





# Super-resolved edge detection in optical microscopy images by superposition of virtual point sources

**GUILLERMO D. BRINATTI VAZQUEZ,<sup>1,2,\*</sup>  SANDRA MARTÍNEZ,<sup>2,3</sup>   
AND OSCAR E. MARTÍNEZ<sup>1,2</sup>**

<sup>1</sup>Laboratorio de Fotónica, Instituto de Ingeniería Biomédica, Universidad de Buenos Aires, Argentina

<sup>2</sup>Consejo Nacional de Investigaciones Científicas y Técnicas (CONICET), Argentina

<sup>3</sup>Departamento de Matemática, Facultad de Ciencias Exactas y Naturales, IMAS, CONICET, Universidad de Buenos Aires, Argentina

\*[gbrinatt@fi.uba.ar](mailto:gbrinatt@fi.uba.ar)

**Abstract:** A new approach to the edge detection problem is presented which is specially designed to achieve high accuracy detection, below instrumental resolution (super resolution) in microscopy images. The method is based in a modified version of a recently published algorithm known as SUPPOSE, which performs a numerical reconstruction of an image as a superposition of virtual point sources. The method was tested in simulated and experimental optical microscopy images and compared to the standard Laplacian of Gaussian algorithm, showing huge differences when the size of the object is smaller than the lateral resolution provided by the instrument.

© 2020 Optical Society of America under the terms of the [OSA Open Access Publishing Agreement](#)

## 1. Introduction

Quantitative analysis from microscopy images is one of the main tools in cell biology but also in nanophotonics and in semiconductor science and applications. In this context, the retrieval of the exact size and shape of the objects under study is of great interest. In order to obtain a precise, repetitive and unbiased definition of the object's contour, it is often necessary to use edge detection algorithms.

This kind of algorithms have a long story and a distinguished role in image processing [1], pattern recognition [2] and computer vision [3]. Its goal is to generate a sparse image from a natural one, where only the large gradients in intensity are preserved. There are two main strategies to solve this problem. The first one is to apply some kind of first order differential operator and then to identify the points of large gradient by setting a threshold. In this family of methods are included the algorithms by Roberts, Prewitt, Sobel [4] and Canny [5], which are widely used. The second approach is to use a second order differential operator (such as the Laplacian) and search for zero-crossing. In this family we found the Marr-Hildreth [6] or Laplacian of Gaussian (LOG) algorithm as the most spread. More recent methods [7,8] make use of complex cost functions that exploit structural information (edge thickness, curvature, fragmentation, etc.) to enhance edge detection for complex images such as the ones obtained in medical imaging applications. These papers use a genetic algorithm to minimize the cost function and obtain the edge that best suits this conditions.

Although these methods were developed several years ago, most published articles that rely on edge detection are found to use one of the most basic algorithms [9–12]. This can be seen as a consequence of most image processing software (ImageJ, Matlab Image processing toolbox, etc) having these methods already implemented in their libraries.

Despite this wide spreading the methods present serious limitations on the precision of the localization, as the detection is pixel limited. Some more recent articles claim to achieve sub-pixel resolution by computing a non linear fit to the image after the differential filter is applied [13,14].

This might increase the precision of the localization in some cases, but in the case where the object is smaller than the image lateral resolution (very usual in microscopy images) the initial guess might be several pixels wrong. Additionally, as these methods are very sensitive to noise, it is usual to perform a previous step where the image is smoothed, leading to another reduction in the accuracy of the detection. A recent paper [15] shows that (in the context of cell migration) the estimations can differ up to a 25% just by changing the edge detection method or one of its parameters.

Some other algorithms claim to achieve super-resolution in image processing [16,17]. Although a similar name is used, the word resolution is used as equivalent to the number of pixels instead of a measure of detail level. Those algorithms use statistical priors to increase the amount of pixels on an image and are an alternative to interpolation methods rather than edge detection algorithms. Despite the risk of confusion, we chose to stay with the term super-resolution to describe our method as it is the usual in the microscopy community [18], meaning to recover details which are below instrumental resolution.

The main assumption on all previous methods (in particular, LOG and Canny, which are the most used) is to assume that when convolving the object with the PSF the contributions from the different edges do not overlap, i.e. that the object is larger than the PSF. To overcome this limitation we propose a new approach to the problem for edge detection which is specially designed to achieve high accuracy detection, not only sub-pixel but also below the instrumental resolution. This algorithm is based in a recently proposed deconvolution algorithm named SUPPOSE [19] (SUPERposition of POint Sources) which was designed to obtain super resolved images (with resolution below diffraction limit) from a standard fluorescence microscope. The idea behind both of these methods is to approximate the image or its gradient by a superposition of point like sources of equal strength that live in a continuous domain. These point sources will generate a synthetic version of the image (in SUPPOSE) or the image's gradient field (in SUPPOSE Edge) when convolved with the instrument's Point Spread Function (PSF). Then the problem is changed to find the best distribution of the sources to fit the image or its gradient in the least square sense. As this is a complex minimization problem, a genetic algorithm is used to find the best solution. But, as we are solving an inverse problem (i.e. finding the edge that is the best fit of the image gradient), the introduction of complex cost functions, as the ones used in previous genetic heuristics implementations [7,8], is unnecessary. And, most important, the use of the PSF information allows a detection accuracy below instrumental resolution.

In section 2 a detailed description of the method is presented and the main features of the algorithm are discussed. In section 3 the experimental setup and the sample preparation methods are described. Then, in section 4, we will show the simulated and experimental images used to test the algorithm's performance. A comparison with the standard Laplacian of Gaussian method is also presented, leading to the article's conclusions in section 5. A detailed description of the genetic algorithm used to solve the minimization problem is given in Appendix A.

## 2. SUPPOSE Edge method

We start by briefly reviewing the deconvolution problem addressed by SUPPOSE. Any microscopy image is blurred by the microscope Point Spread Function (PSF), obtaining a low resolution (LR) image  $S$  from the target image  $R$ . These two are related by a convolution product of the form

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

where  $I$  represents the microscope PSF and where we added the variable  $\eta$  to account for image noise. Here, for simplicity, we assumed a background free image. On the other hand, no supposition is made on the nature of  $\eta$  as it is added for Eq. (1) to hold and it can, for example, depend on  $R$ , as will be the case for Poisson noise (a more detailed discussion on the noise model can be found in the [appendix](#)). The SUPPOSE algorithm transforms the deconvolution problem

(to retrieve  $R$ ) into an unrestricted least square problem by proposing that  $R$  may be approximated by a superposition of virtual point sources of equal intensity. We write

$$R = \alpha \sum_{k=1}^{N_s} \delta(x - a_k, y - b_k), \quad (2)$$

where  $(a_k, b_k)$  are the positions of each source,  $\alpha$  is a constant and  $N_s$  is the total amount of sources used to approximate the image. The positions are unrestricted and different intensities in the image can be achieved by stacking multiple sources in a small area. Then, given a certain  $N_s$  and  $\alpha$ , the problem turns into finding the positions that minimize the difference between the measured image  $S$  and the convolution of the virtual sources with the PSF. This is minimizing the quantity

$$\chi^2 = \sum_{i=1}^{N_x} \sum_{j=1}^{N_y} (S_{ij} - \alpha \sum_{k=1}^{N_s} I(x_i - a_k, y_j - b_k))^2, \quad (3)$$

where the pair of indexes  $i$  and  $j$  sample all the image pixels, being  $N_x$  and  $N_y$  the number of pixels in each dimension. This minimization problem is solved by means of a genetic algorithm. The SUPPOSE method proved to enhance resolution of wide-field fluorescence images up to four times. The resolution enhancement and the convergence speed of the algorithm is better when the image is sparse. This means that, in an application where we only need to identify the edges of a non sparse image, a lot of virtual sources (i.e. a large  $N_s$ ) must be used to account for the parts of the image with no useful information. This results in a poor reconstruction.

