

RESEARCH ARTICLE



SUPPOSE 3Dge: A Method for Super-Resolved Detection of Surfaces in Volumetric Fluorescence Microscopy

Guillermo D. Brinatti Vazquez^{1,2,*} , Axel M. Lacapmesure^{1,2}, Sandra Martínez^{2,3} and Oscar E. Martínez^{1,2}

¹Laboratorio de Fotónica, Instituto de Ingeniería Biomédica, Universidad de Buenos Aires, Argentina

²Consejo Nacional de Investigaciones Científicas y Técnicas, Argentina

³Departamento de Matemática, Facultad de Ciencias Exactas y Naturales, Instituto de Investigaciones Matemáticas Luis A. Santaló, Universidad de Buenos Aires, Argentina

Abstract: We present SUPPOSE 3Dge, a method for the detection of surfaces in volumetric fluorescence microscopy images with resolution better than the diffraction limit. The method works by approximating an arbitrary object surface as a polyhedron formed by small triangular faces, which are thought as virtual point like sources of signal gradient. Then, the super-resolution arises from solving a least square problem only on the positions of the sources, by comparing the measured image gradient with a synthetic one built from the approximated surface and the known point spread function of the instrument. We present the formulation of the method and its characterization using simulated measurements of two different types. We show that the method is able to identify with high precision and accuracy the sizes and shapes of objects smaller than the instrumental resolution. Using different quantifiers that measure the distance between the solution and the target object, we report reconstructions with an accuracy 100 times better than the axial resolution and 40 times the lateral resolution of the instrument.

Keywords: super-resolution, fluorescence, microscopy, image-processing, edge detection

1. Introduction

Determining the complex and dynamic interplay that exists among subcellular compartments, organelles and plasma membrane has become a challenging objective in biology. Shape and sizes have been successfully measured by 3D edge detection methods using electron tomography [1, 2], but this approach is not compatible with living samples, where fluorescence is the preferred source of contrast [3]. Fluorescence image segmentation is another application that relies on 2D and 3D edge detection algorithms [4, 5], but its accuracy is always limited by the diffraction limit. Thus, improving the performance of edge detection in ultra-high resolution fluorescent microscopy [6–8] would allow to achieve competitive results in living systems.

The problem of determining shapes and edges of 3D objects from some imaging technique is transverse to many disciplines, including metrology [9], medical diagnosis [10, 11], geophysics [12], and augmented reality and robotics [13, 14] among many others. Several edge detection methods have been implemented

[15, 16] being the Canny algorithm [17] among the most successful. Although this method was originally proposed for 2D edge detection, its 3D generalization is well known [18, 19] and several implementations are readily available in popular image processing software such as MATLAB and ImageJ. Also, it is very well established as a standard in 3D edge detection, as one can see in recent published articles on the matter [20, 21]. The method is based on smoothing the image and then calculating the gradient. Improved versions and variants have been proposed [13, 16]. All these methods depend on localizing a border that must be located far enough from the other side of the object so that the instrument response function (point spread function (PSF) in the case of a microscope) located on one side does not overlap with the other side. Hence, they cannot be used to super-resolve shapes and sizes of objects smaller than the PSF.

Fluorescence microscopy is characterized by the fact that each point source (molecule) emits incoherently so that the image is built by adding the intensity from each source. Within this paradigm, many deconvolution methods have been developed to recover the 3D structure from stacks of 2D images focused to a different plane. Such methods provide a good 3D reconstruction with background suppression, but the resolution is only improved marginally [22].

*Corresponding author: Guillermo D. Brinatti Vazquez, Laboratorio de Fotónica, Instituto de Ingeniería Biomédica, Universidad de Buenos Aires and Consejo Nacional de Investigaciones Científicas y Técnicas, Argentina. Email: gbrinatt@fi.uba.ar

A method has been recently introduced [23] based on approximating the object to be reconstructed by a superposition of virtual sources of equal intensities and is called SUPPOSE. It provides a resolution enhancement up to 5 times better than the instrument response, but it needs a certain degree of sparsity in the object to be reconstructed. Also, in Lacapmesure et al. [24] the problem of very sparse sources was analyzed and it was shown that SUPPOSE outperforms compressed sensing-based techniques for images with low signal-to-noise ratio (SNR).

In a 2D image edge detection, the object whose edge we want to know is typically not sparse but the very opposite: a dense shape that would require a large number of point sources needed to be filled. To overcome this issue, a variant of SUPPOSE was presented in Vazquez et al. [25] that instead works with the image gradient, since a sharp boundary can be described as a sparse collection of ordered segments that act as point sources of gradient. With this formulation, it was shown that the method can recover shapes of objects smaller than the PSF span with super-resolution, an achievement not possible with Canny or related edge detection methods nor with other deconvolution algorithms. Notice that we use the term super-resolution meaning to recover details which are below instrumental resolution as usual in the microscopy community, and should not be confused with the same term used in image processing to address a sub-pixel or sub-voxel reconstruction.

In this work, we present SUPPOSE 3Dge, a generalization to three spatial dimensions of the edge detection method SUPPOSE Edge [25], that allows for the identification of surfaces and the retrieval of shapes of objects smaller than the instrumental resolution in 3D microscopy images. Additionally, we show that the minimization problem arising from the SUPPOSE 3Dge method can be efficiently solved by the use of a stochastic gradient descent (SGD) method, allowing for a much faster convergence when compared to the original formulation of SUPPOSE that used genetic algorithms, as was also demonstrated for the case of very sparse samples [24].

2. Method

The goal of the method is to retrieve, from a single image stack, an object's surface with enhanced resolution. We start by formulating our imaging model. Let S be our measured image, which is produced by a certain distribution R of incoherent emitters in the sample (as expected for fluorescent molecules). The emitters are blurred by the imaging system PSF I and the measurement is affected by noise. Then, S and R are related by the expression,

$$S = R * I + \eta, \quad (1)$$

where $*$ denotes the convolution product and η represents a general, zero mean noise contribution. As our target is the boundary of our object, we will focus on the gradient of our image

$$\nabla S = \nabla R * I + \eta'. \quad (2)$$

As in the previous methods of the SUPPOSE family [23], the idea is to model our signal (in this case the image's gradient) as produced by a superposition of virtual point-like sources all with the same intensity contribution. After this, a synthetic version of the signal is produced using only the positions of the virtual sources and the PSF, which is then compared with the measurement. The super-resolution information is recovered from the positions of the virtual sources that are the best fit to it. These positions can take any real value and are not limited to the voxels of the image, allowing for sub-voxel resolution. As we are fitting the changes in the intensity of the

image, the resolution will be limited by the intensity fluctuations rather than the PSF width, similar to what occurs in single molecule localization microscopies [26–28]. In the original formulation of SUPPOSE [23], the approximation is applied directly to R , resulting in virtual sources of intensity. In the edge detection problem, our signal is not the image itself but its gradient, and then the SUPPOSE approximation should be adapted to account for the vectorial nature of ∇R . In 2D images a SUPPOSE variant named SUPPOSE Edge [25] was developed, addressing this problem by defining an ordered set of sources that can be thought as the nodes of a line segment discretization of our contour. Each node is then a source of gradient, whose direction is assigned by a local estimation for the vector normal to the contour by using the positions of the source's two first neighbors. Then, the fit proceeds as before, by computing a synthetic version of the image's gradient from the virtual sources and the PSF, and by finding the positions of the sources that minimize the norm between this synthetic gradient and the measured one. Finally, a regularization term must be added to avoid very sinuous edges. For more details of the method, see Equations (2) and (3) and Figure 2 in Vazquez et al. [25].

