Development of Computational Tools for Super-resolution Microscopy

Zhenting Wu

ICA-3558304

Information and Computing Sciences Department

Faculty of Science

Utrecht University

Supervisors: Eugene Katrukha

Lukas Kapitein

Remco Veltkamp

Table of Contents

Abstract

In recent years, super-resolution localization has been a vast developing area in bio-imaging, which aims at overcoming the resolution limit of microscope image due to the diffraction limit of light microscopes. To achieve sub-resolution precision, super-resolution techniques fit single molecules to the PSF (Point Spread Function) of the microscope. Existing software for super-resolution localization either lose substantial number of the true positive in the particle detection phase or doesn`t provide good estimations of the localization error, which is important parameter for reconstructing image with higher resolution. In current work a set of robust computational tools is developed for super-resolution localization analysis. One of the main goals is to keep as many true positive as possible while eliminating the false positive. These tools work well not only with homogeneous background images, but also provide robust analysis of images with non-homogeneous background, which are usually obtained during acquisition. The methods of the particle localization provide statistical estimation of localization precision, so that the real resolution could be determined. All methods are implemented as a plug-in for the free open source software ImageJ that is widely used for microscopy image analysis. It is shown that suggested approach gives substantial improvement in the particle detection and the localization precision compared with existing software and simultaneously gives an estimation of the localization error. Furthermore, drift correction algorithms are studied to get clearer image.

Chapter 1 Introduction

1.1 Motivation

In cell biology domain, fluorescent microscopy is the one of the most important tools for analyzing cell activities. It enables visualization of cell structures at the micrometer scale.

In the past few decades, researchers had been working on improving the resolution of fluorescent microscopy. Efforts had been made to limit the directions of the emitting light, so as to narrow the region from which a pixel receives the light, hence improving the image resolution. A typical example of this kind of microscope is the confocal microscope. Confocal microscope uses scanning point source for illumination, instead of the traditional light source that illuminated the whole specimen at the same time. Moreover, it eliminates out-of-focus light with a pinhole. Images acquired by a confocal microscopy have less out-of-focus signals, compared with traditional wide-filed microscopy.

However, being a type of optical microscopy, fluorescent microscopy has the same limit in resolution, which is roughly half of the wavelength. This limit is caused by a diffraction of light. Any object that is smaller than this diffraction limit will appear as a disk.

Therefore, a development of advanced optical components already reached its limit, while an increasing demand for resolution in cell biology still remains. It is naturally that researchers started to exploit the super-resolution techniques to overcome the bottleneck of diffraction limit in recent years.

Instead of improving image acquisition technique, as the new advanced microscopes do, super-resolution techniques is focused on improving the resolution by means of image analysis. Time-lapse images of the specimen are taken, with the help of special chemical techniques. After that a super-resolution reconstruction image is generated by analyzing the time-lapse images. It will give an image of cell structures at the nanometer scale.

Although a lot of software analysis tools have been developed for super-resolution, they are either not free or not robust enough. With the growing application of super-resolution techniques, a free super-resolution tool that can provide a good super-resolution reconstruction is in great demand.

1.2 Goal

The goal of this project is to develop a set of instruments for single-molecule super-resolution localization. The main tasks are automatic detection of particles in time-lapse images and finding spots centers` coordinates. The localization errors will also be estimated.

1.3 Overview

In the first chapter introduces the motivation and the main goal of our project.

In the chapter two the main principles of super-resolution microscopy and its difference from conventional fluorescence microscopy are briefly described.

Chapter three describes the first phase of super-resolution image analysis - particle detection. To improve the true positive rate detection and eliminate the false positive, specific convolution kernel is used to enhance the particle signal, smooth the noise and remove the gradient background. An optimal threshold is chosen for image binarization. Some filters are applied after that to eliminate the false positive. In comparison with existing super-resolution analysis software RapidStorm, I achieve similar true positive rate on artificial images dataset and number of false positives is much smaller. Comparing with another existing software QuickPALM, our false positive rate is slightly worse but our true positive rate is considerable more than QuickPALM`s.

In the chapter four the second phase of analysis is introduced: particle localization. A set of different localization algorithms is discussed: center of mass method, Gaussian mask method and Gaussian fitting method. The Gaussian fitting method is discussed in details to figure out possible improvements in the localization precision and estimated error of particle localization. The localization precision has been improved by 4% - 76% at signal-to-noise ratio from 2 to 100 comparing with QuickPALM on our artificial dataset (Appendix 1.3.1). In comparison with RapidStorm, our localization precision is up to 7% better at low signal-to-noise ratios (from 2 to 7) and roughly the same at high signal-to-noise ratios (from 7 to 100). At signal-to-noise ratio of 100, our localization error is equal to 0.011 pixels. Since typical microscopy images have dimensions of 100 nm/pixels, the achieved localization precision is around 1 nm at signal-to-noise ratio of 100. The estimated localization error given by proposed algorithm is 13% - 17% more than the real localization error at signal-to-noise ratios from 2 to 7. At the signal-to-noise ratios from 7 to 100 the estimated localization error exceeds “ground-truth” error up to 58%.

The chapter five introduces the implementation of our super-resolution image analysis tool. The general framework of the algorithm is described and the implementation details of the fitting are given. The performances of different fitting library are compared.

The chapter six concludes the whole project and discusses directions of the future works.

Chapter 2

2.1 Conventional Fluorescence Microscopy

Fluorescence microscopy is a branch of optical microscopy, which involves the transmission and reflection of lights through optical lenses together with special staining of a sample. Typical setup contains a light source, a set of filters, a microscope and a camera.

To visualize the cells using fluorescence microscopy, first the specimens should be stained with fluorescent dyes. Fluorescent dyes are fluorescent molecules which can absorb a certain wavelength of lights and emit lights with a longer wavelength. These dyes are chosen to label target structures that carry specific molecules in the cell with specific colors [1]. There are a variety of fluorescent dyes in the fluorescent dyes family. Each type of dye has its specific applications. After staining the specimen is exposed under proper range of light wavelengths that matches this fluorescent dye absorption spectrum and the electron in the fluorescent molecule will absorb the photon and switch from the ground state to a higher energy level, which is unstable. This wavelength range is called the excitation range. At the unstable level the electron has the tendency to fall back to the ground state and at the same time, it emits a photon. The emitted photons are within another wavelength range, which is the emission range. The emission range is always higher than the excitation range, implying lower energy. This phenomenon is called fluorescence, which was first described by Stokes [2]. To separate the emitted photons from photons coming from the light source, the emission filter is used. These emitted photons are detected by the camera and in this way the structures labeled with fluorescent can be depicted.

Since registration of fluorescence does not differ much from registration of other microscopy techniques, the resolution of imaging system is ultimately limited by diffraction. Characteristic scale of resolution depends on optical parameters of system, used wavelength and etc, but it can be roughly estimated to be equal to 200-300 nm.

2.2 Super-resolution Microscopy

During last decades a set of different methods were developed to overcome diffraction limit and obtain super-resolution images.

4Pi microscopy [3] overcomes the fact that the resolution in the axial direction is one third or less of the resolutions in the other directions, because the point spread function is elongated in the axial direction. By illuminating a sample using two objective lenses, the axial resolution can be enhanced.

Stimulated emission depletion (STED) [4] improves the resolution by inhibiting the outer region of excitation point spread function which is equivalent to breaking the diffraction limit. Structured illumination microscopy (SIM) [5] uses images made under different illumination to derive encoded high-resolution information.

The techniques described above increase the resolution by improving the illumination of sample and require hardware modification of microscopy setup. It is not the case with single-molecule localization super-resolution (SR) techniques. These methods work only under the assumption that the images are sparse-particle images and each particle is a single particle.

In conventional fluorescence microscopy, an image of the specimen displays an ensemble of overlapping particles. In these conditions it is very hard to locate individual molecules. Main principle of SR localization is to light up only single molecules and determine the positions of centers with high precision.

To adapt the imaging for the use of single-molecule super-resolution techniques, some special dyes are introduced. Photoactivable fluorescent molecules can be activated by impulse of light within a specific range of wavelengths. Photoswitchable fluorescent molecules are capable of being turn on and turn off stochastically under certain illumination [6]. With these dyes only a few particles are turn on in each frame, producing sparse-particle image. For each sparse-particle image, single particles are detected and localized with a localization precision that is much smaller than the diffraction limit. Finally they are plotted on a super-resolution reconstruction image.

Single Molecule High Resolution Imaging with Photobleach (SHRImP) [7] (sometimes also referred as BaLM [8]) gradually photobleaches molecules and thus is able to separate individual fluorophores. Photo-Activated Light Microscopy (PALM) [9] uses photoactivatable fluorescent molecules to target specific protein. After stochastic activation of subset of molecules, they are imaged, localized and photobleached (inactivated) and cannot be re-activated. Another single-molecule super-resolution technique – Stochastic Optical Reconstruction Microscopy (STORM) [10] is based on photoswitchable fluorescent dyes. The fluorophores are first all switched off. In the later stage they are switched on and off stochastically.

For BaLM and SHRImP, at initial moment when the molecules are active, they are all turned on. At this stage the acquired images are of no big different from the conventional microscopy image. After photobleaching is applied, the molecules keep bleaching, which may be reflected in the image that the particles disappear as time passes. Single molecules are located at image that is subtraction of each frame from the following one. For STORM, the particles in each frame are already expected to be sparse, so they can be directly used for super-resolution technique.

Figure 2-2-1 shows the conventional microscopy image and super-resolution image of the sample containing stained microtubules *in vitro*. If the same region of interest is zoomed in both images, in the conventional microscopy images, we can hardly see fine structure and location of microtubules. In the zoom-in image of the super-resolution image, we can see there are two microtubules crossing each other while in the conventional image, these two microtubules may be count as one microtubule because of the lower resolution.

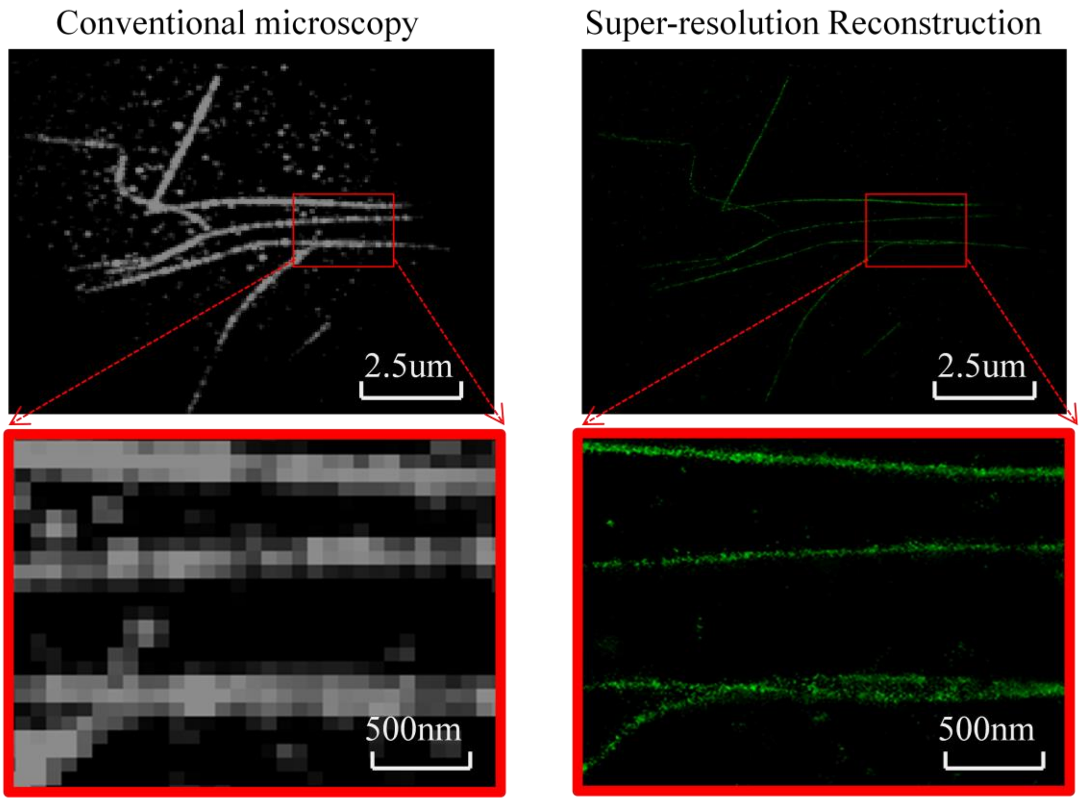


Figure 2-2-1 (1) The two images on the top shows a conventional microscopy image and the super-resolution reconstruction of it. (2) The two images on the bottom are the insets of the same rectangular region of the images above them.

There are a number of existing software for super-resolution image analysis: PALM3D [11], 3Bmicroscopy [12], Octane [13][14], M^{2}LE, RapidSTORM [15] and QuickPALM [16]. Each of the solution has its own drawbacks. PALM3D focuses on 3D super-resolution reconstruction and it has strict limits in the input data format. 3Bmicroscopy aims at analyzing image where many particles may overlap and requires substantial computational resources. Octane doesn`t provide automatic particle detection. M^{2}LE requires large-memory machine for calculation, which is not applicable for common user. RapidSTORM has a good true positive rate and uses Gaussian fitting method for localization, but the false positive rate is as high as 28% at signal-to-noise ratio of 2 (testing on our artificial dataset). QuickPALM use a simple center of mass method for particle localization, which give worse localization precision than Gaussian fitting method and it loses most of the true positive at low signal-to-noise ratios, but its false positive rate has been eliminated to less than 1%. The achieved results are compared results with RapidSTORM and QuickPALM, as widely accepted benchmark software.

Chapter 3 Single Particle Detection

3.1 Introduction

3.1.1 Background and Motivation

In the past few decades we had witnessed a rapid growing of digital image processing. Nowadays it has become a well-developed image analyzing technique that has wide application in the fields of information technology, biology, geosciences, etc.

In spite of the advances in digital image processing, the effectiveness of particle detection is always limited by the quality of the images. Unfortunately, in the application of super-resolution fluorescent microscopy, image quality is often a critical issue. The noise in microscopy images will always increase the difficulty of single particle detection (Appendix 3.1.1). The main noise sources in fluorescent microscopy images are the shot noise and the background noise.

Shot noise origins from the particle nature of light. The camera works by counting the number of photons it receives in a specific region within a short period of time, and transforming it to the grey count of that pixel in the corresponding frame. The photons from a stream of light of constant intensity still arrive at detector randomly, yielding a variation in the number of photons the camera receives. The brighter the light is, the larger the variation we get. This photon counting error contributes to the shot noise in the pixels.

For charged-couple device (CCD) (that is a major technology used in cameras for digital imaging) background noise mainly consists of out-of-focus fluorescence, CCD readout noise and dark current [17].

Dark current is caused by thermally generated electrons in the CCD. It`s directly related to the temperature of the CCD. If the temperature is high enough, dark current will completely fill the image.

Readout noise follows Gaussian distribution. Shot noise and dark current both follow Poisson distribution, which can also be approximated by Gaussian distribution at high values [18].

To measure the ratio of the spot signal and the noise signal the parameter called signal-to-noise ratio (SNR) is introduced (see Appendix 3.1.2 for definition). Fluorescent microscopy images usually have signal-to-noise ratio between 2 and 100.

The performance of particle detection dramatically gets worse in low signal-to-noise ratio region. Unfortunately the fluorescent microscopy images fall in low signal-to-noise ratio region frequently. In low signal-to-noise ratio region the number of true positive dramatically decreases. And fraction of the false positives is in general higher in the low signal-to-noise ratio region than in the high signal-to-noise region. As shown in Figure 3-1-1, testing with the uniform-background images (slope = 0) in our artificial dataset, QuickPalm has only 2% of true positive at SNR = 2, 20% at SNR = 3 and 70% of true positive at SNR = 4. RapidStorm has much better performance of true positive than QuickPALM with uniform-background images. But its false positive rate is as high as 28% at SNR = 2, 17% at SNR = 3 and 10% at SNR = 4. As for gradient-background images (slope = 3), RapidStorm loses its discriminative power, while QuickPALM`s performance is as good as its performance with uniform-background.

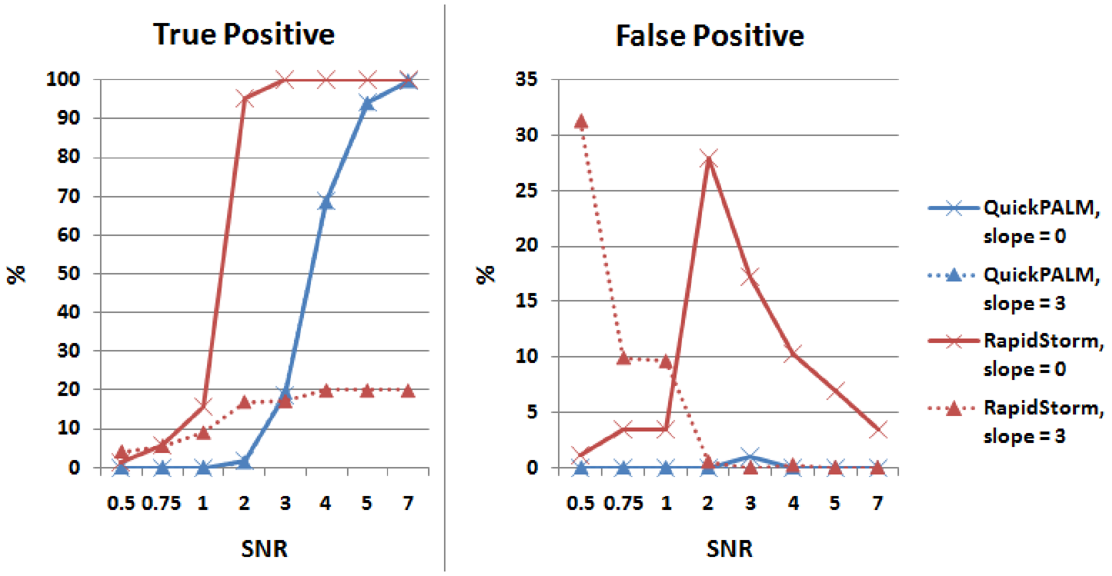


Figure 3-1-1 The left plot shows the true positive rates of QuickPALM particle detection and RapidStorm particle detection at different signal-to-noise ratios, testing with the uniform-background images (slope = 0) and the gradient-background images (slope = 3) of our artificial dataset. The left plot shows the false positive rates of them. In all our tests with QuickPALM in this project, the following parameters are used: Minimum SNR – half of SNR; Maximum FWHM (in px) – 5; Image plane pixel size (nm) – 106.00; other parameters are chosen as default. The test in rapidStorm performed in the casual mode. The parameters setting are chosen to correspond to the parameters in QuickPALM.

3.1.2 Framework

In machine learning object detection can be fulfilled by supervised methods. Additional training dataset is required to train a classification system in an offline phase. Supervised methods may give good results because the particles have certain patterns. But considering that for the reconstruction of super-resolution structures there may be hundreds of thousands of particles, and computation time is also a critical issue, supervised method would be too computation consuming. Moreover, if training dataset is involved for particle detection, automatic particle detection cannot be achieved. Smal [19] has tested fluorescence microscopy particle detection algorithms with the commonly used supervised methods and unsupervised methods. It has been shown that supervised methods didn`t give substantially better results than the best of the unsupervised methods. Therefore in this work the unsupervised method based on particle detection algorithm is chosen.

The implemented unsupervised method follows a preprocessing-detection-post processing scheme.

Given an image ***I***, the first step of particle detection is noise removal. A low-pass filter is applied on the image to remove noise. After that the image is convoluted with a kernel ***K***. The convolution step smoothes the noise and enhances the spot signal that match the kernel. These two steps are pre-processing steps that prepare the image before the particle detection. The next step is binarization with a threshold *T*. It divides the image into the background region and the spot regions, fulfilling initial particle detection. After that a few filters are applied on the binary image for false positive elimination.

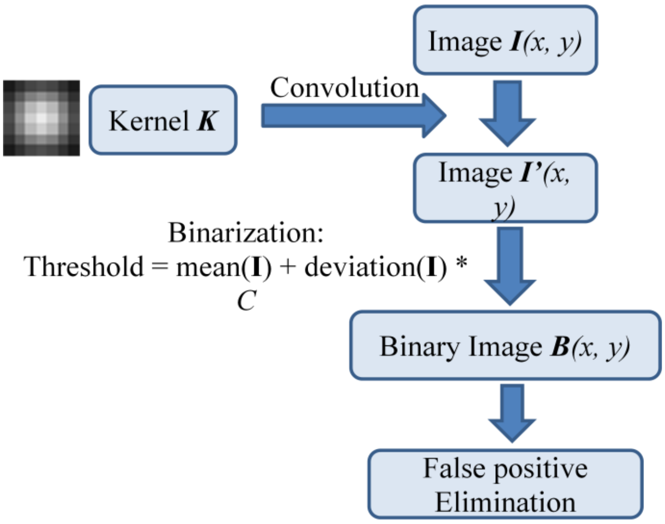


Figure 3-1-2-1 General framework of the particle detection section of our algorithm.

To remove non-homogeneous background, our particle detection algorithm combines the Top-hat filter with a Gaussian kernel. The parameters of these filters are optimized so that they will keep as many true positives as possible while eliminating the false positives. Automatic thresholding is implemented by calculation the statistical characteristics of the images. Our algorithm detect more than 60% of true positive at SNR = 2 and more than 95% of true positive at SNR = 3, for both uniform background images and gradient-background images. The achieved false positive rate is 3.5% at SNR =2 and 0.5% at SNR = 3.

3.1.2 Goal

In this chapter our main goal is to improve the efficiency of particle detection, namely increase the number of true positives, eliminate false positive and make the particle detection invariant to gradient background. Considering that missing some particles among thousands of fluorophores is better than wrongly labeling some non-existing particles while they are not expected, when a balance has to be taken between increasing the true positive and eliminating the false positive, it is preferable to eliminate false positive in sacrifice of losing some true positive.

Apart from improving the performance, another goal is to implement automatic detection. In the particle detection phase, various filters are applied and each of them requires some customized input to fit the testing data. Users should be relieved as much as possible from the difficult choice of complicate user-input parameters. Approach requires that user provides only a few easy-to-measure parameters and evaluates the rest of parameters based on revealed optimal dependences, thus the algorithm can be automatically executed.

3.2 Convolution

The main purpose of convolution is to enhance the spot signal and smooth the noise.

Image convolution (Appendix 3.2.3) is a 2-dimensional discrete convolution of the image $\boldsymbol{I}$ and the kernel $\boldsymbol{K}$.

The kernel $\boldsymbol{K}$ can be viewed as a weight mask. A pixel at the rth row and the cth column in the output image is the sum of $\boldsymbol{K}$ and a square region in the input image. This square region has the same size as $\boldsymbol{K}$, centered at the rth row and the cth column.

Consider an average kernel $\boldsymbol{K}_\mathrm{A}$, whose elements have the same value 1. D is the 2-dimensional pixel region of $\boldsymbol{K}_\mathrm{A}$. When an image is convoluted with $\boldsymbol{K}_\mathrm{A}$, each pixel in the output image is the average of the square region in the input image, which is the process of average blur (Appendix 3.2.4).

If we view it in the frequency domain, a convolution operator in the space domain has the same effect with a multiplication operator in the frequency domain (Appendix 3.2.5):

In frequency domain, kernel $\boldsymbol{K}_\mathrm{A}$ has its average value a_0 in the lowest frequency level. In all the other higher frequency level the coefficients are all zero. If $\boldsymbol{I}$ multiplies $\boldsymbol{K}_\mathrm{A}$ in frequency domain, only the average value of $\boldsymbol{I}$ will be left if $\boldsymbol{K}_\mathrm{A}$ is large enough. The details of $\boldsymbol{I}$ in higher frequency level are all discarded because of the zeros in these levels of $\boldsymbol{K}_\mathrm{A}$. So we can see $\boldsymbol{I}$ will become a plain background image without any textures. The kernel works as a blur filter. This is in accordance with the analysis in space domain.

When the kernel doesn`t have uniformed value, it`s a weighted blur filter if all the values are positive. In our case the purpose is to smooth the noise while avoiding blurring the spot signal, so an average kernel will not be an appropriate choice.

Apart from being a blur filter, the kernel also serves to enhance a certain pattern in the images, which is done by setting weight values in each pixel of the kernel. A kernel enhancing the spot must have a pattern agreeing with the intensity distribution of the spot, which is also the point spread function of the microscope.