The SUPPOSE Edge algorithm overcomes this problem by fitting the gradient of the image, which is sparse for the images of interest. This brings increased complexity, as we should now reconstruct a vector field instead of an image. Computing the gradient in Eq. (1) we obtain

$$\nabla S = \nabla R * I + \eta', \quad (4)$$

where  $\eta'$  is the derivative of the noise.  $\nabla R$  should only be large at the edges of the image, where a large leap in the intensity is present. Then, we want to approximate  $\nabla R$  as a superposition of point sources (sources of gradient rather than intensity), but this time we have to account for the direction of the gradient, i.e. the vector normal to the contour curve. Then we propose to approximate  $\nabla R$  as

$$\nabla R = \alpha \sum_{k=1}^{N_s} \delta(x - a_k, y - b_k) \hat{n}_k, \quad (5)$$

where  $\hat{n}_k$  is the unit length normal vector, computed from the positions of the first neighbours of each source, i.e.  $\hat{n}_k \perp (a_{k+1} - a_{k-1}, b_{k+1} - b_{k-1})$ . Then we should minimize

$$\chi^2 = \|\nabla S - \nabla R * I\|^2 = \sum_{d=1}^2 \sum_{i=1}^{N_x} \sum_{j=1}^{N_y} \left( \left( (\nabla S)_{ij} - \alpha \sum_{k=1}^{N_s} I_{ij}^k \hat{n}_k \right) \cdot \hat{e}_d \right)^2, \quad (6)$$

where  $I_{ij}^k$  is short for  $I(x_i - a_k, y_j - b_k)$ ,  $\hat{e}_d$  denotes the canonical vector and  $(\cdot)$  is the scalar product in  $\mathcal{R}^2$ . At this point the problem is not well conditioned, as for example two sources placed at the same position but with opposite normal vector cancel out their contribution. For this reason a regularization term must be used to avoid rapid changes in the normal of the curve. Then we define the fitness  $f$  as

$$f = \chi^2 + \lambda \sum_{k=1}^{N_s} (1 - \hat{n}_k \cdot \hat{n}_{k+1})^2, \quad (7)$$

where  $\lambda$  is a parameter to choose in the algorithm. Notice that the added term will be small for a slow varying curve and large in the opposite scenario. The SUPPOSE Edge method is based on

an algorithm that searches for the minimizers of  $f$  after proposing an initial parameter  $\lambda$  and a number of virtual sources  $N_s$ . Note that, as the normal vectors are computed from the positions of the sources, the algorithm should only minimize the fitness with respect to these positions. Additionally, we need to notice that the only two hyper-parameters of the method are  $N_s$ , the number of virtual point sources (nodes of the contour) and  $\lambda$  (the weight of the regularization term), as the best  $\alpha$  can be computed for a given contour (see the [appendix](#) for the detailed calculation). A discussion on the election of  $\lambda$  is presented in section 4, where we will see that the solution is not really sensitive to the precise value of this parameter. On the other hand, the number of sources used for the approximation can be decided from a rough estimation of the perimeter or length of the edges. The relationship between the length of the contour and the characteristic length of the instrument's PSF together with the Nyquist criterion gives the minimal number of sources needed to approximate that contour in the instrumental resolution. As we are pursuing resolution greater than that limit, we should incorporate additional sources according to the expected increase in resolution. A factor of 6 times the initial number of sources was found to work in most cases, depending on the quality of the first estimation. This means that, although there are two hyper-parameters, there is not a lot of room to play with them and they can even be computed automatically, making the method more robust and consistent.

As in SUPPOSE, the minimization is achieved by means of a genetic algorithm. A detailed description of the genetic algorithm used to solve the minimization problem and some fine details of the method are presented in the appendix for the interested reader.

Finally, we can compute the expected value of Eq. (6) in the ideal case where Eq. (5) is a perfect reconstruction of the target's gradient. In that case, by looking at Eq. (4), we can see that this value will depend completely on noise. By using that the noise is independent for each pixel, we can relate the ideal value for  $\chi^2$  with the variance of each pixel in the measured image as

$$\langle \chi_{ideal}^2 \rangle = 4 \sum_{i=1}^{N_x} \sum_{j=1}^{N_y} \text{Var}(S_{ij}) \quad (8)$$

where  $\langle \rangle$  is the expectation value over different noise realizations and  $\text{Var}()$  is the variance operator. This means that, if we have a model for the fluctuations of our signal, we can use it to obtain a value for the ideal fitness. In our two photon absorption microscope, the noise has ideally a Poisson statistics (photon counting) meaning that  $\text{Var}(S_{ij}) = \langle S_{ij} \rangle$ . In that scenario, we can estimate the ideal fitness as four times the sum of all the pixels composing the image (here, we approximate the expectation value of  $S$  by using one measurement of the stochastic variable). This value can be used as a part of a stop criterion of the algorithm (see appendix A4 for a short deduction and further discussion on the stop criterion).

Although we choose to perform our measurements using two photon excitation microscopy and widefield epifluorescence microscopy, the method can be applied for any case where Eq. (1) holds, that is, where the image can be written as a convolution product between the target object and the instrument response. This is the case for almost every optical microscopy and many other techniques.

### 3. Experimental

We will show experimental results for two different optical microscopy techniques: two photon excitation microscopy and widefield epifluorescence microscopy.

The first ones were obtained using our homemade Two Photon Excitation Microscope. A Titanium:Sapphire laser delivers 50 fs pulses at 800 nm with a mean power of 450 mW to the microscope. A prism compressor is used to achieve minimal dispersion at the sample in order to maximize two photon absorption. A 40X 0.95 NA microscope objective (Olympus UPLSAPO40X2) is used to focus the laser beam in the sample. Absorptive neutral filters are

used to reduce optical power after the scanning optics, resulting in a mean power at the sample plane between 0.1 mW to 6 mW. The fluorescence signal is collected by the same objective and separated from the laser beam path by using a short pass dichroic mirror with a cutoff wavelength at 650 nm. Short pass filters are used to strongly attenuate any light at wavelengths longer than 700 nm. Finally, the fluorescence signal is detected using a photon counting module (Hamamatsu H7828). The resulting detection window, accounting for all the optics and the detector response curve, is between 400 nm to 650 nm.

