), 41–48.

Depeursinge, A., Al-Kadi, O.S., Mitchell, J.R. (2017). *Biomedical Texture Analysis: Fundamentals, Tools and Challenges*. Academic Press, Cambridge, MA.

Dey, N., Blanc-Feraud, L., Zimmer, C., Roux, P., Kam, Z., Olivo-Marin, J.-C., Zerubia, J. (2006). Richardson–Lucy algorithm with total variation regularization for 3D confocal microscope deconvolution. *Microscopy Research and Technique*, 69(4), 260–266.

Ellenberg, J., Swedlow, J.R., Barlow, M., Cook, C.E., Sarkans, U., Patwardhan, A., Brazma, A., Birney, E. (2018). A call for public archives for biological image data. *Nature Methods*, 15(11), 849–854.

Frangi, A.F., Niessen, W.J., Vincken, K.L., Viergever, M.A. (1998). Multiscale vessel enhancement filtering. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer, New York.

Girshick, R. (2015). Fast R-CNN. In *Proceedings of the IEEE International Conference on Computer Vision*, Cambridge, MA.

Girshick, R., Donahue, J., Darrell, T., Malik, J. (2014). Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Cambridge, MA.

Gómez-de Mariscal, E., García-López-de Haro, C., Ouyang, W., Donati, L., Lundberg, E., Unser, M., Muñoz-Barrutia, A., Sage, D. (2021). DeepImageJ: A user-friendly environment to run deep learning models in ImageJ. *Nature Methods—Techniques for Life Scientists and Chemists*, 18(10), 1192–1195.

Goodfellow, I., Bengio, Y., Courville, A. (2016). *Deep Learning*. MIT Press, Cambridge, MA.

Gustafsson, M.G. (2000). Surpassing the lateral resolution limit by a factor of two using structured illumination microscopy. *Journal of Microscopy*, 198(2), 82–87.

He, K., Zhang, X., Ren, S., Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Cambridge, MA.

Ho, T.K. (1995). Random decision forests. In *Proceedings of 3rd International Conference on Document Analysis and Recognition*. IEEE, Cambridge, MA.

Jacob, M., Blu, T., Unser, M. (2004). Efficient energies and algorithms for parametric snakes. *IEEE Transactions on Image Processing*, 13(9), 1231–1244.

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.

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

Kefer, P., Iqbal, F., Locatelli, M., Lawrimore, J., Zhang, M., Bloom, K., Bonin, K., Vidi, P.-A., Liu, J. (2021). Performance of deep learning restoration methods for the extraction of particle dynamics in noisy microscopy image sequences. *Molecular Biology of the Cell*, 32(9), 903–914.

Kirshner, H., Aguet, F., Sage, D., Unser, M. (2013). 3-D PSF fitting for fluorescence microscopy: Implementation and localization application. *Journal of Microscopy*, 249(1), 13–25.

Krull, A., Buchholz, T.-O., Jug, F. (2019). Noise2void-learning denoising from single noisy images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. IEEE, Cambridge, MA.

Kusumi, A., Tsunoyama, T.-A., Hirosawa, K., Kasai, R., Fujiwara, T. (2014). Tracking single molecules at work in living cells. *Nature Chemical Biology*, 10, 524–532.

Laine, R.F., Jacquemet, G., Krull, A. (2021). Imaging in focus: An introduction to denoising bioimages in the era of deep learning. *The International Journal of Biochemistry & Cell Biology*, 140, 106077.

Lehtinen, J., Munkberg, J., Hasselgren, J., Laine, S., Karras, T., Aittala, M., Aila, T. (2018). Noise2noise: Learning image restoration without clean data. *arXiv*, 1803.04189.

Lowe, D.G. (1999). Object recognition from local scale-invariant features. In *Proceedings of the Seventh IEEE International Conference on Computer Vision*, Cambridge, MA.