As mention before any object that is smaller than the diffraction limit will appear as a disk in microscopy images. This disk is the convolution of the spot itself and the point spread function of the microscope. Point spread function is the function that describes the distribution of the intensity of a spot that is smaller than the diffraction limit under an optical microscope. It is a property of the microscope. Considering the complexity of constructing the true point spread function model of a microscope, in most application, it`s approximated by a Gaussian function for simplification.

Here the kernel $\boldsymbol{K}_G$ is generated (Appendix 3.2.6) that represents a discrete Gaussian function.

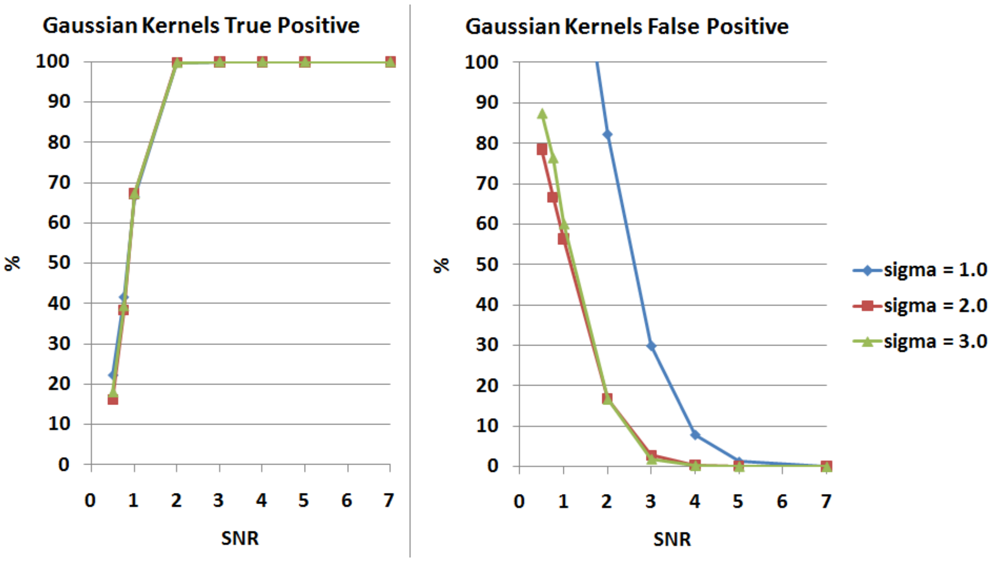


Figure 3-2-1 (1) The plot on the left shows the true positives percentage when particle detection algorithm detects with Gaussian kernels with different sigma (standard deviation) at different signal-to-noise ratio. (2) The plot on the right shows the false positive the particle detection algorithm detects with Gaussian kernels with different sigma (standard deviation) at different signal-to-noise ratio.

Figure 3-2-1 shows that the particle detection has the best result when the standard deviation of the Gaussian kernel is 2.0 (sigma = 2.0). It is exactly the value of the standard deviation of the artificial microscope`s point spread function used to generate images.

To avoid edge effect induced by the convolution, while computing convolution of the pixels near the edges, the pixels outside the image region are padded with pixels of $\boldsymbol{I}$ that has smallest Euclidean distance to the out-of-bound pixels.

3.3 Thresholding

The preprocessing steps prepare the images for particle detection. After that the image will be binarized with a threshold value *T*. Any pixels that have intensity equal to or larger than *T* are the foreground and pixels that have intensity smaller than *T* are considered to be background.

Existing algorithms for image thresholding are mainly histogram-based (Appendix 3.3.1). All of them compute a threshold value *T* that will minimize the detection error.

Histogram approach is fast and convenient way to implement image binarization. However, for single particle detection, our images are preprocessed into sparse-particle images, implying that the number of noise pixels is greatly larger than the number of spot pixels, resulting in a huge difference between the noise histogram and the spot histogram, which makes histogram segmentation impossible.

Figure 3-3-1 shows the histogram of an artificial image containing particles. The x-axis shows the intensity range of this image is [0, 65535]. From the figure one can only recognize a major peak in the histogram, which is the peak of the noise pixels. The pixels belonging to the spots distribute in the high-intensity region of this histogram. But the number of spot pixels is so small compare with the number of noise pixels that one cannot distinguish them from the noise pixels. Therefore, a statistics approach could be taken.

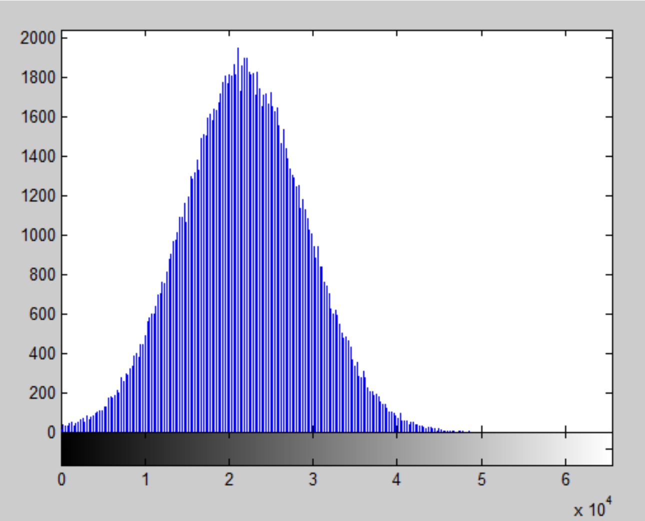


Figure 3-3-1 The histogram of an image with SNR = 3 from our dataset. The x-axis represents the intensity of the pixels and the y-axis represents the counts. It could be seen that one can hardly separate the particle signal from the noise signal in the histogram.

For a wide-field microscope the noise can be modeled by a Gaussian distribution $N(\mu_n, \sigma_n^2)$, where μ_n is the average intensity of the noise, and σ_n is the standard deviation.

As mentioned in section 3.1.1, the distribution of spot signals can be also approximated by Gaussian distributions. Assume that the spot signal follows the distribution $N(\mu_s, \sigma_s^2)$, where μ_s is the average intensity of the spots signal and σ_s is the standard deviation.

Since $(\sigma_s + \sigma_n) \ll (\mu_s - \mu_n)$, it can be assumed that the histograms of the noise and the spot signal are non-crossing. Thus they could be considered separately. If the intensity of a pixel is higher than a intensity value $(\mu_n + C \bullet \sigma_n)$, the probability that this pixel is not a noise pixel (which means it`s a spot pixel, because the background is dark and the noise signal and the spot signal has higher intensity than the background intensity) is:

$$1 - \mathrm{P}(\mu_n + C \bullet \sigma_n) = \int_{\mu_n + C \cdot \sigma_n}^{+\infty} p(i)di \qquad (3\text{-}3\text{-}1)$$

P(x) is the cumulative distribution function of the noise and p(x) is the probability density function. If C is large enough, the intensities of most of the noise pixels are smaller than $(\mu_n + C \bullet \sigma_n)$. So a threshold that divides the noise and the spot T is given by:

$$T = \mu + \sigma \cdot C \qquad (3\text{-}3\text{-}2)$$

where C is a constant parameter. The choice of C will greatly affect the goodness of particle detection. Figure 3-3-2 shows that a value of C within the range of [3, 4] would be appropriate. In current work the value of 3 is chosen to be optimal. With this three-stand-deviation coefficient, there are still 0.2699796% of noise pixels being wrongly classified as spot pixels and it may become false positives of the particle detection phase. These false positive will be removed in later phases.

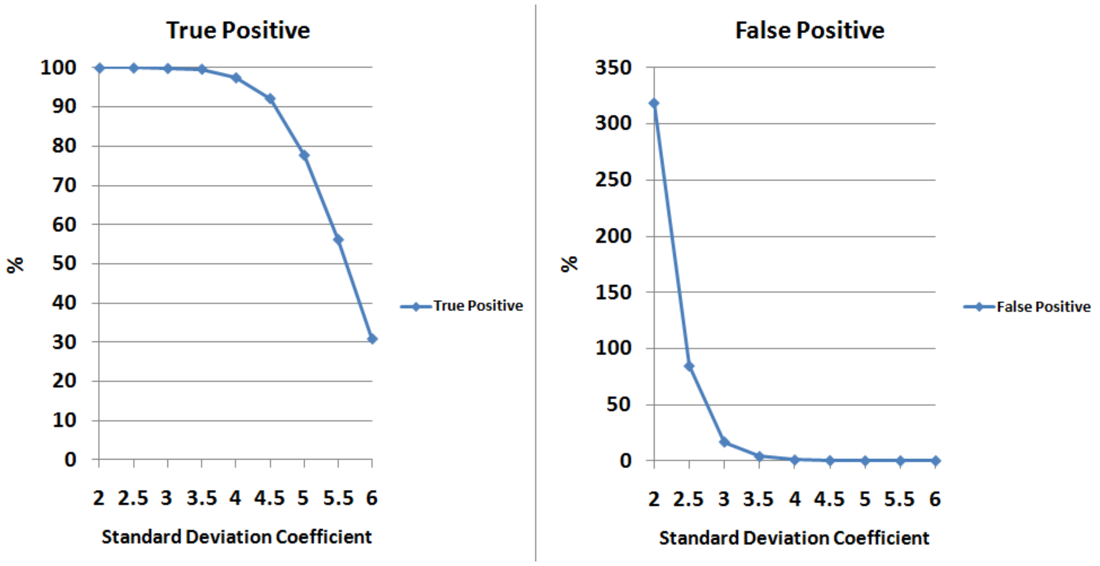
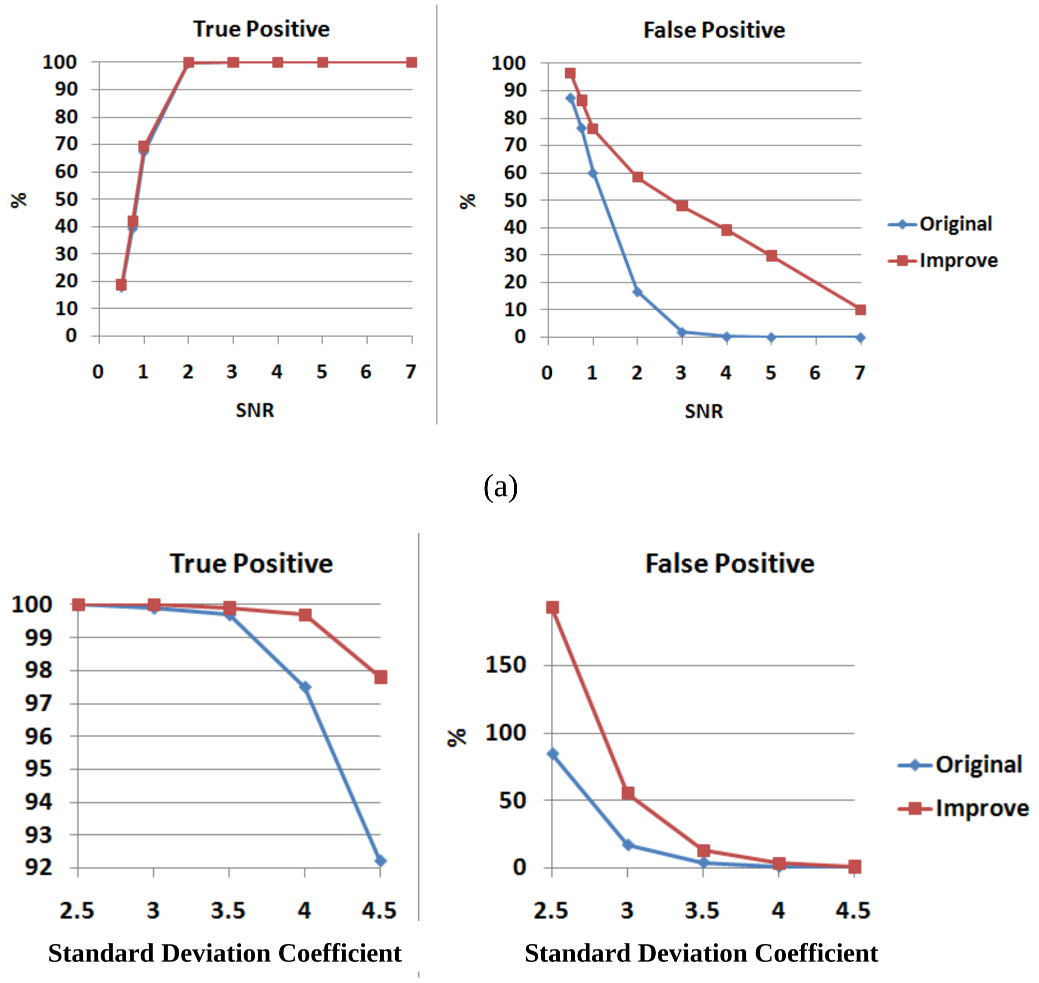


Figure 3-3-2 (1) The plot on the left shows the true positive detected by the algorithm with different standard deviation coefficient on the x-axis. (2) The plot on the right shows the false positive detected by the algorithm with different standard deviation coefficient on the x-axis.

Now a problem arises. The proper thresholding requires values of the mean and the standard deviation of the noise. However, it`s impossible to get the real statistical values of the noise, because before that correct separation of the noise signal and the spot signal is required. But this cannot be done before the image is converted to binary. In current implementation the statistical values of the whole image as an approximation of the statistical values of the noise is calculated. Let these approximated values be $(\hat{\mu}, \hat{\sigma})$ and the real values be (μ, σ).

Assume that using the real value (μ, σ) one can better binary the image. There is a simple way to obtain a better approximation of the real statistics. After thresholding with threshold $\hat{T}$, the segmentation of the noise pixels and the spot pixels is achieved. With that a new mean $\hat{\mu}'$ and a new standard deviation $\hat{\sigma}'$ of the noise pixels can be calculated and thresholding can be performed again with the new threshold $\hat{T}'$, which is calculated with $\hat{\mu}'$ and $\hat{\sigma}'$.



(a)

(b)

Figure 3-3-3 (a) The figure shows the true positive/false positive the original algorithm and the improved algorithm detect at different signal-to-noise ratio. (b) The figure shows the true positive/false positive the original algorithm and the improved algorithm detect with different standard deviation coefficient.

Considering that the spot signal has higher intensity than the noise, one can get the following relation if *C* is fixed:

$$\hat{\mu} > \mu > \hat{\mu}' \tag{3-3-3}$$

$$\hat{\sigma} > \sigma > \hat{\sigma}' \tag{3-3-4}$$

$$\hat{T} > T > \hat{T}' \tag{3-3-5}$$

Therefore, if $\hat{T}$ is used as the threshold the numbers of true positive and false positive are less, and if $\hat{T}'$ is used these numbers are bigger.

As shown in Figure 3-3-3 (a), when C = 3, the difference in true positive is rather small, but the difference in false positive is substantial.

However, the threshold is a single value. Adjusting of the threshold could be made by directly adjusting *C*. By doing this the difference between these two approaches will be wiped out, as shown in Figure 3-3-3 (b). If the curve of the improved approach is shifted for -0.5 along the x-axis, it will overlap the curve of the original approach. Similarly, the shift also works for the curves plotted with the real statistical values of the noise. Therefore the assumption that using the real values gives better thresholding result than using the approximated values doesn`t stand.

3.4 Background Subtraction

A non-specific staining, stray illumination, out-of-focus structures and other factors can induce non-homogeneous background in the SR images. If threshold is directly applied to the image, the regions that have brighter background may be classified as the spots, and the spots that lie on darker background regions may be classified as the background, as shown in Figure 3-4-1.

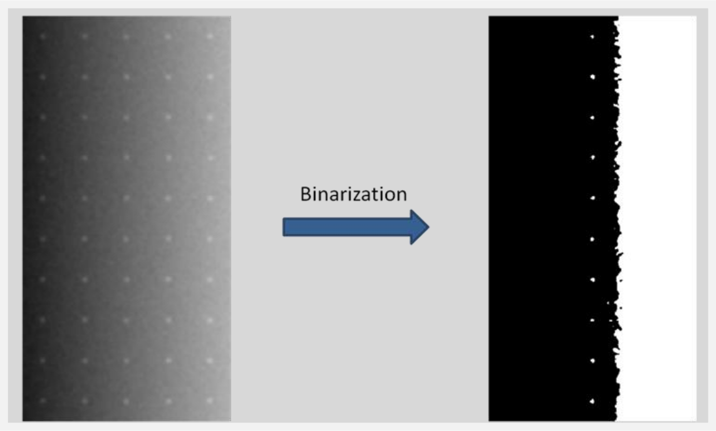


Figure 3-4-1 (1) The left image is an image from our dataset with gradient background. (2) The right image is the binarized image of the left image without proper handling of background.

Figure 3-4-2 shows the true positive rate of particle detection of homogeneous background (left) and non-homogeneous background (right) images. Applying our Gaussian kernel and statistical threshold, the algorithm achieves good particle detection performance while testing with homogeneous background images. But it loses its discriminative power while testing with non-homogenous background image. In the right plot the true positive curve converges to 20%, agreeing with Figure 3-4-1, where only one out of five columns of particles that share similar background intensity is detected.

To solve this problem, background subtraction is introduced in our algorithm to make the convolution kernel invariant to gradient background.

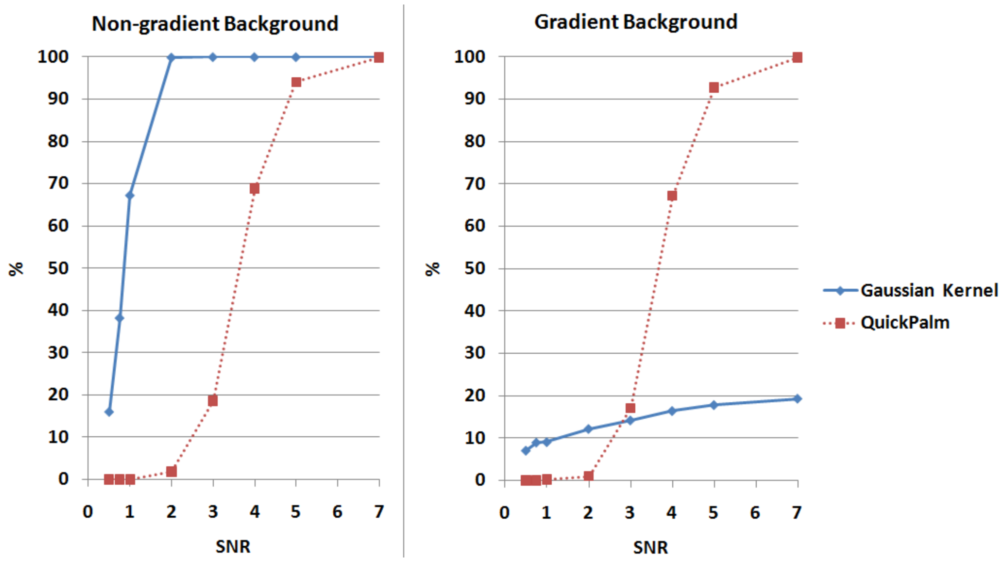


Figure 3-4-2 (1) The left plot shows the true positive detected by the Gaussian kernel and QuickPALM at different signal-to-noise ratio, testing with the non-gradient background images. (2) The left plot shows the true positive detected by the Gaussian kernel and QuickPALM at different signal-to-noise ratio, testing with the gradient background images.

Mashanov and Molloy [20] purposed a simple but effective background subtraction kernel (Table 3-4-1). The sum of weight of this kernel is exactly one.

The cells in the inner region of the kernel have weight value of one. This region serves as an averaging kernel. The zeros cells are ignored. The outer region with minus-value cells is considered as the background. By setting minus value in the outer region, which is presumed as the background region, the background value is subtracted. This kernel immediately improves the performance of particle detection for gradient background images. However, firstly it doesn`t match our particles` pattern very well because it uses an average filter in the middle of the kernel. Secondly, found background region does not give the best evaluation of the background value. Finally, the sum of weight is one, which is not as good as a zero weight for gradient background removal.

To solve the first problem Gaussian kernel is integrated with a background subtraction filter. For the second one, Top-hat filter is used for background evaluation. Finally, I study the influences of different sum of weight values.

0	-1	-1	-1	-1	-1	0
-1	0	1	1	1	0	-1
-1	1	1	1	1	1	-1
-1	1	1	1	1	1	-1
-1	1	1	1	1	1	-1
-1	0	1	1	1	0	-1
0	-1	-1	-1	-1	-1	0

Table 3-4-1 The Mashanov`s convolution kernel.

Before moving to solution of these problems, let’s discuss how background subtraction works.

In digital image processing, the sum of weight of a basic 3 by 3 high-pass spatial filter is (Appendix 3-4-1) zero [21]. After the filter move over the image during convolution, the output image has intensity values close to zero. In this way the zero-frequency term of the input image is eliminated. This can be verified by calculating the average intensity of the image, which shall be zero. If one subtract the original image by the high-pass image, one get the low-pass image, representing the low frequency terms. The corresponding low-pass filter (Appendix 3-4-1) is a typical average filter, which will get the average values of the image it convolutes with. For a pixel in the input image, if we want to evaluate the background value of this pixel with its n by n neighboring pixels, including the center pixel itself, the average filter will be exactly the filter we need, when $n = 3$. Subtracting the original image by the background value has the same effect as applying the high-pass filter to the original image.

But what happens in the Mashanov`s kernel is slightly different, for the sum of weight is one instead of zero. As was mentioned earlier:

High-pass = Original – Low-pass

A high-boost filter is:

High-boost = *C* • Original – Low-pass

where *C* is a user-defined coefficient. When *C* > 1, it enhances the high-frequency terms and at the same time keeps the general appearance of the original image.

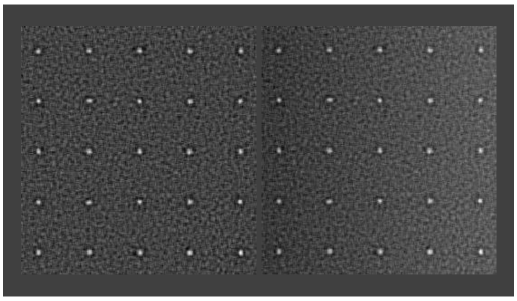
The filter in Mashanov`s kernel can be described by:

$$\left(\frac{50}{49}\cdot \boldsymbol{Orginal}-\boldsymbol{Lowpass}\right)\cdot 49$$

$$=\frac{1}{49}(\boldsymbol{Original}+49\cdot \boldsymbol{Highpass})\cdot 49$$

$$= \frac{1}{49}(\boldsymbol{Lowpass} + 50 \cdot \boldsymbol{Highpass}) \cdot 49$$

A small portion of the low-pass terms, namely the background value is still kept in the output image. If the input image has a steep gradient background, with Mashanov`s kernel the gradient background cannot be completely subtracted. Figure 3-4-3 (a) and (b) show the difference in background subtraction of a high-pass kernel and a high-boost kernel.



(a) (b)

Figure 3-4-3 (a) The result of background subtraction with a high-pass kernel. (b) The result of background subtraction with a high-boost kernel.

To integrate our 7 by 7 Gaussian kernel with a 7 by 7 average filter for background subtraction, both of them are simply normalized and summed up. This new Gaussian subtraction kernel integrates Gaussian kernel and average filter. It has the same effect as sequentially applying a Gaussian filter and the average filter.

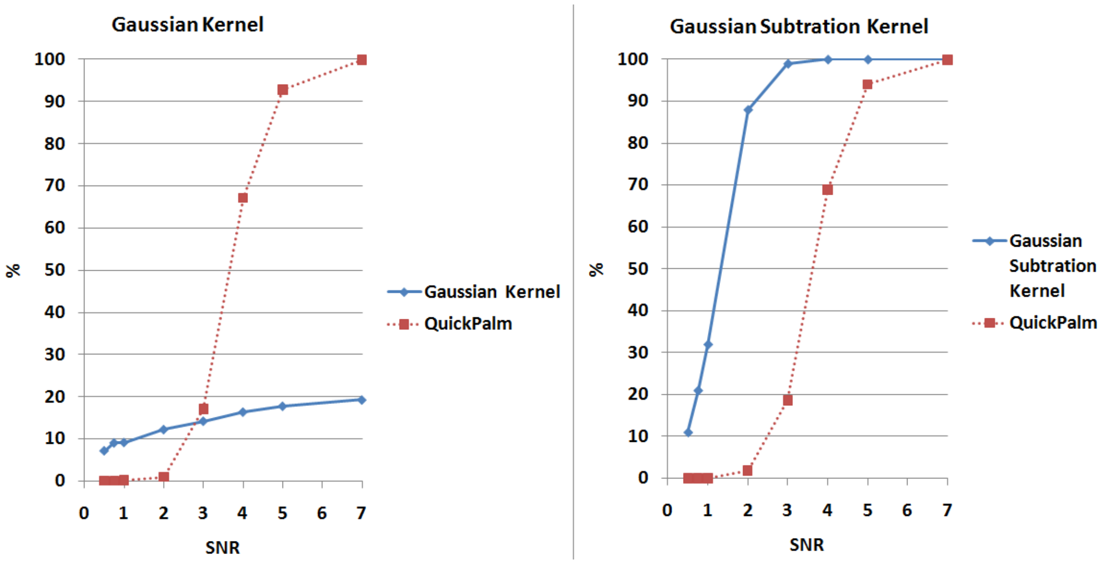


Figure 3-4-3 (1) The left plot shows the true positive detected by the Gaussian kernel and QuickPALM at different signal-to-noise ratio, testing with the gradient background images. (2) The left plot shows the true positive detected by the Gaussian subtraction kernel and QuickPALM at different signal-to-noise ratio, testing with the gradient background images.