For the widefield epifluorescence measurements, a commercial Olympus inverted microscope was used. A 525 nm center wavelength green LED is used as the excitation source. To detect a peak emission wavelength at 580 nm, a fluorescence cube with an 550 nm centered longpass dichroic mirror was chosen. A bandpass excitation filter (525 nm center wavelength, 45 nm FWHM) and an emission filter (565 nm center wavelength, longpass) were used. For these measurements a 10X 0.3 NA microscope objective (Olympus UPlanF110X) and a 40X 0.95 NA microscope objective (Olympus UPLSAPO40X2) were employed.

Two different samples were used in this work. The first one was produced using a polycarbonate membrane filter (Isopore DTTP01300) with 600 nm pores. These filters have the advantage of having a highly repetitive pore size, becoming a good reference sample to test the method. No staining was introduced in the filters, so we measured the autofluorescence of the polycarbonate substrate. This results in challenging low signal, noisy images.

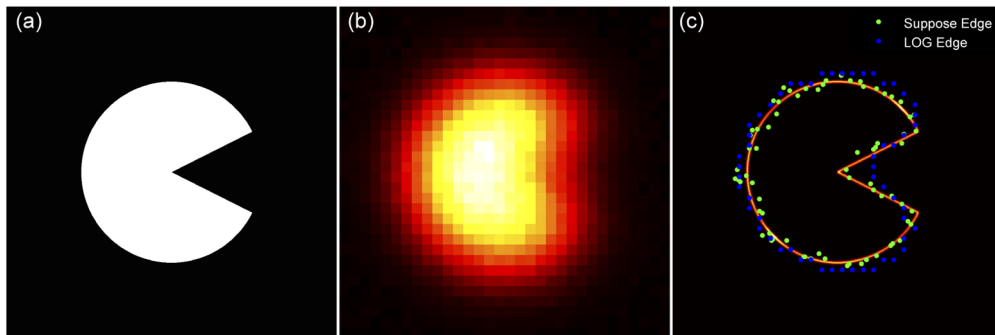
The second type of samples used were stained polystyrene beads (Thermofisher TetraSpeck Fluorescent Microspheres mounted slide) with a nominal diameter of 200 and 500 nm and a fluorescence emission maximum at 580 nm. The advantage of these samples is that, being stained, the SNR of the measurements are much higher compared to the polycarbonate filters. The disadvantage is that the size dispersion is much greater and the sample is embedded and covered with a coverslip and thus it cannot be crosschecked with other techniques.

#### 4. Results and discussion

We start by showing our results using simulated images. The use of simulations allows to test the algorithm with a known ground truth for comparison prior its use in experimental data. This allows to gain confidence in the performance of the fit and to quantify the improvement. The simulated images are produced as follows: first, a high resolution (HR), small pixel image is produced with the target object (1024 x 1024 pixels). The image is then convolved with the instrument PSF and then it is sampled, obtaining a large pixel, low resolution image (32 x 32 pixels). Finally, background and noise is added using realistic values and assuming Poisson statistics. The SUPPOSE Edge algorithm only receives the low resolution, noisy image and the PSF width in units of pixels.

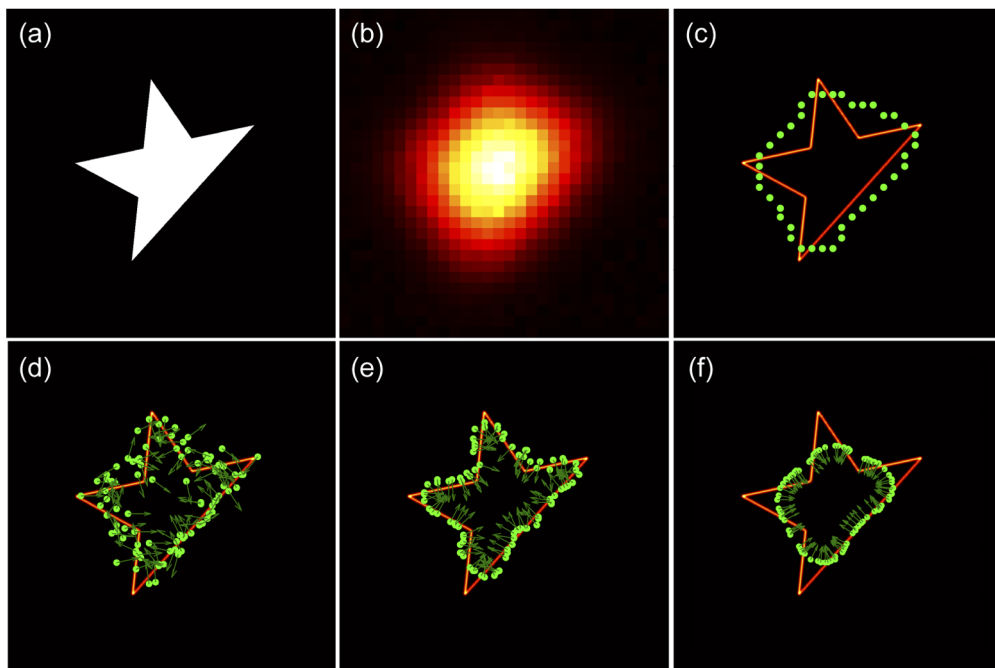
In Fig. 1 we show a comparison between the target object (left, a pizza without a slice or a Pacman) and its simulated measurement (middle). The PSF used for this simulation was Gaussian, with a standard deviation value of  $\sigma = 3.125$  px. The radius of the main circle is  $2.8\sigma$ . We can see that the HR details in Pacman's mouth are completely lost in the LR image. At the right, only the edge of the target image is plotted and compared with two different edge detection algorithms: a standard Laplacian of Gaussian algorithm (LOG edge) and our method (SUPPOSE Edge). We can see that the LOG edge domain is limited to the image pixels and only achieves to reconstruct the basic information in the LR image, without increasing the detail. On the other hand the SUPPOSE Edge algorithm achieves a much greater reconstruction of Pacman's mouth, and a sub pixel detection of the edge all around the shape.

In Fig. 2 we show a comparison of the suppose algorithm for an asymmetric star shape. Picture (a) shows the HR target object. Picture (b) shows the LR simulated measurement. Image (c) shows a Laplacian of Gaussian standard edge detection. We can see that for this particular image all the shape of the star is lost in the edge detection. Pictures (d), (e) and (f) show three different



**Fig. 1.** Edge super resolution by the Suppose Edge algorithm. (a) Target object (Pacman). (b) Simulated measurement image (input of the algorithm). (c) Green: Suppose Edge reconstruction. Blue: Laplacian of gaussian edge reconstruction.

runs of the Suppose Edge algorithm with increasing weight in the regularization term of the fitness ( $\lambda$  in Eq. (7)). The computed normal vector for each source is shown with a green arrow. Taking run (e) as a reference,  $\lambda$  is 100 times smaller in run (d) and 100 times larger in run (f).