For 3D surface detection, we take this idea one step further. Here, we will model the target surface as a set of plane triangular elements stuck together forming a polyhedron, as represented in Figure 1(a). Computationally, this is described using two separate arrays, a first one containing the information on the position of all the vertices of the polyhedron, and a second one containing the order in which the vertices should be connected to construct the faces. Then, the point-like sources of gradient are chosen one at the center of each face, pointing at the direction normal to the triangular element's surface. This is easy to compute from the proposed representation, as the position of each source is the 3D average of the positions of the vertices of that face, and the normal vector is computed from the cross vector product between the directions given by two of the edges of the face (taking care that its direction is consistent with a previously chosen orientation of the surface). This is depicted in Figure 1(b), where the X at the center of each face represents the position of the virtual source and the blue vectors parallel to the edges of the face (used to calculate the normal vector of the face) are highlighted. From this representation, we can generate the synthetic gradient produced by the sources $\nabla \tilde{R}$,

$$\nabla R \simeq \nabla \tilde{R} = \alpha \sum_{t=1}^N \delta(x - a_t, y - b_t, z - c_t) \hat{n}_t, \quad (3)$$

and including the PSF information

$$\nabla \tilde{R} * I = \alpha \sum_{t=1}^N I(x - a_t, y - b_t, z - c_t) \hat{n}_t, \quad (4)$$

where α is an intensity to be determined, N is the number of virtual sources used to approximate the surface, (x, y, z) are the coordinates of the image, and (a_t, b_t, c_t) and \hat{n}_t are the position and the normal vector of the virtual source t . From that, we need to find the positions of the sources that minimize the squared error χ^2 between the synthetic and the real gradient, which results in the least squares minimization problem

$$\chi^2 = \|\nabla S - \nabla \tilde{R} * I\|^2 = \sum_{d=1}^3 \sum_{ijk} \left(\left((\nabla S)_{ijk} - \alpha \sum_{t=1}^N I_{ijk}^t \hat{n}_t \right) \cdot \hat{e}_d \right)^2, \quad (5)$$

where the first sum is over all spatial dimensions of canonical versors \hat{e}_d , the second sum symbol is a contraction of three separate sums over all the image's voxels, and I_{ijk}^t is short for $I(x_i - a_t, y_j - b_t, z_k - c_t)$.

Figure 1

Representation of the surface and its relation with SUPPOSE 3Dge virtual sources. (a) A portion of the surface, represented as a set of plane triangular elements. Black dots represent the vertices, blue lines the edges, and shaded green regions the faces of the polyhedron. (b) The SUPPOSE 3Dge virtual sources are located at the center of each face (X). The direction given by two of the edges of the face (blue arrows) is used to compute the normal vector corresponding to that face and virtual source. (c) Representation of one vertex and its surroundings up to third neighbors. We show an example for the notation used in the calculation of the regularization term (edge e in red and the normal vectors of adjacent faces). If the vertex p is moved, the pink faces change their size and orientation. Blue and green faces are not affected, but the first ones share an edge with the changing ones and are then also relevant for the calculation of the gradient

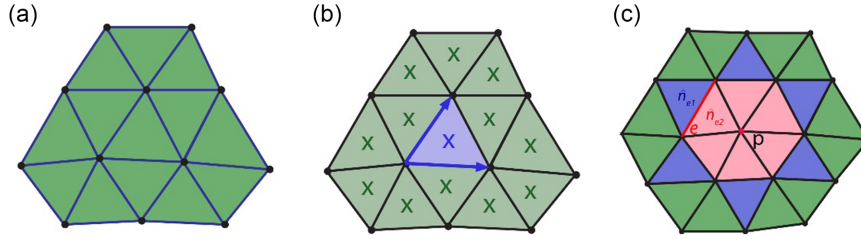
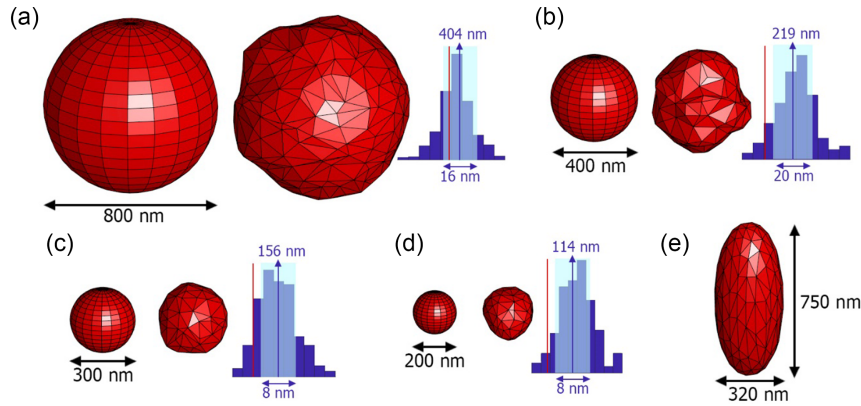


Figure 2

Determination of the sizes of objects smaller than the experimental resolution. For ease of comparison, all the images are at the same scale. (a) to (d), super-resolved surface detection of spheres of different diameters, ranging from 800 nm (similar to the axial resolution) to 200 nm (smaller than the lateral resolution). Each insert is composed of, left to right, the high-resolution object, the SUPPOSE 3Dge reconstruction, and the histogram of adjusted radii. The mean value of the distribution (blue arrow), together with the standard deviation (light blue shading) and the target radius (red line), is indicated. (e) Mid height surface level of the PSF, indicating lateral and axial resolution (FWHM)