The right plot of Figure 3-4-4 shows the true positive performance of Gaussian subtraction kernel, compared with QuickPALM. The left plot is performance of Gaussian kernel without background subtraction. In 3-1-1 it was shown that RapidStorm is not invariant to gradient background, thus here only QuickPALM was used for comparison.

Now the remaining task is choosing the correct pixels for background value evaluation. Assume that the spots have radius *r*. Consider a *n* by *n* kernel size, any pixels within the distance of *r* from the center pixel are the spot pixels, and the others are the background pixels. The background value of the center pixel will be the average intensity of the background pixels (Appendix 3.4.2).

3.5 False Positive Elimination

In Figure 3-3-2 (2) one can see that no matter how the threshold is adjusted, there are always some true positives lost or some false positives existing. Because of the trade-off, a balanced value has to be chosen, so that most of the true positives are kept and at the same time the number of false positive is kept as small as possible. After that additional false positive elimination phase is added to remove remaining false positives.

Four types of filters are applied here for noise elimination: noise filter, symmetry filter, opening filter and area filter, among which the noise filter is the pre-convolution filter and the other three are the post-convolution filter. The pre-convolution filter is applied for image pre-processing before the convolution. It removes the signal in the image that is characterized by the noise model. The other three filters are applied on the binary image after the convolution and the binarization. Each of them make use of a property of the spot model to remove detected spots that don`t match the property.

3.5.1 Noise filters

There are a variety of filters for removing noise. Some of them either work in the space domain, such as average filter and median filter, or in the frequency domain, such as low-pass filter. Others estimate the noise model for image restoration. However, for most of them, smoothing the noise will more or less bluring the image.

In our case it was already shown that the noise follows Gaussian distribution, and there is the initial guess of the spot (signal) model, which is also a Gaussian function. The choice of a filter should be done carefully so that it will smooth the noise but avoid blurring the spot signal too much.

Although the noise signal model and the spot signal model are both Gaussian function, one can still see the difference between them. For each pixel of the image, the noise intensity is a random variable that follows Gaussian distribution. So the noise of the image has the majority of its signal in the high-frequency range of the image spectrum.

The spot signal model is a Gaussian function in the space domain within certain range. So it lies in the lower-frequency range of the image spectrum.

Considering this difference the Gaussian low-pass filter is used for the purpose of noise removal. Since the Fourier transform of a Gaussian function is also a Gaussian function, a Gaussian function is a low-pass filter that will smooth the image [22].

3.5.2 Symmetry

In the spot signal model, no matter what the parameters of the Gaussian distribution are, the spots are always reflection symmetric. This is a feature that can be used to distinguish the spot from the noise.

To measure the degree of symmetry of a spot, a feature descriptor is needed. QuickPALM uses two values S_x and S_y (Appendix 3.5.2.1) for measuring the symmetry with respect to x-axis and y-axis [23]. If the spot is a true positive then it should stand that $S_x \approx 1$ and $S_y \approx 1$. Allowing a portion of error, a tolerance is set to T_s. If a spot has symmetry errors exceeding the T_s, it is a false positive (Appendix 3.5.2.2).

Figure 3-5-2-1 shows the results of choosing different tolerance T_s. When $T_s = 1.0$, the allowing range is [0.0, 2.0], which means no symmetry filter is applied. It can be seen that from $T_s = 0.4$ to $T_s = 0.35$ there is a sudden drop on the false positive, but the loss on true positive is rather small. So $T_s = 0.35$ was chosen as the optimal value.

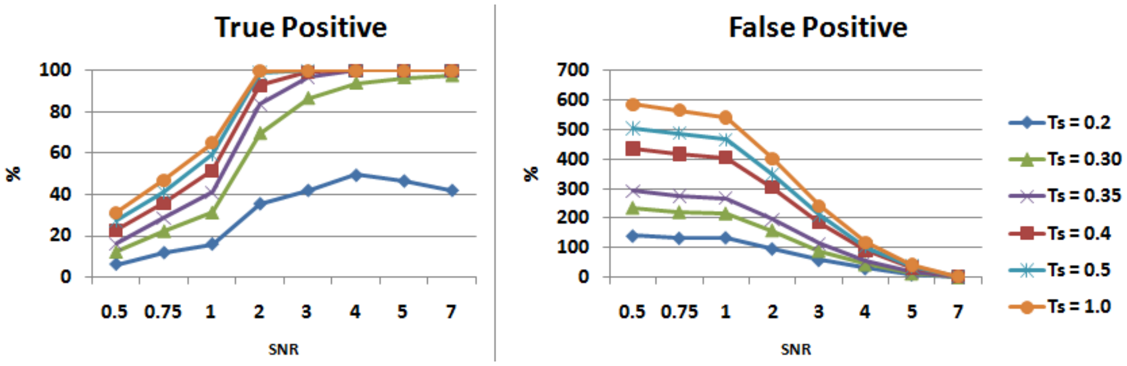


Figure 3-5-2-1 (1) Left: the true positive detected by the particle detection algorithm with different symmetry thresholds T_S. (2) Right: the false positive detected by the particle detection algorithm with different symmetry thresholds T_S.

3.5.3 Open

Open operation is a widely used morphological operation in image processing. It is the sequential combination of erosion and dilation, which are also fundamental morphological operations. Open is often used for noise removal. Here binary open operation is applied on the image after thresholding. It will remove small foreground particles from the image.

Open is an erosion followed by a dilation with the same operator (Appendix 3.5.3.1).

Using a large operator width will lead to loss of some true positives that have small sizes. But with a small operator width the operation will has less eliminating power. In this case the chance to eliminate the false positive is less.

Here the 3 by 3 open operator was chosen with all the elements in the cells equal to one. So the center of the operator is (2, 2).

3.5.4 Area

After binarization, there are many small spots in the image that are originated from the noise in the original microscope image. These spots are false positive which have to be removed. A simple area filter will do the job efficiently.

An area filter defines a threshold T_a. Any spot that has an area smaller than T_a will be treated as a false positive.

The choice of threshold value is a critical issue here. As mentioned before, the plug-in has to choose most of the parameters automatically, including the area threshold T_a.

The initial assumption is that T_a may be affected by the standard deviation of the point spread function, the convolution kernel size and the binarization threshold. Since the binarization threshold is defined by the convoluted image itself, and it shall correctly separate the background and the spots, the influence of binarization threshold can be ignored. Because as long as the background and the spots are correctly separated, the areas of the spots shall be the same. Our tests have shown that T_a = kernelSize + 1 would be appropriate.

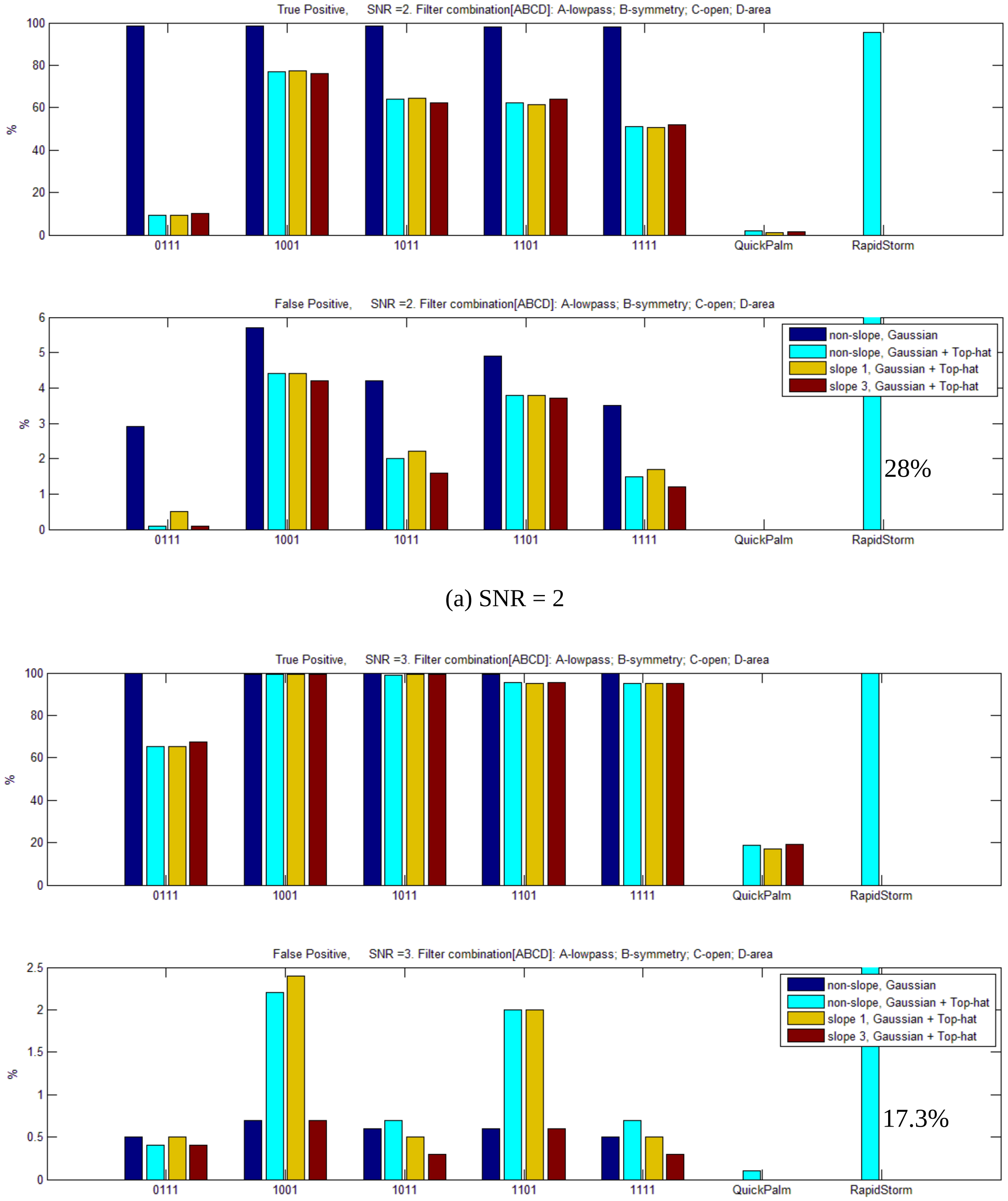
3.6 Experiment Result

The whole process of particle detection has been discussed in this chapter. In this section I am going to integrate all parts and test the performance of proposed approach.

In the previous section, all of the filters discussed are designed for eliminating the false positive. The Gaussian low-pass filter removes the noise in the image before particle detection. Except the Gaussian low-pass filter, the other three filters eliminate the false positive by removing irregular detected spots. It`s not difficult to notice that the functions of some of these filters overlap with each other. For instance, opening operation and area filter both remove small particles from the detected spots. Area filter only deals with the area of a spot, while opening operation not only deals with the area, but also works on the shape of a spot.

Therefore, it is essential to investigate how these filters perform when they are combined with each other. The crucial experiment is designed to test the performance of different combination of filters. The performance is evaluated by the true positive and false positive it gives.

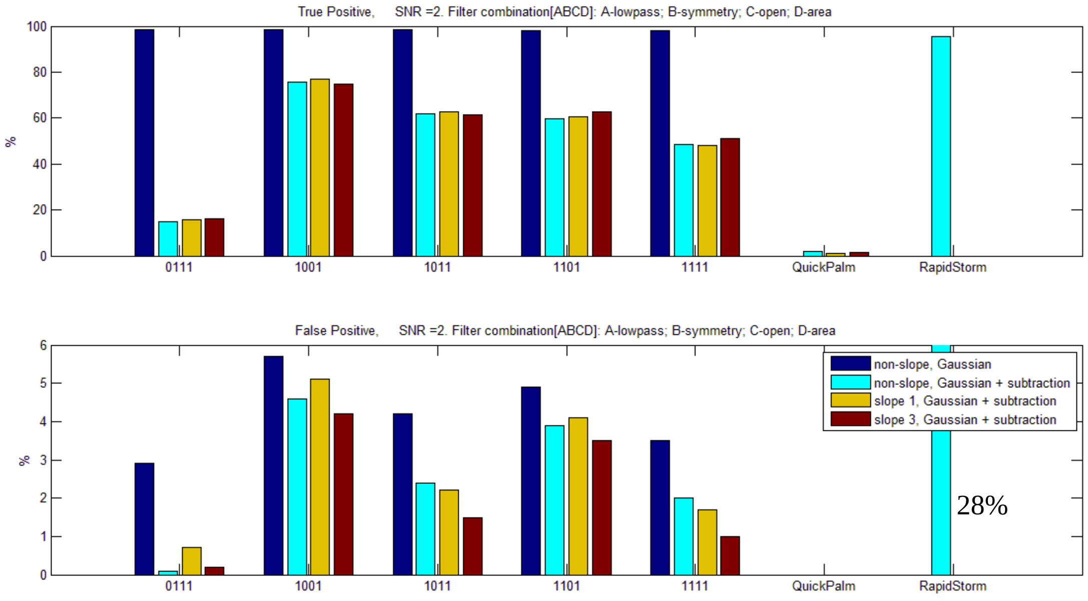
In the experiment, all the combinations of the four filters are tested. The filters are applied in a certain order. The order is arranged considering the functions and the effects of the filters. As a noise removal filter, Gaussian low-pass filter must be a pre-processing step, so it shall be applied before the convolution. Because open operation will change the shapes of the spots, and once the shapes are changed, the



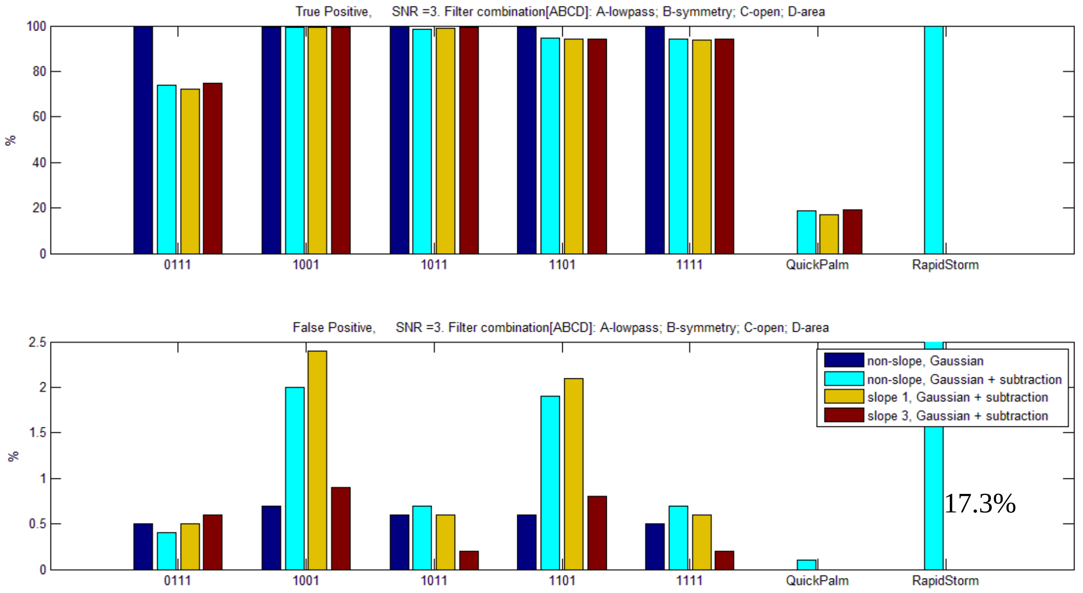
(a) SNR = 2

(b) SNR = 3

Gaussian Top-hat Kernel



(c) SNR = 2



(d) SNR = 3

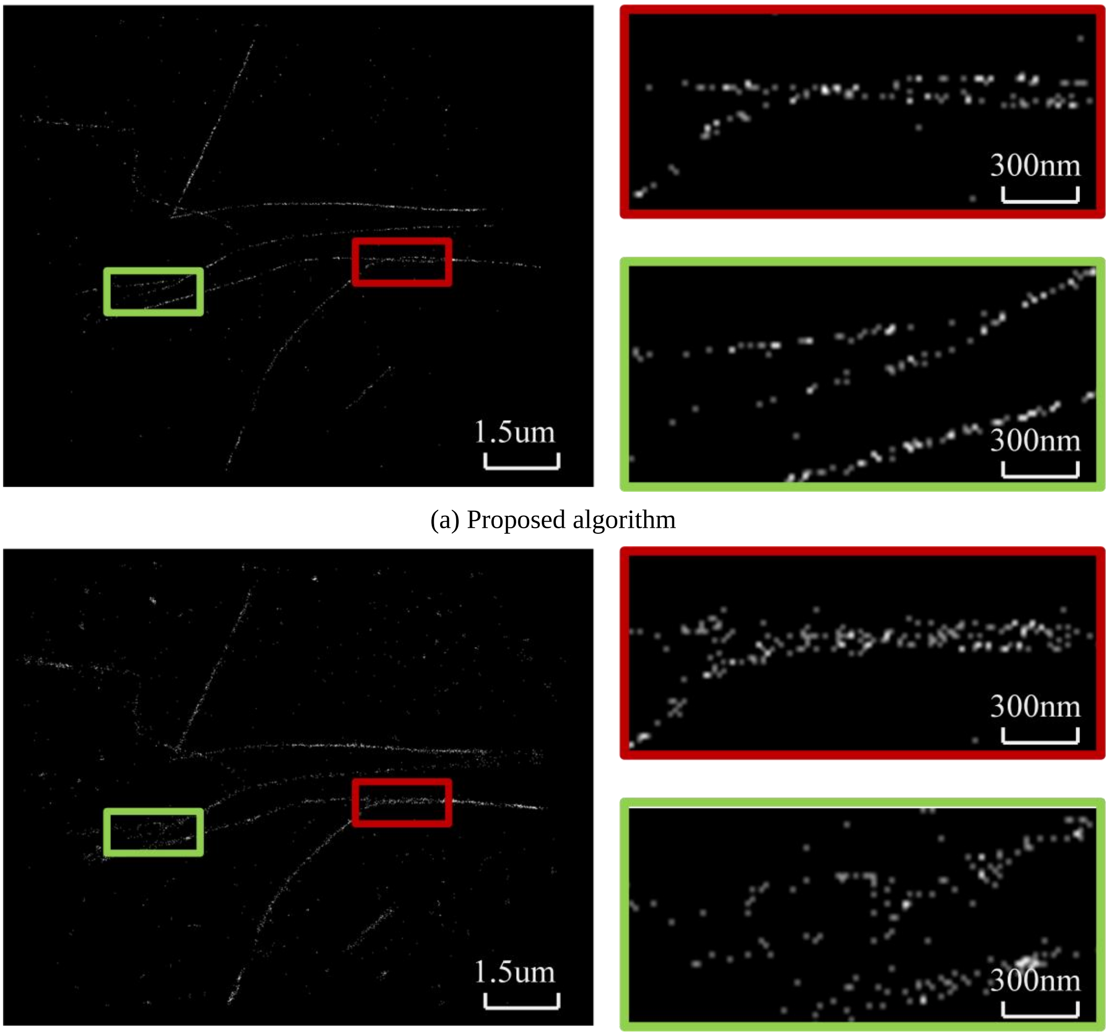
Gaussian Subtraction Kernel

Figure 3-6-1 shows the true positive/false positive rates of the particle detection algorithms with Gaussian Top-hat kernel and different filter combinations, at SNR = 2. (b) is the result of the same algorithm except that SNR =3. (c) and (d) are the results of the same algorithm, except that they use Gaussian subtraction kernel. The performances of QuickPALM and RapidStorm are also shown here for comparison.

Each digit of the combination numbers indicates whether the corresponding filter is applied. The correspondence of the digits and the filters are: 1) Gaussian low-pass filter. 2) Symmetry filter. 3) Opening operation. 4) Area filter.

symmetry may be changed, it has to be applied after symmetry filter. Open operation may separate some improperly connected spots into single spots. If area filter is applied before opening operation, the single spots which are smaller than the area threshold cannot be removed. Concluding all of these conditions, the order of these filters can be summarize.

Firstly, Gaussian low-pass filter is applied on the original image. And then the image will be convoluted with the top-hat Gaussian kernel. After that the image is binarized with a threshold calculated from the image statistics. For the next step, symmetry filter is applied on the binary image, following with opening operation. Finally, area filter will be applied.



(a) Proposed algorithm

(b) QuickPALM

Figure 3-6-2 (a) is the super-resolution reconstruction of a series of dSTORM images with proposed particle detection and (b) is the reconstruction with QuickPALM`s particle detection. For each of them, the left image is the whole reconstruction image. The right images are the details in the rectangular regions of the left image. Comparing them it can be seen that the microtubules in (a) have smaller widths and clearer structures.

Figure 3-6-1 shows the results of the top-5 combinations using Gaussian top-hat kernel, compared with the result of QuickPALM and RapidStorm. All of the combinations here greatly outperform QuickPALM in terms of true positive. Even with SNR = 3 can QuickPALM detect only less than 20% of true positive, but the best of our filters combination can detect more than 95% true positive. However, QuickPalm does an excellent job in eliminating false positive, it has less than 1% false positive both with SNR = 2 and SNR = 3. Our best combination has 0.5% false positive at SNR = 3 and almost 2% false positive when SNR = 2. Compared with rapidStorm our algorithm is slightly better in true positive rate. But RapidStorm has high false positive rate of 28% at SNR = 2 and 17% at SNR = 3.

Comparing the results of Gaussian top-hat kernel and Gaussian subtraction kernel, it is hard to find obvious differences in the true positive plots. But from the plots of false positives it can be seen that Gaussian top-hat kernel is more robust to gradient background image, especially when considering the "non-slope bar" and the "slope 3" bars. This agrees with the conclusion of the section Background Subtraction.

In this experiment combination 1011 shows the best result, which is the combination of Gaussian low-pass filter, opening operation and area filter. If considering only the figure of Gaussian top-hat kernel at SNR = 3, it achieves 99% true positive and 5% false positive.

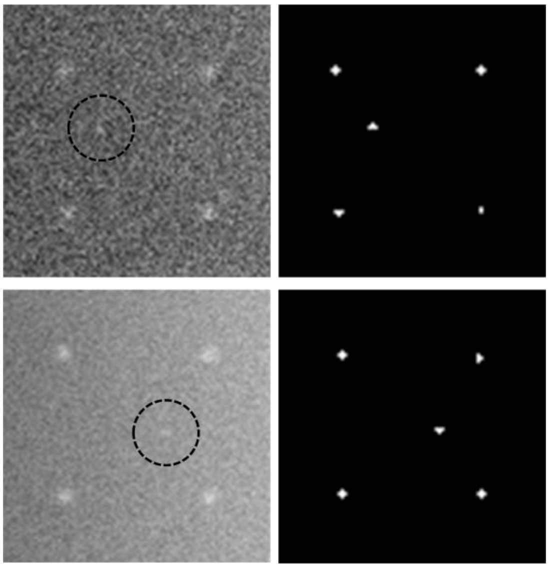


Figure 3-6-3 The left column shows the original image, and the right column shows the binary image with SNR=2. If we check the false positive (the ones in the dashed circle) in the original image, we find that even human visual system will classify them as real spots, but actually they are just spot patterns resembled by noise.

Although it was mentioned before that losing true positive is better than plotting non-existing false positive, but the statistics itself doesn`t make too much sense if there is no reasonable evaluation of the final reconstruction result. Since the super-resolution technique is developed for further investigation of cell structures, the evaluation of performance shall be done by human visual system, with real microscope images.

