An ImageJ/Fiji plugin for segmenting and quantifying sub-cellular structures in fluorescence microscopy images  
Aurélien Rizk, Grégory Paul, Pietro Incardona, Milica Bugarski, Maysam Mansouri, Axel Niemann, Urs Ziegler, Philipp Berger, Ivo F. Sbalzarini

1 MOSAIC Group, Center of Systems Biology, Max Planck Institute of Molecular Cell Biology and Genetics, 01307 Dresden, Germany. (Previously: MOSAIC Group, Department of Computer Science, ETH Zurich, 8092 Zürich, Switzerland.)
2 Paul Scherrer Institute, Biomolecular Research, Molecular Cell Biology, Villigen PSI, Switzerland.
3 University of Zurich, Center for Microscopy and Image Analysis, 8057 Zürich, Switzerland.
4 ETH Zurich, Institute of Molecular Health Sciences, 8093 Zürich, Switzerland.
5 Now at: ETH Zurich, Computer Vision Laboratory, 8092 Zürich, Switzerland

Abstract

Detection and quantification of fluorescently labeled molecules in sub-cellular compartments is a key step in analyzing many cell-biological processes. Pixel-wise colocalization analyses, however, are not always suitable because they do not provide object-specific information and are vulnerable to noise and background fluorescence. Here we present a versatile method for detecting, delineating, and quantifying sub-cellular structures in fluorescence microscopy images. The method is implemented as a freely available, user-friendly plugin for ImageJ and Fiji. It works on both 2D and 3D images, accounts for the microscope optics, computes cell masks, provides sub-pixel accuracy, and can be used on parallel computer clusters. The present method allows both colocalization and shape analyses. We compare it with other colocalization techniques and apply it to studying sub-cellular localization of RAB GTPases and to mitochondria morphology analysis.

1 Introduction

An increasing amount of biological and medical research relies on single-cell imaging to obtain information about the phenotypic response of cells to a variety of chemical, mechanical, and genetic perturbations. While it is possible to distinguish obvious phenotypes by eye, computational analyses allow one to process large datasets, to obtain quantitative and less biased results, and to detect more subtle changes in phenotype by statistical analysis. Finally, the ability to observe and quantify multiple fluorescent markers in the same cell under various conditions and over time opens doors for spatiotemporal modeling of biological processes [1].

Quantifying the colocalization of objects between different channels is an important tool, since the fluorescent probes are usually related to cellular markers. Colocalization can be quantified using various methods that are either pixel-based or object-based [2]. Pixel-based methods compute an overlap measure between the pixel intensities of the different color channels. This includes Pearson’s correlation coefficient [3], the overlap and Manders’ overlap coefficients [4], intensity correlation [5], cross correlation [6], and techniques correcting for unspecific (random) colocalization [7]. Object-based methods first detect and delineate the objects represented in the image and then quantify their overlap [3, 8] or nearest-neighbor distances [9]. This also allows correcting for the cellular context and for unspecific colocalization [8]. Methods based on intensity correlation are not suitable when fluorophores are not in ratiometric numbers [10], while methods looking for co-occurrence are very sensitive to noise and back-
ground levels. A more robust method indicating the fraction of colocalized molecules based on cross correlation and autocorrelation has been proposed, but is not automatic [11].

Object-based analyses provide access to additional features of biological relevance, such as the spatial distribution of objects within the cell, and the shapes and sizes of objects. This allows inferring interactions between objects and testing statistical hypotheses about their distribution (e.g., random vs. non-random) [9]. Object-based methods, however, require that the objects in the image be first detected and delineated, a non-trivial task that may involve image segmentation. Object detection and image segmentation are still frequently done by hand or using ad-hoc heuristics such as thresholding or hand-crafted pipelines of filters.

Recent progress in computer vision, however, has provided well-founded theories that can give justification to the methodology [12]. Here we make use of a novel segmentation method that directly connects the image-segmentation task with biological reality through prior knowledge about the imaged objects, the image-formation process, and the noise present in the image. This allows adapting the same method to a wide spectrum of images by adjusting the prior knowledge. Also, our method provides theoretical performance and robustness guarantees, is independent of manual initialization, and directly corrects for microscope blur and noise, yielding optimally deconvolved segmentations [13].

The latter is achieved by accounting for the microscope’s point-spread function (PSF), improving the capacity to segment objects with sizes close to the resolution limit. Our method provides sub-pixel accurate segmentations in both 2D and 3D. Sub-pixel segmentation was so far only available in two dimensions [14, 15] and has been shown useful, e.g., for studying the live morphology of endosomes [16]. We further extend the method by cell masks that allow restricting the analysis to a sub-population of cells in an image, e.g., to transfected cells.

We evaluate the present approach against other segmentation techniques and colocalization analysis methods. Finally, we demonstrate application results on the colocalization of endosomes with sub-cellular markers and the morphological analysis of mitochondria.