As in SUPPOSE Edge, a regularization term Λ needs to be included to impose some smoothness on the surface that allows for proper convergence of the algorithm. This is achieved by looking at all the pairs of first neighbor sources and computing a measure that penalizes rapidly changing normal vectors. We can write this as

$$\Lambda = \lambda \sum_{e=1}^{N_e} (\hat{n}_{e1} \cdot \hat{n}_{e2} - 1)^2, \quad (6)$$

where the index e spans over the N_e edges of the polyhedron and \hat{n}_{e1} and \hat{n}_{e2} are the normal vectors corresponding to each one of the two faces that share the edge e (see Figure 1(c)). The parameter λ is a constant weight for the regularization term, and its importance (as well as a method for estimating its value) is discussed for the two-dimensional case in Vazquez et al. [25]. The fitness function f , the quantity to minimize, is the sum of the contributions from Equations (5) and (6), resulting in

$$f = \chi^2 + \Lambda. \quad (7)$$

To retrieve the blurred information, we need to find the positions of the sources that minimize Equation (7). For the two-dimensional case, we showed that this can be done using a genetic algorithm. This method shows a great performance in avoiding the characteristic local minima arising from the hundreds of degrees of freedom to optimize. On the other hand, the method has a high computational cost, as we need to evaluate the PSF at each source position (around 10^2) for each pixel of the image ($32 \times 32 \approx 10^3$ pixels) for each candidate solution in a population (around 10^2) that evolves for a certain number of iterations until convergence. The number of required iterations is in the order of 10^4 for SUPPOSE Edge, resulting in approximately 10^{11} PSF evaluations. This takes around half an hour of computations for each image in our MATLAB implementation of SUPPOSE Edge in CPU. For

the three-dimensional case this number will further increase, as the added dimension introduces a geometrical increase in the number of voxels and more virtual sources are required for the description of the surface. This leads to at least another factor of 10^2 evaluations, raising the computational time up to two days. Despite we already shown that this time can be reduced down by factor of 80 with GPU processing technology, here we present an alternative to the genetic algorithm that uses a SGD method, adapted from techniques widely used in machine learning frameworks.

As in any SGD implementation, we need to compute the gradient of our fitness function (Equation (7)) with respect to our surface parameters. Despite our functional being written in terms of the position of the virtual sources (located at the center of each face of the polyhedron), it is more convenient to minimize f with respect of the positions of the vertices that define the faces. Then, we can get an analytical expression for the partial derivatives of our fitness functional with respect to the position $\vec{r}_p = (x_p, y_p, z_p)$ of the vertex p , which will have two contributions, one arising from Equation (5) and another from Equation (6). For the first one, it is convenient to define $\vec{\delta} = \nabla S - \nabla \tilde{R} * I$, such as $\chi^2 = \|\vec{\delta}\|^2 = \vec{\delta} \cdot \vec{\delta}$. With this definition, we get

$$\frac{\partial \chi^2}{\partial x_p} = -2\alpha \sum_{ijk} \vec{\delta} \cdot \left(\sum_{t \in \{F_p\}} \frac{1}{3} \frac{\partial I_{ijk}^t}{\partial a_t} \hat{n}_t + I_{ijk}^t \frac{\partial \hat{n}_t}{\partial x_p} \right), \quad (8)$$

where we defined the set $\{F_p\}$, containing all the faces F of our polyhedron that contain the vertex p . The derivatives with respect to y_p and z_p are analogous. Then, to obtain the gradient of the χ^2 we need to compute the gradient of the PSF and the gradient of the normal vectors of each source. The first one will depend on the imaging system and can be computed analytically or numerically depending on the nature of the PSF used. The derivative of the normal vectors can be easily (and rapidly) computed from the positions of the vertices if we write \hat{n}_t as

$$\hat{n}_t = \frac{(\vec{r}_{t1} - \vec{r}_p) \times (\vec{r}_{t2} - \vec{r}_p)}{\|(\vec{r}_{t1} - \vec{r}_p) \times (\vec{r}_{t2} - \vec{r}_p)\|}, \quad (9)$$

where \vec{r}_{t1} and \vec{r}_{t2} are the remaining two vertices that together with p define the face t of the polyhedron. We also need to compute the gradient of the regularization term ϕ which leads to the expression

$$\begin{aligned} \frac{\partial \Lambda}{\partial x_p} = & \sum_{e \in \{E_p\}} 2(\hat{n}_{e1} \cdot \hat{n}_{e2} - 1) \left(\frac{\partial \hat{n}_{e1}}{\partial x_p} \cdot \hat{n}_{e2} + \hat{n}_{e1} \cdot \frac{\partial \hat{n}_{e2}}{\partial x_p} \right) \\ & + \sum_{e \in \{\bar{E}_p\}} 2(\hat{n}_{e1} \cdot \hat{n}_{e2} - 1) \left(\frac{\partial \hat{n}_{e1}}{\partial x_p} \cdot \hat{n}_{e2} \right), \end{aligned} \quad (10)$$

where we defined the sets $\{E_p\}$ (which contains all the edges E radiating from the vertex p) and $\{\bar{E}_p\}$ (which contains all the edges E that form a face in $\{F_p\}$ but are not included in $\{E_p\}$, i.e., do not contain the vertex p) and the indices 1 and 2 indicate each one of the two faces that share the edge e . This expression is better understood by looking at Figure 1(c). The faces shaded in pink are those whose normal vectors change when moving the vertex p . Then, the first sum in Equation (10) accounts for the derivatives of the terms in Λ that multiply the normal vectors of two pink faces. In the same way, the second sum accounts for the terms in Λ that multiply the normal vectors of a pink face and a blue face in Figure 1(c), that is, a face whose normal vector does not change when displacing the

vertex p but shares an edge with a face that does. Again, the derivatives with respect to y_p and z_p are analogous.

With these ingredients, we have all we need to build our SGD minimization. A similar approach was already shown for the standard SUPPOSE method applied to the very sparse problem of localizing single emitters [24]. Here, we base our algorithm in the Nesterov-accelerated Adaptive Moment Estimation (NADAM) [29] SGD minimization, which differs from the standard gradient descent method in mainly two ways: first, the positions of the sources are updated individually in an order chosen at random before every iteration (i.e., stochastic optimization). Second, the direction of each update is opposite and proportional to the gradient of the fitness functional with respect to that position (as in any gradient based method), but it also incorporates information about the history of each source by taking exponentially decaying averages: first in an inertial component called momentum that favors the movement in the same direction as its previous steps, second in an adaptive factor that modulates the size of the update of each source individually by dividing the gradient by the magnitude of its fluctuations in the previous steps [30, 31]. Overall, the NADAM implementation results in a versatile minimization algorithm capable of adapting to very different optimization scenarios with a low number of well-defined, understandable parameters.