Lucas, A.M., Ryder, P.V., Li, B., Cimini, B.A., Eliceiri, K.W., Carpenter, A.E. (2021). Open-source deep-learning software for bioimage segmentation. *Molecular Biology of the Cell*, 32(9), 823–829.

Luisier, F., Vonesch, C., Blu, T., Unser, M. (2010). Fast interscale wavelet denoising of Poisson-corrupted images. *Signal Processing*, 90(2), 415–427.

Mahečić, D., Stepp, W., Zhang, C., Griffie, J., Weigert, M., Manley, S. (2022). Event-driven acquisition for content-enriched microscopy. *Nature Methods*, 19(10), 1262–1267.

Mathis, A., Mamidanna, P., Cury, K.M., Abe, T., Murthy, V.N., Mathis, M.W., Bethge, M. (2018). DeepLabCut: Markerless pose estimation of user-defined body parts with deep learning. *Nature Neuroscience*, 21(9), 1281–1289.

Meijering, E. (2020). A bird's-eye view of deep learning in bioimage analysis. *Computational and Structural Biotechnology Journal*, 18, 2312.

Meijering, E., Smal, I., Danuser, G. (2006). Tracking in molecular bioimaging. *IEEE Signal Processing Magazine*, 23(3), 46–53.

Meijering, E., Dzyubachyk, O., Smal, I. (2012). Methods for cell and particle tracking. *Methods in Enzymology*, 504, 183–200.

Meijering, E., Carpenter, A.E., Peng, H., Hamprecht, F.A., Olivo-Marin, J.-C. (2016). Imagining the future of bioimage analysis. *Nature Biotechnology*, 34(12), 1250–1255.

Meiniel, W., Olivo-Marin, J.-C., Angelini, E.D. (2018). Denoising of microscopy images: A review of the state-of-the-art, and a new sparsity-based method. *IEEE Transactions on Image Processing*, 27(8), 3842–3856.

Miura, K. and Sladoje, N. (2020). *Bioimage Data Analysis Workflows*. Springer, Berlin.

Myers, G. (2012). Why bioimage informatics matters. *Nature Methods*, 9(7), 659–660.

Ouyang, W. and Zimmer, C. (2017). The imaging tsunami: Computational opportunities and challenges. *Current Opinion in Systems Biology*, 4(9), 105–113.

Ouyang, W., Beuttenmueller, F., de Mariscal, E.G., Pape, C., Burke, T., de Haro, C.G.-L., Russell, C., Moya-Sans, L., de-la Torre-Gutiérrez, C., Schmidt, D. et al. (2022). Bioimage model zoo: A community-driven resource for accessible deep learning in bioimage analysis. *bioRxiv*. doi: 10.1101/2022.06.07.495102.

Püspöki, Z., Storath, M., Sage, D., Unser, M. (2016). Transforms and operators for directional bioimage analysis: A survey. In *Focus on Bio-Image Informatics*, De Vos, W.H., Munck, S., Timmermans, J.-P. (eds). Springer, Cham, 69–93.

Redmon, J., Divvala, S., Girshick, R., Farhadi, A. (2016). You only look once: Unified, real-time object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Cambridge, MA.

Richardson, W.H. (1972). Bayesian-based iterative method of image restoration. *JoSA*, 62(1), 55–59.

Ronneberger, O., Fischer, P., Brox, T. (2015). U-Net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical Image Computing and Computer Assisted Intervention*. Springer, New York.

Russ, J.C. (2002). *The Image Processing Handbook*. CRC Press, Boca Raton, FL.

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.

Sage, D., Unser, M., Salmon, P., Dibner, C. (2010). A software solution for recording circadian oscillator features in time-lapse live cell microscopy. *Cell division*, 5(1), 1–9.

Sage, D., Donati, L., Soulez, F., Fortun, D., Schmit, G., Seitz, A., Guiet, R., Vonesch, C., Unser, M. (2017). DeconvolutionLab2: An open-source software for deconvolution microscopy. *Methods*, 115, 28–41.