2 Results

Image segmentation

Image segmentation is a well-researched topic in computer vision, and many technological advances have successfully been transferred to bio-image analysis [12]. A wealth of user-friendly software tools is available for analyzing and quantifying fluorescence microscopy images [17]. They are either based on applying various filters to the pixels of the image (e.g., Cell Profiler [18]), on using machine-learning techniques to classify pixels as belonging to an object or to the background (e.g., Ilastik [19]), or on including models of the imaged objects and the image-formation process (e.g., Region Competition [15]). Filter-based approaches require the user to construct an appropriate pipeline of filters and to select all filter parameters. Machine-learning approaches require the user to manually label or segment a part of the data for the algorithm to learn the task. Model-based approaches require the user to design or choose a model that appropriately describes the type of image to be segmented.

Model-based methods aim at finding the segmentation that best explains the image. In other words, they compute the segmentation that has the highest probability of resulting in the actually observed image when imaged with the specific microscope used. We choose this framework because it is generic to a wide range of image types and segmentation tasks, and because it provide direct access to probabilistic quantitates that can be used in downstream analyses. We base our work on a recent extension of a family of image-segmentation models that allows including a variety of denoising and deconvolution tasks [13]. Moreover, the present approach is independent of initialization and robustly finds the optimal solution, because the underlying optimization problem is convex and hence has only globally optimal solutions; see Ref. [13] for theoretical and algorithmic details.
We consider an image model where the intensity distribution within each object is homogeneous and where the image consists of two regions: a brighter foreground and a darker background. The noise in the image can be either Gaussian or Poisson. Three quantities are hence to be estimated for each object in a given image: the segmentation, which is a mask assigning to each pixel a score of belonging to the object or to the background, the fluorescence intensity in the local background around the object, and the fluorescence intensity in the interior of the object. The method provides optimal segmentations using prior knowledge about the microscope optics (see Online Methods for details). This strategy hence combines image denoising, deconvolution, and segmentation [13]. It is not necessary to separately denoise or deconvolve the images beforehand. The results provided by the present joint deconvolution-segmentation procedure are of higher quality and are more robust against noise than those obtained by standard deconvolution followed by segmentation [13]. This is because the two tasks naturally regularize each other when considered jointly. This helps cope with very noisy data and with small objects close to the diffraction limit.

We account for spatial variations in the background intensity and for different objects having different foreground intensities by iteratively applying the procedure to local windows in the image around each object. Hence, each object may have a different estimated intensity, and the background around each object may locally be different (see Fig. 1).

We illustrate the present segmentation procedure using an example image of cherry-tagged RAB5 endosomes (Fig. 1a,b). First, object detection and separation is performed (steps 1–4), followed by estimating the local background and object intensities and computing the optimal segmentation (steps 5–7). The individual steps in detail are:

1. **Background subtraction** (Fig. 1c) is performed first, as the segmentation model assumes locally constant intensities. Background variations are non-specific signals that are not accounted for by this model. We subtract the image background using the rolling-ball algorithm [20].

2. **Segmentation** is performed on the whole image (Fig. 1d). This assigns to each pixel a number between 0 and 1, representing a score for the pixel to belong to an object or to the background. For this initial, rough segmentation, we set the object intensity to the maximum intensity present in the image and estimate the background using $k$-means clustering.

3. **Thresholding** of the score mask reveals the initial objects (Fig. 1e). For this preliminary segmentation the threshold value is set to the user-defined minimum intensity of objects to be included in the analysis. All objects with intensities lower than this threshold are discarded. Connected regions are identified as individual objects [15].

4. **Decomposition** of the image into smaller parts is done in order to allow locally different background and foreground intensities for different objects (Fig. 1e). We decompose the image by (possibly overlapping) 2D or 3D boxes around each detected object from the previous step. We ensure that objects in different boxes do not influence each other by computing the Voronoi diagram of the binary mask obtained in step 3. The image region considered for each object is then the intersection of the box around it with the Voronoi cell containing it (Fig. 1e).

5. **Local background and object intensities** are estimated in each image region (Fig. 1e). This determines what will be considered background and foreground in the subsequent segmentation step, and provides local analysis of the image. Intensities are estimated by solving the optimization problem described in the Online Methods.

6. **Individual object segmentation** is obtained by running the algorithm separately for each image
region (Fig. 1f). This step can be done with a segmentation resolution that is higher than the resolution of the image, thus providing sub-pixel accuracy [16]. Since this oversampling is only done in local patches around the objects, rather than on the whole image, it has only moderate impact on the computational cost (cf. Table 1).

(7) The final segmentation (Fig. 1g) is obtained by thresholding the masks obtained in step 6 using the optimal threshold that minimizes the segmentation error, automatically determined as previously described [13].

