University of Technology

Master Thesis

3D Single Molecule Localization Using Micromirrors

Author:
J.P. Cnossen

Supervisors:
dr. D. Dulin
prof. dr. D. de Ridder
prof. dr. N.H. Dekker

May 20, 2012

1 Abstract

3D imaging using fluorescence microscopy is a quickly advancing field of microscopy with many applications. Disadvantages of the various methods include a lack of depth resolution, requiring bright fluorescent samples to achieve a useful accuracy, or setups that are too complex. There is still a lot of room for improvement in this field, and imaging using micromirrors [11] could be a useful tool here.

This thesis project focuses on development and validation of an algorithm to localize fluorescent molecules or particles in a 3D micromirror setup. We can distinguish between localization (finding a molecule in 3D), and tracking (finding a molecule in 3D, knowing it's position at an earlier time). This thesis will focus primarily on localization. After an introduction of the localization process, an image formation model is presented that describes how the light from the sample forms a digital image in the camera. Noise characteristics of this process are also considered. The image formation model is used to create a probabilistic model that defines where the most likely location of a molecule is. Different optimization methods are then used to find this most likely location. The localization software is designed to make use of fast graphics hardware, in order to achieve localization speeds that allow for practical use of the localization method in experimental research.

The optimization methods are tested and compared using simulated data, which is randomly sampled according to the noise that would be typical of experimental conditions. The difficulties arising in localization for a micromirror are explored, and the theoretical the accuracy of the localization method is established by simulation.

Contents

2 Introduction

This chapter will introduce the concepts of microscopy, fluorescence and localization to the reader. After that, the previous work done in 2D and 3D localization will be considered.

2.1 History of light microscopy

In 1674, when the Delft scientist Antonie van Leeuwenhoek discovered micro-organisms with his self-built microscope, he started a new scientific field called microbiology. From that point onwards, scientists have been improving microscopes to study the micro and nano-scale mechanisms and structures used by life.

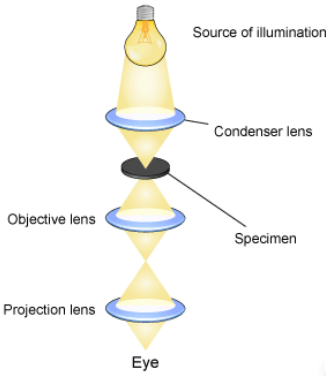


Figure 1: Light microscope schematic[40].

In regular light microscopy, a bright light shines through the sample into the eye piece, shown in Figure 1. The specimen can be seen because it blocks some of the light from the source. The specimen (Figure **??** for example) needs to be opaque, so it can block a part of the light and be in contrast with the bright background. A schematic of this is shown in Figure 1, which also shows some of the lenses involved.

The first microscopes had significant lens *aberrations*, which are essentially defects in the optical system that cause it to produce a less-than-ideal magnified image. An ideal lens should focus all rays from the same source into the same point on the image, but real lenses fail to do this perfectly. High quality lenses are corrected for aberrations only in the region on the focus plane and near the optical axis, and even then aberrations occur. Possible aberrations include:

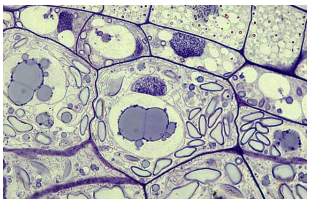


Figure 2: Light microscopy example. Specimen is a section of wheat endosperm (seed). Image source: [46].

- *Chromatic distortion*, when light with different colors follow different paths, Figure 3)
- *Defocus*, when the sample is not located on the focus plane.
- *Spherical aberration*, due to the spherical shape of the lens, the rays far from the optical axis have a different focus point than those close to the optical axis.
- *Astigmatism*, when the refraction of light rays in X direction is different from the ones in Y direction.

Refer to Hecht [13, chapter 6] for a detailed discussion of aberrations.

The field of optical microscopy that van Leeuwenhoek pioneered has since evolved into a very powerful tool for science, and for microbiology in particular (For an overview: [44]). Advances in objective (lens) design have allowed larger magnifications and lower aberrations.

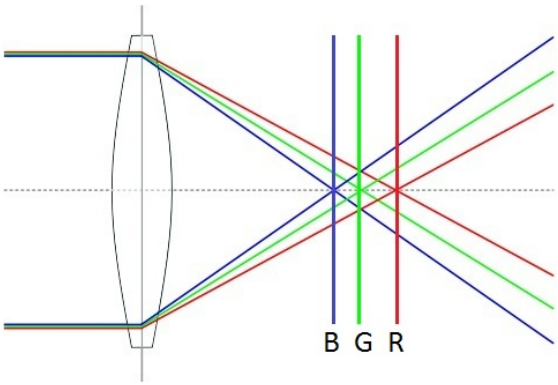


Figure 3: Chromatic aberration: different colors have different focal planes (indicated by B,G and R). Source: [50]

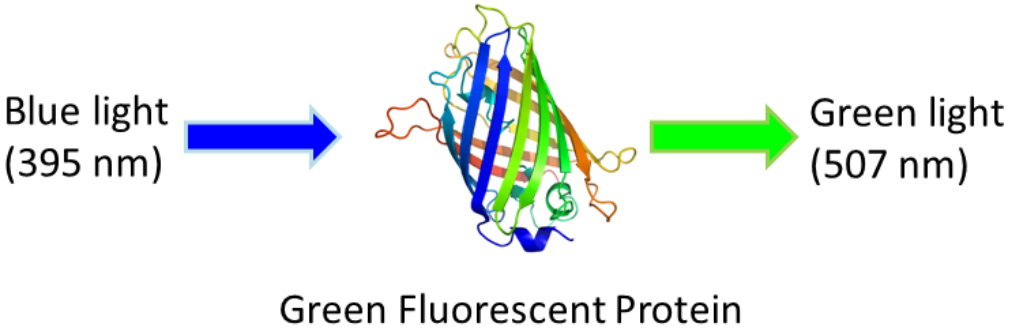


Figure 4: Fluorescence in a green fluorescent protein (GFP). The ribbon diagram is shown as a symbolic representation of the 3D protein structure. Image source: [53].

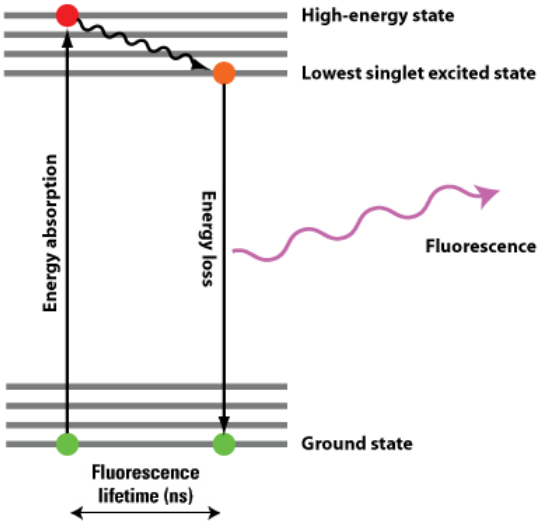


Figure 5: Jablonski diagram, showing different molecular states. After absorption, some energy is released as heat until it fluoresces and jumps back to the original. Image source: [52].

2.2 Fluorescence microscopy

Fluorescent molecules, or fluorophores, are molecules that emit light of a certain color, after being excited by the energy from light of another color (Figure 4). In biological research, these are used to label molecules such as proteins, membranes and DNA (Figure 6), which are otherwise not detectable in the visible spectrum of light. When fluorescent molecules are exposed to light of the right frequency, some of the molecules will switch to an excited state where they hold the energy from the photon (see the Jablonksi diagram: Figure 5). After several nanoseconds (known as the fluorescence lifetime), the energy is released from the molecule as light of another color. Because the output energy cannot be larger than the input energy, the emitted light always has a lower frequency than the absorbed light. The principle of fluorescence was already described by Sir George Gabriel Stokes in 1843, but it developed into a useful biological research tool. For an extensive overview of the history of fluorescence microscopy, see Masters [20].