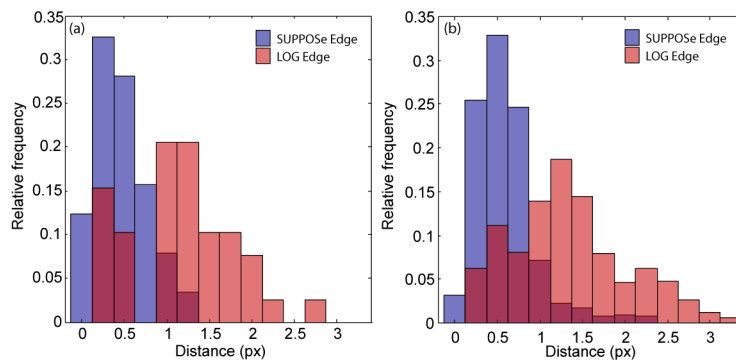
**Fig. 2.** Suppose Edge reconstruction of an asymmetric star. (a) High resolution target object (star). (b) Simulated measurement image (input of the algorithm). (c) Laplacian of gaussian edge reconstruction. (d) to (f): Suppose edge reconstruction for three different regularization weights. (d) is 100 times smaller than (e) and (f) is 100 times larger than (e). Small arrows in dark green show the normal vector for each source.

We can see that as  $\lambda$  increases resolution is lost in the reconstruction of the sharp edges in the star points, as expected for such regularization. Despite this rounding, the normal vectors are much better ordered in run (e) than in run (d), where sources are poorly placed and we can find sources in the middle of the star. In spite of this small loss in resolution, run (e) shows to

be the best and shows a significant increase in the detail of the reconstruction compared with the standard LOG edge shown in image (c). We can see in this last image that the LOG edge algorithm fails in reconstructing the most basic star features, which are present in SUPPOSE Edge reconstruction.

Notice that the increase in  $\lambda$  needed to have a noticeable change in the solution of the algorithm is large (a factor 100). The solution is not very sensitive to this parameter for smaller changes. Experimentally, we have found that the regularization term should weight around 1% in the total fitness in order to achieve good convergence. Of course, this is shape dependent but, as the run is not highly sensitive to the actual value, we can use Eq. (8) to compute an estimation for  $\lambda$  that can be modified in a second run if the result is not satisfactory.

To continue our analysis with the simulated images we can discuss the accuracy of our detection. To characterize this, we developed two complementary metrics that are shown in Fig. 3. In Fig. 3 (a) we plotted an histogram of the distances from each point in the HR edge (for a fine grid) to the nearest virtual source (for example, in Fig. 2 (e), the minimum distance between each point in the orange edge to the nearest green node). In Fig. 3 (b) we have the complementary histogram, i.e. the distance from each virtual source to the real edge. Together they give a measure of how close is our detection to the actual edge and allow a more quantitative comparison with the standard LOG edge algorithm. As an example of how this metrics work we can notice that, if a part of the edge is not precisely detected, it will be noticed in the first histogram (but not necessarily in the second). In the same manner, if a source is misplaced (for example, far away from the real edge) it will be noticed on the second histogram but not necessarily in the first.



**Fig. 3.** Metrics of the detection's precision. Comparison between LOG edge and SUPPOSE Edge algorithms for the asymmetric star (Fig. 2 (c) and (e)). (a) Histogram for the distance from each point in the real edge (ground truth) to the nearest virtual source (detection). (b) Histogram for the distance from each virtual source (detection) to the nearest edge (ground truth).

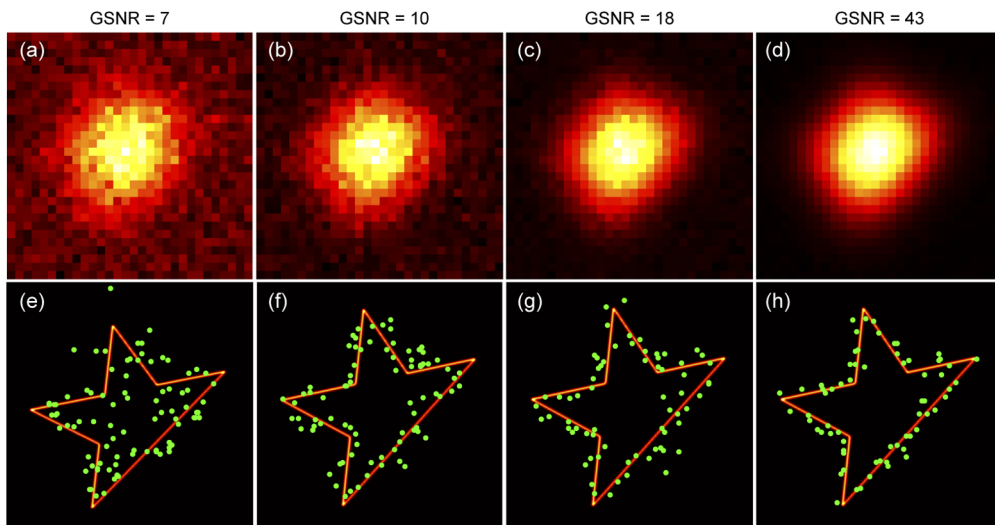
From the first metric (Fig. 3 (a)) we can obtain the mean distance from the edge to the nearest sources. For SUPPOSE Edge, this value is 0.46 px, which is equivalent to  $0.14 \sigma$ . For the LOG edge algorithm that distance is 1.2 px which is equivalent to  $0.4 \sigma$ . We can see that the Suppose Edge method achieved an accuracy of approximately  $\sigma/7$  which is almost three times better than the LOG edge algorithm. For the same metric, the maximum distance in the Suppose Edge is of 1.3 px (the lower left point of the star in Fig. 2 (e)) while the maximum distance for LOG algorithm is 2.7 px (the upper right valley in Fig. 2 (c)) which is over a factor 2 improvement. Also the precision (measured as the standard deviation of the distances) is a factor 2 smaller for SUPPOSE edge and the histogram shows a more consistent behaviour. Similar results are observed for the second metric (Fig. 3 (b)), where we obtain again an enhancement on both accuracy and precision of over a factor 2. The same comparison was performed using Canny's algorithm [5] instead of

LOG, achieving even better improvements. In both cases, the hyper-parameters of the methods were evaluated for different values and we are always comparing with the best results obtained.

Finally, in Fig. 4 we show the performance of the algorithm for images with different signal to noise ratios. To do so, we generated a series of images of the same object but with different maximum intensities. As the noise follows Poisson's statistics, this results in a worse signal to noise ratio as intensity decreases. On the other hand, the background (i.e. the number of counts in the surroundings of the object) was kept constants at 100 counts. This means that the intensity difference between the object and the background is also decreasing (worst case scenario). As this difference diminishes, the edge will be more difficult to identify. Thus we choose to describe the quality of each image by computing a measure of the maximum intensity variation relative to the noise. For this purpose we define the Gradient Signal to Noise Ratio as

$$\text{GSNR} = \frac{\Delta S_{max}}{\sqrt{2} \sigma_{max}}, \quad (9)$$

where  $\Delta S_{max}$  is the difference in counts between the maximum intensity (at the center of the object) and the background and  $\sigma_{max}$  is the noise at the pixel of maximum signal. This is basically a rough estimation of the HR maximum gradient divided by the actual gradient's noise that can be easily computed for most images.