Colocalization analysis

Object-based colocalization is computed after segmenting the objects, using the information about the shapes and intensities of all objects in both channels. This allows straightforward calculation of the degree of overlap between objects from the different channels. We consider three different colocalization measures (see Online Methods for details). The first one, $C_{\text{number}}$, simply counts the number of objects that overlap between the two channels. Objects are considered overlapping if at least 50% of their volumes coincide. The second measure, $C_{\text{size}}$, quantifies the fraction of the total volume occupied by objects that overlaps with objects from the other channel. The third measure, $C_{\text{signal}}$, quantifies colocalization in an intensity-dependent manner. It computes the sum of all pixel intensities in either channel one or channel two in all regions where objects overlap with objects from the respective other channel. These signal-based definitions have been described and used before [8], but here we use the estimated object intensities from the segmentation rather than the raw pixel values. This improves robustness against noise and optical blur from the microscope PSF, because the present segmentation method computes an optimally denoised and deconvolved solution. In all cases, we use one-way ANOVA [21] followed by a Tukey-Kramer [22] test in order to check the statistical significance of the observed colocalization (see Supplementary Figs. 4 and 5).

Cell masks

Often, analysis should be restricted to a single cell or a subset of the cell present in an image. Examples include working with transfected cells or with mixed cells populations. Fig. 2 shows HEK293 cells that were transfected with cherry-tagged RAB5 expression constructs and then stained for the early-endosome marker EEA1. Only one cell in the image was transfected and should be considered in the analysis. Our software provides the option to compute cell masks by thresholding the original image and filling the holes of the resulting binary mask. Analysis is then restricted to the intersection of the image and the cell mask in order to ensure that only positive cells are considered in both channels. Fig. 2c shows the mask for the RAB5-positive cell, and Fig. 2d shows the overlay of the mask with the segmentation results from both channels.

Validation using sub-cellular localization of RAB GTPases

We analyzed the sub-cellular localization of four RAB GTPases (RAB5, 4, 11, and 7) with known sub-cellular distribution and function in order to validate our software. RAB GTPases are good markers for intracellular trafficking routes, since they are sequentially recruited to vesicles [23]. Internalized receptors first appear in early endosome, which are positive for RAB5. To become a recycling endosome, early endosome first acquire RAB4 and loose RAB5, and then they replace RAB4 by RAB11 [24]. In order to mature to late endosomes and finally lysosomes, they replace RAB5 by RAB4 followed by RAB7 [25,26]. We transfected fluorescently tagged RAB GTPases in HEK293 cells and determined their sub-cellular colocalization with well-known markers for the following compartments: EEA1 as a marker for early
endosomes, LBPA for late endosomes, LAMP-2 for lysosomes, and PDI for the endoplasmic reticulum (Fig. 3).

We observed that most EEA1-positive vesicles were also RAB5-positive. As expected, there was less colocalization with RAB4 vesicles and even less with RAB11 and RAB7 (Fig. 3a), suggesting a gradual loss of EEA1 during endosome maturation. LBPA is a phospholipid that is important for forming multi-vesicular bodies in late endosomes [27]. However, multi-vesicular bodies were also found in RAB5-positive compartments [27]. This is in line with our observation that colocalization of LBPA increased from RAB5 to RAB4 to RAB7. Recycling endosomes (RAB11) showed little colocalization with LBPA (Fig. 3b). LAMP-2 is used as a marker for lysosomes that are formed from late endosomes in an irreversible process. As expected, we found the highest colocalization of LAMP-2 with RAB7 (Fig. 3c). As a negative control, we used PDI, a marker for the endoplasmic reticulum. None of the tested RAB GTPases has previously been shown to localize to this compartment, and we confirmed this result (Fig. 3d). In summary, our analysis method confirmed previously described localization data of RAB GTPases. In addition, it allows one to quantitatively monitor the maturation of endosomes, either by loss of colocalization with EEA1 or gain of colocalization with LBPA.

Infection of insect cells with baculovirus

We infected Sf21 cells with different amounts of YFP-expressing baculovirus in order to obtain a dose-response curve. At low titers, cells are infected by single viruses. The algorithm should therefore find a linear correlation. We indeed observed a linear correlation between the amount of virus and the infection rate (Fig. 3d).

Mitochondria morphology analysis

Mitochondria are elongated organelles in eukaryotic cells. Their length is controlled by fusion and fission factors. GDAP1 is a fission factor that leads to fragmented mitochondria [28]. In cells over-expressing GDAP1, we observed reduced mitochondria lengths and increased numbers of mitochondria (Fig. 3e). The total volume of mitochondria remained constant within the statistical noise (Fig. 3d). This is in line with previous manual observations [28]. Example images from both conditions are shown in Fig. 3f.

Comparison with other methods

We benchmark our segmentation method on a test set of synthetic images previously used to compare eleven segmentation methods [29]. Noisy synthetic images of cells with a nucleus and circular vesicles are considered with out-of-focus blur and uneven background illumination. The ground truth of all vesicles outlines is available and used to evaluate the F-score, a combination of accuracy and recall, of the segmentation methods. With an F-score of 0.62, the present method performs as well as the Multiscale Wavelet method, which was the previous winner on this benchmark. This is remarkable, since the Multiscale Wavelet method is a spot detector, particularly tuned to detect round objects like those present in the benchmark images, whereas our method makes no assumptions about the shapes of the objects.