Figure 3-6-2 shows the super-resolution reconstructions of a dSTORM sample of microtubules. The QuickPALM reconstruction has more isolated spots that are not connected to any microtubule. And its microtubules have worse continuity and larger width than results acquired with suggested approach, which means the former has lower resolution. So suggested particle detection procedure is better than QuickPALM`s.

Notice that there are still 5% false positive while doing particle detection with developed algorithm. These false positive are inevitable rare noise patterns resembling spots (Figure 3-6-3). Some of these false positive can be removed in the next chapter – Particle Localization.

Chapter 4 Particle Localization

4.1 Introduction

4.1.1 General Introduction

Once particle detection is done, it provides a list of spots, each of which the intensity and the coordinate of each pixel are provided. Our remaining task is to find the accurate position of the maximum intensity peak in each spots.

The simplest approach one can come up with is calculating the geometric center of the spot. This approach is fast but crude. It gives a rough estimate of the spot position and it`s too rough for super-resolution reconstruction.

A better approach is calculating center of mass, as QuickPALM does. It`s also fast and it gives an acceptable precision. In this chapter a new method for center of mass calculation is introduced, that improves the localization accuracy in the low SNR range comparing with QuickPALM. However, when the image quality is extremely good, center of mass will give an accurate result, but with the SNR range that general microscope images have, center of mass is not as reliable as with high SNR range, even though the accuracy has been improved.

Therefore further localization is necessary for higher resolution reconstruction. What is known is the spot signal follows Gaussian distribution, and this is the information one can use for further localization.

With the information of the spot pixels known, a non-linear least square fit to the Gaussian distribution can be used to estimate the mean of the Gaussian distribution, which will be the position of the spot. Since it`s a two-dimensional fitting, each spot may need a considerable number of iterations before it is fitted, it`s much more computational expensive than computing center of mass.

A compromise is Gaussian mask [16], which is a simplified Gaussian Fitting. Although it is a simplified Gaussian fitting, it has a form similar to center of mass, which makes it much faster than Gaussian fitting.

4.1.2 Motivation

As shown in Figure 2-2-1, conventional microscopy images do not have enough resolution to show the details of many subcellular structures. The center of mass localization method used by QuickPALM resolves much higher details than conventional microscopy images. RapidStorm uses Gaussian fitting for localization, which gives better localization precision than QuickPALM (Figure 4-1-2-1). But again RapidStorm has much worse performance with gradient-background images than with uniform background images. A robust particle localization algorithm shall give small localization errors with both uniform background images and gradient-background images.

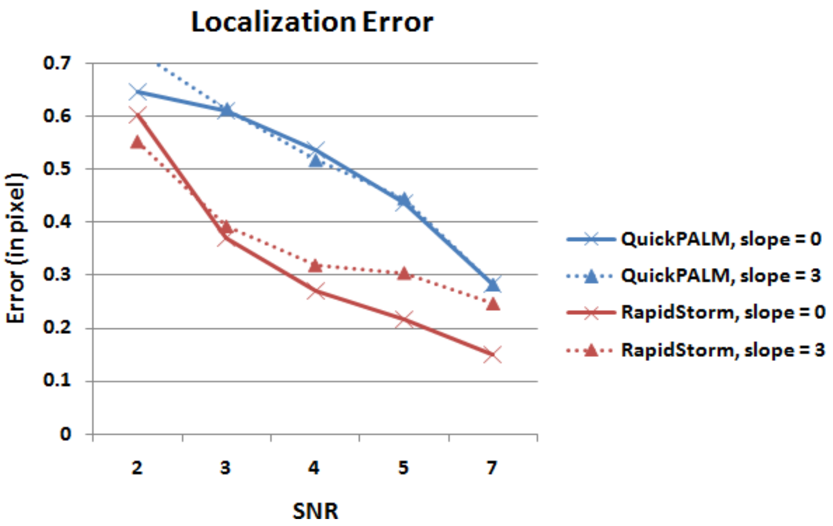


Figure 4-1-2-1 The localization error of QuickPALM and RapidStorm at different signal-to-noise ratios, testing with the uniform-background images (slope = 0) and the gradient-background images (slope = 3) of our artificial dataset.

After the particles are localized, they will be plotted on the super-resolution reconstruction image. To accurately plot the sub-pixel coordinate of a particle on the image, simply adding the average intensity of the particle to the pixel that is closest to the particle`s coordinate is not sufficient. Instead, the particle shall be displayed with a Gaussian spot whose standard deviation is the localization uncertainty of the particle. The localization uncertainty is evaluated by the localization error.

When testing is performed on artificial synthetic dataset, since the ground true is already known – the real position of the spots, the localization error can be calculated directly. But if this algorithm is going to be applied to real microscopy data, since the ground true is not provided, another way to estimate the error should be found.

4.1.3 Framework

The general framework of particle localization is simple (Figure 4-1-3-1). For a spot detected by the phase particle detection, the pixels belonging to the spot region are used for calculating center of mass. With center of mass as the initial spot position, Gaussian Mask or Gaussian fitting will be done to get a more accurate spot position.

A question here is how is the spot region defined? QuickPalm and gSHRImP [25] both find the local maximum of a spot and cut out a predefined shape from the original microscopy image with the local maximum as the center. But here a more representative spot region is used, which will be introduced in the next section --

Center of Mass. Gaussian fitting will be implemented by Levenberg-Marquardt method. Before that the background value of a spot is evaluated for background removal. Weight fitting, estimated parameters constraint and different Jacobian evaluation methods are discussed to improve the localization precision. The method of estimating localization error is introduced.

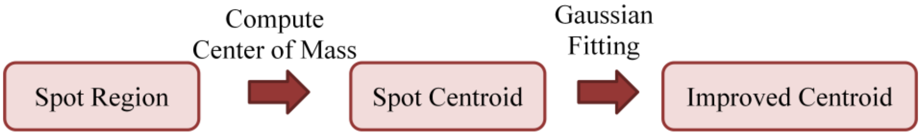


Figure 4-1-3-1 The general framework of particle localization

With our Gaussian fitting algorithm, achieved localization errors is 0.011 pixel at SNR = 100 and 0.6 at SNR = 2, which is a considerable improvement compared with QuickPALM center of mass method. In low SNR region, our localization errors are up to 7% better than RapidStorm`s Gaussian fitting method. Our implementation also estimated errors 13% - 58 % higher than the real errors at SNR from 2 to 100.

4.1.4 Goal

The main goal here is developing an algorithm for single-molecule super-resolution particle localization. The localization shall give small localization errors. It shall also give an estimation of the localization error, which approximates the real error.

4.2 Center of Mass

The output of particle detection is some pixels with intensity above the threshold. Let P_i be the pixels belonging to the *i*th spot and p_{ij} be the *j*th pixel of the *i*th spot. A commonly used method to find the spot region is to find the local maximum first. With the local maximum as center, a spot region is cut out.

Let p_m be a local maximum in the microscope image. QuickPALM defines the brightest spot in P_i be the local maximum and define a spot width w_{s}. The spot region is a square with a length w_{s} and a center p_m . gSHRImP has a more conventional way of finding local maximum. It takes all the pixels that are brighter than their eight surrounding neighbors as local maximums and defines a spot with radius r_{s}. The spot region includes all the pixels within a distance of r_{s} to p_m. Since a normal distribution will result in a round disk in the image, defining a radius is more reasonable than defining a width. But on the other hand, is more time consuming to cut out a circle than a square. In the center of mass method, the center coordinate is calculated with the intensity of each pixel as the weight (Appendix 4.2.1).

With the interference of the shot noise and the background noise, it`s natural that some of the pixels` intensity are not as reliable as others`. As shown in Figure 4-2-1, the Gaussian-distribution spot has been damaged by the noise.

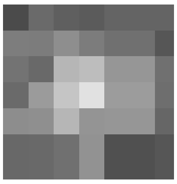


Figure 4-2-1 A spot cut out from the image.

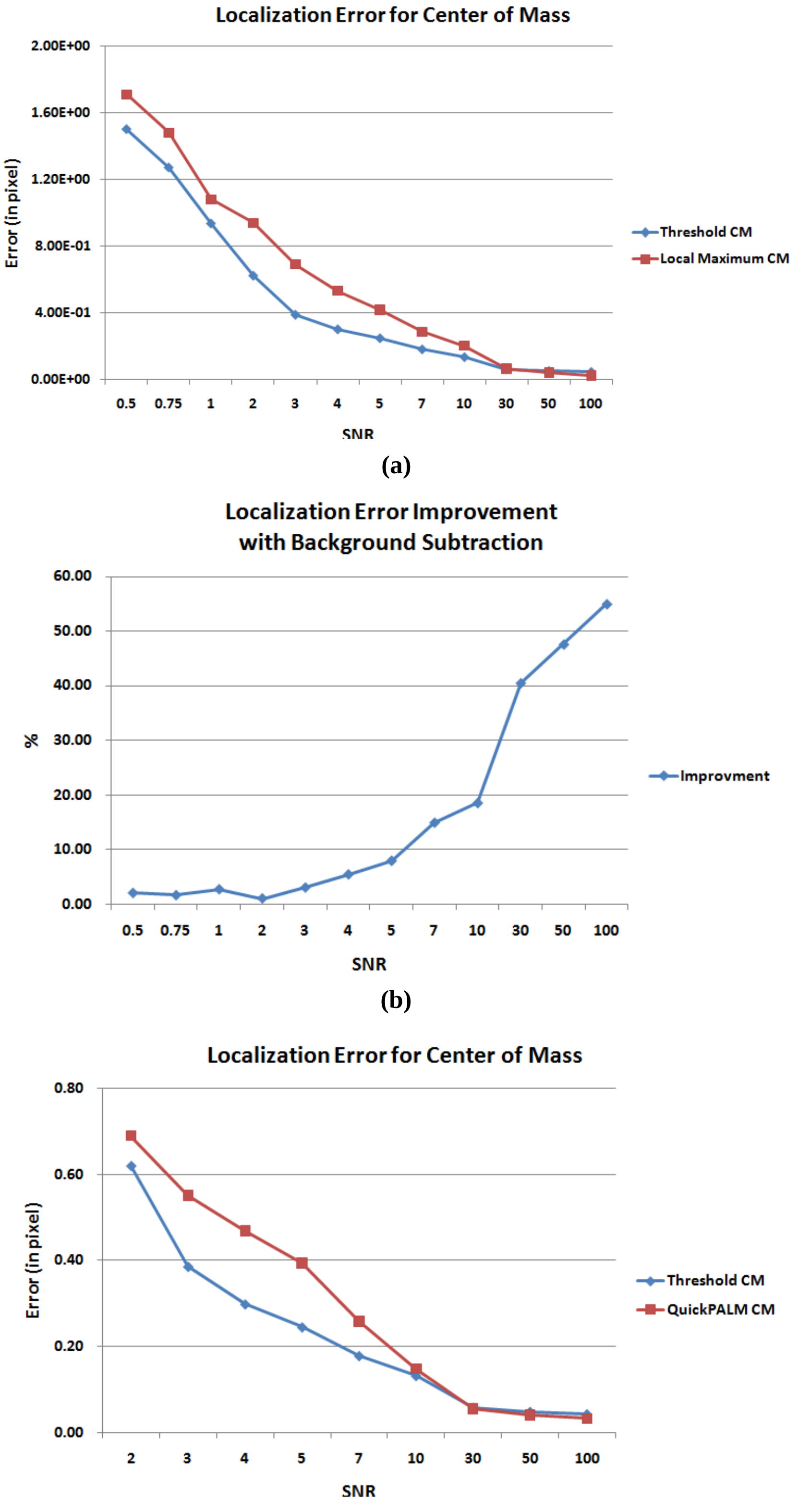
Shot noise is caused by the fluctuation in photon arrival times. The standard deviation of the shot noise is equal to the square root of the photon number. And the number of photons is proportional to the light intensity. Let *s* be the intensity of a pixel. The absolute uncertainty is: $\sigma = k\sqrt{s}$. The relative uncertainty with respect to the intensity is $k/\sqrt{s}$. So the larger the intensity is, the smaller the uncertainty.

For background noise, since the background noise model on each pixel is the same and it`s irrelevant to the intensity of the pixel, the absolute uncertainty caused by background noise is the same on different pixels regardless of the variety of the intensity. So the relative background noise uncertainty is also smaller on a larger intensity pixel.

From all above it can be concluded that with the interference of noise, the pixels with larger intensity are more reliable for localization. If this is true, using only the large-intensity pixels to calculate center of mass will give us a better accuracy at low Signal-to-noise area.

Under this assumption I purpose another approach for calculating center of mass. Instead of finding the local maximum as the spot center, I directly use P_i to calculate center of mass of the *i*th spot. By doing this only the pixels with intensities larger than the threshold are taken into account. This will avoid the main disadvantage of local maximum approach that with the existence of background noise and shot noise, p_m may not be the center of the spot. The result (Figure 4-2-2 (a)) has proven that our assumption is correct. The localization error has been decreased by up to 45% in the interested SNR range. The two curves cross at SNR = 30 and at SNR larger than that, our approach lose its power. That`s because at high signal-to-noise area, the initial spot center defined by local maximum is accurate enough. And since the noise is ignorable compared with the spot signal, the pixels around the local maximum are also reliable.

In equation (A-22) and equation (A-23), the coordinate of each pixel of a spot is multiplied by the weight of this pixel. The weight of a pixel shall be equal to the probability that a photon will hit this pixel, which can be estimated by the number of photons hitting this pixel over the total number of photons in the spot. In a microscopy, the number of photons hitting a pixel is proportionally converted to the intensity of the pixel. When there`s only the spot signal with a completely dark background in the image, taking the pixel intensity over the sum of intensity of the spot as the weight



(a)

(b)

(c)

Figure 4-2-2 (a) Comparison of localization errors of threshold center of mass method and local maximum method. (b) The improvement of adding background subtraction in the center of mass method in terms of localization error. (c) Comparison of localization errors of threshold center of mass method and QuickPALM center of mass method.

will give needed result. However, in a microscope image the background is not completely dark, because of the existence of all sorts of noise. Moreover, these noises change over different parts of the image, which make the background intensities of different parts vary.

To get the proper weight for each pixel, the background intensity has to be subtracted from the pixel intensity. Assume that all the pixels of a spot have the same background intensity, let the intensity of the darkest pixel of a spot be s_l and the background intensity of the spot be s_b. The background intensity of the spot is estimated by:

$$s_b = s_l \tag{4-2-1}$$

This estimation works because the binarization of particle detection separates the spots and the background. Let s_l' be the intensity of the darkest pixel of the spot if there is no noise on the image. One will get:

$$s_l' + s_b = s_l \tag{4-2-2}$$

Because the point spread function of the microscope is a Gaussian function:

$$s_l' \approx 0 \tag{4-2-3}$$

Combining equation (4-2-2) and (4-2-3), equation (4-2-1) can be derived.

The results Figure 4-2-2 (b) shows that at high signal-to-noise condition the background subtraction approach gives better improvement. With high-quality image (SNR = 100), 55% of improvement has been seen. Background subtraction doesn`t show too much of its power at low signal-to-noise condition because compared with the spot signal intensity, the background intensities on the spot pixels are large, resulting in large differences among the background intensities of different spot pixels. And these differences are so large that the assumption that all the pixels of a spot have the same background intensity can no longer be true.

Figure 4-2-2 (c) shows that our threshold approach of center of mass with background subtraction has better localization accuracy than QuickPALM center of mass. Especially within the SNR range of microscope image, the improvement is up to 36% (at SNR = 4).

4.3 Gaussian Mask

Gaussian mask is a simplified Gaussian fitting introduced by Thompson et al in 2002 [24]. Gaussian fitting incorporates not only background noise but also shot noise. But as a simplification of Gaussian fitting, Gaussian mask only incorporates background noise.

Gaussian mask approach defines a two-dimensional Gaussian mask on the spot, with the spot center (x_0, y_0) as the Gaussian center. The values on the cells of the mask follow the Gaussian distribution. It`s a discrete form of Gaussian distribution. It`s generated by integrating the Gaussian function over each pixel. By adjusting the Gaussian distribution model, the Gaussian mask is changed, so that the χ^2 sum is minimized (Appendix 4.3.1). The initial value of (x_0, y_0) will be set to the center of mass of the spot. Iteration after iteration it shall converge and give a more accurate center.

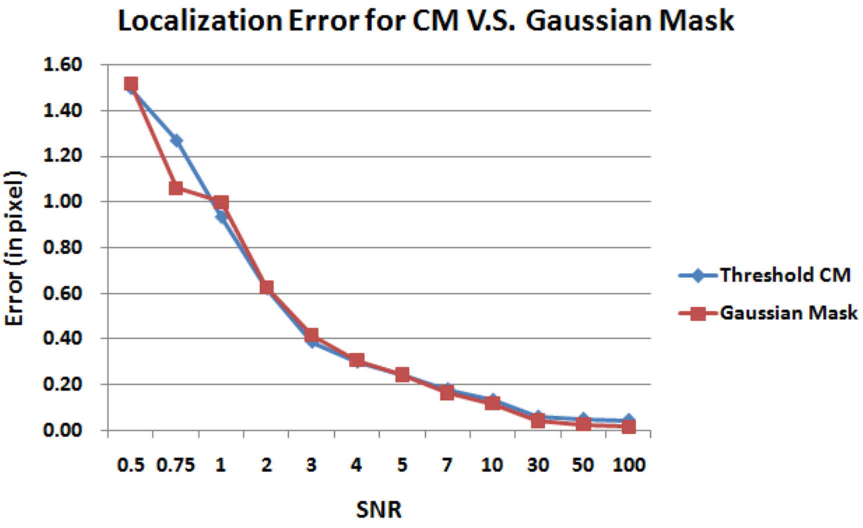


Figure 4-3-1 Comparison of the localization errors of the threshold center of mass method and the Gaussian Mask method.

Comparing the localization errors calculated by the center of mass and the Gaussian mask Figure 4-3-1, it was found that the Gaussian mask doesn`t show any improvement over the center of mass. The center of mass approach already achieves an excellent accuracy, considering its low computation cost. Though the Gaussian mask approach is a simplified Gaussian fitting, it still has a form closer to the center of mass. It updates the coordinate of the spot center with the intensities of the pixels as weights, and incorporates with the discrete Gaussian mask. So the Gaussian mask is actually a sort of adjusted center of mass. Therefore it shall not be surprising to see that the localization error s of the center of mass and the Gaussian mask are so close.

4.4 Gaussian Fitting

4.4.1 Non-linear Least Square Fit

Least square is a widely used method in machine learning to evaluate the fit of a model to the data. When the objective function is non-linear on parameters to be estimated, the widely used fit is a non-linear least square fit. It`s not as commonly used as linear least square fit, which is often seen while fitting a linear regression model.

Here in our application, the objective function of non-linear least square fit has the same form as the point spread function of the microscope, which is approximated by Gaussian function. Because the parameters of the Gaussian functions of different spots vary, the Gaussian functions of different spots vary. So the full non-linear least square fit has to be done for each spot. The fit is a minimization problem of the least square loss function, which minimizes the error between the objective function and the spot pixels signal (Appendix 4.4.1.1).

4.4.2 Levenberg-Marquardt

Commonly used method for solving non-linear least square problem includes gradient descent method, Gauss-Newton method, Levenberg-Marquardt, EM algorithm, etc [28].

Gradient descent method minimizes the least square in the direction along the steepest gradient of the objective function with respect to the parameters. The steepest gradient direction is the direction to the current estimated global minimum. But in general case it is not necessary that the direction of the current steepest gradient will coincide with the shortest direction to the real global minimum.

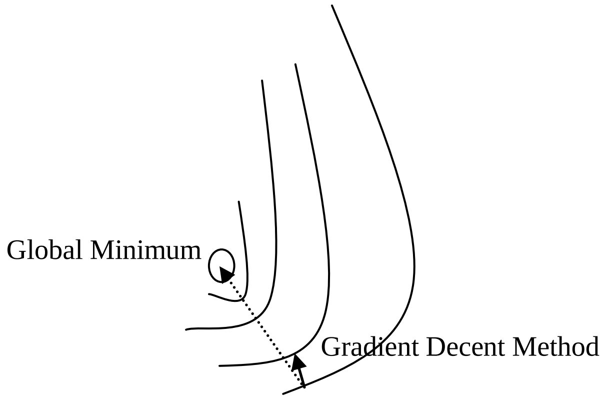


Figure 4-4-2-1 The steepest gradient direction of gradient decent method

Gauss-Newton method updates the parameters with the differentiation equation of the objective function. Compared with gradient descent method Gauss-Newton method is faster but it`s more sensitive to the initial values. The Gauss-Newton method works under the presumption that the current location of the objective function is quadratic in the parameters away from the global minimum, which means that the initial values shall be already close to the optimal values. That explains why the Gauss-Newton method is so sensitive to the initial values.

EM algorithm is a maximum-likelihood based algorithm. The iteration is performed by alternating an expectation estimation step and a log-likelihood maximization step.

A maximum likelihood based method has been proposed [26] and they claimed that their localization error had achieved Cramer-Rao lower bound.

Many existing methods for solving non-linear least square problem don`t provide any way to estimate the localization error. But fortunately Levenberg-Marquardt does. Our algorithm uses Levenberg-Marquardt, mainly because correct analysis requires not only high localization accuracy and short computation time, but also a good estimation of the localization error. Levenberg-Marquardt is a combination of gradient descent method and Gauss-Newton method (Appendix 4.4.2.1). It can adjust the converging step dynamically, avoiding trapping into local minimums. Existing implementation has provided a variety of options for improving the algorithm, such as weight fitting, different Jacobian approximation methods, independent variable scaling, constrained minimization, etc. Moreover, it`s fast compared with many other methods. As mentioned before, the number of particles to be localized for a super-resolution reconstruction can be hundreds of thousands, which makes the time issue critical. A fast-converge least square fit method will remarkably improve the speed. In practice, Levenberg-Marquardt has become a standard method for non-linear least square problem.

A damping coefficient λ is introduced Levenberg-Marquardt algorithm. In equation (A-36), with the addition of the term $\lambda\boldsymbol{I}$, when the current location is far away from the global minimum, λ is relatively large and the method will act more like the gradient decent method. When the current location is near the global minimum, the presumption of the Gauss-Newton method is valid. λ will decrease and the method works like the Gauss-Newton method.

The square errors of the estimated parameters are at the diagonal of the covariance matrix, which is convenient for localization error estimation (Appendix 4.4.2.2).

4.4.3 Background Parameter

Unlike the center of mass approach and the Gaussian Mask approach, Gaussian fitting approach doesn`t require background subtraction before localization starts. In the center of mass approach and the Gaussian Mask approach, the background value of a spot is pre-calculated, and the value is subtracted from the intensity of each pixel of the spot. However, in the Gaussian fitting approach, the background value can be incorporated in the Gaussian fitting model, so there is no need to subtract the background value beforehand.

Many existing super-resolution techniques setup the Gaussian fitting model without considering the background. As in gSHRIMP, the fitting model is:

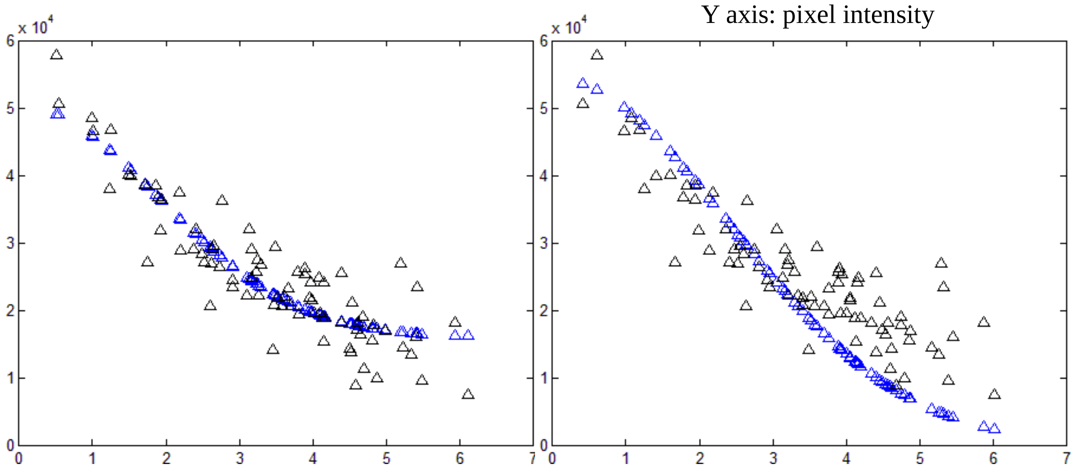
$$g_{gS}(x_i, y_i) = A \cdot e^{-\frac{1}{2\sigma^2}[(x_i - x_0)^2 + (y_i - y_0)^2]} \qquad (4\text{-}4\text{-}3\text{-}1)$$

Compared with the model mentioned before Equation (A-29), which is also used by RapidStorm, the background parameter A_0 is missing in the gSHRIMP model. Without the background parameter, the dark current is not incorporated. The existence of background parameter in the model implies the assumption that all the pixels of a spot have the same background value, as it was proposed before in the section Center of Mass.