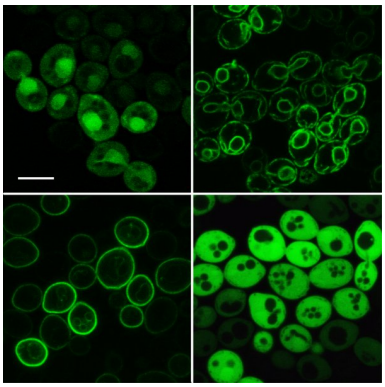


Figure 6: 4 examples of tagging cell parts using GFP. Top left: cell nucleus, Top right: endoplasmatic reticulum, Bottom left: plasma membrane. Bottom right: cytosol. The white scale bar is 5 μm. Image source: [45]

In fluorescence microscopy, fluorescent molecules (also known as dyes) are used to highlight processes and structures within the sample. Fluorescent dyes have different excitation and emission spectra (Figure 7). This allows filters within a fluorescence microscope to only present the user with light that is emitted by one of the dyes. This filtering is done using a dichroic mirror, first applied by Brumberg in 1948. A dichroic

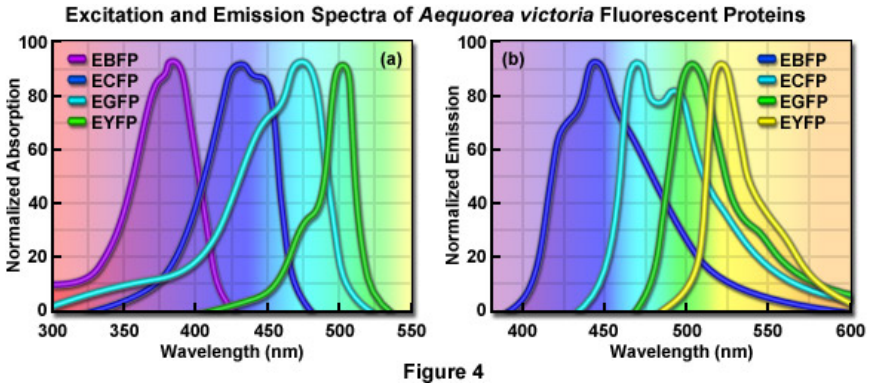


Figure 7: The absorption and emission spectra for 4 different fluorescent proteins (Blue, Cyan, Green, Yellow). Image source: [47]

mirror reflects some wavelengths, while being transparent to others. This allows separation of the exciting light from the emitted light coming from the fluorophores. This process is depicted in Figure 8 below.

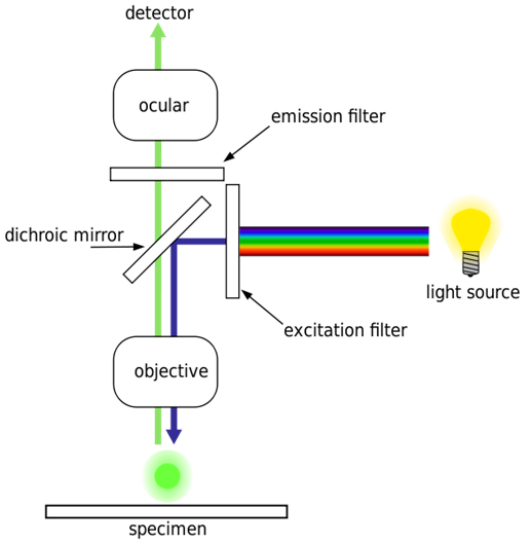


Figure 8: A fluorescence microscope uses a dichroic mirror which reflects the excitation light (blue) to the sample, but lets the emitted light (green) pass through to the detector, such as a CCD camera. Image source: [54].

Following the discovery of fluorescence, its usability for biological research was initially very limited. This changed with the contribution of Gregorio Weber [43] who, in 1952, first used synthetic fluorescent dyes bound to proteins. This allows to make observations about very specific proteins. A similar milestone that improved the specificity of fluorescence microscopy was the discovery and isolation of the Green Fluorescent Protein (GFP) gene from jellyfish DNA [37]. The GFP is a small protein that can fluoresce without additional substrates and has no protein folding requirements (it folds automatically, so no species-specific chaperones are needed). This

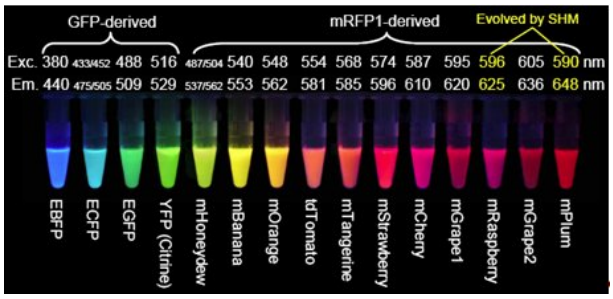


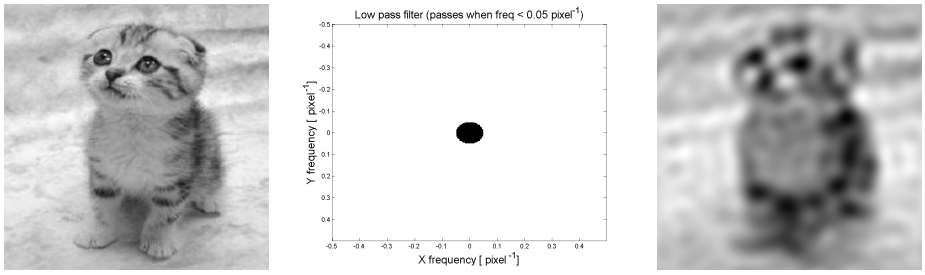
Figure 9: A large number of fluorescent proteins have been developed since the discovery of wild-type GFP. Image source: [48]

.

discovery was awarded with the 2008 Nobel Prize, because it resulted in an extremely practical tool for molecular biologists. Fluorescence markers based on the GFP gene can be built into the genome of an organism, so no external dyes have to be added. Following the discovery of wild-type GFP, modifications have been made to the GFP gene to adjust the absorption and emission spectra of the fluorescent protein. Fluorescent proteins are now available to in a variety of colors (Figure 9). This made it possible to simultaneously monitor different proteins or genes.

All these developments made fluorescence microscopy very well suited to imaging live samples and seeing biological processes at work, something that is more complex or even impossible with other techniques (such as electron microscopy)

One of the limiting factors of fluorescence microscopy (and light microscopy in general), is the wavelength of visible light. The physical process called *diffraction* will cause the light waves to blur the positions of the molecules, which results into a lower limit on the size of the details that can be imaged with fluorescence microscopy. Because of this effect, the lens system in a microscope setup behaves as a low-pass frequency filter, in which only the lower image frequencies are retained in the measured image (see Figure 10 for an example). Diffraction is a wave phenomenon that cannot be prevented in a microscope setup. For example, increasing the magnification of the optical system will not help, because it does not add any high frequency patterns to the image. Instead, it will just enlarge the patterns that are already there.



(a) Original image

(b) Frequency-domain filter

(c) Filtered image

Figure 10: The effects of a low pass filter(b) are shown applied to an image of a cat (source: [39]). The patterns with frequencies above 0.05pixel^{-1} (or a wavelength lower than 20 pixels) are lost in the filtered image(c).