The method starts by computing an initial guess for the surface, which can be done by using a standard edge detection algorithm or, as in our case, by computing a level surface from the volumetric image. From this guess, we can select the required number of virtual sources by using a rough estimate of the surface area, from where we can compute the minimum number of sources needed to describe the surface at the native resolution using the Nyquist criterion. As we expect to increase our image resolution, we resample our initial guess to contain a number of faces which exceeds Nyquist by a certain parameter. In our case, we use a factor of four times the minimum number of sources.

Once the initial surface is established, an iterative optimization begins, where each one of the vertices of our polyhedron is updated once every iteration but in a random order by the update rule

$$\vec{r}_{p,q+1} = \vec{r}_{p,q} + \frac{\eta}{\sqrt{\vec{v}_{p,q}} + \epsilon} \left(\beta_1 \vec{m}_{p,q} + \frac{1 - \beta_1}{1 - \beta_1^q} \nabla_p f \right), \quad (11)$$

where the index q represents the iteration number, $\nabla_p f = \left(\frac{\partial f}{\partial x_p}, \frac{\partial f}{\partial y_p}, \frac{\partial f}{\partial z_p} \right)$, $\vec{m}_{p,q}$ is the momentum vector, which is computed on each iteration as

$$\vec{m}_{p,q+1} = \frac{\beta_1}{1 - \beta_1^q} \vec{m}_{p,q} + \frac{1 - \beta_1}{1 - \beta_1^q} \nabla_p f, \quad (12)$$

$\vec{v}_{p,q}$ is the uncentered variance vector, which is computed at each iteration as

$$\vec{v}_{p,q+1} = \frac{\beta_2}{1 - \beta_2^q} \vec{v}_{p,q} + \frac{1 - \beta_2}{1 - \beta_2^q} (\nabla_p f)^2 \quad (13)$$

and where η , β_1 , β_2 , and ϵ are constant parameters chosen in advance. Notice that the $\vec{\cdot}$ notation represents \mathbb{R}^3 vectors and all of the operations (square roots, powers, etc.,) are computed element-wise. We can see from Equations (12) and (13) that the parameters β_1 and β_2 are related to the extent of an exponentially decaying memory of the momentum and the variance vectors. On the other hand, ϵ is chosen

as a small number to avoid divergences caused by small number division. The parameter η is usually known as the learning rate and defines the initial size of our updates. In all of our runs, we chose $\eta = 10^{-5}$, $\epsilon = 10^{-8}$ and β_1 and β_2 such that the half-decay time of the momentum and variance vectors is 5 and 50 iterations, respectively.

It is important to notice that the NADAM minimization is computed only on the positions of the sources. Between iterations, the intensity factor α (present in Equations (3) and (5)) is optimized at fixed positions (where it becomes a linear, closed formula, least squares problem) to best fit the experimental gradient's magnitude.

3. Simulations

We tested the performance of our method for two different types of simulated samples. First, we will focus on the ability of the method to resolve the size of structures smaller than the diffraction limit (PSF width). Second, we will quantify the performance of the method when it is applied to complex structures containing high spatial frequency features, such as spiky corners. The model for the instrument is the one presented in Equation (1), where the high-resolution image is blurred by the PSF and corrupted by noise. For the latter, a Poisson statistic is chosen, as it is usually the limiting factor in experiments where fluorescence is detected by using photon counting devices, such as photomultipliers or single photon avalanche detectors. Other kinds of more specific and setup dependent effects on the images, such as optical aberrations or more exotic noise contributions (usually related to camera detection), are not considered in this work but deferred to future research.

It is important to notice that the PSF profile used for the reconstruction should resemble as close as possible the actual PSF of the experiment. Both expression-based and numerical (look-up tables) PSF models can be used for this purpose, with the first option having computation speed advantages. The choice of model will ultimately depend on the axial sectioning technique used to acquire the stack, where light sheet, laser scanning confocal, and multiphoton microscopies are the most relevant techniques.

For a light sheet microscopy experiment, the axial profile will be that of the pump beam transverse profile, typically a Gaussian or Bessel beam. PSFs obtained by physical modeling of the instrument were shown to be appropriate for deconvolution purposes in this technique [32].

Regarding laser scanning confocal microscopy, the computed theoretical PSF must consider the Airy iris used at the detection [33–35] and filters [36]. If measured using fluorescent beads, the size of the bead is critical [37] and the protocol for the 3D PSF measurement [38] has shown that a smooth and symmetric axial profile is achieved. In accordance with this, Gaussian approximations for the confocal microscope PSF have been proposed [39, 40], where a relative squared error $< 0.5\%$ was estimated. More complex analytical expressions involving superposition of Gaussian functions [35] or Bessel functions [34] were also investigated.

Multiphoton microscopes provide another axial stacking method and multiparametric Gaussian fittings have proven a valuable approximation to the 3D PSF [41, 42]. In our 2D edge detection version [25], it was experimentally shown that a Gaussian approximation was sufficient for multiphoton and widefield fluorescence images despite the expected ringing in the PSF. Hence, what is relevant for the technique is to have a PSF model that approximates well enough the experimental PSF. Then, we define

our PSF model I, as proportional to the squared intensity profile of a propagating gaussian beam,

$$I = \frac{1}{(1 + z^2/z_r^2)^2} e^{-\frac{(x^2 + y^2)}{2\sigma^2(1 + z^2/z_r^2)}}, \quad (14)$$

as it will be expected for a two-photon excitation microscope. In this context, σ and z_r are two parameters related to lateral and axial resolution, respectively. We can see that this PSF model is Gaussian in the lateral directions but Lorentzian for the axial dimension, leading to heavier tails than a multivariate Gaussian model, and thus it is more demanding for the deconvolution method. To produce the synthetic samples, we start by defining our object as a binary volumetric image in a high-resolution grid (intensity is 1 inside the object and 0 outside of it). After this, we compute the low-resolution image by convolving with the instrumental PSF as introduced in Equation (1) with the model proposed in expression 14. For this, we choose to use $\sigma = 136$ nm and $z_r = 582$ nm, corresponding to a full width half maximum resolution of 320 nm (2.35σ) and 750 nm ($1.29 z_r$), respectively, values which are typical for laser scanning fluorescence microscopies.

Once the convolution product was evaluated, the image is resampled to a much larger voxel size by summing the contributions of each one of the fine voxels grid in a 40 nm side cube, resulting in a $64 \times 64 \times 64$ voxels volumetric image. Then, the intensities are scaled to a desired number of detected counts (in our case, 10^4 at the maximum) and passed to a Poisson random number generator to account for shot noise in the detection (resulting in a SNR of 40 dB at the brightest pixel). This process results in a low resolution, sampled, and noisy volumetric image of our object.