Figure 4-4-3-1 shows the fitting with and without the background parameter. Figure 4-4-3-2 shows the localization error given by these two different models. The Gaussian fitting is performed with the Levenberg-Marquardt library Levmar. Figure 4-4-3-3 shows the improvement on localization error after applying background parameter in the Gaussian model.

X axis: distance to the spot center

Y axis: pixel intensity



With Background parameter

Without Background parameter

Figure 4-4-3-1 The best fit of Gaussian function to a synthetic generated 2D spot. X-axis represents the distance from the data point to the spot center. Y-axis represents the intensity of the data point. The blue triangles are the Gaussian function data points and the black triangles are the spot data points. Since 2D Gaussian function is used, the data points of it shall outline a half-1D-Gaussian-curve.

From Figure 4-4-3-1 and Figure 4-4-3-2, it`s obvious that with the background parameter the data is better fitted to the Gaussian model. Again from the improvement (Figure 4-4-3-3) it was found that in low signal-to-noise region the background parameter doesn`t have too much power. This agrees with the conclusion made in the section Center of Mass -- in low signal-to-noise region the amplitude of the noise is too large, resulting in large different in the background values of different pixels of a spot, so the assumption that all the pixels of a spot have the same background value is not as appropriate as in high signal-to-noise region.

It`s convenient to set a background parameter in the Gaussian model for handling background noise, but subtracting the background value before the localization (as it was done for the Center of Mass approach and the Gaussian Mask approach) is also possible for the Gaussian fitting approach. The only different is how the background values are acquired. The background parameter estimates the background value by fitting. The background subtraction takes the minimum intensity of all the pixels as the background value. The background value estimated by fitting shall be larger than the minimum intensity of the pixels.

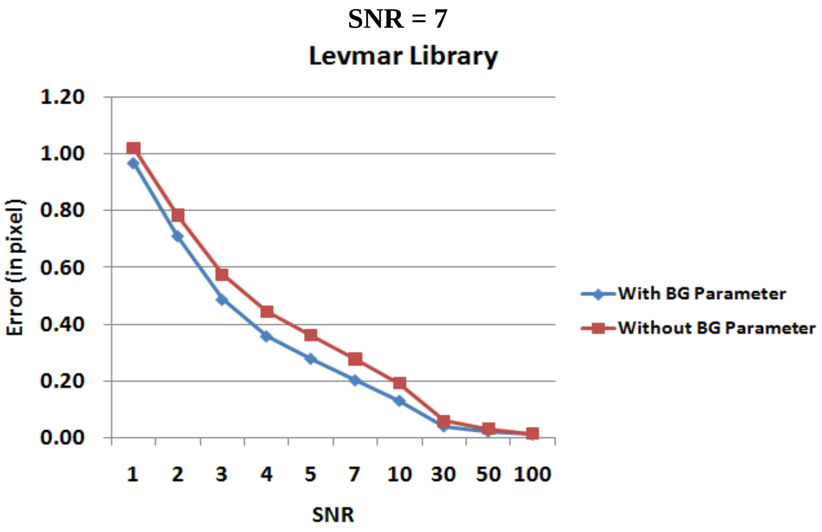


Figure 4-4-3-2 The comparison of the localization errors of background parameter particle localization and non-background-parameter particle localization. The fitting is implemented with Levmar library.

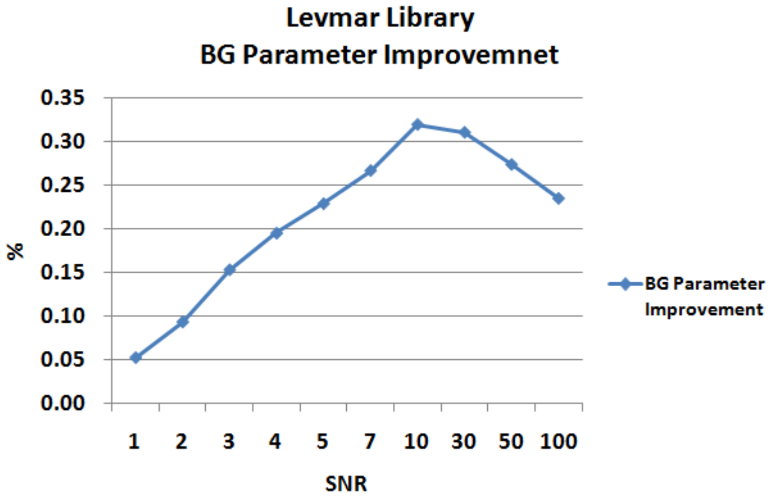


Figure 4-4-3-3 The improvement of the localization error after adding background parameter estimated in particle localization. The fitting is implemented using Levmar library.

For comparison of the background parameter and the background subtraction, Figure 4-4-3-4 shows the ratio of the background subtraction localization error and the background parameter localization error. The result shows that the background subtraction has a smaller localization error than the background parameter at signal-to-noise ratio smaller than 10.

At high signal-to-noise ratio, the background parameters estimated by the two approaches are extremely close. It can be seen that the ratio of the background subtraction localization error and the background parameter localization error is almost 1. As the signal-to-noise decreases, the amplitude of the noise increases, leading to larger uncertainties of the pixels` intensities. So the background value estimated by the background parameter gradually differs from that of the background subtraction`s, which is supposed to be a good estimation of the background value. Therefore, the background subtraction approach begins to outperform the background parameter approach. As the signal-to-noise reaches 3 and continues to decrease, the assumption that supports the background subtraction is not valid anymore. The improvement of the background subtraction approach compared with the background parameter approach reduces, so the background subtraction-background parameter ratio increases again.

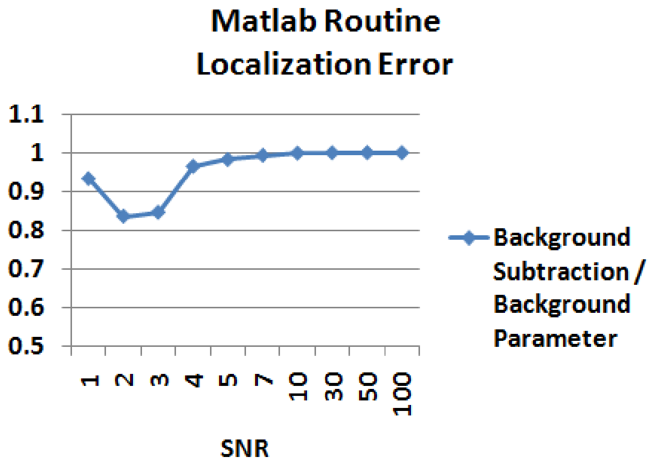


Figure 4-4-3-4 The ratios of the localization errors using background subtraction and particle localization with background value as a parameter.

The test above clearly shows that the background subtraction approach gives better localization accuracy than the background parameter approach. So instead of doing the fitting with the model of Equation (A-21), Equation (4-4-3-1) shall be used, and a background subtraction step shall be added before the localization.

4.4.4 Weight

It is mentioned before in the section Center of Mass that the intensity of a pixel represents the number of photons hitting that pixel. And the intensity is proportional to that number. Each pixel of the spots is a sample data of the spot signal model. However, in a series of repeat samplings, the measured intensities at the same pixel are not always the same [27]. This variation of measured intensities at different samplings is caused by the shot noise.

The shot noise has casted uncertainty on the intensity of each pixel. If s is a measurement of the intensity at a pixel, the uncertainty can be measured by the standard deviation of the intensity (Appendix 4.4.4.1).

From equation (A-38) follows that a data point with a large intensity will have a small relative uncertainty, giving us more confidence on the signal.

Since the data points have different quality, introducing weights in the fitting model may enable the data points with high quality to have more influence on the parameter estimation than the data points with low quality. The weight of a data point shall correspond to the confidence of the data point`s intensity (Appendix 4.4.4.2). By giving different weight to the residual of each data point, these data points will have different influences on the parameter estimation.

In theory, with weight fitting, the localization shall be more precise.

Figure 4-4-4-1 shows that using a fitting with weights, the localization precision calculated with ground true barely improves, especially at low signal-to-noise level, which means the real localization precision doesn`t improve. However, the localization error estimated by the covariance matrix does reduce, as it is expected. A better covariance estimate error may indicated that weight fitting has improved the goodness of the fitting itself. However, if one considers the χ goodness (Appendix 4.4.4.3) of the fitting with and without the fitting (Figure 4-4-4-2), it is found in agreement with the covariance estimate error that the fitting has improved with the existence of the weight.

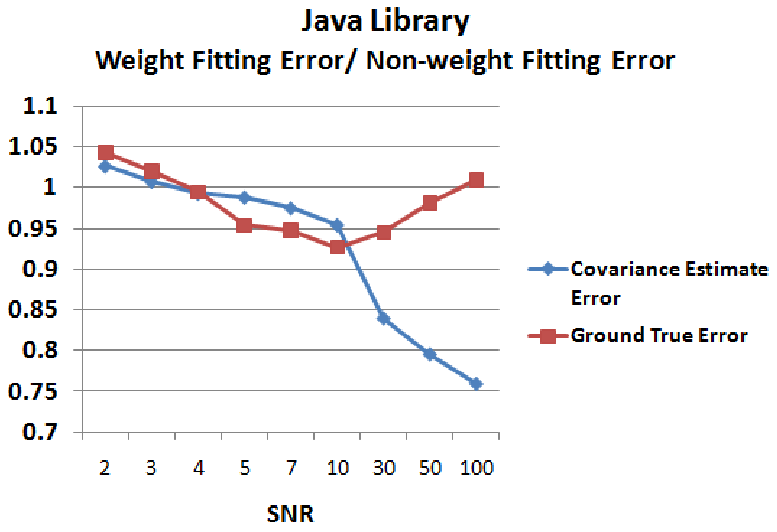


Figure 4-4-4-1 The ratios of the localization errors of weighted and non-weighted fitting. The y-axis represents the ratio of the corresponding errors of weighted fitting and non-weighted fitting.

The goodness is evaluated by the fitting process itself, so it could be understood that when the goodness has reduced, the covariance estimated error will also reduced. Small values of χ goodness indicate that the data is well fitted to the model. In Figure 4-4-4-1 and Figure 4-4-4-2 it is shown that the covariance estimate error and the chi goodness of the fitting have showed a conflict with the real error.

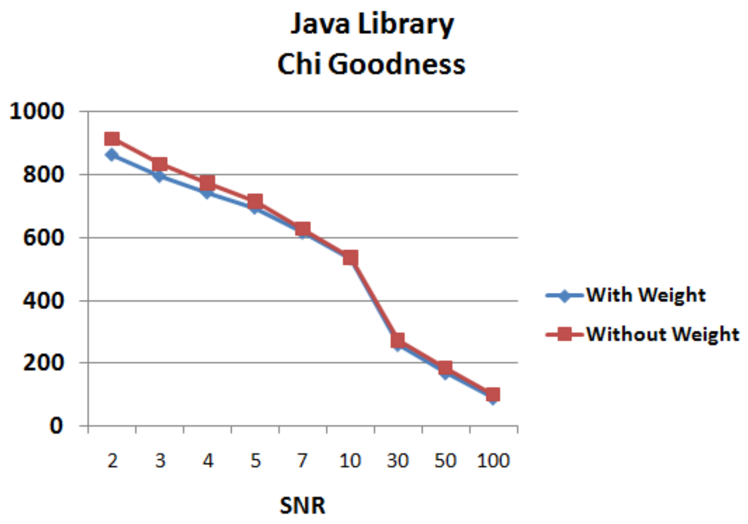


Figure 4-4-4-2 Dependence of χ goodness of the fitting with weight and the fitting without weight on SNR.

Now let's look into the detail of how the covariance matrix is computed and how the estimated parameters are updated. To add weight in the fitting, we have to do some change to the Equation (A-36). The equation for updating the estimated parameters will be:

$$[\boldsymbol{J}^T\boldsymbol{W}\boldsymbol{J} + \lambda\boldsymbol{I}]\boldsymbol{h}_{gn} = \boldsymbol{J}^T\boldsymbol{W}(\boldsymbol{g} - \boldsymbol{S}) \qquad (4\text{-}4\text{-}4\text{-}1)$$

And the covariance matrix calculated with weight is:

$$\boldsymbol{C} = [\boldsymbol{J}^T\boldsymbol{W}\boldsymbol{J}]^{-1} \qquad (4\text{-}4\text{-}4\text{-}2)$$

The diagonal elements of the covariance matrix, namely the variances of the estimated parameters can also be represented in this form [28]:

$$\sigma^2(p_j) = \sum_{k=0}^{n-1} \sigma_i{}^2 \left(\frac{\partial p_j}{\partial s_i}\right)^2 \qquad (4\text{-}4\text{-}4\text{-}3)$$

s_i is the intensity of the *i*th data point of the spot and p_j is the *j*th estimated parameter. σ_i is the absolute measurement error of the intensity of the *i*th data point.

In Equation (4-4-4-1) it can be seen that the data points with larger weight will play more important roles in updating the new parameters. These data points with larger weight are pixels that have smaller uncertainty at their intensities. The fitting will take the differences between the intensities of these pixels and the value given by the Gaussian model at these pixels as priority. So the updating of the estimated parameters will give smaller differences at pixels with larger weights. When the covariance matrix is calculated, it also give more weight to these pixels, whose intensity have smaller differences with the value given by the Gaussian model. So again, these data points with smaller errors play more important roles on the total errors of the estimated parameters than other data points, and the data points with larger errors contribute less in the total errors, resulting in smaller total errors of the estimated parameters than when all the data points have equal weights. So the covariance estimate error can be smaller while the real error is actually larger.

The weight fitting is supposed to have better performance than the fitting without weight. Since the covariance estimate error and the chi goodness agree that weight fitting has improve the data fitting, it can be believed that the fitting itself has improved, in the sense of minimize the loss function. But the real localization error is not improved as expected. A possibility is that the fitting is not improved in the expected way. It doesn't direct the fitting to the real spot center.

In Figure 4-4-4-1, it is shown that for both the covariance estimate error and the ground true error, at low signal-to-noise ratio the weight fitting gives especially bad result. At higher signal-to-noise level, weight fitting actually gives slightly improvement on the ground true error. We may consider that at low signal-to-noise level, evaluating the uncertainty of the data point only with shot noise is not appropriate. Actually, at low signal-to-noise level, the background noise will dominate the uncertainty of a data point.

Since weight fitting doesn`t give considerable improvement on the localization precision, we apply uniform weight in our fitting.

4.4.5 Jacobian evaluation

When all the steps of the algorithm are settled, the least-square fitting options themselves are the main factors that affect the localization precision. As a matter of fact, the Levenberg-Marquardt algorithm provides a variety of options for different applications.

During fit iterations Equation (A-36) is solved to get the perturbation of the estimated parameters. But in the equation the Jacobian matrix ***J*** is unknown. Before we are able to solve this equation, we have to evaluate the Jacobian matrix, i.e. get derivatives of optimization function with respect to parameters.

There are two ways to evaluate the Jacobian matrix. The first one is called the analytic Jacobian, which takes the user defined partial derivative functions to evaluate the Jacobian matrix in the iteration. The elements of the Jacobian matrix are calculated with these derivative functions, when the information of the data points is provided. The second way is called the difference approximated Jacobian. It approximates the partial derivatives dynamically by calculating the difference between the consecutive iterations. No explicit partial derivative function is need. Actually, the derivative is approximated by computing the slope defined by the point of the current iteration and the point of the previous iteration, instead of computing the tangent line of the current point, which is the derivative at the current position [29].

Since the difference approximated Jacobian calculates the approximation of the partial derivatives, while the analytic Jacobian gives the exact value of partial derivative, we may expect that the Jacobian matrix evaluated by the analytic Jacobian should be closer to the Jacobian matrix we need. On the other hand, the difference approximated Jacobian approximates the partial derivatives with the dynamic values calculated in the iterations. It shall be more reliable than the values computed by the derivative function of a theoretical objective function.

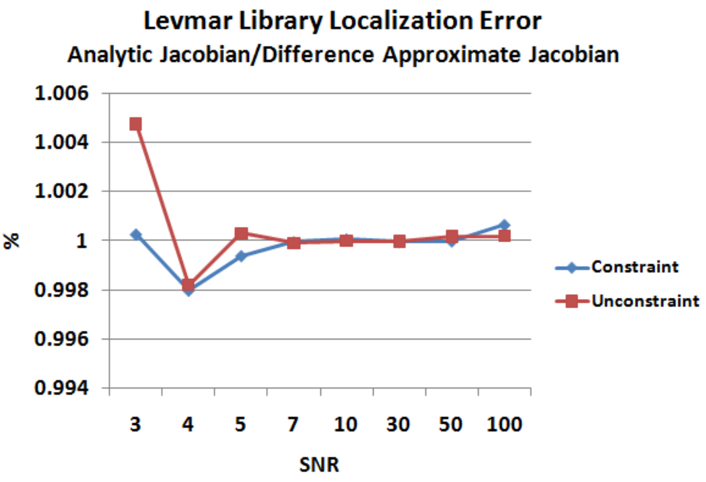


Figure 4-4-5-1 The ratio of localization errors of the fittings with analytic Jacobian and the fitting with difference approximate Jacobian. The red curve is the fitting with constraint on estimated parameter and the blue curve is the fitting without constraint. The y-axis represents the ratio of the corresponding errors of Analytic Jacobian difference Jacobian.

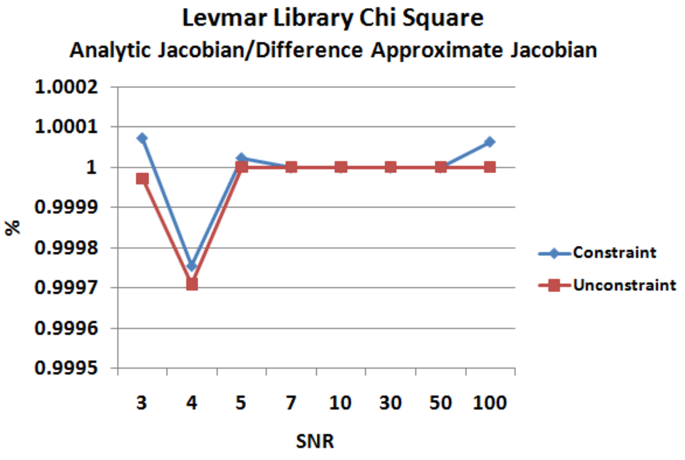


Figure 4-4-5-2 The ratio of χ goodness of the fittings with analytic Jacobian and the fitting with difference approximate Jacobian. The red curve is the fitting with constraint on estimated parameter and the blue curve is the fitting without constraint.

Figure 4-4-5-1 compares the localization error of fittings using analytic Jacobian and difference approximated Jacobian.

The analytic Jacobian tends to outperform the difference approximated Jacobian as the signal-to-noise ratio increase and becomes worse than the difference approximated Jacobian at lower signal-to-noise levels. But when the image quality is extremely bad, the situation becomes complicated and none of them is better than the other. The chi square goodness agrees with the result given by comparing the localization errors (Figure 4-4-5-2).

To study more about this result we have to go back to the theory. Equation (A-33) and Equation (A-35) show how the perturbations are computed for the gradient decent method and the Gauss-Newton method. For the gradient decent method, the steepest decent direction $\boldsymbol{J}^T(\boldsymbol{g} - \boldsymbol{S})$ is derived by the giving the derivative of the chi square a zero value:

$$\begin{aligned}\frac{\partial\chi^2}{\partial\boldsymbol{p}} &= (\boldsymbol{g}-\boldsymbol{S})^T\frac{\partial(\boldsymbol{g}-\boldsymbol{S})}{\partial\boldsymbol{p}} \\ &= (\boldsymbol{g}-\boldsymbol{S})^T\frac{\partial\boldsymbol{g}}{\partial\boldsymbol{p}} \\ &= (\boldsymbol{g}-\boldsymbol{S})^T\boldsymbol{J} \qquad \text{(4-4-5-1)}\end{aligned}$$

So the Jacobian matrix shall be evaluated by:

$$\boldsymbol{J} = \frac{\partial\boldsymbol{g}}{\partial\boldsymbol{p}} \qquad \text{(4-4-5-2)}$$

For the Gauss-Newton method, the Taylor expansion for deriving Equation (A-35) is:

$$\begin{aligned}\boldsymbol{g}(\boldsymbol{p}+\boldsymbol{h_{gn}}) &\approx \boldsymbol{g}(\boldsymbol{p}) + \frac{\partial\boldsymbol{g}}{\partial\boldsymbol{p}}\cdot\boldsymbol{h_{gn}} \\ &= \boldsymbol{g} + \boldsymbol{J}\boldsymbol{h_{gn}} \qquad \text{(4-4-5-3)}\end{aligned}$$

Note that the first line is an approximation. Actually if the analytic Jacobian on the right side is replaced by the difference approximated Jacobian, the left side is equal to the right side.

Therefore, for the decent gradient method it`s better to use the analytic Jacobian and for the Gauss-Newton method, the difference approximated Jacobian will be more appropriated.

When the signal-to-noise ratio is high, as the fitting start, the damping coefficient will decrease fast, and the fitting acts more like the Gauss-Newton method. So the difference approximated Jacobian is expected to give smaller localization error. But since the image quality is good, there will not be much error difference between the two methods. The localization will finally stop at where it shall be for both methods. Therefore at high signal-to-noise level the analytic Jacobian and the difference approximated Jacobian give very close errors. When the signal-to-noise ratio is low, the spot model is not so smooth. Hence the damping coefficient may increase frequently, slowing down the speed of convergence. Compared with at higher signal-to-noise level, the fitting acts more like the gradient decent method. So the analytic Jacobian may sometimes give better localization precision. But when the image is noisy, the prediction of global minimum is not so reliable. The gradient decent method may also give bad result. So at very low signal-to-noise ratio we cannot expect one method will obviously perform better than the other.

4.4.6 Parameter Constraint

During the fitting, it is possible to restrict the values of the estimated parameters, so that they will not go too far to some impossible values. It is especially useful when the image signal-to-noise ratio is very low. It prevents the fitting from trapping in a local minimum that`s far away from the expected range.

With parameter constraint, whenever an updated parameter is out of the range during the fitting, the updated parameter will be bounded to the border of the range.

From the localization errors (Figrue 4-4-6-1) and covariance estimated errors (Figrue 4-4-6-2) of fittings with and without parameter constraint, we can see when the signal-to-noise ratio is lower than three, the fitting with parameter constraint helps avoiding huge localization errors. But if we compare the chi goodness of the fittings with and without parameter constraint, we will find that in contrast with the results shown by the localization error and the covariance estimated error, the chi goodness of the fitting without parameter constraint is better than the fitting with parameter constraint. Actually, with parameter constraint, the constraint will restrict the values the parameters can be, hence eliminating some of the possible values that will give lower chi square error. Therefore the chi goodness of constraint fitting is worse while the parameter constraint help decreasing the localization error. At high signal-to-noise level, because the distribution of the data signals are so good that the fitting will barely go to the positions that are out of bound, the fitting with parameter constraint will not show much power.

In the unconstraint parameter test we find that at signal-to-noise ratio lower than or equal to one, the localization error is always more than one and sometimes it goes to more than a thousand. However, in Figure 4-2-2 (a) we can see the localization errors given by the center of mass method at low signal-to-noise ratios are still less than 1.6 pixel. A solution to overcome the large localization errors at low signal-to-noise level is combining the center of mass method and the fitting method. At low signal-to-noise ratio, the center coordinates given by the Levenberg-Marquardt could be compared with the center coordinates given by the center of mass method. If their difference is larger than a certain pixel number, we may get the conclusion that the Gaussian fitting has failed and instead the center of mass coordinates is more reliable.