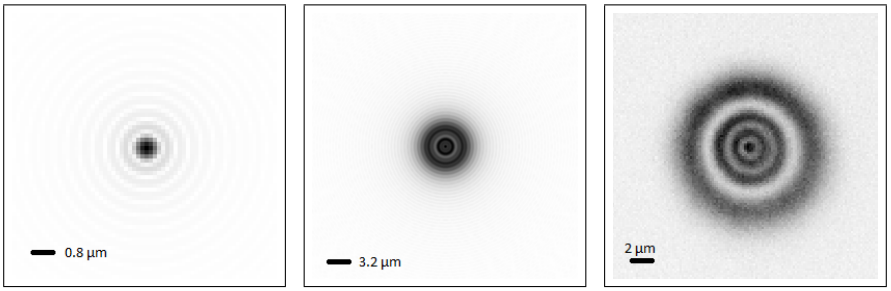
This diffraction limit is proportional to the wavelength of the light. Light coming from an emitter on the focus plane of a microscope is blurred according to the Abbe diffraction limit: $d = \frac{\lambda}{2NA}$, where d is the minimum distance at which 2 features in the image can be distinguished from each other, NA is the numerical aperture, and λ is the wavelength. For visible light, this means that diffraction prevents details smaller than $250nm$ from being distinguishable in the measured microscope image. A mathematical model for the diffraction process in a microscope will be discussed later in this thesis, as it is an important problem in this thesis.

With the advance of very sensitive camera electronics, such as Charge Coupled Device (CCD) microchips, it also became possible to record movies of biological experiments using fluorescence microscopes. These movies can then be processed with algorithms to extract useful information. Usually, this involves following the fluorophores through all the frames of the movie, which is a process known as *tracking*. However, in order to follow fluorophores in time, we first need to know where they are, a process known as *localization*. This will be discussed after introducing the Point Spread Function.

2.3 The Point Spread Function

The effects of diffraction and aberration are often combined into a single function called the *Point-Spread-Function* (PSF), which represents the way that the optical system transforms the light waves from the specimen into a measured image. The PSF describes how each point at the source is spread out on the measured image. Figure 11 shows some examples of PSFs. The PSF can be explained as a convolution filter that is applied on the wavefield coming from the fluorophores in the specimen.

The PSF is a result of complicated wave physics, and can be modelled and approximated with various degrees of complexity and accuracy. This subject will be discussed in section 4.



(a) In-focus PSF (b) Out-of-focus PSF (c) Real PSF

Figure 11: Some examples of point-spread-functions. (a) An ideal in-focus PSF, to which only diffraction contributes (no aberrations). (b) A simulated PSF of an out-of-focus particle. Except for defocus, no aberrations are included. (c) A experimental PSF image, from a fluorescent bead that is stuck in an agarose gel, measured in our setup. The presence of multiple rings indicate that this bead is not located at the focus plane.

2.4 2D localization

2.4.1 Introduction

Although diffraction will remove the high-frequency spatial information in the light wavefield, methods have been proposed to recover some of this information, given that the number of emitters in an area is small enough. If it is known that received light is emitted by a single emitter, then this knowledge allows computation of the position of that emitter with a higher accuracy than stipulated by the diffraction limit. This is known as *localization*.

By 2D localization, we mean computing the (X,Y) position of a particle (Figure 12). It is convention to define the image axes as X and Y, and call the optical axis (normal to the plane) Z.

2.4.2 Noise contributions

Any 2D localization method has to successfully deal with a number of problems:

- Localization methods are algorithms that run on a computer with finite memory and computational speed. This creates limits on the number of calculations for computing a position.
- Diffraction, as previously mentioned, will blur or transform the light pattern when it is projected by the lenses. The more the photons are spread out, the more photons are required to accurately determine the emitter location. Additionally, the diffraction is often modelled in a simplified way, which also introduces errors in localization.
- The photon detection (camera) chip has finite size pixels. This means that the square pixel area in which the photon hit is known, but not the exact location. This is referred to as the *pixelation* error.
- Fluorescence microscopy often deals with fluorophores that emit a small number of photons. This creates a noisy images, which will be discussed in detail later.
- Dark-current noise. A CCD camera will detect photons due to random thermal fluctuations.
- A discretization error: the electrical noise from converting detected light intensity to a digital value can be significant, if the CCD chip is not sensitive enough. In high-end microscopes such as used for this micromirror setup, an electron-multiplying CCD camera is used which greatly reduces the relative electrical noise, as the light signal is amplified while the electrical noise is not.

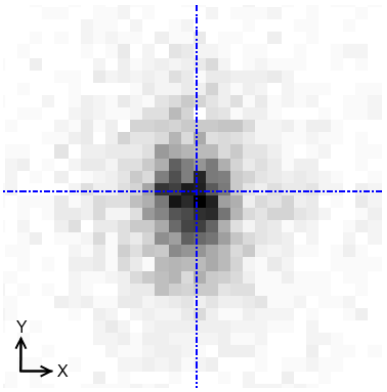


Figure 12: 2D localization: going from a noisy image of a fluorophore to an accurate subpixel position. The intersection of the blue lines shows a possible subpixel location at which the emitter could be. The image was generated using the micromirror simulation software written for this thesis.

2.4.3 Algorithms

Several localization algorithms have been proposed, and each of them has advantages and disadvantages. A straightforward method is to simply calculate the center-of-mass (COM) of an image *. However, this assumes that the particle is actually located at the center of mass, but does not account for noise that can shift the COM. The COM method does not model noise in any way.

What makes localization a complex topic is the need to extract as much information from the image as possible. The performance of localization algorithms in this extraction is explored by Cheezum et al. [7] and Ober et al. [26] by doing localization on simulated measurements. By using methods from *estimator theory*[17], Mortensen et al. [23] show that maximum-likelihood based fitting is able to localize focussed 2D fluorophores in an information-optimal way, whereas a least-squares fitting (such as done by [35]) will squander a considerable part of the information in the image.

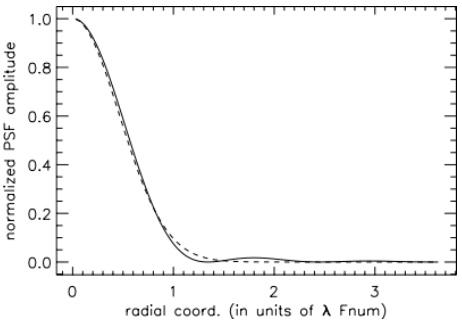


Figure 13: The analytical solution for the PSF in an aberration-free optical system can be described using the Airy function. At the focal plane, this Airy function (solid line) is often approximated using a Gaussian (dashed line) [49].

Next to the aforementioned noise contributions, any localization algorithm will make certain assumptions about the point spread function of the particle. The true PSF is a result of wave physics, but can be approximated with a Gaussian function (see Figure 13). This Gaussian approximation is commonly used to reduce the computational complexity of localization.

The implications of the use of the Gaussian PSF in localization has been extensively studied by Stallinga and Rieger [32], and statistically optimal methods have been created to fit the Gaussian PSF to image data using a maximum likelihood estimation (MLE) approach by Smith et al. [31].

2.5 3D localization

Because biological processes occur in a 3D environment, it would be ideal to have accurate localization in all 3 dimensions. Since the microscope projects the 3D world onto a 2D image, finding the Z coordinate requires additional measurements and processing. A variety of methods have been developed to perform 3D imaging and localization using fluorescence microscopy, but there are significant differences between them.

First of all, a number of methods for 3D imaging using fluorescence are based on confocal microscopy (such as Hell et al. [14]), in which a scanning laser is used to excite only a small part of the sample. This greatly reduces the background noise coming from out-of-focus fluorophores. Compared to wide-field microscopy, confocal microscopy is well suited to 3D scanning of larger structures, but suffers from higher photon damage to the sample and a slower speed (as it takes time to move the beam back and forth over the sample).

*To compute the Center-Of-Mass, multiply the value of each pixel by its position, and then divide by the sum of all pixel values:

$$\mathrm{COM}_x = \frac{\sum xI(x,y)}{\sum I(x,y)} \quad \mathrm{COM}_y = \frac{\sum yI(x,y)}{\sum I(x,y)}$$

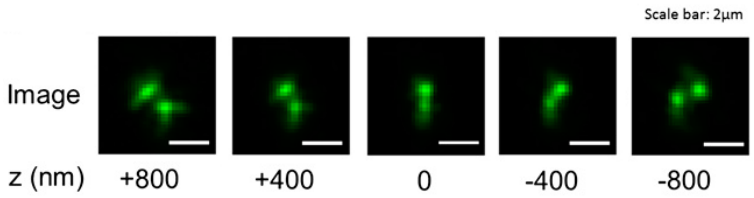


Figure 14: A double-helix point-spread-function, which rotates when the emitter moves in the Z direction compared to the focal planeRama et al. [30]. The double-helix shape is created using a spatial light modulator (SLM) in the back focal plane of the objective.