We benchmark the present object-based approach to colocalization analysis by comparing it with results from a pixel-based Pearson correlation analysis. Fig. 4a provides a comparison of the results obtained for the colocalization of LAMP-2 with the different RAB GTPases. As LAMP-2 is a marker for lysosomes, colocalization is mainly expected with RAB7. While both approaches capture high colocalization with RAB7, the pixel-based method results in more spurious colocalization with RAB4, 5, and 11. Introducing a first segmentation component, the cell masks, improves the picture. However, the fully object-based approach yields the best result. The differences are statistically significant. To explain these differences we take a detailed look at the results obtained in the single image shown in Fig. 4c. Two
phenomena affect the Pearson-correlation results: First, many objects in the LAMP-2 channel overlap
with objects in the RAB7 channel, but not vice versa (compare the two $C_{\text{size}}$ in Fig. 4b). In such asym-metric situations, pixel-based colocalization analysis is not appropriate, as it lumps both sides together.
Second, pixel-based analysis is more sensitive to imaging noise than object-based analysis. This is shown
in Fig. 4b, where the Pearson correlation score tends to zero with increasing noise level. This is consistent
with previous reports [3]. As pixel-based correlation methods directly consider pixel values, they have no
way to differentiate noise from signal. 

We benchmark the accuracy of sub-pixel segmentation on a synthetic image obtained by blurring a
previously determined segmentation of RAB5 endosomes with the PSF of the microscope and adding
(modulatory) Poisson noise of various magnitudes. We assess the ability of the sub-pixel segmentation
method to recover the ground-truth objects by computing the $F$-score of the obtained segmentation with
respect to ground truth. Sub-pixel segmentation improves the segmentation accuracy of the outlines of
the objects for signal-to-noise ratios (SNR; defined for Poisson noise according to Eq. (19) in Ref. [30])
avove four (Fig. 4c). Figure 4e shows the sub-pixel segmentation results in comparison with normal,
pixel-level segmentation on a synthetic image with SNR=12.

3 Discussion

We presented a method to segment sub-cellular objects in fluorescence microscopy images and perform
colocalization and morphological analysis. The algorithm is implemented as a free open-source plugin for
ImageJ and Fiji and is available on all major platforms, including Linux, MacOS, and Windows. It allows
automatic analysis of large numbers of images and only has a small number of intuitive parameters. By
considering the point-spread function of the microscope and a noise model adapted to the acquisition
technique, the present method provides more justification to the analysis than ad-hoc filter-based meth-
ods. We have shown that the presented method reproduces previously known colocalization results with
high accuracy, is robust against noise, and segments sub-cellular objects at least as well as the previously
best method tested on the same benchmark.

Thanks to the use of ImageJ and open-source software, the method can be customized in a number of
ways. Using ImageJ macros, the procedure can be integrated into larger workflows that also include other
ImageJ functions and plugins, for instance to compute additional object features after segmentation or to
use different methods of background subtraction. Possible extensions of the model-based segmentation
approach include shape priors to segment objects of priorly known shape, and extensions to objects with
inhomogeneous intensities. Results from the present method can also be used to perform more refined
analyses of the spatial patterns of sub-cellular object distributions [16].
Online Methods

Image segmentation

We use the two-region piecewise constant image model with a convolution kernel as previously described \[13\]. The optimal segmentation is the one that maximizes the posterior probability of having generated the actually observed image (in the Bayesian sense). The algorithm determines the segmentation mask \( M \), which assigns to each pixel a value between 0 and 1, related to the probability that the pixel belongs to an object. In addition, the algorithm estimates for each object the fluorescence intensity within, \( c_{\text{int}} \), and the local background intensity around it, \( c_{\text{ext}} \). We use the alternating split-Bregman algorithm described in Ref. \[13\] to solve the following optimization problem:

\[
\min_{(M, c_{\text{ext}}, c_{\text{int}})} \left[ D(u_0, K \ast u(M, c_{\text{ext}}, c_{\text{int}})) + \lambda L(M) \right]
\]  

This strategy combines image denoising, deconvolution, and segmentation (Paul 2013). The data-fitting term \( D(u_0, K \ast u(M, c_{\text{ext}}, c_{\text{int}})) \) quantifies the discrepancy between the observed image \( u_0 \) and the image one would expect to observe if \((M, c_{\text{ext}}, c_{\text{int}})\) was really the correct segmentation. The latter is computed by convolution with the microscope’s PSF \( K \) in order to account for the diffraction blur introduced by the optics. \( D(\cdot) \) denotes the discrepancy measure; in the present case it is a Bregman divergence that depends on the noise model of the image \[13\]. For optimal denoising, different divergences are used for Gaussian and Poisson noise models as described \[13\]. The regularization term adds a penalty on the total length of the contour \( L(M) \) to encode the biological prior that sub-cellular objects tend to have smooth boundaries. This helps cope with very noisy data. The user-defined regularization parameter \( \lambda \) tunes the trade-off between fitting the image data and producing smooth results, i.e., not overfitting noise in the image.