Sage, D., Pham, T.-A., Babcock, H., Lukes, T., Pengo, T., Chao, J., Velmurugan, R., Herbert, A., Agrawal, A., Colabrese, S. et al. (2019). Super-resolution fight club: Assessment of 2D and 3D single-molecule localization microscopy software. *Nature Methods*, 16(5), 387–395.

Sarder, P. and Nehorai, A. (2006). Deconvolution methods for 3-D fluorescence microscopy images. *IEEE Signal Processing Magazine*, 23(3), 32–45.

Schindelin, J., Rueden, C.T., Hiner, M.C., Eliceiri, K.W. (2015). The ImageJ ecosystem: An open platform for biomedical image analysis. *Molecular Reproduction and Development*, 82(7–8), 518–529.

Schmidt, U., Weigert, M., Broaddus, C., Myers, G. (2018). Cell detection with star-convex polygons. In *International Conference on Medical Image Computing and Computer Assisted Intervention*. Springer, New York.

Sibarita, J.-B. (2014). High-density single-particle tracking: Quantifying molecule organization and dynamics at the nanoscale. *Histochemistry and Cell Biology*, 141.

Simonyan, K. and Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. *arXiv*, 1409.1556.

Soille, P. (2013). *Morphological Image Analysis: Principles and Applications*. Springer Science & Business Media, Berlin.

Stringer, C., Wang, T., Michaelos, M., Pachitariu, M. (2021). Cellpose: A generalist algorithm for cellular segmentation. *Nature Methods*, 18(1), 100–106.

Terzopoulos, D., Witkin, A., Kass, M. (1988). Constraints on deformable models: Recovering 3D shape and nonrigid motion. *Artificial Intelligence*, 36(1), 91–123.

Tikhonov, A.N. (1963). Solution of incorrectly formulated problems and the regularization method. *Soviet Math Dokl*, 4, 1035–1038.

Ulman, V., Maška, M., Magnusson, K.E., Ronneberger, O., Haubold, C., Harder, N., Matula, P., Matula, P., Svoboda, D., Radojevic, M. et al. (2017). An objective comparison of cell-tracking algorithms. *Nature Methods*, 14(12), 1141–1152.

Valli, J., Garcia-Burgos, A., Rooney, L.M., de Melo E Oliveira, B.V., Duncan, R.R., Rickman, C. (2021). Seeing beyond the limit: A guide to choosing the right super resolution microscopy technique. *Journal of Biological Chemistry*, 297(1), 100791.

Van der Maaten, L. and Hinton, G. (2008). Visualizing data using t-SNE. *Journal of Machine Learning Research*, 9(11), 2579–2605.

Weigert, M., Schmidt, U., Boothe, T., Müller, A., Dibrov, A., Jain, A., Wilhelm, B., Schmidt, D., Broaddus, C., Culley, S. et al. (2018). Content-aware image restoration: Pushing the limits of fluorescence microscopy. *Nature Methods*, 15(12), 1090–1097.

Xiao, Y. and Yang, G. (2017). A fast method for particle picking in cryo-electron micrographs based on fast R-CNN. *AIP Conference Proceedings*, 1836, 020080.

Xing, F., Xie, Y., Su, H., Liu, F., Yang, L. (2017). Deep learning in microscopy image analysis: A survey. *IEEE Transactions on Neural Networks and Learning Systems*, 29(10), 4550–4568.

Yuan, P., Rezvan, A., Li, X., Varadarajan, N., Van Nguyen, H. (2019). Phasetime: Deep learning approach to detect nuclei in time lapse phase images. *Journal of Clinical Medicine*, 8(8), 1159.

Zimmermann, T., Rietdorf, J., Pepperkok, R. (2003). Spectral imaging and its applications in live cell microscopy. *FEBS Letters*, 546(1), 87–92.