We implemented SUPPOSE 3Dge under NADAM SGD optimization using MATLAB (code available upon request). We found that 2000 minimization iterations are more than enough to ensure convergence. The algorithm receives only the low-resolution simulated measurement and the PSF model and returns a list of locations for the vertices and faces that describes the super-resolved surface that is the best fit to the object. Again, these locations are not limited to the voxels of the image, as each one of them can take any real value. The regularization weight was chosen using ranges similar to those of the 2D case [25] for these particular noise levels. An average run lasts for approximately 4 hours in our Intel Core i7-8700K (6 cores, 3.70 GHz), depending on the sample and the number of virtual sources used. As most of the computational time is spent evaluating the convolutions, the method would benefit greatly from GPU acceleration. Since the solutions lie in a continuous space, the typical routines based on matrix formulations and Fourier transform do not apply here. Therefore, our group has already worked on an open source library that performs a parallel computation of 2D and 3D convolutions for the original SUPPOSE method in a time and resource-efficient way using the texture mapping units of a GPU and exploiting the translational invariant property of the images (code available upon request). This 3D-convolution library could be applied to implement SUPPOSE 3Dge with the GPU.

4. Results

The results of the first experiment, regarding the detection of the size of structures, are depicted in Figure 2. For this, we simulated the volumetric fluorescence signal of spherical objects of different sizes. We selected diameters ranging from 800 nm (same order as axial

resolution) down to 200 nm (much smaller than both axial and lateral resolution). In Figure 2, we can see a 3D representation of the surfaces of the objects in decreasing size (a) to (d) and the half-maximum level surface of the PSF in (e). For ease of comparison, all the images (including the PSF) are at the same scale. For each one of the figures corresponding to the simulated samples we can see (from left to right): the high-resolution sample, the SUPPOSE 3Dge reconstruction obtained from the simulated measurement, and a histogram that measures the distance from each face of the polyhedron to the geometrical center of the same. From these distances, we can compute the mean value (that we call the fitted radius of the reconstruction, marked with a blue arrow pointing to its value over each histogram) reporting to the accuracy of the method. In light blue shading, we highlight one standard deviation of the distribution to each side of the mean value (numerical value underneath), reporting on the precision of the solution. Finally, and for comparison we indicate the true sample radius with a vertical red line.

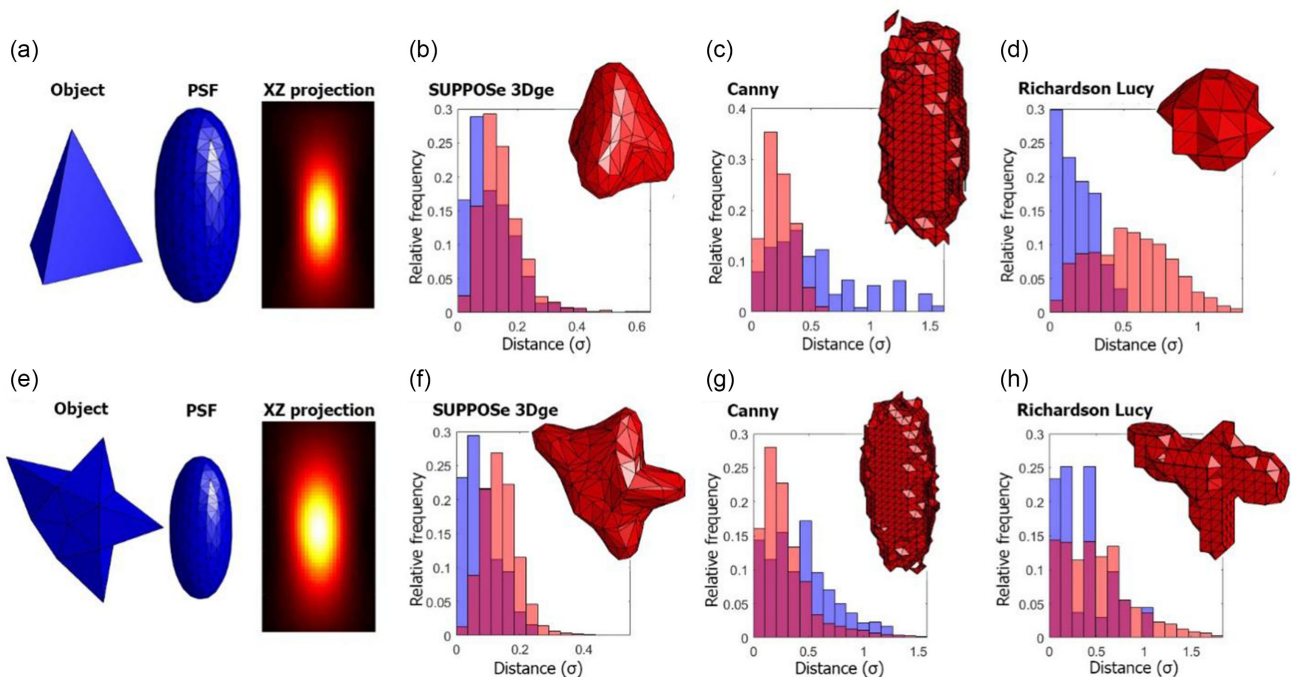
We can see that the solutions match the shape and size of the target, and that the fitted radii are in good agreement with the sample. When comparing the fitted radius with the real one (blue and red lines over the histograms) we can see that there is a bias toward the right (bigger radius). This is a direct consequence of the method used to choose the initial surface from where the minimization starts. As we use a diffraction limited detection method to set this initial guess, the resulting surface will always be roughly the size of the PSF when the structure is smaller than the resolution, leaving the target surface completely inside of it. As a result of this, the minimization algorithm will always approach the solution from the outside. As the optimization is getting closer to the target, the gradient of our fitness

functional becomes smaller and smaller until it starts competing with the noise in our measurement and the solution stops improving upon iterations. This reports strongly on the accuracy of the technique, as it is a measure of how close we can get to the target by optimization. If we look carefully, we can see that the histograms are quite narrow, as the 2σ interval, which contains roughly 70% of the virtual sources, yields a precision between 8 nm and 19 nm (1% to 7% of the sphere diameter) and an accuracy between 4 nm and 16 nm (2% to 5% of the sphere diameter) for the different sizes. These numbers correspond to an accuracy and precision between 16 and 80 times better than the lateral resolution of the instrument and between 40 and 180 times the axial resolution.