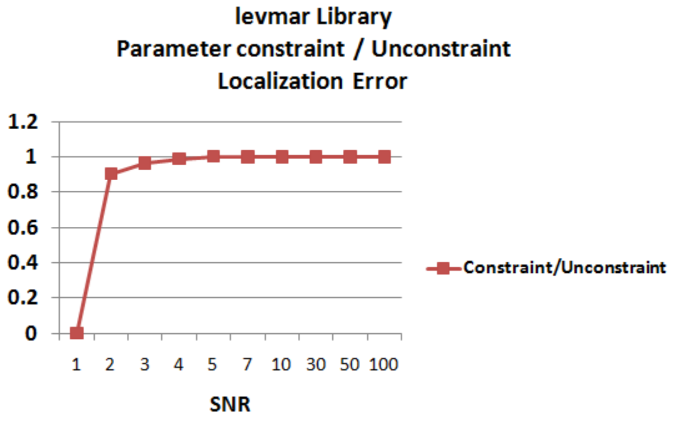


Figure 4-4-6-1 The ratios of the localization errors of the fitting with parameter constraint and the fitting without parameter constraint.

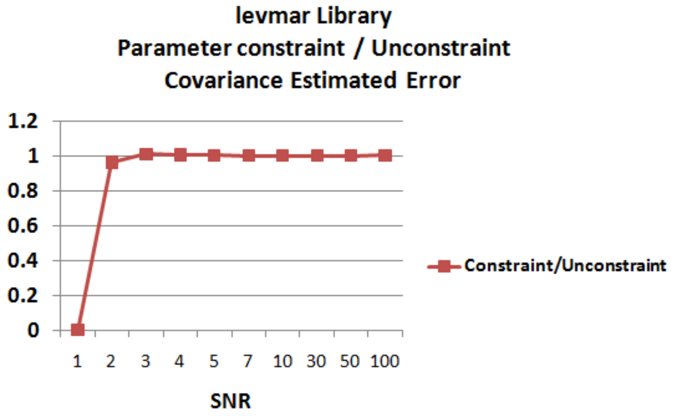


Figure 4-4-6-2 The ratios of the estimated localization errors of the fitting with parameter constraint and the fitting without parameter constraint.

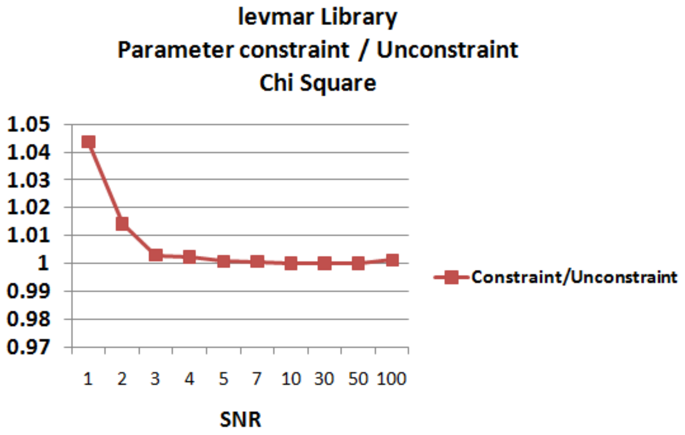


Figure 4-4-6-3 The ratios of the χ squares of the fitting with parameter constraint and the fitting without parameter constraint.

4.4.7 Covariance Estimated Error

Estimating the localization error is one of our main goals. Fortunately, Levenberg-Marquardt provides the covariance matrix recording the variance of each estimated parameter. The localization precision of a estimated parameter will be the square root of the parameter`s variance.

Considering our application, an ideal covariance estimated error shall approximate the real localization error, and at the same time it shall be the upper bound of the real localization error.

We test with difference Levenberg-Marquardt libraries on how good the covariance matrix can achieve in estimating the localization error. The result is quite satisfying. All of them give covariance estimated errors slightly more than the "ground-truth" localization errors.

The Levmar C++ library and the Matlab routine both give estimation errors of less than 12 percents within the signal-to-noise range of [2, 100]. The LMA Java library gives up to 58 percents` estimation error at high signal-to-noise ratio, but we usually will not achieve so high a signal-to-noise ratio with the microscope, so even the estimation given by LMA Java library is acceptable. All of the three libraries share the same characteristic: the estimation errors increase as the signal-to-noise ratio increases.

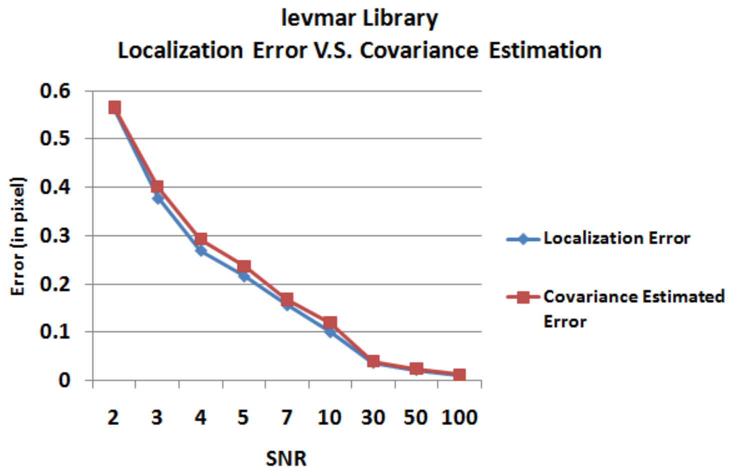


Figure 4-4-7-1 The comparison of the ground-truth localization errors and covariance estimated error of the particle localization at different SNR. The fitting is implemented with Levmar Library.

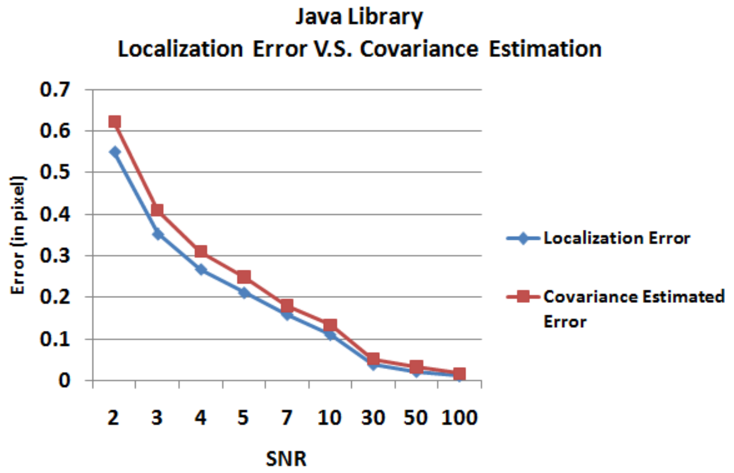


Figure 4-4-7-2 The comparison of the ground-truth localization errors and covariance estimated error of the particle localization at different SNR. The fitting is implemented with Java LMA Library.

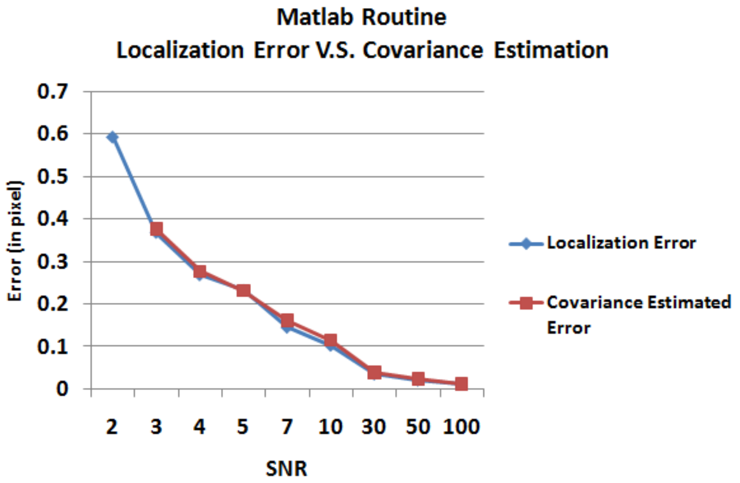


Figure 4-4-7-3 The comparison the ground-truth localization errors and covariance estimated error of the particle localization at different SNR. The fitting is implemented with the Matlab routine.

4.5 Drift Correction

4.5.1 Background and Motivation

A critical issue of super-resolution reconstruction is the drift correction. During the whole period of the time-lapse images, the specimen may have continuous drift on x-axis or on y-axis. The main cause of the drift is the temperature variation, which is induced by the light source, the change of the room temperature and other factors that may affect the temperatures of the microscope and the materials. As the microscope cools down or warming up, the drift goes in one direction and another.

For conventional microscope imaging, the drift will not cause any problem because an image is taken within a short period. The drift will be too trivial to lower the revolution of the images. However, for single-molecule super-resolution imaging, obtaining all the fluorophores in the specimen by only allowing sparse particles in the images, the experiment will take up to several minutes, which causes considerable drift.

The drift produces different errors on the localization at different frame of the microscopy images. As time passes the cell structure floats on the images, producing thick lines on the reconstruction image, hence reducing the resolution (Figure 4-5-1-1).

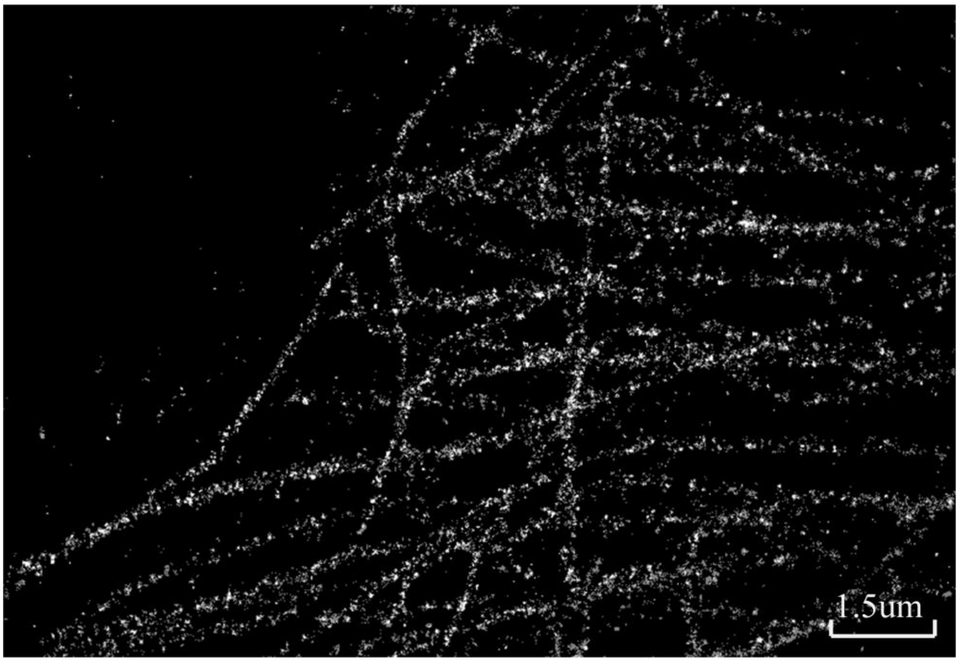


Figure 4-5-1-1 The super-resolution reconstruction image of a specimen of microtubules without drift correction. Because of the drift, the width of the microtubules is larger than it shall be.

Drift correction estimates the displacement of the images taken at different times and corrects the localization positions.

A common and reliable method to track the drift is introducing markers in the specimen. Unlike the fluorophores in the images, which blink in the whole period of the imaging, the markers appear in each frame and it is always at the same relative position in the specimen. The constant existence of the markers enables us to track the trajectory of them. Since the drift is mainly in the horizontal direction and the vertical direction in the x-y plane, we could assume that the trajectory of the markers is also the trajectory of the drift. Despite the high precision of drift estimation of this method, introducing markers may interfere the imaging of the cell structures. To solve the problem without introduction of markers, so called “period drift correction” was developed [31]. The whole microscopy images stack is divided into many periods. For each period T_k, a reconstruction image I_k is generated with all the particles switching on in this period. The lengths of these periods are chosen that the number of stochastic particles in each period is large enough to reconstruct the general cell structure of the specimen. The first period T_0 is the reference period, whose drift is zero. For a period T_k ($k \neq 0$), the drift of period T_k is estimated by finding the maximal cross-correlation function of I_0 and I_k`s shifted images. This method is generalized but the effectiveness has some dependency on the particles number in each period. The optimal particles number in each period varies for different specimen, depending on the complexity of the cell structure.

In this section we test our drift correction algorithm with a training set of dSTORM microscopy images of labeled microtubules. There are 40,000 frames in total and every 2,000 frames form a drift correction period.

It should be noted that results presented below are very preliminary, since they were tested only on one dataset.

4.5.2 Cumulative Drift Approach and Direct Drift Approach

There are two ways of calculating the drift of a period T_k. The first one is direct drift approach, which directly calculating the absolute drift of T_k with T_0 as the reference. The second one is cumulative drift approach that calculates the relative drift of T_k with T_{k-1} as the reference, which is obtained by finding the maximal cross-correlation function of I_{k-1} and I_k`s drift images. And then accumulate the relative drifts from T_0 to T_k.

The advantage of the cumulative drift approach is that the amount of drift is expected to be much smaller for consecutive periods than for a random period and T_0. Smaller potential drift means smaller searching window for the maximal cross-correlation function. Defining a square searching window, if the length of the searching window for the cumulative drift approach is l, the length of the searching window for the direct drift approach shall be $(n-1)\ l$, where n is the total period number. With a brute force algorithm the complexity of the direct drift approach is $(n-1)^2$ times of the complexity of the cumulative drift approach. Using the cumulative drift approach would save a large amount of time. However, a concern about the cumulative drift is, with the same drift estimation precision, smaller drift may yield larger relative error of drift estimation. As the cumulative drifts accumulate for getting the absolute drift, the errors may also accumulate.

However, in our tests we find that the drift correction of the direct drift approach doesn`t give clearer reconstruction image of the whole specimen than the drift correction of the cumulative drift approach. We compare the width of line segments in the same region of the two drift corrected reconstructed images (Figure 4-5-2-1). The direct drift approach gives slighter thinner line segments than the cumulative drift approach, but the little improvement doesn`t worth such a huge computation cost. For shorter computation time, the cumulative drift approach shall be used.

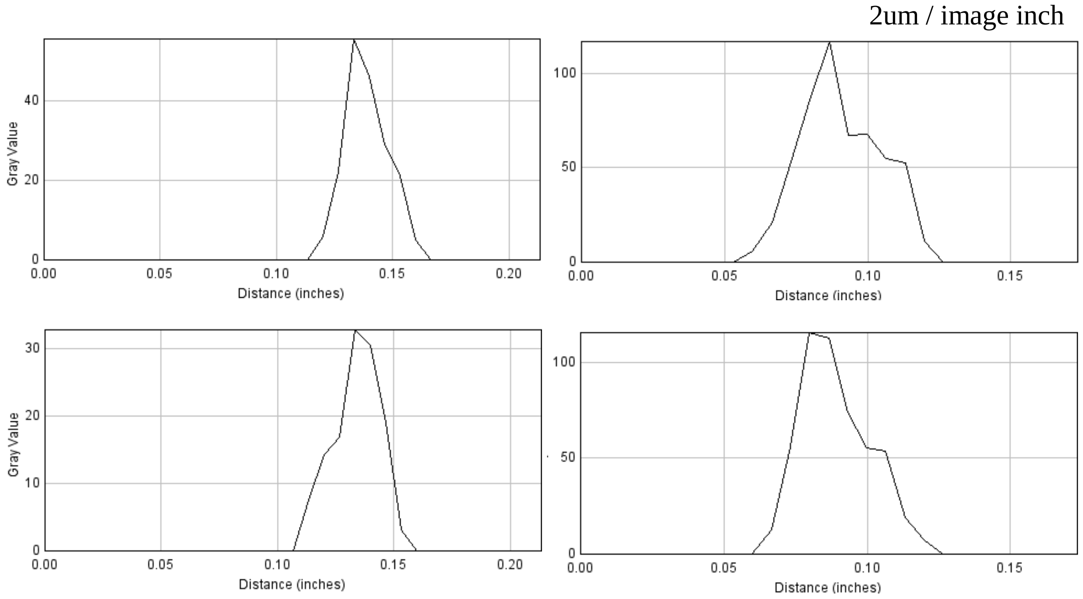


Figure 4-5-2-1 Comparison of the widths of the microtubules` line segments in the drift corrected reconstructed images. Each plot shows the integrated intensity of a 15-pixel-length line segment along the width. The x-axis represents the distance along the line width and the y-axis represents the integrated intensity of 15 pixels. The plots on the first row are the integrated intensity of two random line segments in the cumulative drift reconstruction image. The second row shows the integrated intensity of the same line segments in the direct drift reconstruction image. The plots on the same column represent the same line segment from different drift reconstruction images.

4.5.3 Drift Linear Interpolation

As stated before, the drift of the specimen is a continuous process. Drift happens within a period as well as between different periods. However, the drift correction algorithm we use only calculates the drift for each period. Applying uniform drift correction to all the frames within a period will imply that the drift happens only at the border of each pair of consecutive periods like a step function. The error of drift correction will be at least the drift of half a period. To rectify this problem, linear interpolation is applied for drift correction of each frame.

Assume that the number of frames in each period is *N* and the drift of period *T* is *drift*(*T*).

For the *n*th frame of period *T*, if $n \leq N/2$, its drift is:

$$drift(T,n) = drift(T-1) + \frac{n+\frac{N}{2}}{N} \cdot \left(drift(T) - \ drift(T-1)\right) \quad (4\text{-}5\text{-}3\text{-}1)$$

If $n > N/2$, the drift is:

$$drift(T,n) = drift(T) + \frac{n-\frac{N}{2}}{N} \cdot \big(drift(T+1) - \, drift(T)\big) \qquad \text{(4-5-3-2)}$$

So if *n* = *N* /2, *drift*(*T, n*) exactly equals to *drift*(*T*).

Linear interpolation works under the assumption that the overall drift of period T is exactly the same as the drift of the frame at the half of the period, and the specimen is moving linearly between each two consecutive periods.

2um / image inch

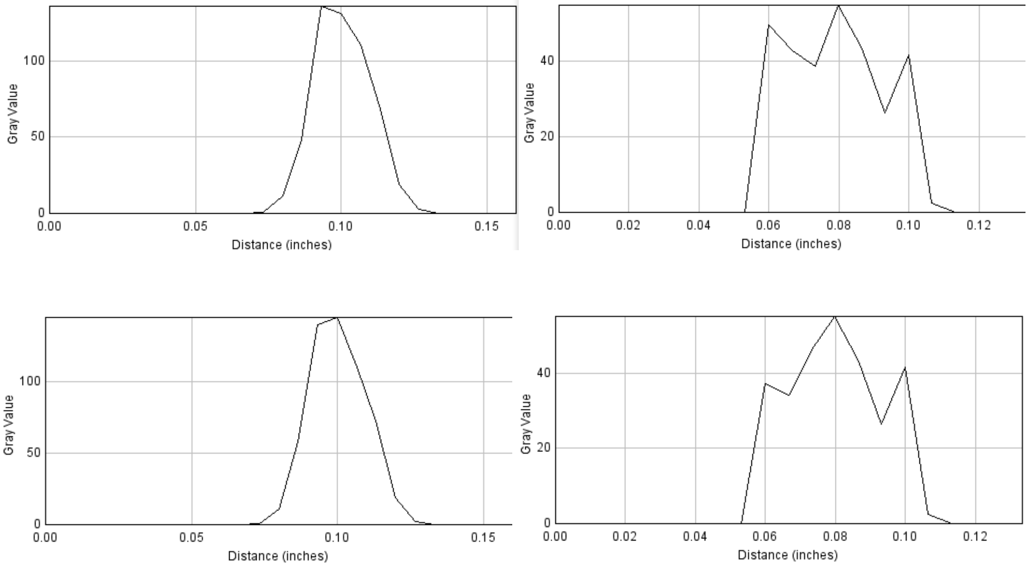


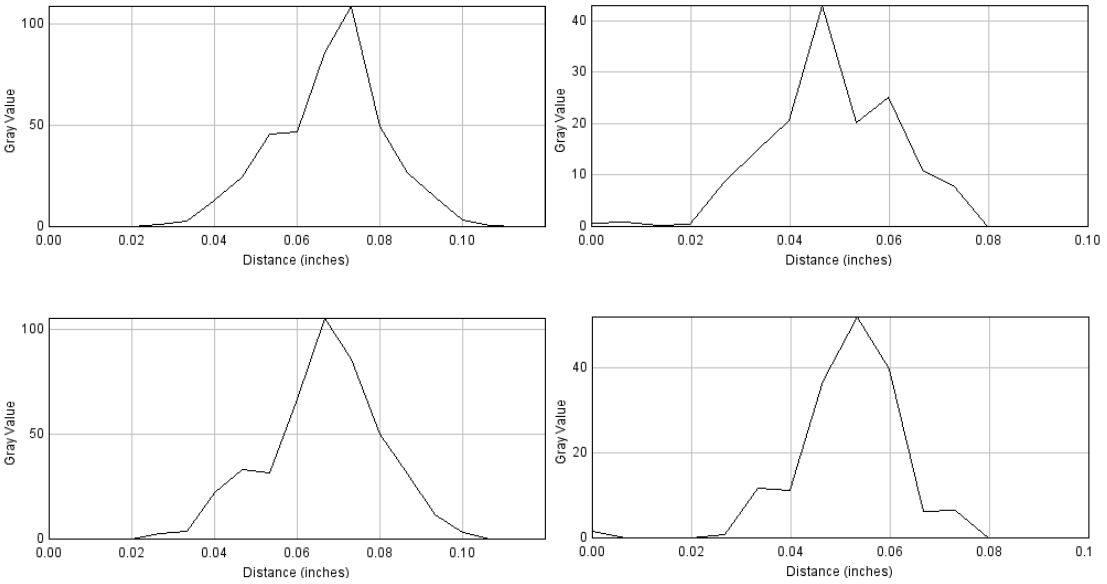
Figure 4-5-3-1 Comparison of the integrated intensity of segments from the uniform drift reconstructed image and linear interpolation drift reconstructed image. The first row is for uniform drift and the second row is for the linear interpolation drift.

In Figure 4-5-3-1 we find that with our only microscopy image dataset the improvement of applying linear interpolation is extremely subtle. We need to have more samples to make a better conclusion.

4.5.4 Particle-dependent Period

The reason why the drift shall be estimated at each period instead of at each frame is the cross-correlation function shall be calculated for two images that displays the same cell structure. Only with reconstructed images that contain large amounts of particles could the general cell structure be displayed. Considering that the particles switch on stochastically and the number of particles varies from frame to frame, it shall be more appropriate to define the length of each period with the number of particles instead of the number of frames.

Again we test these two approaches with our microtubule images stack. For time-dependent period drift, each 2,000 frames form a period. With our 40,000 frames there are 20 periods. For particle-dependent period drift, there are roughly 440,000 particles in the image stack. To keep the periods number the same as the time-dependent period drift, we define 22,000 particles a period. Figure 4-5-4-1 compares the integrated intensity of segments from the same regions of the time-dependent period drift reconstructed image and the particle-dependent period drift reconstructed image.



2um / image inch

Figure 4-5-4-1 Comparison of the integrated intensity of segments from the time-dependent period drift reconstructed image and particle-dependent period drift reconstructed image. The first row is for time-dependent period drift and the second row is for the particle-dependent period drift.

In our sample we could see very small improvement from the plots of particle-dependent period drift reconstructed image.

4.6 Experiment Result

In this chapter we have discussed three different particle localization approaches - the center of mass method, the Gaussian mask method and the Gaussian fitting method. It has been shown before that our threshold center of mass method with background subtraction considerably outperforms the regular center of mass method at low signal-to-noise ratios, because it only considers the data points with high quality. The Gaussian mask is a form of improved center of mass that integrates a discrete Gaussian function. However, to our disappointment, its performance is merely as good as the threshold center of mass.

For the Gaussian fitting, despite the convenience of setting a background parameter in the Gaussian model, handling the background value before the fitting actually starts is more effective. The weight fitting that copes with the uncertainty induced by the shot noise doesn`t show any improvement on the real localization precision, although it does reduce the covariance estimate error.

Figure 4-6-1 shows the localization error for QuickPALM center of mass method, the threshold center of mass method and the Gaussian fitting method. Compared with QuickPALM center of mass method, the threshold center of mass method remarkably reduces the localization errors at signal-to-noise ratios from 2 to 10. The Gaussian fitting outperforms both of the center of mass approaches at any signal-to-noise ratio.