**Fig. 4.** Performance of the algorithm for different noise levels. The first row (images a to d) corresponds to the same object (Fig. 2(a)) for different signal to noise ratios (background is at a constant value of 100 counts). Each image in the second row (figures e to h) corresponds to the SUPPOSE Edge solution of the image above it. The HR contour is shown in orange for comparison. The title on top of each column represents a measure of the signal to noise ratio in the image's gradient (GSNR).

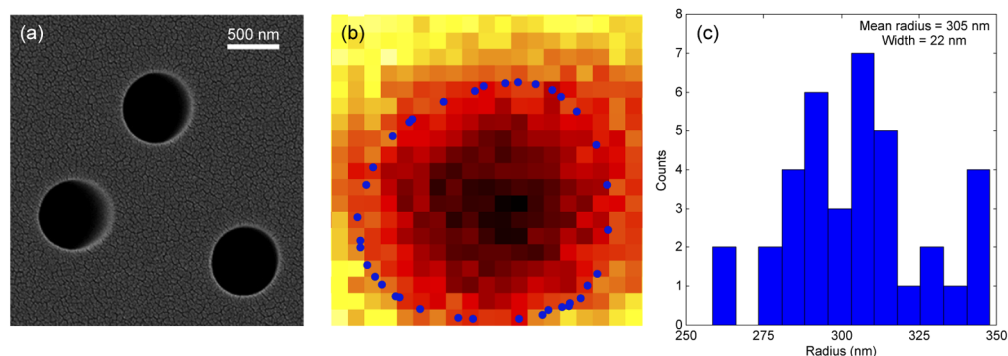
For a very noisy image, like the one shown in Fig. 4 (a) (GSNR = 7), we can see that the reconstruction (Fig. 4 (e)) is poor. As we increase the intensity, we can see that the solution starts to get closer to the real contour. At GSNR = 10, the solution is a better approximation of the object (Fig. 4 (f)) compared to (e)) but some oscillations are observed around the true edge. By increasing the intensity we found that this oscillatory behaviour disappears at GSNR = 18. At even higher intensities, the solution becomes much closer to the real edge. If we use the metrics developed in Fig. 3 to quantify these results, we get that the distribution of the distances from each virtual source to the edge (written in the form mean  $\pm$  standard deviation, in units of the



PSF radius  $\sigma$ ) is:  $(0.3 \pm 0.2)\sigma$  for solution (e),  $(0.19 \pm 0.15)\sigma$  for solution (f),  $(0.19 \pm 0.17)\sigma$  for solution (g) and  $(0.12 \pm 0.08)\sigma$  for solution (h). Besides (f) and (g) having an equal value (the oscillations in (f) fall both inside and outside the real contour), we see that this metric also shows an enhancement for increasing GSNR.

To be able to process the experimental images, a detailed characterization of the instrument PSF must be accomplished. This one-time, auxiliary measurement was done by capturing images of fluorescent beads which are much smaller than the instrumental resolution (in this case 100 nm in diameter). Several images were used to overcome poor signal to noise ratio. A Gaussian model was used to fit the PSF. For the two photon excitation microscope PSF, this process resulted in a standard deviation value of  $\sigma = 189$  nm, which corresponds to an optical resolution defined as  $2\sigma = 378$  nm. For the widefield epifluorescence microscope we obtained an optical resolution of  $1.18 \mu\text{m}$  using the 10X 0.3 NA microscope objective. A detailed description of this method for the calibration of the PSF can be found in [19].

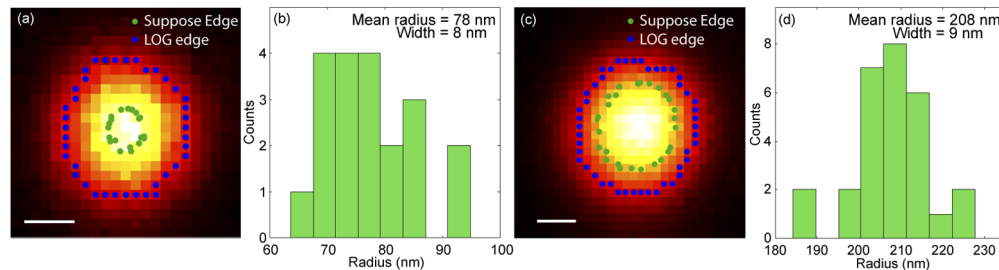
After characterizing the microscope PSF, we can start using the SUPPOSE Edge algorithm. In Fig. 5 we can see the results of using this method to detect the diameter of the pores of a porous membrane. In figure (a) we can see a SEM image of the sample (polycarbonate filter), showing a good regularity in the 600 nm diameter pores. In figure (b) we show the 2P microscopy image of a single pore (pixel size is 40 nm). We can see that the image is noisy (GSNR = 13). In blue dots we show the SUPPOSE Edge estimation for its border. To test the method, we performed a histogram of the radius of each source (blue dots) to the center of mass of the group. We obtained a distribution with a mean value of 305 nm and a standard deviation of 22 nm. This means that the measured diameter of the pore is  $(610 \pm 44)$  nm which is in agreement with the SEM images and has a precision of  $0.23 \sigma$ , which is over 8 times better than the instrumental resolution.



**Fig. 5.** SUPPOSE edge detection applied to pores of known diameter 600 nm. (a) SEM images of the membrane's pores, showing high regularity in size. (b) Color gradient: two photon excitation microscopy images of the pores (autofluorescence). Blue dots: virtual sources for the SUPPOSE edge detection. (c) Histogram of the distance of each source to the center of mass of the group.

In Fig. 6 we can see two photon excitation microscopy images for fluorescent beads of two different sizes (200 nm and 500 nm nominal diameter, picture (a) and (c) respectively). As the sizes of the particles are in the order and below the microscope resolution, both images look really similar. If we use a LOG edge detection algorithm (blue dots), we get a result which is almost identical for both sizes (estimated diameter  $(534 \pm 40)$  nm and  $(629 \pm 29)$  nm respectively). In green dots we show the result obtained with the Suppose Edge algorithm, where the difference with the standard method is evident. In pictures (b) and (d) we show histograms of the radii of each source in the same way we did in Fig. 5. Here, the dispersion in the sizes of the beads is much greater, but we obtain diameters of  $(156 \pm 16)$  nm and  $(416 \pm 18)$  nm. The fact that