In Figure 3, we show the result of a different simulated experiment, where we explore the performance of the method on retrieving complex shapes. For this, we simulated the volumetric fluorescence image of a tetrahedron (Figure 3(a)) and an asymmetric star (Figure 3(e)) that we show next to the PSF used, and a projection of the volumetric image in the xz direction (all the intensity values in the y direction summed). Because of how the latter compares to the size of the structures, most of the details that are characteristic of the objects (sharp edges and corners) are completely lost in the convolution, as we can see in the projections, from which the shape of the object cannot be identified. We decided to compare our solutions with a 3D Canny edge detection method available since 2017 as a MATLAB function (edge3) and moreover adding a 3D deconvolution that incorporates the PSF information not used by Canny. In Figure 3 for each sample we compare the solution from SUPPOSE 3Dge (Figure 3(b) and (f)) to the 3D Canny edge detection method applied directly to the volumetric image (Figure 3(c) and (g)) and after using a Richardson–Lucy deconvolution with the same PSF information

Figure 3

Determination of high-resolution details in unresolved objects. For ease of comparison, each row (a)–(d) and (e)–(h) are displayed at the same scale (except for projections of the volumetric images). (a) Sample (tetrahedron) compared to the PSF and XZ projection. (b) SUPPOSE 3Dge reconstruction and performance histograms. (c) Same for Canny edge detection method. (d) Same for Canny method after Richardson–Lucy deconvolution. In all cases, blue histogram corresponds to the first metric (distance from solution to target) and red histogram corresponds to second metric (distance from target to solution). (e) to (h), same experiment and analysis but for an asymmetric star shaped sample



supplied to SUPPOSE 3Dge (Figure 3(d) and 3(h)). In Vazquez et al. [25], we define a quantifier that measures the similarity between two sets of coordinates. That is, if $A = \{a_1, \dots, a_k\}$ and $B = \{b_1, \dots, b_k\}$ are two sets of coordinates, for each position of a source in A then the distance to B is



$$d_k^{AB} = \min_{1 \leq l \leq L} |a_k - b_l|, \quad \forall 1 \leq k \leq K \quad (15)$$

If we take A as the solution of one of our 3D edge detection methods and B as the target object we define $d_1 = d^{AB}$ (i.e., the distance from the solution to the object) and $d_2 = d^{BA}$ (i.e., the distance from the object to the solution). We show this information by generating histograms for the quantifiers d_1 (blue) and d_2 (red) for each one of our solutions. In this way, the first measures how close is the reconstruction to the object while the second is useful for detecting unrecovered features (pieces of the target not well identified in the solution).

The information loss can be clearly seen in the diffraction limited Canny detection. For both shapes, the detected surfaces roughly match the PSF, showing none of the special attributes of the objects. The same information can be seen in the blue histogram, corresponding to d_1 . The heavy tails show that the detected surface is far from the target in distances on the order of the instrumental resolution. As the original Canny's method does not include information on the PSF, we take it a step further and we apply a Richardson–Lucy deconvolution before the surface detection. For the tetrahedron (Figure 3d), we can see the axial resolution greatly improved, but the shape of the object is still not retrieved, as the solution is roughly a sphere. We can also easily see this in the red histogram, measuring the distance from each point in the solution to the nearest point in the detection. There, a broad and heavy tailed histogram evidences how the sharp edges in the tetrahedron found its closest detection at distances on the order of the resolution. Something similar happens with the star shape, where we can see some retrieval of the shape, but very distorted and with the upper part missing, also reflecting in a long-tailed histogram for d_2 .

When we compare all of this with the SUPPOSE 3Dge solution the enhancement is evident. In Table 1, we show a numerical comparison of the distance quantifiers for the different reconstructions. Our method succeeds in retrieving the shape of both the tetrahedron and the star shape with no missing features. As expected, the sharp corners appear rounded due to the presence of the regularization term that forces some smoothness in the solution. Despite that, we see that SUPPOSE 3Dge recovers most of the information regarding the shape of the objects that was well below the instrumental resolution. If we look at the distance quantifiers, we can see that both histograms are narrow, well-behaved and with mean values below 1/12 of the lateral resolution and 1/27 of the axial resolution of the instrument. As shown in Table 1, the mean values of the histograms also show that SUPPOSE 3Dge outperforms previous methods by a factor between 3 and 5 and shows consistently smaller standard deviations. A quantitative study about the effects of SNR and the regularization weight on the performance of SUPPOSE 3dge would require to perform a statistically significant number of runs, a task for which a prior optimization on the computational time of the algorithm is mandatory, since, in its current state, each run takes several hours with our equipment. In this regard, GPU parallelization is a promising approach for speed improvement. Nevertheless, accuracy degradation due to different SNR levels was previously studied in the 2D edge detection method [25], together with the effect of different regularization weights. A similar dependence is thus expected for SUPPOSE 3Dge.

Table 1
Mean and standard deviation (Std) of the histograms shown in Figure 3, corresponding to the distance quantifiers d_1 and d_2 , used to measure the performance of each method (in units of PSF σ)

		SUPPOSE 3Dge		Canny		Richardson– Lucy	
		Mean	Std	Mean	Std	Mean	Std
	d_1	0.11	0.07	0.52	0.37	0.19	0.12
	d_2	0.15	0.08	0.24	0.11	0.54	0.27
	d_1	0.08	0.05	0.43	0.30	0.35	0.28
	d_2	0.14	0.06	0.29	0.23	0.50	0.35

5. Conclusions

We presented SUPPOSE 3Dge, a method for the detection of surfaces in volumetric fluorescence microscopy images with resolution better than the diffraction limit. We tested the performance of the method with simulated images for two different situations: first, the accurate detection of the size of objects smaller than the PSF and second, the retrieval of high-resolution details that are lost in the convolution with the instrumental response. In the first simulated experiment, we showed that the method can accurately identify the size of spherical objects more than three times smaller than the expected axial resolution. For the second simulated experiment, we compared SUPPOSE 3Dge with Canny's method generalized to three dimensions with and without using Richardson–Lucy deconvolution for increased resolution. The result of these experiments shows that our method successfully retrieves the shapes of objects where the other methods fail, yielding a precision and accuracy as good as 1/100 of the instrument resolution for smooth structures and 1/10 of the resolution for complex spiky surfaces.

Funding Support

This work was partially financed by grant PICT-2020-SERIEA-00617 from Agencia Nacional de Promoción Científica y Tecnológica, grant PIP-2021-2023-11220200101184CO from Consejo Nacional de Investigaciones Científicas y Técnicas (CONICET), grant ubacyt-2018-20020170100137BA from Universidad de Buenos Aires, and grant FA9550-18-1-0470-P00001 from AFOSR.

Ethical Statement

This study does not contain any studies with human or animal subjects performed by any of the authors.