Implementation

The present method is implemented as open source and in pure Java as a plugin for ImageJ and Fiji. It is thus compatible with the wide range of image file formats supported by ImageJ and runs on most computer platforms. The plugin reads images files containing one or several color channels. We provide a .lif extractor macro for ImageJ in order to convert data from .lif files. In addition, we provide a script for the R statistical analysis software to produce plots and perform statistical analyses (see Supplementary Figs. 4–7 for example output). The plugin provides a graphical user interface and supports ImageJ macros. It has also been tested with the headless ImageJ mode running in parallel on computer clusters. In order to allow fast segmentation and parameter testing on desktop computers, the software is also parallelized within single images. Multiple threads are created according to the number of available processor cores, and computation is parallelized across image sub-domains for steps 1–4 of the method and across the extracted blocks for steps 5–7. Supplementary Table 1 provides execution times and memory usage to analyze images of various sizes on a dual core 2.3 GHz Intel Core i5 processor with 8 GB RAM.

Usage hints and workflow

Image acquisition

1. Care has to be taken to select spectrally separated fluorophores in order to avoid cross-talk between the color channels, which would lead to spurious colocalization. Before computing colocalization between different color channels, it is important to correct for chromatic aberration as well as possible, as this biases the results \[9\].

File and parameter input
2. The plugin works with any image format supported by ImageJ, but all stacks and channels of a single image have to be in the same file. As part of the present package, we provide the LifExtractor ImageJ macro to automatically extract dual-channel TIFF image files from .lif files. It uses BioFormats [31] and the channel rearrangement tools of ImageJ. The plugin works both on single files and on folders containing multiple files for batch processing.

3. In order to correct for diffraction blur, the software needs information about the PSF of the microscope. We model the PSF as a Gaussian [32]. The parameters of the PSF can either be estimated from the image and the objective lens parameters by clicking the “Estimate PSF” button (see Supplementary Fig. 1), or they can be input by the user in the lateral \( x - y \) and axial \( z \) directions separately. Airy units only have to be entered for confocal microscopes.

4. Reduce background fluorescence using the rolling ball algorithm by selecting “Remove Background” and entering the window edge-length in units of pixels. This length should be large enough so that a square with that edge length cannot fit inside the objects to be detected, but is smaller than the length scale of background variations.

5. Set a regularization weight on the total objects perimeter. Use higher values to avoid segmenting noise-induced small intensity peaks (see Supplementary Fig. 2). Typical values are between 0.05 and 0.25.

6. Set the threshold for the minimum object intensity. Intensity values are normalized between 0 for the smallest value occurring in the image and 1 for the largest value.

7. Select “sub-pixel segmentation” to compute segmentations with sub-pixel resolution. The resolution of the segmentation is increased by an over-sampling factor of 8 for 2D images and 4 for 3D images. Activating this setting requires more computation time and memory (see Supplementary Table 1).

Cell masks

8. A cell mask allows restricting the analysis to a certain region of an image, e.g., a transfected cell. Select “cell mask channel 1” and/or “cell mask channel 2” to compute cell masks based on the respective channel. Cell masks are computed by thresholding the respective channel and filling holes in the obtained binary mask. Set a value between 0 and 1 for the threshold in “threshold channel 1” and/or “threshold channel 2”. Click “preview cell masks” for a live preview of the resulting masks.

Software output

9. The software offers four options to visualize the segmentation result (see Supplementary Fig. 3): With “colorized objects”, each object is visualized in a different random color. “Object labeling” assigns to all pixels of an object a value that is identical to the label (index) of that object in the result file. “Object intensities” displays objects in their estimated fluorescence intensities. “Object outlines” shows an overlay of the original image with red object outlines. For two-channel images an additional visualization is provided attributing a distinct color to each channel in order to reveal colocalization. “Intermediate steps” allows visualizing the result both after background subtraction (step 1, Fig. 1c) and segmentation (step 2, Fig. 1d). Checking “save results” stores the resulting visualizations in .zip files, which can be opened by ImageJ without prior decompression.

10. Click “OK” to start the analysis. The number of objects found is shown in the log window of ImageJ. The mean features of all objects from each image are stored in a .csv file ending in “Images_data.csv”. Individual, per-object features for each channel are stored in additional .csv files.
For each object (indexed by its segmentation label), the software stores the size (area or volume), surface (or perimeter), length (maximum extension in the most extended direction), fluorescence intensity, and the position of the center of mass. For two-channel images, we additionally store the amount of overlap with objects from the other channel, and the sizes and intensities of all colocalizing objects from the other channel. These features can then be used for object classification or colocalization analysis.