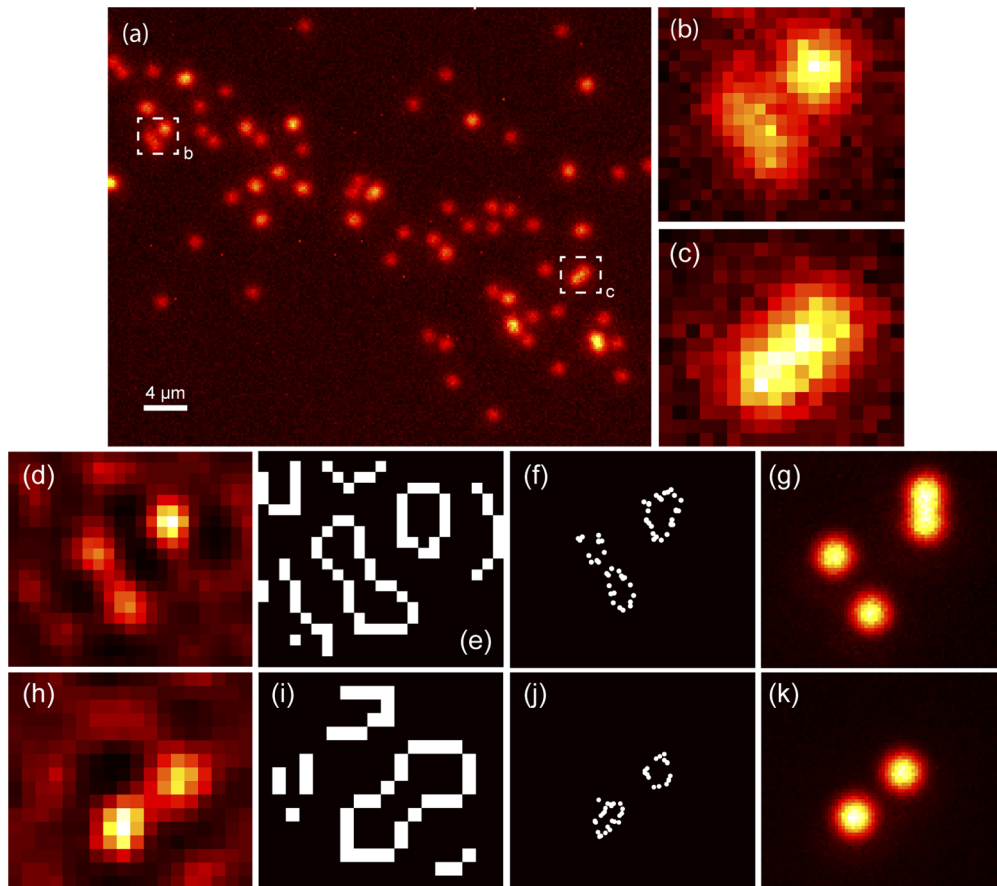
the diameters are smaller than the nominal values can be due to a loss (or bleaching) of the molecules near the surface (in both cases the measured value is 20% smaller than expected). Although the precise size of the particles is unknown, the SUPPOSE Edge algorithm gives a superior reconstruction compared to the standard LOG edge detection algorithm (which returns a diameter 2.7 times greater than the nominal for the small bead). The precision in the diameter determination is better than  $0.1\sigma$ , showing the power of the method in detecting sub-PSF shapes with high precision.



**Fig. 6.** Two photon microscopy image of fluorescent beads. (a) 200 nm nominal diameter bead. LOG edge detection algorithm identifies a diameter of  $(534 \pm 40)nm$  and Suppose Edge  $(156 \pm 16)nm$  (c) 500 nm nominal diameter bead. LOG edge detection algorithm identifies a diameter of  $(629 \pm 29)nm$  and Suppose Edge  $(416 \pm 18)nm$ . (b) and (d) display the histograms corresponding to the radius distribution of each source for the image at its left. Scale bars are 200 nm long.

In Fig. 7 we show the ability of the SUPPOSE Edge algorithm to resolve the edges of pairs of particles that are closer than the instrumental resolution. As a proof of this in experimental images we measured 500 nm nominal diameter fluorescent beads using the 10X 0.3 NA and the 40X 0.95 NA microscope objectives in our widefield epifluorescence microscope. This allows to use the edge detection algorithm in the 10X image and to use the 40X image as the ground truth. The 10X objective PSF was measured as described before, resulting in an optical resolution of  $1.18 \mu m$ . In Fig. 7(a) we show a 10X widefield image of the fluorescent beads, where we selected two different regions of interest (b and c) that exhibit bead clusters. The high resolution images of the same regions (acquired with the 40X 0.95 NA objective) are shown in figures (g) and (k) respectively. In both cases we can identify two beads that are indistinguishable in the 10X image whose centers are  $1 \mu m$  apart. This means that the resulting gaps between the inner edges of the beads are around 500 nm, well below instrumental resolution. Additionally, the region of interest (b) and its high resolution counterpart (g) show a bead cluster of probably three beads that are in full contact.

We have already seen in Fig. 6 that the standard edge detection algorithms fail to reproduce sub-PSF edges. In this image we first applied a Richardson Lucy deconvolution algorithm which is fed with the same PSF information that SUPPOSE Edge uses. In figure (d) and (h) we show the result of 50 iterations of the Richardson Lucy algorithm, where we can see that the resolution is enhanced but many artifacts are present. The number of iterations was chosen to achieve the best edge detection performance. If more iterations are performed, the importance and number of the artifacts grows even more, losing all useful information on the image. Also the use of a damping factor (normally used in this type of algorithms to suppress artifacts) was tested and discarded, as its use reduces the resolution. Then we applied a LOG edge algorithm on the RL deconvolved image, obtaining the results shown in (e) and (i). We can see that a lot of false positives are present in the solution, meaning that the algorithm is detecting edges that are not present in the image and are a consequence of the previous deconvolution process. Here it is important to notice that the combination of Richardson Lucy with LOG edge detection has at



**Fig. 7.** SUPPOSE Edge reconstruction of 500 nm nominal diameter fluorescent beads acquired using a 10X 0.3 NA microscope objective. As the ground truth a 40X 0.95 NA image of the same beads is shown. The SUPPOSE Edge solution is compared to the LOG method applied to the output of a Richardson-Lucy deconvolution algorithm (RL). Figures d to g correspond region b, as figures h to j correspond to input image c. (a) Widefield image of 500 nm nominal diameter fluorescent beads. Two regions of interest are marked and shown independently on b and c with more detail. (d) RL solution for input b. (e) LOG detection applied to the RL output. (f) SUPPOSE Edge solution. (g) HR resolution image of the same region used as reference for comparison (40X image of region b). (h) Result of RL applied to image c. (i) LOG detection applied to the RL output. (j) SUPPOSE Edge solution. (k) HR resolution image of the same region used as reference for comparison (40X image of region c).

least 4 hyper-parameters (number of iterations and damping factor of the RL filter, and threshold and sigma for the LOG detection) as opposed to the two needed for SUPPOSE edge (number of virtual sources and weight of the regularization). For the LOG algorithm, if the threshold is increased less of the artifact edges are detected, but eventually the edges of the beads become undetected. The value of this threshold was chosen in a way that the contour of the beads was closed (i.e. there are no gaps in the detection of the edges). If the hyper-parameter sigma is diminished, the resolution slightly increases but the presence of false negatives grows even more. No combination of these 4 parameters yielded a structure with the beads separated. Additionally, despite the incorporation of the deconvolution algorithm, the LOG detection resolution is still limited by the pixel size.