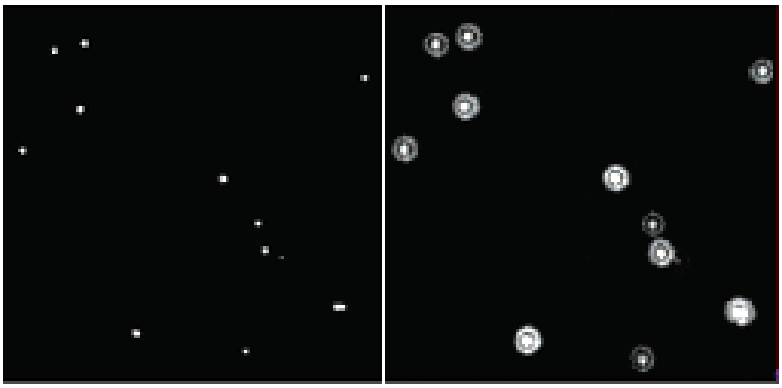


Figure 15: The method of Toprak et al. [36] splits the light into 2 separate beams with different focus depths (in-focus (left) and out of focus (right)). The sizes of the spots are used to infer the Z positions.

We are interested in using 3D localization for tracking in both in-vitro and in-vivo measurements. Because of this, confocal methods for 3D localization will be skipped, as wide-field microscopy is faster and therefore the preferred method for tracking.

In recent years, a few common methods for acquiring Z positions of fluorophores have emerged. The main idea behind these methods is to modify the PSF using various optical methods in such a way that it contains information about the Z position that can be extracted.

- Astigmatic lenses can be used to create an elliptical point-spread-function, whose shape depends on the Z position. The Z position can then be extracted by fitting this PSF to the image. This method has been used in 3D tracking of quantum dots (Holtzer et al. [15]) and localization microscopy, by Huang et al. [16] in their STORM method (Stochastic optical reconstruction microscopy).
- Toprak et al. [36] use bifocal microscopy, in which two light paths with a different focal depth produce two images (Figure 16). The Z position of emitters is then inferred from the sizes of the spots on each image (Figure 15).
- Sun et al. [33] also use a bifocal approach, but unlike Toprak et al. [36] their approach is set up in such a way that a displacement in Z of the molecule causes a displacement in Y in the measured images.
- Quirin et al. [29] and Rama et al. [30] use a spatial light modulator (SLM) device to create

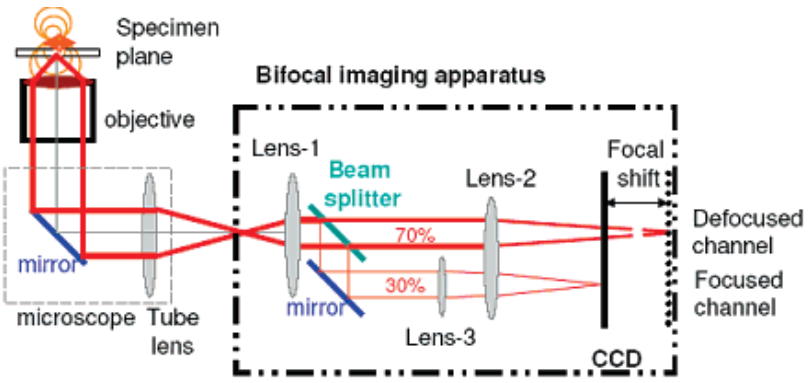


Figure 16: Toprak et al. [36] splits the light beam into 2 beams, each ending up on the camera with a different focus depth.

a point-spread-function with 2 spots that rotate when the Z coordinate of the emitter is changed (see Figure 14, imagine a cross section of the double-helix of two base-paired DNA strands).

Interestingly, these localization methods can be combined with fluorophore blinking techniques, to be able to isolate and localize each molecule individually (this is referred to as *super-resolution* fluorescence microscopy, used in methods such as PALM [5], STORM [2], or their successors [6]).

As mentioned, each of the listed methods has modified the regular optical system in such a way that the PSF depends on Z. However, this information about Z comes at a cost. Some of the considerations are:

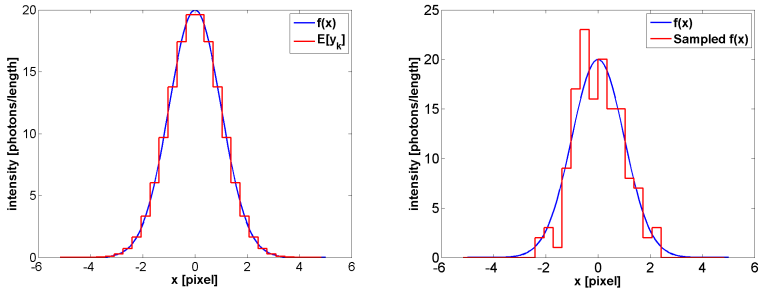
- The Z accuracy can be much lower than the X & Y accuracy (e.g. in the astigmatic method),
- Splitting or modification of the optical path results in a loss of photons along the way (in the bifocal methods or methods using an SLM).
- It complicates the optical system, leaving less room for additional features (such as in multicolor imaging)

Concluding, room for improvements in 3D localization remain.

2.6 Parameter estimation

This section shows an example of *maximum likelihood estimation* (MLE) of parameters, to introduce the method applied in this thesis. Localization, inferring the actual location of a particle in an image, is about estimating the parameters (X, Y, Z) that caused the particular signal (the image). MLE is used here as it makes optimal use of the information embedded in the image data (Mortensen et al. [23]).

This section will explain MLE for a set of samples measured from a 1D Gaussian signal, explaining localization in a 1D space. Later on, MLE will be used for the micromirror localization algorithm as well.



(a) (b)

Figure 17: Comparing the noise-free $f(x)$ in (a) with its Poisson-sampled version (b). Parameters were set to: $\sigma = 1$, $A = 20$, $\mu = 0$. The stepwise graph (a) shows λ_k for each pixel k. The sampled graph (b) has 30 pixels with length 1/3. The blue curve on both plots shows the continuous Gaussian profile itself.

This example problem starts with a signal defined by a 1D Gaussian function:

$$f(x) = A \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right) \tag{1}$$

Thinking of this as 1D localization, A defines the brightness of a particle, μ is the position and σ is known by assuming that $f(x)$ approximates the Airy disk. *. $f(x)$ has units of photons per length, or photons per area in the 2D case.

We take n samples $y_{1..n}$ from this function, at different x positions $x_{1..n}$ (see Figure 17). The samples are modelled as pixels in a CCD camera, which are commonly modelled with a Poisson probability distribution. This distribution is defined as $p(x|\mu) = \frac{\mu^x e^{-\mu}}{x!}$, where x is the discrete variable and μ its expected value $E[x]$. The Poisson distribution models a stochastic process with events happening at a certain average rate, with the time of each event being independent of the other events. Section 4.5 discusses the camera noise in more detail.