11. Optional: Use the R statistical software to plot the results into a PDF file and for analyzing their statistical significance. We provide a R script that generates colocalization and object-feature graphs, performs one-way ANOVA followed by Tukey test analysis to evaluate the statistical significance, and allowing filtering of objects based on intensity and size thresholds. The output from this script is shown in Supplementary Figs. 4–7 for the RAB study presented here.

Parameter values
Since PSF parameters are determined by the microscope characteristics, only three parameters are left for the user to set: (1) the window size for background subtraction, the regularization parameter, and the minimum intensity threshold. Window size for background removal should be set larger than the diameter of the objects of interest, but smaller than the diameter of structures that should be removed (for instance the cell nucleus). There are thus only two parameters, regularization and minimum intensity, that have to be tuned to provide the desired segmentation. We provide in the above workflow ranges of values that work well in most cases. Nevertheless, some manual tuning is necessary for a given set of images. The effects of varying the minimum intensities and the regularization parameter are illustrated in Supplementary Fig. 2.

Colocalization measures
We provide here the precise mathematical formulas for the colocalization measures used in the present analysis. The first measures are based on counting the number of objects in one channel that overlap with objects from the other channel:

\[
C_{\text{number}}(O_{1O_{2}+}/O_{1}) = \frac{|\{o \in O_{1} : O_{o} > 0.5\}|}{|O_{1}|}
\]

\[
C_{\text{number}}(O_{2O_{1}+}/O_{2}) = \frac{|\{o \in O_{2} : O_{o} > 0.5\}|}{|O_{2}|}
\]

Here, \(O_{1}\) is the set of all objects detected in channel 1 and \(O_{2}\) the set of objects detected in channel 2. \(\cdot\) denotes the total number of objects in the given set. \(O_{o}\) is the fraction of object \(o\) that overlaps with any other object from the other channel. The notation \(O_{1O_{2}+}\) means “objects in channel one that are positive for objects in channel two”.

The size-based colocalization coefficients are defined as:

\[
C_{\text{size}}(O_{1O_{2}+}/O_{1}) = \frac{\sum_{o \in O_{1}} S_{o} O_{o}}{\sum_{o \in O_{1}} S_{o}}
\]

\[
C_{\text{size}}(O_{2O_{1}+}/O_{2}) = \frac{\sum_{o \in O_{2}} S_{o} O_{o}}{\sum_{o \in O_{2}} S_{o}}
\]

where \(S_{o}\) is the size of object \(o\) given by the number of pixels belonging to it. Hence, \(\sum_{o \in O_{1}} S_{o} O_{o}\) is the total size of the colocalizing regions and \(\sum_{o \in O_{1}} S_{o}\) is the total number of pixels covered by any object in that channel.

The intensity-based colocalization coefficients are:
\[ C_{\text{signal}}(O_{1O_2+}/O_1) = \frac{\sum_{o \in O_1} I_o S_o O_o}{\sum_{o \in O_1} I_o S_o} \]
\[ C_{\text{signal}}(O_{2O_1+}/O_2) = \frac{\sum_{o \in O_2} I_o S_o O_o}{\sum_{o \in O_2} I_o S_o} \]

where \( I_o \) is the estimated intensity of object \( o \). Hence, \( \sum_{o \in O_1} I_o S_o O_o \) is the total signal in the colocalizing region in channel 1, while \( \sum_{o \in O_1} I_o S_o \) is the total signal present in channel 1 altogether. These intensity-based definitions are the same than those used in (Ramirez, 2010), except that we use the object intensities estimated by the segmentation procedure, rather than raw pixel intensities inside the object. This leads to better noise robustness.

**Cell culture and microscopy**

HEK293 cells were maintained in 10% FBS/DMEM/Penicillin/Streptomycin. For transfection, cells were plated at a density of approximately 40% on glass coverslips coated with poly-L-lysine (Sigma-Aldrich P4707) and then transfected with the indicated mCherry-tagged RAB GTPases [33] with FugeneHD (Promega) according to the manufacturer’s recommendations. Cells were fixed by transferring the coverslips into 4% formaldehyde/PBS for 10 minutes and then washed three times with PBS. Cells were permeabilized with 1% NP-40 in PBS for 10 minutes and then incubated with the following primary antibodies for two hours: rabbit anti-EEA1 (Cell Signaling, #3288, 1:500), mouse anti-LBPA (kindly provided by Dr. Peter Greimel, 1:20), mouse anti-LAMP-2 (H4B4, Developmental Studies Hybridoma Bank, Iowa, 1:20), and mouse anti-PDI (1D3, 1:60). Secondary antibodies were labeled with DyLight488 or Cy5, (Jackson ImmunoResearch, Suffolk, UK). All antibodies were diluted with PBS. Samples were mounted in Gelvatol and analyzed on a Leica SP5 confocal microscope. We combined mCherry (excitation with 543nm, emission collected from 585nm to 620nm) with DyLight488 (488, 500–535) and Cy5 (633, 640–730). The sequential scanning mode was chosen and the number of overexposed pixels was kept at a minimum. At least twenty images with 10 to 25 z-sections and a resolution of 512×512 pixels were taken (voxel size 120×120×420 nm, 16 bit resolution). Supplementary Figs. 8 to 11 give examples of the images obtained, showing maximum projections. The levels and gamma settings of images used for illustration were adjusted in Adobe Photoshop. All operations were applied to the entire image.