The results of the SUPPOSE Edge detection is shown in Figs. 7 (f) and (j). First of all we note that the algorithm is artifact free, resulting in a detection with no false positives. Additionally, we see that both pairs of beads are separated and identified as different contours. The cluster in the upper right of figure (f) is still identified as a single contour, as it is seen in its high resolution counterpart (g). The GSNR was estimated in 8.6 for roi (b) and 12 for roi (c). By comparing with the results of the simulations shown in Fig. 4, we can see that both regions have a similar GSNR to the image shown in Figs. 4 (b), and then some distortion in the shape of the contours is expected. Nevertheless, if we estimate the distance between the beads using the center of each contour and we compare this value with the information provided by the HR image we can see that the difference is of about 5% for roi (b) and around 3% for roi (c).

## 5. Conclusions

We presented a novel algorithm to detect borders in microscopy images with a resolution better than the instrumental limit. The method is based on the SUPPOSE algorithm, whose effectiveness was already proved in achieving super resolution in widefield fluorescence images. Our method proposes to approximate the gradient of the image with a superposition of virtual sources of equal intensity. After this the problem becomes to find the best positions of the virtual sources that, after convolving with the instrument PSF, best fits the image gradient. For this purpose, a regularization term is proposed to allow the algorithm to converge to a proper solution. The minimization problem is solved by means of a genetic algorithm. With this method, the only thing needed to achieve super resolution in edge detection is a good characterization of the microscope's point spread function. The method was tested in simulated images and in experimental data, obtaining promising results in both cases. The benefits of the method are greater when the objects to detect are smaller than the diffraction limit, where the standard algorithms drastically fail, but an increase in accuracy and precision of over a factor two was also found in larger structures with small details.

In the simulations the method retrieves details which are not obvious in the blurred image and are not found with the standard Laplacian of Gaussian edge detection. The importance of the regularization term was tested, showing that it is necessary for proper convergence but the result is not very sensitive to its precise value.

In the experimental images, the method successfully measured the diameter of the pores of a membrane and of fluorescent beads with sizes smaller than the instrument PSF. Also, the algorithm shows capable of separating structures that are not resolved (appear merged) in the low resolution image.

## Appendix A: Detailed description of the algorithm

In this appendix we will describe in more detail the algorithm to address the minimization problem in SUPPOSE Edge. The algorithm is similar to the used for the SUPPOSE deconvolution algorithm [19], but some changes need to be addressed to work with the increased complexity of the SUPPOSE edge method. As mentioned in the section 2 the goal of the genetic algorithm is to

minimize the fitness with respect to the positions of the virtual sources. We choose to program our edge detection software using MATLAB. The global picture of the algorithm goes as follows: first, an initial guess of the edge is used to generate a first population of 100 solutions. We will call each one of these candidate solutions as an individual and the group of 100 individuals is called a population. The virtual sources inside an individual are always ordered following the edge. Each time a population is built, the fitness function is computed for each individual (see fitness computation below) so that they can be placed inside the population from better to worse (lower to higher fitness). Here the genetic algorithm (GA) starts. Each iteration of the GA produces a new population from the last one. We call each iteration a generation, and we number the generations starting from 0, corresponding to the initial guess. In each iteration, 100 new individuals are produced from the last population and ordered by its fitness value in a new population. An elite of the 5 best individuals is preserved unchanged in each population in order to ensure that, in each new generation, the fitness of the best individual is monotonously decreasing (i.e. cannot get worse). After a certain number of iterations the GA stops and the best individual in the last population is declared as the solution of the algorithm. A stop criterion based on the ideal fitness is also used (see A4. Noise model and stop criterion).

### A1. Generation of the initial population

The initial population must start with a guess of the edge. This guess might be the output of a standard edge detection algorithm, but we found that the best result is produced by using an algorithm that computes level curves. In that manner, we can check level curves at different levels, compute the fitness value for each one and use the best one as the initial guess. For this purpose we use the `countourc` function of the MATLAB environment. Once the best level is selected, the number of virtual sources to use has to be decided. This is achieved by computing the length of the guessed edge and by placing a fixed number of virtual sources for each unit of  $\sigma$ . In all our runs this value was set to 6 sources per unit  $\sigma$  which seems an appropriate value for all the images tested.

After this first individual is produced, an initial population should be generated from it. This is accomplished by taking the individual and perform random mutations on its coordinates. This means to sum a normally distributed random number to each one. The amplitude of those *kicks* should be small, around 1% of the sigma value.

### A2. Fitness computation

The central part of the algorithm is how the fitness is computed. Here is where the presented theory for the SUPPOSE edge method is implemented. For this, we start by computing the gradient of the image using a second order finite differences algorithm. Notice that no smoothing or additional processing is used in the computation of the gradient. After this, we need to built  $\nabla R$  for each individual as proposed in Eq. (5). This starts by computing the normal vectors for each source as described before. The value of  $\alpha$  is chosen as the one that minimizes the  $\chi^2$  given the positions of the sources. As the expression is linear in  $\alpha$ , this minimization is analytical, resulting in

$$\alpha = \frac{\sum_{i=1}^{N_x} \sum_{j=1}^{N_y} \left( (\nabla S)_{ij} \cdot \sum_{k=1}^{N_s} I_{ij}^k \hat{n}_k \right)}{\sum_{i=1}^{N_x} \sum_{j=1}^{N_y} \left\| \sum_{k=1}^{N_s} I_{ij}^k \hat{n}_k \right\|^2} \quad (10)$$

where  $\| \cdot \|$  denotes the standard norm in  $\mathcal{R}^2$ . After computing  $\alpha$  we get the value for the fitness using Eqs. (6) and (7).

### A3. Genetic operators

The genetic operators are used to produce a new population from the current one. As we discussed in section 2, these operators depend on a number of parameters that affect mainly the speed of the convergence and are fixed and equal in all the images. The selection and the cross algorithms are the ones used for SUPPOSE [19]. The first is based on rescaling the fitness and by repeating the individuals based on its fitness value. The second consists in selecting randomly pairs of individuals and exchanging half of their coordinates (once again, randomly chosen). The mutation algorithm consists of two stages which contribute to the coarse and the fine movements of the virtual sources. The coarse mutations correspond to rigidly shifting the position of a segment compound of a few sources. The position and length of the segment and the size and direction of the displacement are randomly chosen (segments are built using 3 to 6 virtual sources and moved a distance generated by two normally distributed random numbers of amplitude equal to 2.5% of the PSF width,  $\sigma$ ). The fine mutations are independent shifts of the individual sources. The particles to be shifted and the distances are chosen randomly (20% of the virtual sources and 0.1% of  $\sigma$  is used for the displacement amplitude). Notice that these shifts are much smaller than the ones corresponding to segments, and are designed to explore the fine details of the edges.

### A4. Noise model and stop criterion

The last step is to choose a stop criterion for our algorithm. Here we have two different possibilities: after a certain number of iterations, that depends on how close the initial guess was to the solution, it is observed that the fitness stops improving (or its improvement is negligible over several iterations). If this is the case, there is no sense in computing further iterations and we say that the genetic algorithm has converged. The second possibility is that the fitness function reaches a value that is below the ideal fitness (if available). If this is the case, further iterations will be adjusting image noise rather than image structure and because of that the algorithm should be stopped. However, the estimation of the ideal fitness is not always available as it relies upon certain assumptions on our noise model.

