Subpixel super resolution algorithm for *PALM microscopy*

M. Gostiaux Gabriel
M. Gabriel Gostiaux, Master of Science student, Institute of Optics,
Palaiseau, 91 120, France
gabriel.gostiaux@institutoptique.fr

https://github.com/GabrielGst?tab=repositories

Abstract: In this study, we introduce a computationally efficient approach for generating super-resolution frames in Photoactivated Localization Microscopy (PALM) by accelerating the processing pipeline. The method is designed to optimize both the localization accuracy and processing speed, making it particularly suitable for medium-large datasets.
Our results demonstrate that this streamlined approach achieves robust performance in generating super-resolution PALM frames with significant reductions in processing time compared to the basic method, making it a good start for high-throughput super-resolution imaging applications.
Keywords: PALM Microscopy, sub-pixel superresolution, python.
OCIS codes: (000.0000) General.

References and links

1. K. Gallo and G. Assanto, “All-optical diode based on second-harmonic generation in an asymmetric waveguide,” J. Opt. Soc. Am. B **16,** 267–269 (1999).

1. Introduction

PALM (Photoactivated Localization Microscopy) is a high-resolution imaging technique that surpasses the diffraction limit. It relies on the isolation of emitters selectively tagged with a fluorescent molecule, and the fitting of their Point Spread Function (PSF). The fluorescent molecules emit light when stimulated by a laser. This approach enables the precise localization of isolated emitters with an accuracy beyond the Rayleigh criterion, thus providing ”super-resolution” down to around 10 nm. This technique, developed by Eric Betzig and his team in 2006, was awarded the Nobel Prize in Chemistry in 2014. It is of great importance due to its ability to enhance the resolution of existing images. Because this method is implemented with noisy frames, one should rely on estimation and detection theory to retrieve the precise coordinates of the emitters.

The workflow begins with a rigorous noise analysis of acquired frames, enabling the subsequent likelyhood computation steps to operate with enhanced precision. Our algorithm leverages a matched filter for likelihood calculation across frames, a critical step in improving computation time.

To accelerate the localization process, we identify initial guesses for molecule positions using a simple peak detection strategy via the numpy.max function. This initial estimation significantly reduces the computational complexity at the cost of sacrificing accuracy. For precise sub-pixel localization, we then employ the Maximum Likelihood Estimator (MLE) method, with optimization executed through the scipy.fmin function. This choice provides a computational efficiency and high-resolution localization, as it can then swiftly converges to optimal coordinates.

2. Precision analysis

For simplicity reasons and without any loss of generality, we consider the location from a 1D point of view. Let x be the spatial coordinate, θ the position of an emitter. It is then assumed that the signal given by this single emitter on the image plan can be written as $s(x,\theta) = ar(x,\theta) + b$ where r is a Gaussian function centered on θ so that

$$r(x,\theta) = \frac{1}{\sqrt{2\pi}\sigma_r} \exp\left[-\frac{(x-\theta)^2}{2\sigma_r^2}\right] \tag{1}$$

where σ_r is expressed as a function of the full width at half maximum (FWHM) w of the signal on the sensor. In this labwork, b is assumed to be an additive normal white noise of standard deviation σ_b.

P1 We define r_i the integration of $r(x,\theta)$ over the pixel i with $i \in [0, N-1]$

$$r_i = \int_{i\Delta x}^{(i+1)\Delta x} r(x,\theta)\mathrm{d}x \tag{2}$$

where Δx is the discretization step. Give the expression of r_i as a function of the parameters θ and w. We remind that

$$\frac{2}{\sqrt{\pi}} \int_0^z e^{-u^2}\,\mathrm{d}u = \mathrm{erf}(z)$$

Let's compute r_i with a integration by substitution :

$$r_i = \int_{i\Delta x}^{(i+1)\Delta x} \frac{1}{\sqrt{2\pi}\sigma_r} \times \exp\left[-\left(\frac{x-\theta}{\sqrt{2}\sigma_r}\right)^2\right] dx \tag{3}$$

$$= \frac{1}{\sqrt{2\pi}\sigma_r} \times \sqrt{2}\sigma_r \int_{u^-}^{u^+} e^{-u^2} du \tag{4}$$