Conflicts of Interest

The authors declare that they have no conflicts of interest to this work.

Data Availability Statement

Data are available on request from the corresponding author upon reasonable request.

Author Contribution Statement

Guillermo D. Brinatti Vazquez: Conceptualization, Methodology, Software, Validation, Formal analysis, Investigation, Data curation, Writing – original draft, Writing – review & editing,

Visualization. **Axel M. Lacapmesure**: Software, Writing – review & editing. **Sandra Martínez**: Conceptualization, Methodology, Writing – review & editing, Supervision, Project administration, Funding acquisition. **Oscar E. Martínez**: Conceptualization, Methodology, Writing – review & editing, Supervision, Project administration, Funding acquisition.

References

- [1] Ali, R. A., Landsberg, M. J., Knauth, E., Morgan, G. P., Marsh, B. J., & Hankamer, B. (2012). A 3D image filter for parameter-free segmentation of macromolecular structures from electron tomograms. *PLOS One*, 7(3), e33697. <https://doi.org/10.1371/journal.pone.0033697>
- [2] Volkman, N. (2010). Methods for segmentation and interpretation of electron tomographic reconstructions. *Methods in Enzymology*, 483, 31–46. [https://doi.org/10.1016/S0076-6879\(10\)83002-2](https://doi.org/10.1016/S0076-6879(10)83002-2)
- [3] Cuny, A. P., Schlottmann, F. P., Ewald, J. C., Pelet, S., & Schmoller, K. M. (2022). Live cell microscopy: From image to insight. *Biophysics Reviews*, 3(2), 021302. <https://doi.org/10.1063/5.0082799>
- [4] Carpenter, A. E., Jones, T. R., Lamprecht, M. R., Clarke, C., Kang, I. H., Friman, O., ..., & Sabatini, D. M. (2006). CellProfiler: Image analysis software for identifying and quantifying cell phenotypes. *Genome Biology*, 7(10), R100. <https://doi.org/10.1186/gb-2006-7-10-r100>
- [5] Wahlby, C., Sintorn, I. M., Erlandsson, F., Borgefors, G., & Bengtsson, E. (2004). Combining intensity, edge and shape information for 2D and 3D segmentation of cell nuclei in tissue sections. *Journal of Microscopy*, 215(1), 67–76. <https://doi.org/10.1111/j.0022-2720.2004.01338.x>
- [6] 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. <https://doi.org/10.1046/j.1365-2818.2000.00710.x>
- [7] Harke, B., Keller, J., Ullal, C. K., Westphal, V., Schönle, A., & Hell, S. W. (2008). Resolution scaling in STED microscopy. *Optics Express*, 16(6), 4154–4162. <https://doi.org/10.1364/OE.16.004154>
- [8] Hell, S. W., & Wichmann, J. (1994). Breaking the diffraction resolution limit by stimulated emission: Stimulated-emission-depletion fluorescence microscopy. *Optics Letters*, 19(11), 780–782. <https://doi.org/10.1364/OL.19.000780>
- [9] Yagüe-Fabra, J., Ontiveros, S., Jimenez, R., Chitchian, S., Tosello, G., & Carmignato, S. (2013). A 3D edge detection technique for surface extraction in computed tomography for dimensional metrology applications. *CIRP Annals*, 62(1), 531–534. <https://doi.org/10.1016/j.cirp.2013.03.016>
- [10] Chitchian, S., Weldon, T. P., Fiddy, M. A., & Fried, N. M. (2010). Combined image-processing algorithms for improved optical coherence tomography of prostate nerves. *Journal of Biomedical Optics*, 15(4), 046014. <https://doi.org/10.1117/1.3481144>
- [11] Rathnayaka, K., Sahama, T., Schuetz, M. A., & Schmutz, B. (2011). Effects of ct image segmentation methods on the accuracy of long bone 3D reconstructions. *Medical Engineering & Physics*, 33(2), 226–233. <https://doi.org/10.1016/j.medengphys.2010.10.002>
- [12] Di, H., & Gao, D. (2014). Gray-level transformation and canny edge detection for 3D seismic discontinuity enhancement. *Computers & Geosciences*, 72, 192–200. <https://doi.org/10.1016/j.cageo.2014.07.011>
- [13] Gao, T., & Yang, Z. (2020). 3D object recognition method based on improved canny edge detection algorithm in augmented reality. In *IEEE 5th International Conference on Image, Vision and Computing*, 19–23. <https://doi.org/10.1109/ICIVC50857.2020.9177488>
- [14] Zhang, Y. J. (1993). Quantitative study of 3D gradient operators. *Image and Vision Computing*, 11(10), 611–622. [https://doi.org/10.1016/0262-8856\(93\)90057-N](https://doi.org/10.1016/0262-8856(93)90057-N)
- [15] Brejl, M., & Sonka, M. (2000). Directional 3d edge detection in anisotropic data: Detector design and performance assessment. *Computer Vision and Image Understanding*, 77(2), 84–110. <https://doi.org/10.1006/cviu.1999.0811>
- [16] Guo, Y., Bennamoun, M., Sohel, F., Lu, M., Wan, J., & Kwok, N. M. (2016). A comprehensive performance evaluation of 3D local feature descriptors. *International Journal of Computer Vision*, 116(1), 66–89. <https://doi.org/10.1007/s11263-015-0824-y>
- [17] Canny, J. (1986). A computational approach to edge detection. In *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-8(6), 679–698. <https://doi.org/10.1109/TPAMI.1986.4767851>
- [18] Monga, O., & Deriche, R. (1989). 3D edge detection using recursive filtering: Application to scanner images. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 28–35. <https://doi.ieeecomputersociety.org/10.1109/CVPR.1989.37825>
- [19] Monga, O., Deriche, R., & Rocchisani, J. M. (1991). 3D edge detection using recursive filtering: Application to scanner images. *CVGIP: Image Understanding*, 53(1), 76–87. [https://doi.org/10.1016/1049-9660\(91\)90006-B](https://doi.org/10.1016/1049-9660(91)90006-B)
- [20] Ali, R. A., Mehdi, A. M., Rothnagel, R., Hamilton, N. A., Gerle, C., Landsberg, M. J., & Hankamer, B. (2017). RAZA: A rapid 3D z-crossings algorithm to segment electron tomograms and extract organelles and macromolecules. *Journal of Structural Biology*, 200(2), 73–86. <https://doi.org/10.1016/j.jsb.2017.10.002>
- [21] Zhang, L. Y., Lin, P., Pan, J., Ma, Y., Wei, Z., Jiang, L., ..., & Yang, G. Y. (2018). CLARITY for high-resolution imaging and quantification of vasculature in the whole mouse brain. *Aging and Disease*, 9(2), 262–272. <https://doi.org/10.14336%2FAD.2017.0613>
- [22] Sage, D., Donati, L., Soulez, F., Fortun, D., Schmit, G., Seitz, A., ..., & Unser, M. (2017). DeconvolutionLab2: An open-source software for deconvolution microscopy. *Methods*, 115, 28–41. <https://doi.org/10.1016/j.ymeth.2016.12.015>
- [23] Martínez, S., Toscani, M., & Martínez O. E. (2019). Superresolution method for a single wide-field image deconvolution by superposition of point sources. *Journal of Microscopy*, 275(1), 51–65. <https://doi.org/10.1111/jmi.12802>
- [24] Lacapmesure, A. M., Vazquez, G. D. B., Mazzeo, A., Martínez, S., & Martínez, O. E. (2022). Combining deep learning with SUPPOSE and compressed sensing for SNR-enhanced localization of overlapping emitters. *Applied Optics*, 61(7), D39–D49. <https://doi.org/10.1364/AO.444610>
- [25] Vazquez, G. D. B., Martínez, S., & Martínez, O. E. (2020). Super-resolved edge detection in optical microscopy images by superposition of virtual point sources. *Optics Express*, 28(17), 25319–25334. <https://doi.org/10.1364/OE.397125>
- [26] Chao, J., Ward, E. S., & Ober, R. J. (2016). Fisher information theory for parameter estimation in single molecule microscopy: Tutorial. *Journal of the Optical Society of America A*, 33(7), B36–B57. <https://doi.org/10.1364/JOSAA.33.000B36>
- [27] Lelek, M., Gyparakis, M. T., Beliu, G., Schueder, F., Griffie, J., Manley, S., ..., & Zimmer, C. (2021). Single-molecule