The presentation given in 1 has the following origin. Let's define the ideal image expected from  $R$  as  $S_{id} = R * I$ . The measured image will differ from  $S_{id}$  by a residue

$$r = S - S_{id}, \quad (11)$$

where  $r$  is a stochastic variable that accounts for the different sources of noise. It includes additive noise from dark counts, readout noise and other eventual electronic noise contributions (depending on the sensor used). It also includes the noise arising from the photon detection that will be a stochastic function determined by the parameter  $S_{id}$  and will typically have a Poisson distribution with average value  $S_{id}$ . It is a reasonable assumption that the noise for different pixels is not correlated and that the additive noise contribution is the same for all pixels. Hence the average value for the residue  $r$  will contribute to a constant background and defining

$$\eta(x, y) = r(x, y) - \langle r(x, y) \rangle \quad (12)$$

the expression given in Eq. 1 is recovered except for a constant background that makes no contribution to the gradient of the image used for the edge detection. From this also follows that the variance of  $\eta$  is equal to the variance of  $S$  (and the same for  $\eta'$  and  $\nabla S$ ). We can use these results to deduce Eq. (8). By using Eq. (6) in the ideal case where our approximation exactly recovers  $S_{id}$  we obtain

$$\langle \chi_{ideal}^2 \rangle = \langle \|\eta'\|^2 \rangle = \sum_{i=1}^{N_x} \sum_{j=1}^{N_y} \sum_{d=1}^2 \langle (\eta'_{ij} \cdot \hat{e}_d)^2 \rangle. \quad (13)$$

As  $\text{Var}(\eta') = \text{Var}(\nabla S)$  we get

$$\langle \chi_{ideal}^2 \rangle = \sum_{i=1}^{N_x} \sum_{j=1}^{N_y} \left( \text{Var} \left( \frac{\partial S_{ij}}{\partial x} \right) + \text{Var} \left( \frac{\partial S_{ij}}{\partial y} \right) \right) = 4 \sum_{i=1}^{N_x} \sum_{j=1}^{N_y} \text{Var}(S_{ij}) \quad (14)$$

where, for the second equality, we used that the noise is independent for each pixel, so the variance of the partial derivatives of  $S$  are equal to twice the variance of  $S$ .

#### A5. Solution and final considerations

After any of the stop criteria occur, the best individual of the final population is named the solution and the GA is completed. For our images this happens in around  $10^4$  generations. The computation time of the algorithm is linearly dependent on the number of sources and the number of pixels on the image. For the sizes of  $32 \times 32$  px and around 50 sources the algorithm takes 40 min to achieve  $10^4$  iterations on an Intel i7-9700 processor. Nevertheless, the algorithm would benefit greatly from parallel computing, as the most time consuming step is the fitness calculation, and its computation is independent for each individual. Increase in speeds from 10 to 80 times were registered for the SUPPOSE algorithm when computed in a GPU architecture and, due its similarity, the same behaviour is expected for the SUPPOSE edge method. With that experience, we expect convergence in the minute order.

#### Funding

Air Force Office of Scientific Research (FA9550-18-1-0470 P00001); Universidad de Buenos Aires (ubacyt2018 20020170100137BA); Agencia Nacional de Promoción Científica y Tecnológica (PICT 2015-1523).

#### Disclosures

The authors declare that there are no conflicts of interest related to this article.

#### References

1. W. K. Pratt, *Introduction to digital image processing* (CRC, 2013).
2. R. O. Duda and P. E. Hart, *Pattern classification and scene analysis*, vol. 3 (Wiley, 1973).
3. D. Man and A. Vision, "A computational investigation into the human representation and processing of visual information," (1982).
4. J. Kaur, S. Agrawal, and R. Vig, "A comparative analysis of thresholding and edge detection segmentation techniques," *IJCA* **39**(15), 29–34 (2012).
5. J. Canny, "A computational approach to edge detection," *IEEE Trans. Pattern Anal. Mach. Intell.* **PAMI-8**(6), 679–698 (1986).
6. D. Marr and E. Hildreth, "Theory of edge detection," *Proc. R. Soc. Lond. B* **207**(1167), 187–217 (1980).
7. M. Gudmundsson, E. A. El-Kwae, and M. R. Kabuka, "Edge detection in medical images using a genetic algorithm," *IEEE Trans. Med. Imaging* **17**(3), 469–474 (1998).
8. S. M. Bhandarkar, Y. Zhang, and W. D. Potter, "An edge detection technique using genetic algorithm-based optimization," *Pattern Recognit.* **27**(9), 1159–1180 (1994).
9. L. Alexopoulos, G. Erickson, and F. Guilak, "A method for quantifying cell size from differential interference contrast images: validation and application to osmotically stressed chondrocytes," *J. Microsc.* **205**(2), 125–135 (2002).
10. J. Starkey and A. Samantaray, "Edge detection in petrographic images," *J. Microsc.* **172**(3), 263–266 (1993).
11. T. Smith Jr, W. Marks, G. Lange, W. Sheriff Jr, and E. Neale, "Edge detection in images using marr-hildreth filtering techniques," *J. Neurosci. Methods* **26**(1), 75–81 (1988).
12. Y. Al-Kofahi, W. Lassoued, W. Lee, and B. Roysam, "Improved automatic detection and segmentation of cell nuclei in histopathology images," *IEEE Trans. Biomed. Eng.* **57**(4), 841–852 (2010).
13. I. Overington and P. Greenway, "Practical first-difference edge detection with subpixel accuracy," *Image Vis. Comput.* **5**(3), 217–224 (1987).
14. D. Woolford, B. Hankamer, and G. Ericksson, "The laplacian of gaussian and arbitrary z-crossings approach applied to automated single particle reconstruction," *J. Struct. Biol.* **159**(1), 122–134 (2007).
15. K. K. Treloar and M. J. Simpson, "Sensitivity of edge detection methods for quantifying cell migration assays," *PLoS One* **8**(6), e67389 (2013).

16. Y.-W. Tai, S. Liu, M. S. Brown, and S. Lin, "Super resolution using edge prior and single image detail synthesis," in *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition* (IEEE, 2010), pp. 2400–2407.
17. J. Sun, Z. Xu, and H.-Y. Shum, "Image super-resolution using gradient profile prior," in *2008 IEEE Conference on Computer Vision and Pattern Recognition* (IEEE, 2008), pp. 1–8.
18. C. G. Galbraith and J. A. Galbraith, "Super-resolution microscopy at a glance," *J. Cell Sci.* **124**(10), 1607–1611 (2011).
19. M. Toscani, S. Martínez, and O. E. Martínez, "Single image deconvolution with super-resolution using the suppose algorithm," in *Single Molecule Spectroscopy and Superresolution Imaging XII*, vol. 10884 (International Society for Optics and Photonics, 2019), p. 1088415.