$$= \frac{1}{\sqrt{\pi}} \left[\int_0^{u^+} e^{-u^2} du - \int_0^{u^-} e^{-u^2} du\right] \tag{5}$$

$$r_i = \frac{1}{2}\left[\mathrm{erf}\left[\frac{(i+1)\Delta x - \theta}{\sqrt{2}\sigma_r}\right] - \mathrm{erf}\left[\frac{i\Delta x - \theta}{\sqrt{2}\sigma_r}\right]\right] \tag{6}$$

$$P\left(s_{;}\right) = \frac{1}{\sqrt{2\pi}\sigma_b} \exp\left[-\frac{(s_i - ar_i)^2}{2\sigma_b^2}\right] \tag{7}$$

$$l_\theta = -N \times ln\left(\sqrt{2\pi}\sigma_b\right) - \frac{1}{2\sigma_b^2} \sum_{s_i \in \Omega} (s_i - ar_i)^2 \tag{8}$$

$$\frac{\partial l_\theta}{\partial \theta} = \frac{1}{\sigma_b^2} \sum_{s_i \in \Omega} (s_i - ar_i) \times a \times \frac{\partial r_i}{\partial \theta} \tag{9}$$

$$\frac{\partial r_i}{\partial \theta} = -\frac{1}{\sqrt{2\pi}\sigma_r} \times \left[\exp - \left(\frac{(i+1)\Delta x - \theta}{\sqrt{2}\sigma_r} \right)^2 - \exp - \left(\frac{i\Delta x - \theta}{\sqrt{2}\sigma_r} \right)^2 \right] \tag{10}$$

$$\frac{\partial^2 l_\theta}{\partial^2 \theta} = -\frac{1}{\sigma_b^2} \sum_{s_i \in \Omega} \left(-a \left(\frac{\partial c_i}{\partial \theta} \right)^2 - (s_i - r_i) \frac{\partial^2 r_i}{\partial^2 \theta} \right) \tag{11}$$

$$\left\langle \frac{\partial^2 l_\theta}{\partial^2 \theta} \right\rangle = -\frac{1}{\sigma_b^2} \sum_{s_i \in \Omega} \left(-a \left(\frac{\partial r_i}{\partial \theta} \right)^2 - (\langle s_i \rangle - \alpha r_i) \frac{\partial^2 l_\theta}{\partial^2 \theta} \right) \tag{12}$$

$$\left\langle \frac{\partial^2 l_\theta}{\partial^2 \theta} \right\rangle = \frac{a}{\sigma_b^2} \sum_{s_i \in \Omega} \left(\frac{\partial r_i}{\partial \theta} \right)^2 \tag{13}$$

$$CRLB = -\frac{\sigma_b^2}{1} \times \left(\sum_{s_i \in \Omega} \left(\frac{\partial r_i}{\partial \theta} \right)^2 \right)^{-1} \tag{14}$$

$$CRLB = 2\pi \sigma_b^2 \sigma_r^2 \times \left(\sum_{s_i \in \Omega} \left(e^{-u_+^2} - e^{-u_-^2} \right)^2 \right)^{-1} \tag{15}$$

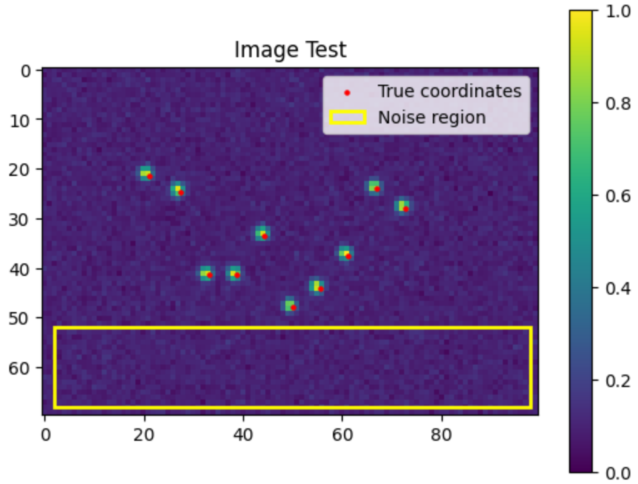
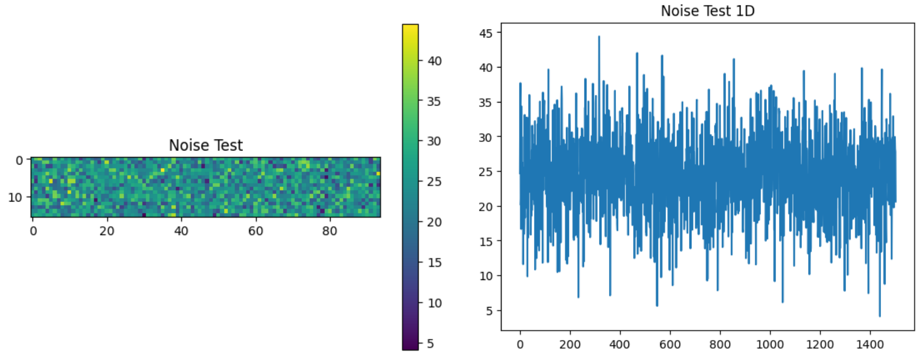