PAE cells were co-transfected with a MitodsRed (Clontech) and a GDAP1 expression plasmid [28]. Cells were fixed with 4% formaldehyde/PBS. Pictures were taken as described above. Sf21 cells were cultured in serum-free SF-4 Baculo Express ICM medium (Animedia). Cells were plated on glass coverslips and infected with different amounts of a baculovirus that expresses YFP (Multibac) [34]. Cells were fixed after 36 hours and the nuclei were stained with Hoechst 33258 (1 µg/ml in PBS for 15 minutes). Samples were mounted with gelvatol and analyzed on an Olympus IX-81 wide-field microscope. Image size was 2219×1674 µm (1376×1038 pixels, 8 bit). Each image contained approximately 3000 cells. Fig. 3d shows a part of a typical image.
Supplementary material

Computation time and memory requirement

The software runs on desktop computers and computer clusters. Supplementary Table 1 provides typical time and memory requirements for 2D and 3D images with and without sub-pixel refinement, measured on a dual-core 2.3 GHz Intel Core i5 with 8 GB RAM. The computational cost only depends on image size for steps 1 to 3 of our method. All further steps do not depend on image size, but rather on the sizes and the number of objects detected in the image. The data in Supplementary Table 1 was obtained for images containing approximately 100 objects each. For more objects, the times scale proportionally. The overhead incurred by sub-pixel oversampling mainly depends on the number and the sizes of the objects and ranges from 10% to 40% for the images used here. All pixel intensity values are normalized between 0 and 1 and stored as a 64-bit Java double variables. This renders the computational performance of the software independent of the bit depth of the original images.

Acknowledgments

This work was supported by SystemsX.ch, the Swiss initiative in systems biology under grant IPP-2011-113, evaluated by the Swiss National Science Foundation (SNF). The research leading to these results has received funding from the European Community’s Seventh Framework Programme (FP7/2007-2013) under grant agreement #290605 (PSI-FELLOW/COFUND).

References


Figure 1. Working principle of the software illustrated on endosome segmentation. (a) Original image; scale bar is 10 µm. (b) Closeup of the region highlighted by the white square in (a); scale bar is 1 µm. (c) The same closeup after background subtraction. (d) Object model image found by the present method. Intuitively, this is a denoised and deconvolved version of (c) taking into account the microscope’s point-spread function. (e) Objects (white) obtained by thresholding image (d). The image is decomposed into regions (blue boxes) and objects are separated by Voronoi decomposition (red lines). (f) Refined sub-pixel object model images obtained by applying the present method inside each region with individual estimates for the local object and background intensities. (g) Final segmentation with the estimated object intensities displayed in shades of green.

Figure 2. Segmentation and colocalization of EEA1 and RAB5. All images show maximum-intensity projections along the optical axis. (a) Raw image in the RAB5 channel. (b) Raw image in the EEA1 channel. (c) Mask of a transfected cell determined using the RAB5 channel. (d) Segmentation results of the vesicles from both channels overlaid with the cell mask. EEA1 vesicles are shown in green, RAB5 vesicles in red. The colocalization coefficient computed for this image is: $C_{\text{number}}(\text{EEA1}^{\text{RAB5+}}/\text{EEA1}) = 0.36$ ($C_{\text{number}}(\text{EEA1}^{\text{RAB5+}}/\text{EEA1}) = 0.02$ when not using a cell mask); Scale bar is 10 µm. (e) Closeup view of the raw image data in the area highlighted by the black square in (d); scale bar is 2 µm. (f) Object segmentation and overlaps in this area.
Figure 3. Validation of the method. (a) Colocalization results for sub-cellular markers with different RAB GTPases. Means and standard errors of the mean over 20 images per condition of the fraction of sub-cellular marker colocalizing with the RAB channel, as determined by the present method. (b) Example image with closeup of EEA1 (green) and RAB5 (red) (c) Example image with closeup of EEA1 (green) and RAB7 (red). Examples images for all conditions as well as statistical analyses are provided in the Supplementary Material. (d) SF21 cells were infected with YFP-expressing baculo-virus. Cells stained with Hoechst33258. A linear correlation between the amount of virus and the fraction of infected cells is observed. (e) Morphological analysis of mitochondria without and with over-expression of the fission factor GDAP1. Means and standard errors of the mean over 8 images per condition. (f) Example images from both conditions; scale bar is 10 µm.
Figure 4. Benchmarks of the present method. (a) Pixel-based Pearson correlation (range −1 to +1) and object-based colocalization (range 0 to 100%) results for LAMP-2 with RABx. For object-based analyses, images were segmented using the present software. LAMP-2 is known to colocalize mainly with RAB7. Means and standard errors of the mean are over 20 images per condition. (b) Pearson correlation and object-based colocalization coefficients for the image shown in (c) corrupted with increasing amounts of Gaussian noise. (c) Example image with both channels shown; scale bar is 10 µm. (d) F-score of segmentation accuracy for 4-fold sub-pixel and normal pixel-level segmentation. We show the mean values over five random images for each signal-to-noise ratio. (e) Example of a benchmark image with signal-to-noise ratio 12. The closeup views below show the ground truth outlines in green and those reconstructed by the present method in red. The middle panel uses normal pixel-level segmentation, whereas the right panel uses 4-fold sub-pixel oversampling; scale bar is 1 µm for top panel and 0.5 µm for closeup panels.
Supplementary Table 1. Computer time and memory requirements on a dual core 2.3 GHz Intel Core i5 with 8 GB RAM.