Maximum likelihood estimation implies maximization of a function called the *likelihood*. The likelihood of a parameter value is defined as the probability of getting the measurement given that parameter value. In other words, the likelihood is $p(\text{sample} \mid \text{model parameters})$. In this

*The central lobe of the Airy disk can be approximated with a Gaussian that has $\sigma \approx 0.42\frac{\lambda}{2\text{NA}}$ [49].

case, the model is the Poisson distribution, which has just one parameter (its expected value γ). If we have a single sample y, the likelihood of γ will be:

$$p(y|\gamma) = \frac{\gamma^y e^{-\gamma}}{y!} \tag{2}$$

The next step is to estimate the parameter μ and A of $f(x)$ by doing a MLE on the collection of pixels. If we have n samples, then the likelihood is computed by multiplying all the probabilities (assuming each sample is independent of the other samples):

$$p(y_{1..n}|\mu, A) = \prod_{k=1}^{n} \frac{\gamma_k^{y_k} e^{-\gamma_k}}{y_k!} \tag{3}$$

To compute this likelihood, the expected value γ_k of the Poisson process for each pixel needs to be computed. As mentioned earlier, $f(x)$ defines the number of photons per unit length, so the expected value γ_k (or $E[y_k]$) of a single pixel with length Δ is computed by integrating $f(x)$ over the pixel segment:

$$\gamma_k = E[y_k] = \int_{x_k-\Delta/2}^{x_k+\Delta/2} f(x)dx = \int_{x_k-\Delta/2}^{x_k+\Delta/2} A \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right) dx \tag{4}$$

Now that the likelihood contribution of each sample is known, the likelihood of the parameters given the complete measurement can be calculated by multiplying all the probabilities (assuming the measurements are independent):

$$p(y_{1..n}|\mu, A) = \prod_{k=1}^{n} \frac{{\gamma_k}^{y_k} e^{-\gamma_k}}{y_k!} \tag{5}$$

To better deal with numerical accuracy problems*, this product can be turned into a sum by using the *log-likelihood*†. An additional advantage is that the sum of logs can be differentiated more easily than a product (which would require endlessly applying the product rule)

$$\log p(y_{1..n}|\mu, A) = \sum_{k=1}^{n} \log \frac{{\gamma_k}^{y_k} e^{-\gamma_k}}{y_k!} \tag{6}$$

Modification of the likelihood to a log-likelihood does not pose a problem for the optimization algorithm: because $log(x)$ is monotonically increasing, the log-probability has its maximum at the same position as the probability. A log-likelihood plot of the parameter μ, the mean of the Gaussian, is shown in Figure 18.

*For example, 1000 samples each contributing a probability of 0.2 will result in a joint probability of 0.2^{1000}, which cannot be represented by the standard floating point hardware in a processor. However, $log(0.2^{1000}) \approx 301$, which does not have this problem.

†We use the natural logarithm, but any base would suffice as it is only a constant factor

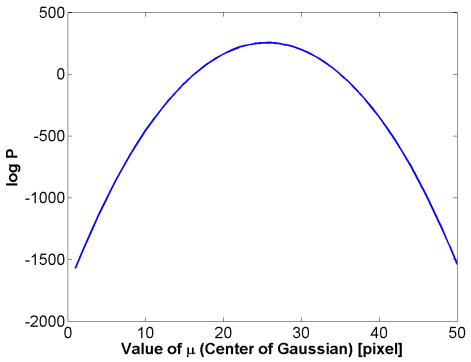


Figure 18: A plot of the log-likelihood computed over a range of values for μ, given the sample shown in figure 17b
. A and σ are set to the true values of the sample ($A = 20$ and $\sigma = 1$).

We have now turned a 1D localization problem into a MLE problem. The 1D pixels are modelled by a Poisson distribution, and the PSF is modelled by a 1D Gaussian function.

At this point, MLE consists of choosing μ and A such that the highest possible likelihood $p(y_{1..n}|\mu, A)$ is obtained. One way of solving this is to compute the derivative of the likelihood with respect to the parameters, and use a gradient based method to find the maximum. This is demonstrated for particle localization by Smith et al. [31]. However, the problem described above is a general *optimization* problem for which many algorithms have been proposed. Choosing a suitable algorithm for this will be explained in more detail as part of the micromirror localization algorithm.

3 Micromirrors

3.1 Introduction

This chapter will introduce the subject of micromirrors and discuss the work done in this area. The main concept is shown in Figure 19. A micro-meter sized mirror is placed in the sample, at an angle to the XY plane. The light from the particle is reflected by the mirror, and in this way the particle can be seen from two different vantage points simultaneously. Therefore, the maximum localization accuracy in Z is , at least in theory, as good as the accuracy in X and Y axes. The micromirrors are produced from a silicon wafer and a vapor-deposited aluminium layer, using methods from electronic chip technology, such as etching and lithography. A micromirror fabrication protocol can be found in Hajjoul et al. [11].

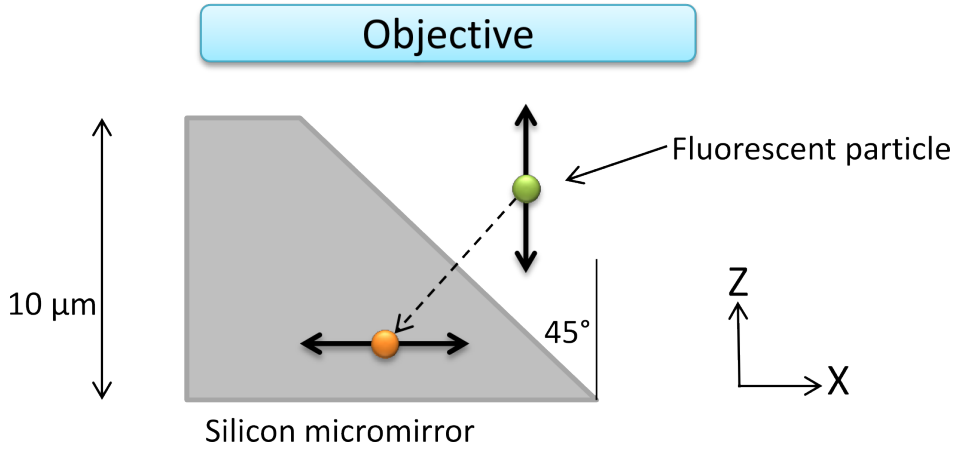


Figure 19: A simple micromirror concept is shown. When the particle moves up and down in the Z direction, the reflected image of the particle moves in the X direction.

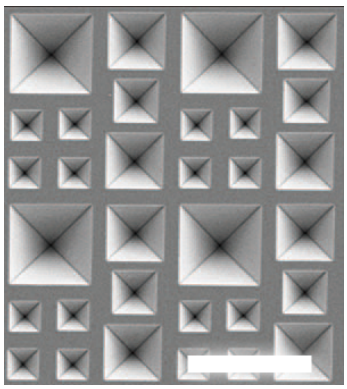


Figure 20: A Scanning-Electron-Microscope (SEM) image of the pyramidal micromirror wells (PMW) created by McMahon et al. [21]. The scale bar is 50 μ m.

As well as provide two additional vantage points to view the emitter, the micromirror also

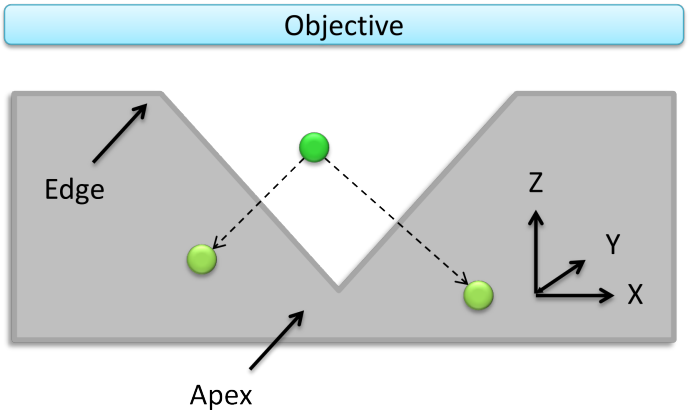


Figure 21: Schematic of the V-groove micromirror used in this thesis. The apex is also the origin of the space ($Z = 0$ and $X = 0$).

increases the amount of light received from the emitter compared to imaging the same emitter on a non-reflective surface. Depending on the reflectivity of the surfaces and on the numerical aperture of the objective, near 3 times as much light may be detected from the emitter (as the light is captured from two additional directions).

For reference, we define the coordinate system and micromirror geometry that is used throughout this thesis in Figure 21.