Fig. 1: Test Image

3. Implementation of the algorithm

4. Conclusion

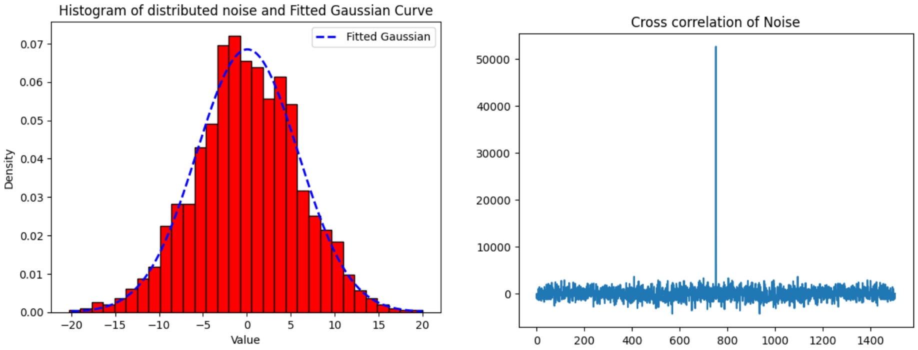
This study presented a specific workflow to optimize computation time for PALM microscopy. The proposed method consist in using a matched filter to compute and save the log-likelyhood arrays from the acquired frames, then use the numpy max function on those to find pixel-wide initial guesses for the scipy fmin optimization function. The time efficiency is then improved by a factor of 10, leading to compute the super-resoluted image presented at the end of section



(a) Noise

(b) Noise 1D

Fig. 2: Noise



(a) Noise histogram

(b) Noise correlation

Fig. 3: Noise properties

2 in 10 minutes, while it was expected to be computed in 100 minutes with the basic fmin MLE optimization method.

The workflow produces a super-resoluted image on which one could read a passage of the famous ”Romeo and Juliette” piece written by M. William Shakespeare, proving the accuracy of the method.