- localization microscopy. *Nature Reviews Methods Primers*, 1(1), 39. <https://doi.org/10.1038/s43586-021-00038-x>
- [28] Ober, R. J., Ram, S., & Ward, E. S. (2004). Localization accuracy in single-molecule microscopy. *Biophysical Journal*, 86(2), 1185–1200. [https://doi.org/10.1016/S0006-3495\(04\)74193-4](https://doi.org/10.1016/S0006-3495(04)74193-4)
- [29] Dogo, E. M., Afolabi, O. J., Nwulu, N. I., Twala, B., & Aigbavboa, C. O. (2018). A comparative analysis of gradient descent-based optimization algorithms on convolutional neural networks. In *IEEE International Conference on Computational Techniques, Electronics and Mechanical Systems*, 92–99. <https://doi.org/10.1109/CTEMS.2018.8769211>
- [30] Kingma, D. P., & Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv Preprint: 1412.6980*. <https://doi.org/10.48550/arXiv.1412.6980>
- [31] Reddi, S. J., Kale, S., & Kumar, S. (2019). On the convergence of Adam and beyond. *arXiv Preprint: 1904.09237*. <https://doi.org/10.48550/arXiv.1904.09237>
- [32] Becker, K., Saghafi, S., Pende, M., Sabdyusheva-Litschauer, I., Hahn, C. M., Foroughpour, M., . . . , & Dodt, H. U. (2019). Deconvolution of light sheet microscopy recordings. *Scientific Reports*, 9(1), 17625. <https://doi.org/10.1038/s41598-019-53875-y>
- [33] 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. <https://doi.org/10.1111/j.1365-2818.2012.03675.x>
- [34] Li, J., Xue, F., & Blu, T. (2017). Fast and accurate three-dimensional point spread function computation for fluorescence microscopy. *Journal of the Optical Society of America A*, 34(6), 1029–1034. <https://doi.org/10.1364/JOSAA.34.001029>
- [35] Samuylov, D. K., Purwar, P., Szekely, G., & Paul, G. (2019). Modeling point spread function in fluorescence microscopy with a sparse Gaussian mixture: Tradeoff between accuracy and efficiency. *IEEE Transactions on Image Processing*, 28(8), 3688–3702. <https://doi.org/10.1109/TIP.2019.2898843>
- [36] Neil, M. A. A., Juskaitis, R., Wilson, T., Laczik, Z. J., & Sarafis, V. (2000). Optimized pupil-plane filters for confocal microscope point-spread function engineering. *Optics Letters*, 25(4), 245–247. <https://doi.org/10.1364/OL.25.000245>
- [37] Song, I., Yoo, H., Choo, J., & Gweon, D. G. (2005). Measurement of point-spread function (PSF) for confocal fluorescence microscopy. *Advanced Characterization Techniques for Optics, Semiconductors, and Nanotechnologies II*, 5878, 58781B. <https://doi.org/10.1117/12.616701>
- [38] Cole, R. W., Jinadasa, T., & Brown, C. M. (2011). Measuring and interpreting point spread functions to determine confocal microscope resolution and ensure quality control. *Nature Protocols*, 6(12), 1929–1941. <https://doi.org/10.1038/nprot.2011.407>
- [39] Zhang, B., Zerubia, J., & Olivo-Marin, J. C. (2006). A study of Gaussian approximations of fluorescence microscopy PSF models. *Three-Dimensional and Multidimensional Microscopy: Image Acquisition and Processing XIII*, 6090, 60900K. <https://doi.org/10.1117/12.645650>
- [40] Zhang, B., Zerubia, J., & Olivo-Marin, J. C. (2007). Gaussian approximations of fluorescence microscope point-spread function models. *Applied Optics*, 46(10), 1819–1829. <https://doi.org/10.1364/AO.46.001819>
- [41] Chouzenoux, E., Lau, T. T. K., Lefort, C., & Pesquet, J. C. (2019). Optimal multivariate Gaussian fitting with applications to PSF modeling in two-photon microscopy imaging. *Journal of Mathematical Imaging and Vision*, 61(7), 1037–1050. <https://doi.org/10.1007/s10851-019-00884-1>
- [42] Lefort, C., Chouzenoux, E., Magnol, L., Massias, H., & Pesquet, J.-C. (2020). Multi-parametric 3D-point-spread function estimation in deep multiphoton microscopy with an original computational strategy dedicated to the reconstruction of muscle images. *Optical Sensing and Detection VI*, 11354, 113541I. <https://doi.org/10.1117/12.2554742>

How to Cite: Brinatti Vazquez, G. D., Lacapmesure, A. M., Martínez, S., & Martínez, O. E. (2024). SUPPOSe 3Dge: A Method for Super-Resolved Detection of Surfaces in Volumetric Fluorescence Microscopy. *Journal of Optics and Photonics Research*. <https://doi.org/10.47852/bonviewJOPR42022350>