3.2 Previous work

The use of this idea in fluorescence microscopy was introduced by McMahon et al. [21] who used a pyramidal micromirror well (see Figure 20) to do fast 3D particle tracking. Focusing on a fast method, their localization algorithms were limited to Center-Of-Mass and similar [3] methods, which have limited accuracy even in 2D (see section 2.4.3) In addition to 3D tracking, 3D live cell imaging using silicon micromirrors was done by Hajjoul et al. [11] (see Figure 23). This group used V-groove micromirrors, so life cells can be flooded into the V-groove channels after construction (see Figure 22).

Some of the work done in micromirror research addresses problems specific to micromirror localization. This section will explain some of the problems and proposed solutions to them.

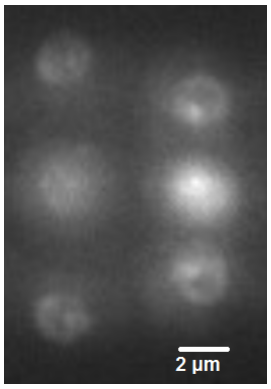
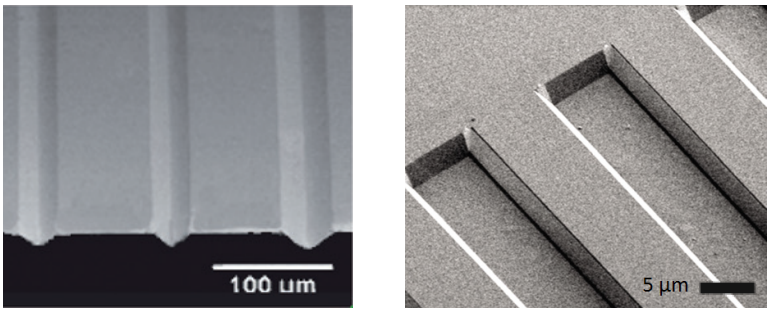


Figure 22: Results of Hajjoul et al. [11], showing a two yeast cells with GFP expression in a V-groove micromirror. The cells are noticably at a different Z position, as the left yeast cell has reflections that are further spaces apart than the right cell. This indicates that left cell is positioned closer to the objective than the right cell, as the distance to the mirror surfaces is larger for the left cell.

3.2.1 Secondary reflections

If reflecting mirror surfaces are in close proximity, light from the particle will be reflected multiple times between the mirror surfaces, creating several spots that are highly out-of-focus



(a) (b)

Figure 23: Electron Microscope images of silicon V-groove micromirrors (a) and single-facet micromirrors (b), as made by Hajjoul et al. [12].

(effectively a higher background noise level). Hajjoul et al. [12] solved this problem by using single-facet micromirrors (SF-MM) (Figure 23), that produces only one reflection of the particle (Figure 24).

3.2.2 Asymmetric aperture cut-off

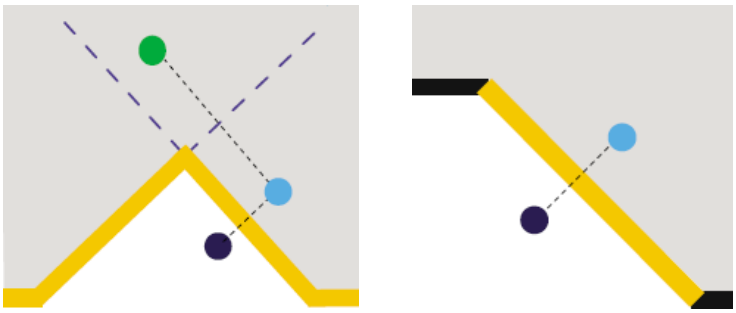
The spot image, in other words the shape of the PSF, is altered when some of the light cannot reach the objective (see Figure 25). The spot image turns from an Airy-disk into an asymmetric spot that cannot accurately be approximated with a simple analytic function, but needs a more complex model that takes the wave nature of light into account. Because it is asymmetric the actual particle position will not be at the center of the shape, so a Center-Of-Mass method or even Gaussian PSF is does not work accurately for localization. Berglund et al. [3] have analysed these localization accuracy problems using a *Fourier optics* simulation model. For a range of positions in the micromirror, they computed the COM a simulated PSF and subtracted it from the actual position to find the COM error. However, their model was only used to estimate the errors, no localization was tested using it.

3.2.3 Defocus for reflected spots

Creating a mirrored image of a particle creates an additional challenge of focussing on this mirror image (Figure 26). Tang et al. [34] solve this by using a bifocal microscope in combination with a micromirror. However, there is a clear trade-off here, because splitting the optical path in two will lose part of the photons, and make the localization process more complex (the localization algorithm needs to process two images).

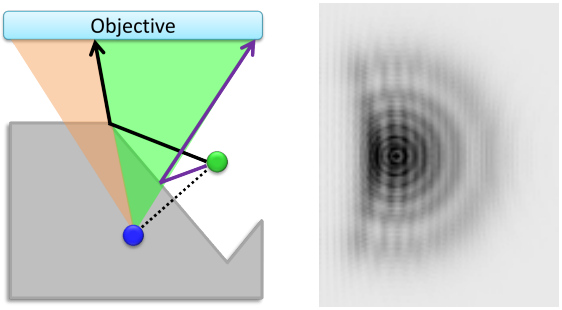
3.3 Research goal

Previous micromirror research demonstrates the potential of this method, but it also shows that additional efforts are required to be able to use it for localization microscopy. Therefore in this research, we set out the goal to create a localization method for micromirrors that can maintain



(a) (b)

Figure 24: Comparison of V-groove (a) reflections with single-facet micromirror (b) reflection. In the V-groove micromirror, the reflection (blue) of the particle (black) in one of the surfaces is itself also mirrored in the opposing surface (green). The single-facet micromirror solves this by eliminating one of the mirror surfaces. The black edges indicate non-reflecting surfaces, whereas the yellow edges are coated with a reflective layer. (Images from Hajjoul et al. [12])



(a) (b)

Figure 25: Light is cut-off by the micromirror shape in an asymmetric way (a), causing the point-spread-function to differ from the regular Airy disk (b). The red triangle indicates the light paths that would normally travel to the objective, but are now cut-off due to the finite size of the left mirror surface.

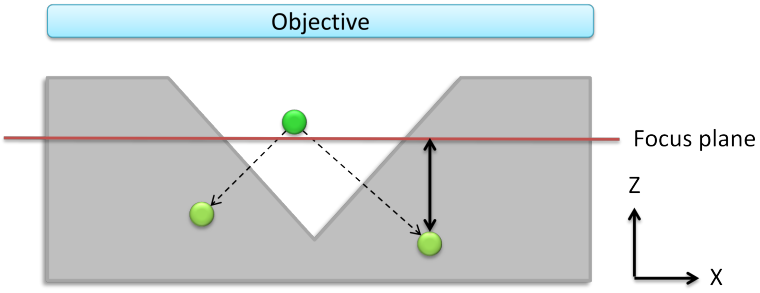


Figure 26: A trade-off needs to be made in micromirror microscopy: focussing more on the real particle will move the reflected image out of focus.

accuracy when the PSF is truncated. Ideally, it is comparable with 2D localization methods in terms of accuracy (e.g., make optimal use of the information in the image), but provide an increased Z axis accuracy.

Development of this localization method consists of:

- Defining the optical model. As mentioned in the section describing previous work, the conventional Gaussian PSF model is not suitable for localization in a micromirror setup. Instead, the method uses a *fourier optical* model based on the work done by Berglund et al. [4], which is explained in section 4.
- As the model is more complicated, so is the optimization method required to fit it. Development and characterization of the optimization methods is done in section 5.
- Evaluation of the performance of the method on simulated measurements is done in section 6

4 Optical model

This chapter discusses the optical model used for calculating the PSF for a particular fluorophore position. This PSF is fitted to a measured image in order to estimate the position. The optical model defined here describes what a particle that emits photons uniformly in all directions looks like when it is imaged in a micromirror microscope setup. Some considerations of optical modelling are given, and the concepts of Fourier optics and the angular spectrum are explained. The micromirror optical model is then defined in continuous form, and subsequently explained how this is discretized for numerical simulation. For a full reference on Fourier optics, the reader is refered to the books by Goodman [10] and Voelz [38].