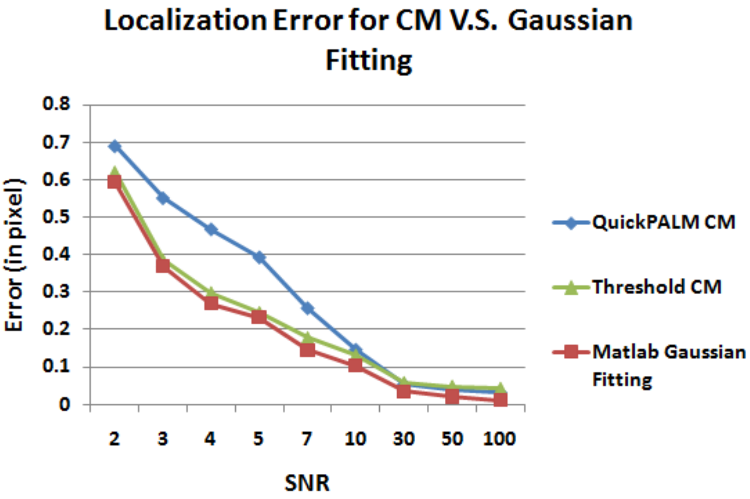


Figure 4-6-1 The localization errors with QuickPALM center of mass method, the localization with the threshold center of mass method and the localization with the Gaussian Fitting implemented with the Matlab routine.

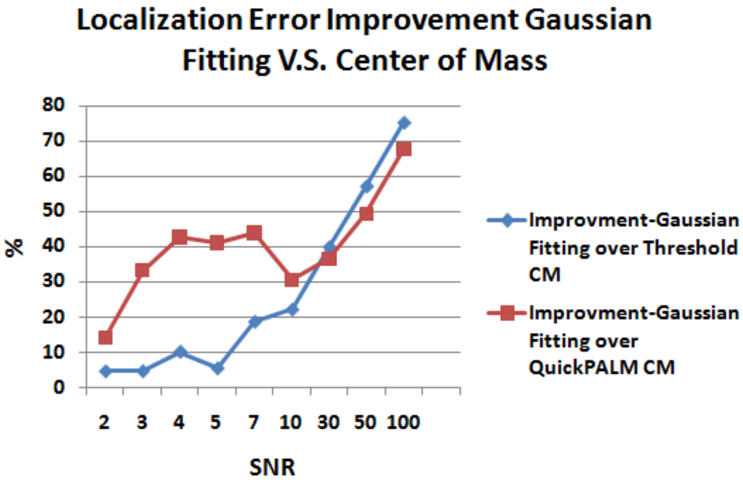
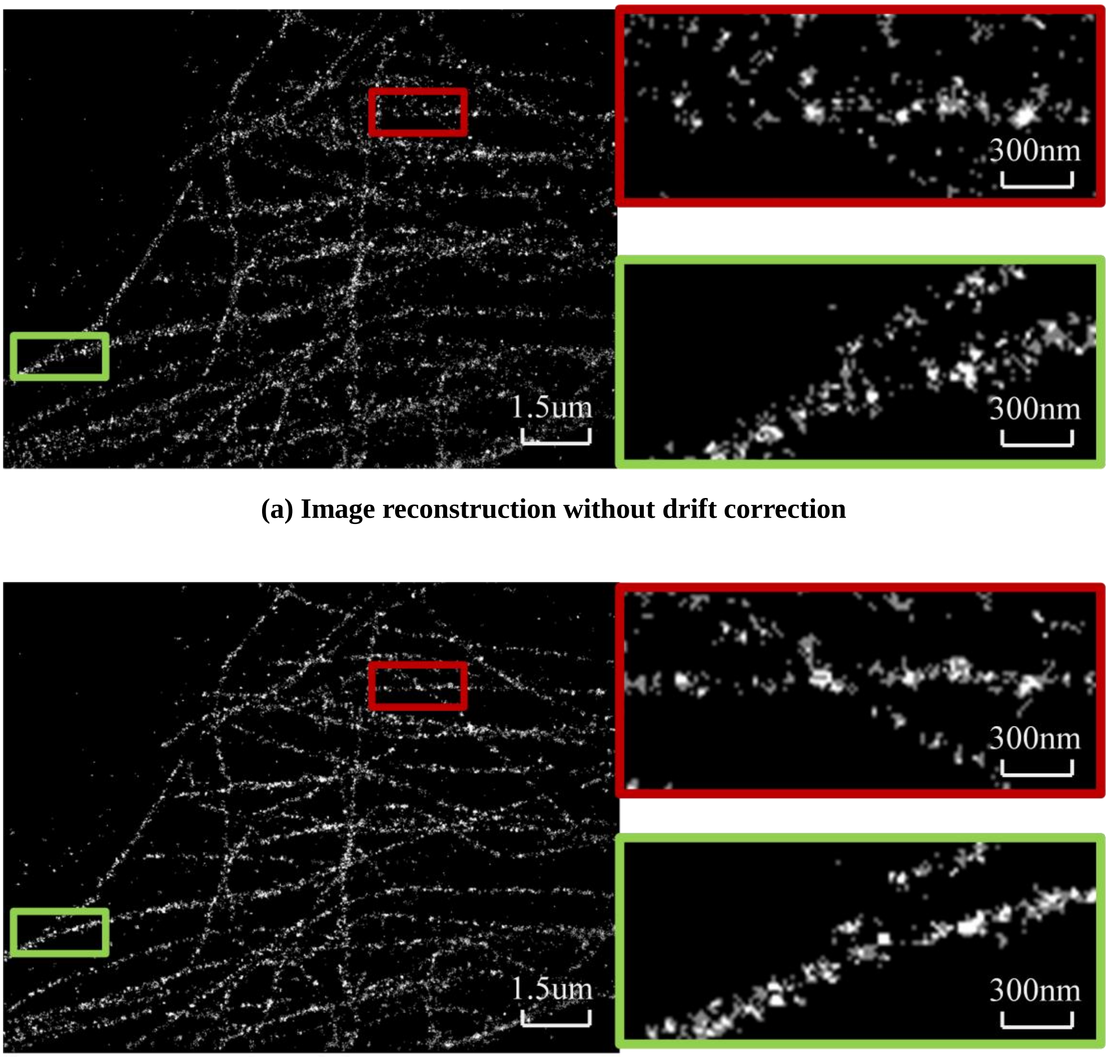


Figure 4-6-2 The red curve shows the improvement of the localization with Gaussian fitting over the localization with the threshold center of mass, in terms of the localization errors. The blue curve shows the improvement of the localization with Gaussian fitting over the localization with QuickPALM center of mass.

Figure 4-6-2 shows the improvement of the Gaussian fitting method over these two center of mass approaches. For both comparisons, the improvements tend to be more substantial as the signal-to-noise increases. At low signal-to-noise ratios, the noise has disturbed the image quality so much that even the Gaussian fitting has lost its power.

As the image quality gets better, the Gaussian fitting method has shown its superiority over the center of mass method. The improvement reaches 70% at our signal-to-noise ratio of 100. Even at a low signal-to-noise of 2, there is still 5% of improvement over the threshold center of mass method and 14% over the QuickPALM center of mass method.



(a) Image reconstruction without drift correction

(b) Image reconstruction with drift correction

Figure 4-6-3 (a) is the super-resolution reconstruction image without drift correction particle detection and (b) is the reconstruction with drift correction. For each of them, the left image is the whole reconstruction image. The right images are the details in the rectangular regions of the left image. Comparing them we can see that the microtubules in (a) have smaller widths and clearer structures.

Figure 4-6-3 compares the reconstruction images without drift correction and without drift correction. Clearly with drift correction the width of the line segments on the cell structure has decrease. In Figure 4-6-4 we randomly choose a region and compute the integrated intensity of the same line segment in that region in the two reconstructed images. The width of the line segment in the non-drift correction reconstruction image is approximately 0.1 inch, while in the drift correction reconstruction image the width is approximately 0.075 inch, which has decrease by 25%.

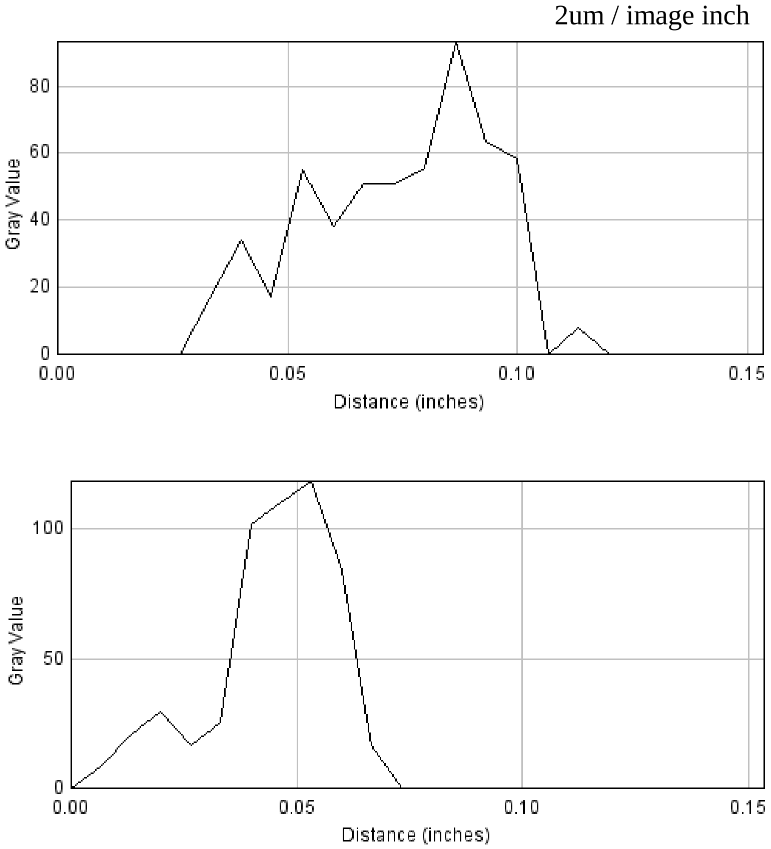


Figure 4-6-3 Comparison of the integrated intensity of segments from the non-drift correction reconstructed image and the drift correction reconstructed image. The plot on top is for non-drift correction reconstructed image and the plot at the bottom is for the drift correction reconstructed image.

Chapter 5 Implementation

5.1 Framework

Main purpose of current project is to develop a tool for automatic single-molecule super-resolution images analysis. Algorithms discussed above were implemented in a Java plug-in. The plug-in is developed for ImageJ that is an image processing software commonly used for biological image analysis. ImageJ provides easily accessed interface for Java plug-in. A great number of image analysis plug-in, including QuickPALM, have been developed for ImageJ.

When a specimen is set under the fluorescent microscope, a series of consecutive images (SR dataset) are taken. Developed plug-in deals with each of these images individually. When an image is inputted, it goes through the steps of particle detection. The output of particle detection is a number of spots in the image whose pixels intensities and pixels coordinates are indicated. In the phase particle localization, each spot is handled individually. For a spot, the center of mass with background subtraction is computed. Then the center of mass is used as the initial center parameters of the spot in the Gaussian fitting.

When the localization is finished, the estimated parameters are checked in case some noise patterns are detected as spots. If a detected spot has a standard deviation much larger or much smaller than the point spread function of the microscope, this spot is a false positive and it will be removed.

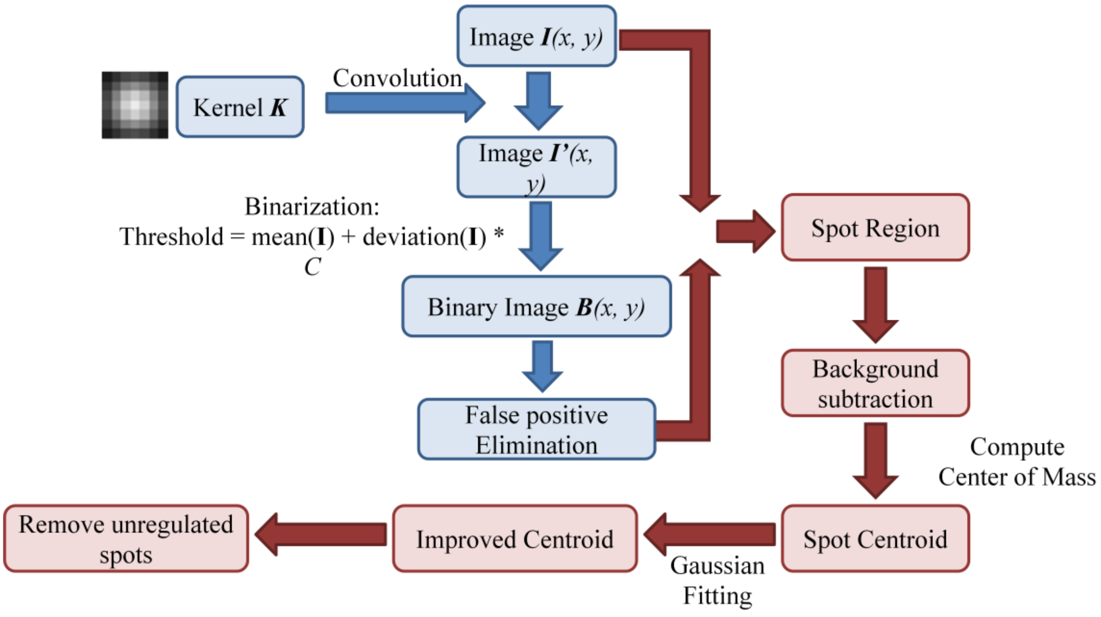


Figure 5-1-1 The general frame work of our whole algorithm.

After all of these are done, for each image we get the coordinates of all the spots in the image. Then all of spots detected from every image will be plot on the final reconstruction image.

5.2 Levenberg-Marquardt Fitting

Before the Levenberg-Marquardt starts its fitting, it needs to be initialized. The following parameters are required to initialize: the fitting function, the partial derivative function, the data matrix of the dependent variables, the data matrix of the independent variables, the weights, the iteration stop criteria and the initial guess of the estimated parameters and the damping coefficient.

First, the fitting model shall be provided, which is a Gaussian function. To evaluate the Jacobian matrix, the partial derivative function (Appendix 5.2.1) of the fitting model with respect to each of the estimated parameters shall be indicated.

The dependent variables are the intensities of the pixels of the spot, whose values largely depend on the relative positions of the pixels with respect to the spot center. If the background subtraction is needed to be done before the fitting, the dependent variables shall be the intensities of the pixels minus the background value of the spot.

The independent variables are the pixels` coordinates of the spot, whose values do not depend on any other variables. But the values of the dependent variables depend on the values of the independent variables.

As stated before, the weights shall be calculated by taking the square roots of signal values, namely the square root of pixels` intensities minus the background value of the spot.

The initial guess of the estimated parameters includes the spot center coordinate, the spot standard deviation and the intensity amplitude. The initial value of spot center coordinate is set as the center of mass. The standard deviation of the spot is provided by the user, which is the standard deviation of the point spread function of the microscope. If a background parameter is set in the fitting model, its initial value will be the minimum intensity value of all the pixels of the spot. Because if there`s no background in the image, the intensity of a pixel of the spot will approach 0 as it goes further away from the spot center, the minimum intensity of the pixels is expected to approximate the background value of the spot. Because the exponent of the Gaussian model is a negative value, the range of the exponential term is [0, 1]. The initial value of the amplitude shall be the maximum intensity minuses the minimum intensity of the pixels.

There are a few convergence criteria for stopping the iteration of the Levenberg-Marquardt algorithm.

- When the derivative of the chi square $\boldsymbol{J}^{\mathrm{T}}\boldsymbol{e}$ is below ε_1.
- When perturbation $\boldsymbol{h}$ drops below ε_2.

- When the chi square $\boldsymbol{e}^{\mathrm{T}}\boldsymbol{e}$ drops below ε_3.
- When the iterations number exceeds $i_{\max}$.

If any of the criteria above is satisfied, the iteration will stop.

The initial guess of the damping coefficient λ is 0.001. The steps of the iteration are as follows:

- Solve the Equation (A-36) for a perturbation.
- If the new chi square error with the updated parameters is smaller than the old chi square error, the update is accepted and the damping coefficient is decreased by a factor of 10. The fitting goes towards the Gauss-Newton method.
- If the new chi square error with the updated parameters is equal to or larger than the old chi square error, the update is rejected and the damping coefficient is increased by a factor of 10. The fitting goes towards the gradient decent method.

5.3 Libraries

Except for the time for convergence, fitting goodness, the stability is also a critical factor influencing our choice of library.

There are a variety of libraries implemented for Levenberg-Marquardt algorithm. The Matlab routine we use is developed by Henri Gavin [30], modified from the Levenberg-Marquardt non-linear regression function of Octave–Forge. It requires constraints on the estimated parameters. Using difference approximated Jacobian, it doesn`t ask for the partial derivative functions explicitly.

Another library is a C++ library Levmar. It provides different options of the Jacobian evaluation methods, parameters constraint, independent variable scaling, computation precision, etc. The library is fast and robust, with numbers of options adapting to different applications.

LMA is a Java library with analytical Jacobian. It doesn`t provide any parameter constraint.

Table 5-3-1 shows the options provided by these three libraries.

In chapter three, we conclude that the problem caused by fitting without parameter constraint at low signal-to-noise level can be overcome by combining the center of mass. Therefore whether the libraries provide parameter constraint doesn`t make any big difference. We also find that the fitting with approximated Jacobian and the fitting with analytical Jacobian have quite similar general performance. Although at low signal-to-noise level, analytical Jacobian is better. So a library using analytical Jacobian is more welcomed but a library with approximated Jacobian is also acceptable. Concluding all above, all the three libraries have satisfied our requirements but the C++ library Levmar and the Java library LMA are preferred.

	Matlab routine	Levmar(C++)	LMA(Java)
Parameter Constraint	Yes	Yes/No	No
Approximated Jacobian	Yes	Yes	No
Analytical Jacobian	No	Yes	Yes

Table 5-3-1 The table shows compare the options provided by the three libraries we use.

Now we will compare the performance of the three libraries. Table 5-3-2 shows the localization errors of the three libraries at difference signal to noise ratios. Note that due the different options provided by the libraries, the Matlab routine uses difference approximated Jacobian while Levmar and LMA use analytical Jacobian. LMA doesn`t provide parameters constraint while the other two do. Except these, all the other settings are the same for the three libraries in this test.

Lib\SNR	1	2	3	4	5	7	10	30	50	100
Matlab	2.067	0.593	0.369	0.269	0.232	0.145	0.103	0.035	0.021	0.011
Levmar	1.150	0.749	0.378	0.270	0.216	0.156	0.100	0.036	0.0213	0.0109
LMA	8.180	0.573	0.360	0.267	0.202	0.151	0.102	0.036	0.021	0.011

Table 5-3-2 The table shows the localization errors of the three libraries at different SNR.

From the results we can see the localization errors given by these three libraries are quite the same at high signal-to-noise levels. As the signal-to-noise ratio decreases, the differences of the three libraries become larger, because of the differences in parameter constraint and Jacobian approximation. The worse the signal-to-noise ratio is, the more likely that the parameters will go to unexpected values that are out of the parameters constraint boundaries. Therefore the differences in localization error of the three libraries become obvious. And as discussed in the section Jacobian Evaluation in chapter three, at low signal-to-noise level the analytical Jacobian starts to outperform the difference approximated Jacobian. That explains why both having parameter constraint, the Matlab routine and Levmar library gives so different result at SNR = 1.

If we look at the performance of LMA library, we find that its localization error is the best among the three libraries, except that at SNR = 1 it unexpectedly gives a huge error, because it doesn`t use parameter constraint. We have stated in the section

Parameter Constraint in chapter three that at signal to noise less than or equal to one, the localization result may be replaced by the center of mass coordinate if the fitting fail to fit the spot center. The localization error given by the center of mass coordinates is 0.93 at SNR = 1, which is even better than the best fitting error of the three libraries. Therefore combining with center of mass, LMA library has the best performance among the three libraries.

Apart from finding the spot centers` coordinate, another goal of our algorithm is to estimate the localization error with the covariance matrix. In the section 4.4.7, we show that all the three libraries give acceptable estimation of the errors. They all give the upper bounds of the real errors and approximate the real errors with only small differences. Although the covariance estimated error given by LMA is generally worse than the other two libraries, at signal-to-noise ratio less than or equal to seven, the difference is trivial and the estimated errors are only up to 13% more than the real errors. So LMA still satisfies our needs.

Since our ImageJ plug-in is written in Java, a Java library gives easier interface than the libraries in other language. We have shown that LMA gives the best localization error and its error estimation is only slightly worse than the other two libraries in our interested region. Therefore, we use LMA library in our plug-in for Gaussian fitting.

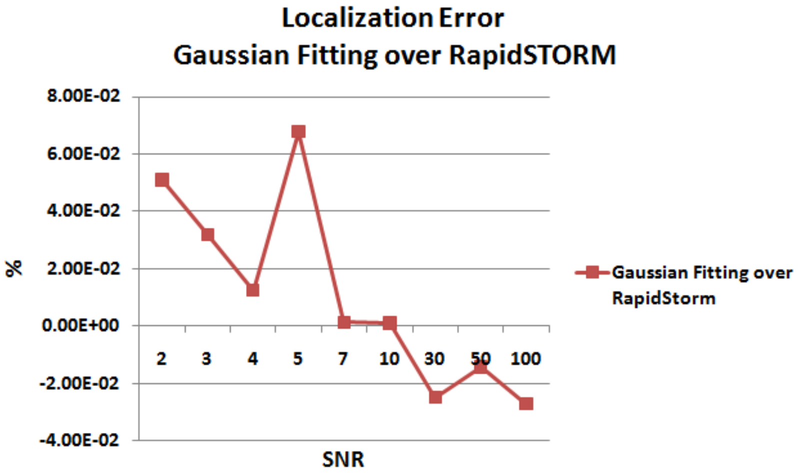


Figure 5-3-1 This figure shows the ratio of the localization errors given by our Gaussian fitting method and the RapidSTORM Gaussian fitting method.

Figure 5-3-1 compare the localization errors of our Gaussian fitting method and the RapidSTORM Gaussian fitting method. At signal-to-noise ratios from 2 to 10, the localization errors given our method are up to 7% smaller than the RapidSTORM Gaussian fitting method. Improving the localization error at low signal-to-noise ratios is exactly our main task.

Chapter 6 Conclusion and Future Works

Main aim of this project is developing a tool for automatic detection and localization of particles for single-molecule super-resolution images including correct estimation of the localization errors. The particle detection is mainly implemented by proper convolution and thresholding that is invariant to gradient background. The false positives are eliminated by several filters. Suggested particle detection substantially outperform QuickPALM`s and RapidSTORM`s as shown in the section 3.6. Then the center of mass coordinates of the particles are calculated, in a way that is different from general center of mass. This improved center of mass has greatly improved the localization error (section 4.2 - Center of Mass) and achieved the localization precision of Gaussian Mask – an iterative method that is the simplification of Gaussian fitting. The center of mass coordinates serve as the initial center value of the Gaussian fitting. The Gaussian fitting is implemented by LMA library. In section 5.3 we show that LMA library has smaller localization error than the other two libraries we test. And it also gives acceptable estimation of the localization errors. Our whole localization algorithm finally reduces the localization errors of QuickPALM by 10% to 70% within SNR range of [2, 100]. At low signal-to-noise ratios, our algorithm gives up to 7% improvement on localization errors compared with RapidSTORM. With drift correction algorithm, we have decreased the thickness of the line segment in the cell structure by 25% in our sample.

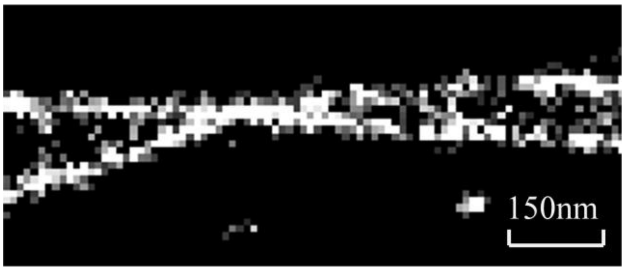


Figure 6-1 In this image two microtubules cross each other. The single-particle-property of the general super-resolution technique may induce blur at the cross.

There are still many details to be improved in the localization algorithm. For instances, in chapter three it was assumed that any two particles are far away from each other. Whole implemented algorithm is working under the assumption of sparse single molecule. Despite the single-molecule imaging techniques, occasionally one can still see overlapping molecules in the images. The problem of detecting these particles and separating them is still not well solved. As shown in Figure 6-1, at the place where two microtubules cross, un-separated overlapping particles bring especially bad effects. The algorithm may detect two close particles that belong to different microtubules as only one particle and plot it in the middle of them. This induces error in the localization of the microtubules. A simple crude solution is to ignore these irregular particles, but obviously this will cause more false negatives in the images. Another solution is to analyze these particles and separate them into single particles. Since crosses may be everywhere in a microscope image, if this process is too computation consuming, it will slow down the speed of the whole localization algorithm.