<table>
<thead>
<tr>
<th>$x \times y \times z$ dimensions &amp; sub-pixel oversampling</th>
<th>computer time (s)</th>
<th>memory (MB)</th>
</tr>
</thead>
<tbody>
<tr>
<td>512×512</td>
<td>17.2</td>
<td>275</td>
</tr>
<tr>
<td>512×512, 4× pixel oversampling</td>
<td>23.7</td>
<td>407</td>
</tr>
<tr>
<td>1024×1024</td>
<td>55.6</td>
<td>478</td>
</tr>
<tr>
<td>512×512×15</td>
<td>180.7</td>
<td>1116</td>
</tr>
<tr>
<td>512×512×15, 4× pixel oversampling</td>
<td>201.3</td>
<td>1536</td>
</tr>
</tbody>
</table>

Supplementary Figure 1. Screenshot of the graphical user interface of the present software. The parameter input masks consist of the following areas: (A) data input; (B) segmentation parameters; (C) Parameters for computing the cell masks; (D) Visualization options; (E) Noise and intensity models; (F) Sub-mask for estimating the microscope’s point-spread function from the image using a Gaussian model.
Supplementary Figure 2. Illustration of how the method parameters affect segmentation results for an endosome example. (A) Segmentation with the minimum intensity threshold set to 0.100 and the regularization weight set to 0.250. (B) Segmentation with the minimum intensity threshold decreased to 0.050, resulting in a larger number of dimmer objects detected. (C) Segmentation with a minimum intensity threshold of 0.050 and the regularization weight decreased to 0.075, resulting in a closer fit of the segmentation with the image data, but an increased sensitivity to noise (see, e.g., the small islands segmented, which may correspond to noise pixels).

Supplementary Figure 3. Illustration of the different ways the software provides to visualize the results. From left to right: original image, segmented outlines overlaid with the original image, detected objects represented in false colors, and detected objects represented with an intensity heat-map, i.e., the intensity of green corresponds to the estimated fluorescence intensity of the object.
Supplementary Figure 4. Example output from the provided R analysis script, page 1/4. The script produces plots showing the object-based colocalization results (here of LAMP-2 with RABx) using number-, size-, and signal-based colocalization coefficients. In the bottom row, the pixel-based Pearson correlations with and without cell masks are shown. The bar graphs display the means and standard errors of the mean over all (here, 20) images. P values are computed using one-way ANOVA test.
Supplementary Figure 5. Example output from the provided R analysis script, page 2/4. 95% confidence intervals of pairwise mean colocalization and Pearson correlation differences. Confidence intervals and P values are computed using a Tukey test.
Supplementary Figure 6. Example output from the provided R analysis script, page 3/4. Mean numbers and sizes (areas in 2D, volumes in 3D, in units of pixels/voxels) of objects segmented in both channels (here, LAMP-2 and RABx vesicles). The total size is the sum of the sizes of all individual objects in condition. The total size ratio is the ratio between the total area occupied by objects in channel 1 and the total area occupied by objects in channel 2.
Supplementary Figure 7. Example output from the provided R analysis script, page 4/4. Mean lengths (in units of pixels) and intensities (in the scaled image) of objects segmented in both channels (here, LAMP-2 and RABx vesicles). Moreover, this last page of the analysis output shows the colocalization coefficients when excluding objects below different size and intensity thresholds.
Supplementary Figure 8. Example images showing the colocalization of LPBA with different RAB GTPases.
Supplementary Figure 9. Example images showing the colocalization of EEA1 with different RAB GTPases.
Supplementary Figure 10. Example images showing the colocalization of LAMP-2 with different RAB GTPases.
Supplementary Figure 11. Example images showing the colocalization of PDI with different RAB GTPases.