4.1 Optical simulation

The simulation proposed in this chapter takes a set of parameters, and produces an image intensity $I(x, y)$, which defines what the expected image of a point light emitter in a micromirror setup. For reference, all the parameters used in the simulation a listed here:

- Emitter position: pos_x, pos_y, pos_z
- Focal depth: f_z
- Emitter intensity: I_0
- Background noise level: I_{bg}
- Mirror properties: apex position on x-axis (m_{apex}), depth (m_h), edge positions on x-axis (m_{left}, m_{right}) and reflectivity m_r
- Magnification: M
- Camera pixel size: w_{pixel}
- Numerical aperture: NA
- Medium refractive index: n_s
- Light wavelength: λ, we use $0.6nm$ in this thesis.

4.2 Model choice and assumptions

Modeling the image formation process of the micromirror experiments is one of the key elements in this thesis. Which model is the right model? The answer to this question depends on a number of requirements:

- The model needs to be detailed enough to extract the information that is in the image. For example, if a straight line is fitted to data points taken from a quadratic curve, one throws away part of the information.
- The model needs to be simple enough to prevent overfitting (e.g. having too many parameters compared to the information that is in the measurement).
- The computation required to process an image has to stay within practical limits. This computation time depends on both the complexity of the model itself, but also on how

many parameters have to be estimated. A larger number of parameters also creates a larger search space, which requires more iterations of the optimization algorithm.

The behaviour of light can be modelled with various levels of detail, ranging from simulating the electromagnetic wave field using Maxwell's equations (known as computational electromagnetics, which is highly accurate even on the nano-scale), or all the way up to geometrical optics (only valid when the wave nature of light can be ignored). Within this range many possible approximations exist.

However, if we consider the context of the application of this model, not many solutions remain. This model is applied to localisation of fluorophores in a typical fluorescence microscopy setup. Such a microscope setup produces many noisy images of a fluorophores, captured over a timescale of milliseconds. Solutions for problems defined in computational electromagnetics take hours to compute with today. hardware, which is unpractical for a localisation method that needs to process thousands of images. Geometrical optics is not appropriate for localisation of objects that are smaller than the wavelength of light. Finally Gaussian optics, which is used for localisation of fluorophores with an Airy disk PSF, is, is not sufficient to deal with the optical effects arising from aperture cut-off by the micromirror. Hence, we are left with a Fourier optical approach.

To apply Fourier optics, we have to make some assumptions about the light and geometry in our setup, to reduce the required complexity of the model. The book Goodman [10, chapter 3.2] mention that the following set of assumptions have to be made if we want to use scalar diffraction theory. In this context, scalar is used as opposed to vector diffraction theory, and indicates that the polarization of the waves are ignored.

- We assume a *linear* medium, which means superposition can be used for the wave field quantities. In the context of light waves, an example would be that the combined image of 2 light sources is equal to the images of each individual light source added together afterwards.
- We assume a *isotropic* medium, which means that a wave propagation speed is independent of its direction.
- We assume a *homogeneous* medium, in which the permittivity of the medium is independent of the position.
- We assume a *non-dispersive* medium, in which the permittivity is independent of the wavelength. All fluorescent emitters have a spectrum of light frequency they emit, so if the medium is dispersive then each different frequency will have a different path through the medium, comparable to how a prism works.

Especially the assumption of homogeneity is tricky, as it is usually not valid in a typical live cell sample. It is however an assumption that needs to be made if any computationally feasible simulation is to be done.

Furthermore, we also assume monochromatic light, and that diffraction within the micromirror sample (such as shown in Figure 27) is negligible. The latter assumption appears justified because of the large mirror size (10 μm) relative to the wavelength of light (~ 0.6 μm). At this moment however, no quantitative study has been done of the diffraction effects for fluorescence in the micromirror, as only Center-Of-Mass based methods have been used for tracking so far, and the Fourier-optical method of Berglund et al. [4] does not mention any experimental comparison of the generated PSF. Hence, it remains to be seen if this kind of diffraction makes any significant contribution to the PSF.

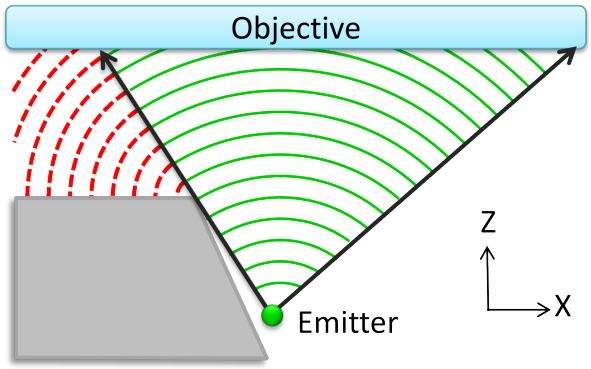


Figure 27: Diffraction at the edge of the micromirror could have an influence on the PSF. The figure shows the light beam that is produced by reflection into the mirror surface. In our current model, only the green part is accounted for, while the dotted red waves that could be present due to diffraction at the edge are not accounted for by our Fourier optical model.

Figure 28: We assume the particle emits a spherical wave.

Additionally, we assume that the fluorescent particles can be represented by a point light sources that emit a photons with a spherical wave (Figure 28), and that the photons have a random polarization. While single emitted photons can have direction and polarization, the collective result of imaging thousands of photons is approximately the same as imaging a particle that emits a uniform spherical wave. According to Stallinga and Rieger [32], this is true for rotating dipole emitters (such as a GFP), but not for dipole emitters with a fixed orientation. Ignoring polarization is not specifically required for Fourier optics, but allows us to simplify the calculations of image formation, by representing the waves with scalar values rather than 3D vectors.

The model we use to study the performance and behaviour of the 3D localization algorithm in simulation does not include aberrations, as the parameters defining the aberration would need to be estimated from experimental data. However, it is likely that the behaviour of the localization algorithm does not depend on small differences in the shape of the simulated PSF. For example, it is likely that only the existence, and not the exact location, of the fringes in the PSF's are important for testing the localization algorithm, as those can influence the complexity of the optimization problem (for example increase the number of local minima). In other words, the aberration constants are not known, but also not needed to assess localization performance in simulation.

4.3 Fourier optical modelling

As the term Fourier optics implies, the optics are defined in the Fourier domain, or *frequency domain* version of the optical signal. In numerical computation, this frequency domain repre-

sentation is often stored in a complex valued image that stores the phase and amplitude of each frequency. To illustrate the frequency domain in a simple way, figure 29b shows the frequency domain version of a regular image (29a). The center of the image contains the intensities of the lower frequencies (around $(0, 0)$), and the pixels near the edges define the intensity of the higher frequencies in the image.

(a) Original (b) Frequency-domain

Figure 29: Comparing spatial-domain (a) and frequency-domain (b) representation of an image. (b) shows the absolute value of the frequency-domain representation (the Fourier transform of an image is complex valued). The bright pixels in the center are indicative of significant low-frequency patterns in the image. Original image source: [51].

Mathematically, the 2D frequency domain signal $F(f_x, f_y)$ is generated by the 2D Fourier transform of the spatial domain signal $f(x, y)$:

$$F(f_x, f_y) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(x, y)e^{-2\pi i(f_x x + f_y y)}\mathrm{d}x\mathrm{d}y \tag{7}$$

It can be converted back to spatial domain using the inverse 2D Fourier transform:

$$f(x, y) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} F(f_x, f_y)e^{2\pi i(f_x x + f_y y)}\mathrm{d}f_x\mathrm{d}f_y \tag{8}$$

A similar use of Fourier theory can be applied to waves in optics. The Fourier representation of wave fields in the context of optics is explained in the following sections.