Another problem is that for the application of photoswitching, the particles switch on and off, blinking repeatedly in the time period, which most of the super-resolution software, including QuickPALM haven`t taken in to consideration. The algorithm will detect the same repeatedly blinking particles as two different particles and they will be labeled twice in the reconstruction image. Since there are errors in the localization and many particles are very close to each other, we are not sure if they are the same particle switching on twice or two different particles that are close to each other. A possible solution to determine this will be constructing a probability model, which combines the factor of the estimated localization error, the signal-to-noise ratio, the distribution of particles in the space, etc.

In the microscope image, actually the particles may be ellipses. The ratio of the maximal axis length and the minimal axis length is the focus length, which indicates the position of the particle in the 3D z-direction [32]. With this z-axis position and the x/y positions we acquire from the Gaussian fitting localization, we are able to make a 3D reconstruction of the sample.

Acknowledgement

I want to thank Eugene Katrukha, Lucas Kapitein and Remco Veltkamp for their supervision and instructions and Robert van den Berg for providing the dSTORM imaging datasets and constructive comments and feedback. Also I’m grateful to Professor Anna Akhmanova for providing me with opportunity to do my project in her lab. This work is a master thesis project of Department of Information and Computing Science, Faculty of Science, Utrecht University. It is also a part of the super-resolution project of Cell Biology group in Department of Biology, Faculty of Science, Utrecht University.

Appendices

Appendix 1.3.1

For microscopy images with low signal-to-noise ratio, even human visual system cannot distinguish all the particles from the noise. This makes difficulties in the evaluation of the particle detection algorithm. To properly evaluate the algorithm, an artificial dataset is generated.

In the dataset there are three main categories of image with different types of backgrounds, one with uniform background and the other two with gradient backgrounds. In each main category there are twelve categories of images with signal-to-noise ratio from 0.5 to 100. Each SNR category contains twenty images.

For each image, 50 spots with predefined coordinates are generated, during which shot noise is added to the signal. It follows Poisson distribution. After that the signal is add up with background noise which is described by another Poisson distribution.

For generating the noises, each pixel is randomly assigned to a value in the Poisson distribution:

$$f(\lambda) = \frac{\lambda^x e^{-\lambda}}{x!} (x \geq 0) \qquad \text{(A-1)}$$

where x is a random variable. For the background noise, λ is 2 times the standard deviation of the noise. For the shot noise λ is equal to the original spot intensity of that pixel, which is generated by the Gaussian function.

Appendix 3.1.1

For single-molecule super-resolution techniques, particle detection is performed under the assumption that the images we obtain are sparse particle images, so that no two particles overlap each other in the space. With this assumption, for any two particles A and B in the image, the following condition should be satisfied:

$$(x_A - x_B)^2 + (y_A - y_B)^2 > r_A + r_B \qquad \text{(A-2)}$$

where r_A and r_B are the radiuses of spot A and spot B, and (x_A, y_A) and (x_B, y_B) are the coordinates of the spot centroids.

Appendix 3.1.2

The signal-to-noise ratio is defined by:

$$SNR = \frac{A_{signal}}{\sigma_{noise}} \qquad \text{(A-3)}$$

where A_{signal} is the amplitude of the spot signal and σ_{noise} is the standard deviation of the noise.

Appendix 3.2.3

The mathematics expression is:

$$I(x,y) * K(x,y) = \sum_{\mathrm{u,v}} I(x+u, y+v) \cdot K(u,v) \tag{A-4}$$

Appendix 3.2.4

An kernel K_A for average blur can be defined by:

$$\mathbf{K_A}(\mathbf{x},\mathbf{y}) = \mathbf{1}, \qquad [\mathbf{x}\ \mathbf{y}]\,\boldsymbol{\epsilon}\ \mathbf{D} \tag{A-5}$$

Appendix 3.2.5

$$FFT(I(x,y) * K(x,y)) = FFT\big(I(x,y)\big) \cdot FFT\big(K(x,y)\big) \tag{A-6}$$

where *FFT*() is the function that transform an image or a kernel from space domain to frequency domain.

Appendix 3.2.6

The discrete Gaussian kernel generator is implemented by:

$$K(x,y) = \int_{\mathrm{x}-\frac{1}{2}}^{\mathrm{x}+\frac{1}{2}} \int_{\mathrm{y}-\frac{1}{2}}^{\mathrm{y}+\frac{1}{2}} \mathrm{g(u,v)dvdu} \tag{A-7}$$

where (x, y) is the center of a pixel in the kernel. g(x, y) is a 2 dimensional Gaussian function, which is given by:

$$\mathrm{g(u,v)} = A \cdot e^{-\frac{1}{2}[\left(\frac{u-x_0}{\sigma}\right)^2 + (\frac{v-y_0}{\sigma})^2]} \tag{A-8}$$

σ is the estimated standard deviation of the spot and (x_0, y_0) is the center of the Gaussian kernel.

Appendix 3.3.1

Figure A-1 shows how most of histogram algorithms implement image segmentation or object detection. Let C_1, C_2 be the two classes to be segmented in the image, and e(C_2| C_1) be the number of pixels belonging to C_1 that is miss-classified as C_2. The detection error of the whole image is:

$$\mathrm{e}(\boldsymbol{I}) = \mathrm{e}(C_2|\ C_1) + \mathrm{e}(C_1|\ C_2) \tag{A-9}$$

which is represented as the shading area in the figure. The threshold that minimizes e(***I***) is at the crossing point of the C_1 and $C_{2.}$

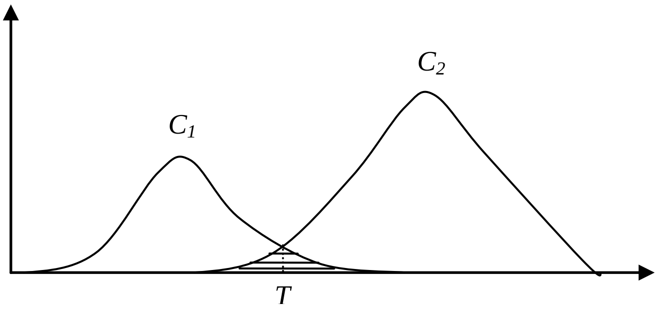


Figure A-1 This figure shows the histogram of two classes C_1 and $C_{2.}$ The optimal threshold between them will be T.

Appendix 3.4.1

The basic 3 by 3 high-pass filter has a form as shown in Figure A-2 (a). The corresponding low-pass filter is shown in Figure A-2 (b).

-1/9	-1/9	-1/9
-1/9	8/9	-1/9
-1/9	-1/9	-1/9

(a)

1/9	1/9	1/9
1/9	1/9	1/9
1/9	1/9	1/9

(b)

Figure A-2 (a) The high-pass spatial filter. (b) The corresponding low-pass spatial filter.

Appendix 3.4.2

For background subtraction, the background value of the center pixel will be the average intensity of the background pixels.

$$bg_c = \frac{1}{\sum_{i,j,(i-i_c)^2+(j-j_c)^2>r^2} 1} \sum_{i,j,(i-i_c)^2+(j-j_c)^2>r^2} I(i,j) \qquad \text{(A-10)}$$

Where (i_c, j_c) is the position of the center pixel.

Appendix 3.5.2.1

QuickPALM a simple symmetry evaluation method for measuring the symmetry with respect to x-axis and y-axis:

$$\sigma_l = \frac{\sum_{i,j} S_{i,j}(c_x - x_{i,j})}{\sum_{i,j} S_{i,j}}, \text{ where } (i,j) \in \{(i,j)|x_{i,j} < c_x\} \qquad \text{(A-11)}$$

$$\sigma_r = \frac{-\sum_{i,j} S_{i,j}(c_x - x_{i,j})}{\sum_{i,j} S_{i,j}}, \text{ where } (i,j) \in \{(i,j)|x_{i,j} \geq c_x\} \quad \text{(A-12)}$$

$$\sigma_a = \frac{\sum_{i,j} S_{i,j}(c_y - y_{i,j})}{\sum_{i,j} S_{i,j}}, \text{ where } (i,j) \in \{(i,j)|y_{i,j} < c_y\} \quad \text{(A-13)}$$

$$\sigma_b = \frac{-\sum_{i,j} S_{i,j}(c_y - y_{i,j})}{\sum_{i,j} S_{i,j}}, \text{ where } (i,j) \in \{(i,j)|y_{i,j} \geq c_y\} \quad \text{(A-14)}$$

$S_{i,j}$ is the intensity of the pixel (i, j) and ($x_{i,j}$, $y_{i,j}$) is the coordinate of it. (c_x, c_y) is the spot center of mass.

The x-axis, y-axis symmetry is measured by:

$$S_x = 1 - \frac{|\sigma_l - \sigma_r|}{\sigma_l + \sigma_r} \quad \text{(A-15)}$$

$$S_y = 1 - \frac{|\sigma_a - \sigma_b|}{\sigma_a + \sigma_b} \quad \text{(A-16)}$$

If the spot is a true positive we will get $\sigma_l \approx \sigma_r$ and $\sigma_a \approx \sigma_b$, which yields $S_x \approx 1$ and $S_y \approx 1$.

Appendix 3.5.2.2

The following symmetry criterion eliminates a false positive:

If $S_x < 1 - T_s$ or $S_x > 1 + T_s$ or

$S_y < 1 - T_s$ or $S_y > 1 + T_s$,

the spot is a false positive.

Appendix 3.5.3.1

The definitions of erosion and dilation are [33]:

$$\text{Erosion: } \alpha \ominus \beta = \cap_{b \epsilon \beta} \alpha_{-b} \quad \text{(A-17)}$$

$$\text{Dilation: } \alpha \oplus \beta = \cap_{b \epsilon \beta} \alpha_b \quad \text{(A-18)}$$

where

$$\alpha_x = \{a + x : a \epsilon \alpha\} \quad \text{(A-19)}$$

α is the binary image and β is the operator.

The center of the operator is:

$$c_x = \text{floor}[(w + 1) / 2] \quad \text{(A-20)}$$

$$c_y = \text{floor}[(h + 1) / 2] \quad \text{(A-21)}$$

where w is the width of the operator and h is the height of the operator.

Appendix 4.2.1

Let P_i' be the pixels of the spot region of the *i*th spot, and p_{ik}' be the *k*th pixel of P_i'.

Let S_{ik}, ($x_{ik,}$ y_{ik}) be the intensity and the coordinate of p_{ik}'. The center coordinate of the *i*th spot calculated by center of mass is:

$$x_{ic} = \frac{\sum_k x_{ik} \cdot S_{ik}}{\sum_k S_{ik}} \tag{A-22}$$

$$y_{ic} = \frac{\sum_k y_{ik} \cdot S_{ik}}{\sum_k S_{ik}} \tag{A-23}$$

Appendix 4.3.1

Let S_{i} be the *i*th pixel of a spot and N_{i} be the corresponding cell of the Gaussian mask. The least square measures the error between spot pixels` intensities S_i and the discrete Gaussian function N_i [24]:

$$\chi^2 = \frac{1}{2}\sum_i (S_i - N_i)^2 \tag{A-24}$$

where:

$$N_i = \int_{y_i-\frac{1}{2}}^{y_i+\frac{1}{2}} \int_{x_i-\frac{1}{2}}^{x_i+\frac{1}{2}} A \cdot e^{-\frac{1}{2}\left[\frac{(u-x_0)^2}{\sigma_x{}^2}+\frac{(v-y_0)^2}{\sigma_y{}^2}\right]} dudv \tag{A-25}$$

(x_{i}, y_{i}) is the coordinate of the *i*th pixel of the spot.

The following equation shall be satisfied:

$$\frac{d\chi^2}{dx_0} = 0 \tag{A-26}$$

These yields:

$$x_0 = \frac{\sum_i x_i S_i N_i}{\sum_i S_i N_i} \tag{A-27}$$

$$y_0 = \frac{\sum_i y_i S_i N_i}{\sum_i S_i N_i} \tag{A-28}$$

Appendix 4.4.1.1

The objective function of the non-linear least fitting will be:

$$g(x_i, y_i) = A_0 + A \cdot e^{-\frac{1}{2}\left[\frac{(x_i-x_0)^2}{\sigma_x{}^2}+\frac{(y_i-y_0)^2}{\sigma_y{}^2}\right]} \tag{A-29}$$

x_0 and y_0 are the parameters to be estimated, which are non-linear in the function.

The fit is a minimization problem of the least square loss function:

$$\min_{x_0,y_0,\sigma,A_0,A} f\left(x_0, y_0, \sigma_x, \sigma_y, A_0, A\right) = \sum_i [g(x_i, y_i) - S_i]^2 \qquad \text{(A-30)}$$

S_i is the signal of the *i*th pixel, which is the intensity. In this minimization equation, x_i, y_i and S_i are observable variables, which are the pixels information provided by the spot. x_i, y_i are the independent variables and S_i is the dependent variable. $g(x_i, y_i)$ is the prediction and $g(x_i, y_i)$ - S_i is the residual.

A weighted least square is given by:

$$\min_{x_0,y_0,\sigma,A_0,A} f\left(x_0, y_0, \sigma_x, \sigma_y, A_0, A\right) = \sum_i w_i [g(x_i, y_i) - S_i]^2 \qquad \text{(A-31)}$$

where w_i is the weight for the *i*th pixel.

Appendix 4.4.2.1

The chi-square of the fit is also the objective function to be minimized, which will be:

$$\chi^2 = \frac{1}{2} \sum_i (g(x_i, y_i) - S_i)^2 \qquad \text{(A-32)}$$

The gradient decent method defines a step length α in the steepest descent direction. The perturbation $\boldsymbol{h}_{gd}$ is the step length times the steepest decreasing direction of the objective function [35]:

$$\boldsymbol{h}_{gd} = \alpha \boldsymbol{J}^T (\boldsymbol{g} - \boldsymbol{S}) \qquad \text{(A-33)}$$

where $\boldsymbol{g}$ is a *n* by 1 matrix of the intensities calculated by the function $g(x_i, y_i)$ (i = 1, 2, …, n) and $\boldsymbol{S}$ is a *n* by 1 matrix of the intensities of *n* spot pixels. $\boldsymbol{J}$ is the Jacobian matrx:

$$\boldsymbol{J} = \frac{\partial \boldsymbol{g}}{\partial \boldsymbol{p}} \qquad \text{(A-34)}$$

$\boldsymbol{p}$ is the *m* by 1 parameters matrix. *m* is the number of estimated parameters.

The Gauss-Newton method always directs the perturbation to the point $\partial \chi^2 / \partial \boldsymbol{h}_{gn} = 0$, where $\boldsymbol{h}_{gn}$ is the perturbation. Approximate $g(\boldsymbol{p} + \boldsymbol{h}_{gn})$ with the Taylor expansion yields:

$$[\boldsymbol{J}^T \boldsymbol{J}] \boldsymbol{h}_{gn} = \boldsymbol{J}^T (\boldsymbol{g} - \boldsymbol{S}) \qquad \text{(A-35)}$$

The equation for calculating the perturbation of Levenberg-Marquardt is very similar to the one of Gauss-Newton method`s:

$$[\boldsymbol{J}^T \boldsymbol{J} + \lambda \boldsymbol{I}] \boldsymbol{h}_{gn} = \boldsymbol{J}^T (\boldsymbol{g} - \boldsymbol{S}) \qquad \text{(A-36)}$$

Appendix 4.4.2.2

The covariance matrix is defined by:

$$\boldsymbol{C} = [\boldsymbol{J}^{\boldsymbol{T}}\boldsymbol{J}]^{-\mathbf{1}} \tag{A-37}$$

Appendix 4.4.4.1

Measurement error is also the relative uncertainty in the intensity of the data point. If σ_S is the measurement error, it is calculated by:

$$\sigma_s = \frac{1}{\sqrt{s}} \cdot s \tag{A-38}$$

Appendix 4.4.4.2

A least square with weight is:

$$\chi^2 = \frac{1}{2}\sum_i w_i(g(x_i, y_i) - S_i)^2 \tag{A-39}$$

The weight w_i of the *i*th data point shall correspond to the confidence of the data point`s intensity. Hence w_i is given by:

$$w_i = \frac{1}{{\sigma_s}^2} \tag{A-40}$$

Appendix 4.4.4.3

The χ goodness is a value for evaluating the goodness of the fitting. It`s χ^2 divide by the degree of freedom, which is:

$$\sigma^2 = \frac{\chi^2}{m-n} \tag{A-41}$$

(m - n) is the degree of freedom, where m is the data points number and n is the number of parameters to be evaluated. σ^2 is also called the unbiased estimated of the error [36].

Appendix 5.2.1

Let the fitting model be y. The partial derivative functions are:

$$f(p_i) = \frac{\partial \hat{y}}{\partial p_i}(i = 1, 2, \dots m) \tag{A-42}$$

m is the number of parameters to be estimated and p_i is the *i*th estimated parameter.

References

[1] Wolfe, S. L.. Molecular and Cellular Biology. Univeristy of California at Davis. Wadsworth Pulishing Company, Belmont, California. p. 117, (1993).

[2] Stokes, G. G.. On the Change of Refrangibility of Light. Philosophical Transactions of the Royal Society of London 142: p. 463–562, (1852).

[3] Hell, S. and Stelzer, E. H. K.. Fundamental improvement of resolution with a 4Pi-confocal fluorescence microscope using two-photon excitation. Optics Communications 93, p. 277-282, (1992).

[4] Hell, S. W. and Wichmann, J.. Breaking the diffraction resolution limit by stimulated emission: stimulated-emission-depletion fluorescence microscopy. Optics Letters, Vol. 19, Issue 11, p. 780-782, (1994).

[5] Gustafsson, M. G. L.. Surpassing the lateral resolution limit by a factor of two using structured illumination microscopy. Journal of Microscopy, 198, p. 82-87, (2000).

[6] Patterson, G. H. and Lippincott-Schiwartz, J.. Selective photolabeling of proteins using photoactivatable GFP. Methods 32, p. 445-450, (2004).

[7] Gordon, M. P., Ha, T. and Selvin, P. R.. Single-molecule high-resolution imaging with photobleaching. PNAS, Vol. 101, No. 17, p6462-6465, (2004).

[8] Burnette D.T., Sengupta P., Dai Y., Lippincott-Schwartz J., Kachar B.. Bleaching/blinking assisted localization microscopy for superresolution imaging using standard fluorescent molecules. PNAS 108(52), p. 21081-6 (2011).

[9] Betzig, E., Patterson, G. H., Sougrat, R., Lindwasser, O. W., Olenych, S., Bonifacino, J. S., Davidson, M. W., Lippincott-Schwartz, J. and Hess, H. F.. Imaging Intracellular Fluorescent Proteins at Nanometer Resolution. Science, Vol. 313, p. 1642-1645, (2006).

[10] Rust, M. J., Bates, M and Zhuang X.. Sub-diffraction-limit imaging by stochastic optical reconstruction microscopy (STORM). Nature Methods, Vol.3 No. 10, p. 793, (2006).

[11] York, A. G., Ghitani, A., Vaziri, A., Davidson, M. W. and Shroff, H.. Confined activation and subdiffractive localization enables whole-cell PALM with genetically expressed probes. Nature Methods 9, p. 327-333, (2011).

[12] Cox, S., Rosten, E., Monypenny, j., Jovanovic-Talisman, T., Burnette, D. T., Lippincott-Schwartz, J., Jones, G. E. and Heintzmann, R.. Bayesian localization microscopy reveals nanoscale podosome dynamics. Nature Methods, 9(2), p. 195-200, (2011).

[13] Niu, L. and Yu, J. Investigating Intracellular Dynamics of FtsZ Cytoskeleton with Photoactivation Single-Molecule Tracking. Biophysical Journal 95(4), p. 2009-16, (2008).

[14] Tatavarty, V., Kim, E.-J., Rodionov, V., and Yu, J. Investigating Sub-Spine Actin Dynamics in Rat Hippocampal Neurons with Super-Resolution Optical Imaging. PLoS ONE 4, e7724, (2009).

[15] Wolter, S., Schuttpelz, M., Tscherepanow, M., Van de Linde, S., Heilemann, M. And Sauer, M.. Real-time computation of subdiffraction-resolution fluorescence images. Journal of Microscopy, Vol 237, pt 1, p. 12-22, (2010).

[16] Henriques, R., Lelek, M., Fornasiero, E. F., Valtorta, F., Zimmer, C. And Mhlanga, M. M.. QuickPALM: 3D real-time photoactivation nanoscopy image processing in ImageJ. Nature Methods, vol. 7, no. 5, p. 339 – 340, (2010).

[17] Mark O’Malley and Eon O’Mongain, “Charge-coupled devices: frame adding as an alternative to long integration times and cooling”, Optical Engineering 31(3), p. 522-526 (1992).

[18] Zhang, B. Contributions to fluorescence microscopy in biological imaging: PSF modeling, image restoration, and super-resolution detection. Ecole Nationale Superieure des Telecommunications, p. 53 – 55, (2007).

[19] Smal, I.. Particle Filtering Methods for Subcellular Motion Analysis. Eramus MC–University Medical Center Rotterdam (Rotterdam, the Netherlands). p. 45–46, (2009).

[20] Mashanov, G. I. and Molly, J. E.. Automatic Detection of Single Fluorophoresin Live Cells. Biophysical Journal, Volume 92, (2007).

[21] Gonzalez, R. C., Woods, R. E.. Digital Image Processing. Addison-Wesley Publishing Company, Inc. (1992).

[22] Castleman, K. R.. Digital Image Processing. Prentice Hall, Inc. Upper Saddle River, New Jersey 07458, p. 209, (1996).

[23] Henriques, R., Lelek, M., Fornasiero, E. F., Valtorta, F., Zimmer, C. And Mhlanga, M. M.. QuickPALM: 3D real-time photoactivation nanoscopy image processing in ImageJ. Nature Methods, vol. 7, no. 5, p. 6 – 7, (2010 suppl).

[24] Thompson, R. E., Larson, D. R. and Webb, W. W.. Precise Nanometer Localization Analysis for Individual Fluoresent Probes. Biophysical J, Vol. 82, p. 2775 – 2783, (2002).

[25] Simonson, P. D., Rothenberg, E., Selvin P. R.. Single-Molecule-Based Super-Resolution Images in the Presence of Multiple Fluorophores. Nano Letters. 11(11): p. 5090 – 5096, (2011).

[26] Smith, C. S., Joseph, N., Rieger, B. & Lidke, K. A.. Fast, single-molecule localization that achieves theoretically minimum uncertainty. Nature Methods, vol. 7

No. 5, (2010).

[27] De Groot, M. H. and Schervish, M. J.. Probalility and Statistics. Fourth Edition. Pearson Education, Inc. (2012).

[28] Press, William. H., Teukolsky, S. A., Vetterling, W. T. And Flannery, B. P.. Numerical Recipies, the Art of Scientific Computing, third edition. Cambridge University Press, p. 790, (2007).

[29] Dennis, J.E., Schnabel, R. B. Jr.. Numerical Methods for Unconstrained Optimization and Nonlinear Equations. Society for Industrial and Applied Mathematics, p. 28, (1996).

[30][35] Gavin, H, The Levenberg-Marquardt method for nonlinear least squares curve-fitting problems, Department of Civil and Environmental Engineering, Duke University: Abr, (2011).

[31] Mlodzianoski, M. J., Schreiner, J. M., Callahan S. P., Smolkova, K., Dlaskova, A., Santorova, J., Jezek, P. and Bewersdorf, J.. Sample drift correction in 3D fluorescence photoactivation localization microscopy. Optics Express, Vol. 19, Issue 16, p. 15009-15019, (2011).

[32] Huang, B., Wang, W., Bates, M. and Zhuang, X.. Three-dimensional super-resolution imaging by stochastic optical reconstruction microscopy. Science 319, p. 810-813, (2008).

[33] Jankowski, M. Erosion, dilation and related operators. 8th International Mathematica Symposium. Avignon, (2006).

[34] Cheezum, M. K., W. F. Walker, and W. H. Guilford. Quantitative comparison of algorithms for tracking single fluorescent particles. Biophys. J. 81: p. 2378–2388, (2001).

[36] Nikulin, M. S. and Voinov, V. G.. Unbiased estimators of multivariate discrete distributions and chi-square goodness-of-fit test. Questiio, Vol. 17, 3, p. 301-326, (1993).