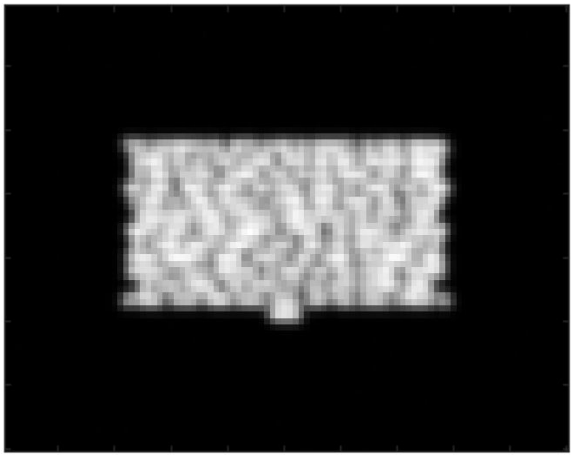


Fig. 4: Image floue

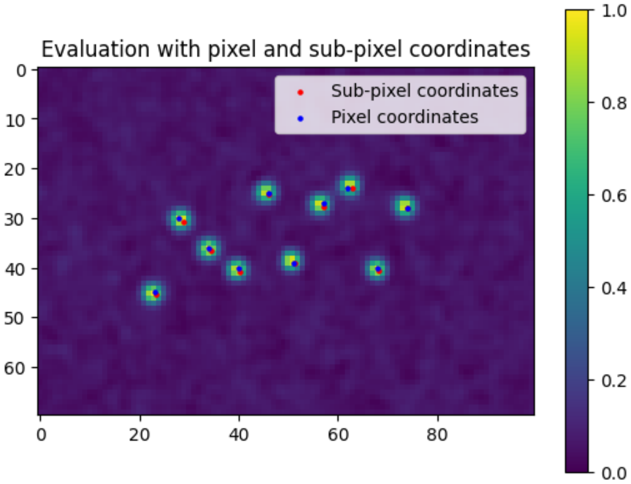
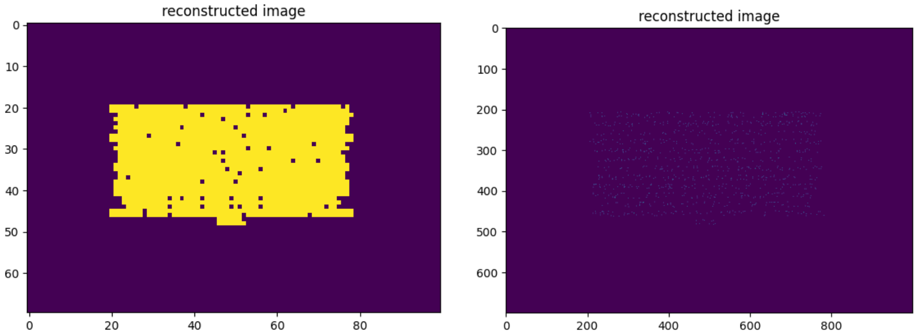


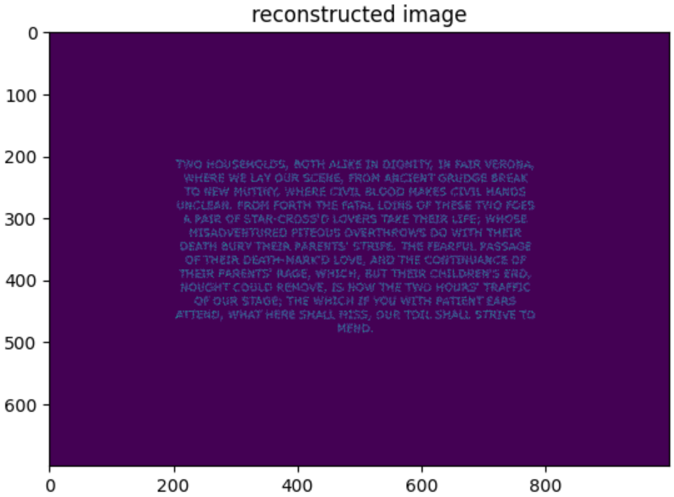
Fig. 5: Evaluation of algorithm acccuracy



(a) Initial guesses

(b) Solving...

Fig. 6: Computing pix and subpix coordinates



TWO HOUSEHOLDS, BOTH ALIKE IN DIGNITY, IN FAIR VERONA,
WHERE WE LAY OUR SCENE, FROM ANCIENT GRUDGE BREAK
TO NEW MUTINY, WHERE CIVIL BLOOD MAKES CIVIL HANDS
UNCLEAN. FROM FORTH THE FATAL LOINS OF THESE TWO FOES
A PAIR OF STAR-CROSS'D LOVERS TAKE THEIR LIFE; WHOSE
MISADVENTURED PITEOUS OVERTHROWS DO WITH THEIR
DEATH BURY THEIR PARENTS' STRIFE. THE FEARFUL PASSAGE
OF THEIR DEATH-MARK'D LOVE, AND THE CONTINUANCE OF
THEIR PARENTS' RAGE, WHICH, BUT THEIR CHILDREN'S END,
NOUGHT COULD REMOVE, IS NOW THE TWO HOURS' TRAFFIC
OF OUR STAGE; THE WHICH IF YOU WITH PATIENT EARS
ATTEND, WHAT HERE SHALL MISS, OUR TOIL SHALL STRIVE TO
MEND.

Fig. 7: Super resolution image