4.3.1 Wave superposition

To explain the Fourier theory applied to optics, we start with a 3D complex function (such as Figure 30), for example a wavefield that describes light waves. A cross section of the function, such as indicated by U is a 2D complex function: $U(x, y)$. Similar to a 2D image of pixels, one can Fourier-transform a 2D complex function such as $U(x, y)$ into the frequency domain $A(f_x, f_y)$:

$$U(x,y) = \int_{-\infty}^{\infty}\int A(f_x, f_y) e^{2\pi i(f_x x + f_y y)} \mathrm{d}f_x \mathrm{d}f_y \quad (9)$$

where f_x and f_y are known as the *spatial frequencies*.

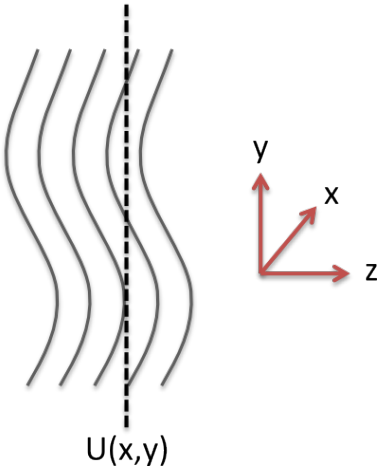


Figure 30: The constant phase lines of a 3D wavefield is shown. The reader has to imagine the wavefield extending into the paper, following the X axis. $U(x, y)$ is a cross-section of the 3D wavefield, which mathematically is just a complex valued 2D function.

4.3.2 Angular spectrum: Wave angles to spatial frequencies

Fourier optics comes into play if make an assumptions about the waves in the 3D wavefield: if all the waves have the same wavelength, we can consider $A(f_x, f_y)$ as the *angular spectrum* (see Goodman [10, chapter 3.10]). Using this assumption of uniform wavelength, the spatial frequencies defined by $A(f_x, f_y)$ can be directly converted into the angles of plane waves in the original 3D wavefield. In that case, each spatial frequency corresponds to a single plane wave angle, illustrated in the following example (see Figure 31).

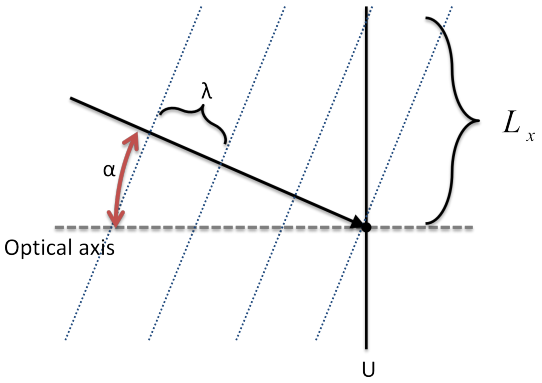


Figure 31: Plane wave hitting a plane perpendicular to the optical axis. λ is the wavelength of light, L_x is the wavelength of the spatial pattern at the plane U.

In the example, there is a wave moving towards a plane at some angle α to the optical axis. The wave has a wavelength of λ. With some trigonometry we can calculate the wavelength L_x of the

pattern on the screen:

$$L_x = \frac{\lambda}{\sin \alpha} \tag{10}$$

Writing it as frequency results in:

$$f_x = \frac{1}{L_x} = \frac{\sin \alpha}{\lambda} \tag{11}$$

This f_x term is a frequency in the planar wave projected on the U plane, with units of $[m^-1]$, which could be the plane of the camera CCD chip for example*.

From Fourier theory, it is known that a linear combination of sines and cosines can represent every 1D signal possible. Hence, a linear combination of planar waves (which are 2D cosines and sines) can represent any 2D shape on the screen. Goodman [10, chapter 3.10] contains a similar description for the plane wave in a 3D space.

4.3.3 Wavefield for an emitter

Now that we have discussed the concept of Fourier optics, the mathematics of the actual simulation can be defined. We will describe the optical simulation as a set of filters or transfer functions that are expressed in the frequency domain $A(f_x, f_y)$.

At the emitter position, we define $A_{emitter}(f_x, f_y) = 1$. In other words, all spatial frequencies are present at the emitter position†.

This $A_{emitter}(f_x, f_y)$ is then multiplied by all the functions defined in the steps below, which will be further explained in the following sections.

(A) We filter out all spatial frequencies that are blocked by the aperture.

(B) We filter out all spatial frequencies that are blocked by the micromirror.

(C) We add a phase difference to simulate a propagation of the wave to the focal plane, to account for the fluorophore z coordinate relative to the focal depth.

(D) Finally, we calculate a phase shift given the x and y coordinates of the fluorophore, relative to the optical axis.

The microscope will magnify the field at the focal plane and record the image intensity on its camera. The objective is a complex system of various lenses, designed to correct for all the optical effects that magnification involves. With the exception of possible aberrations (i.e., an imperfectly corrected objective), no simulation of the optics of the microscope itself is needed, because the magnification is simply a scaling of the units. Hence, magnification by itself does not require any simulation.

For reference, the f_x and f_y variables indicate the spatial frequencies of the wavefield inside the sample space. Because of magnification M, the spatial frequencies at the plane of the camera chip are $\frac{f_x}{M}$ and $\frac{f_y}{M}$.

*The image frequency in pixels can be derived from the spatial frequency as well, by multiplying with the pixel size. This can be seen if we look at the units: $\left[\frac{1}{m}\right]\left[\frac{m}{pixel}\right] = \left[\frac{1}{pixel}\right]$.

†If the aperture is infinite in size, the resulting inverse Fourier transform of the $F(f_x, f_y) = 1$ is a Dirac pulse $f(x, y) = \delta(0, 0)$, which for a point light source makes sense.

4.3.4 (A) Frequency limits due to the numerical aperture

Some part of the light waves emanating from the emitter will reach the objective and travel towards the camera chip, other parts will be blocked. The spatial frequencies in this light that will be transmitted are defined by the numerical aperture of the objective and possible aberrations in the microscope. The numerical aperture is defined by the refractive index n_s multiplied by the sine of maximum light angle $\sin\theta_{\max}$: $\mathrm{NA} = n_s \sin\theta$. Any light wave with sample-space angle larger $\theta_{\max}$ will not reach the camera chip:

$$\sin\theta_{max} = \frac{NA}{n_s} \tag{12}$$

Using Equation 11 ($\sin\alpha = \lambda f$) we can convert this angular limit into a spatial frequency $f_{\max}$:

$$f_{x,max} = \frac{NA}{\lambda n_s} \tag{13}$$

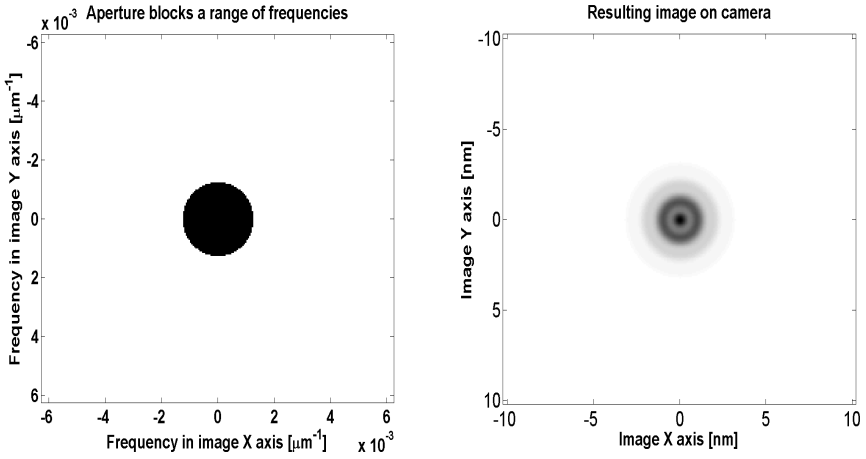
Since there are 2 axes (x, y) and it is a circular aperture, any total frequency has to fulfil the numerical aperture constraint:

$$\sqrt{f_x^2 + f_y^2} \leq \frac{\mathrm{NA}}{\lambda n_s} \tag{14}$$

We can rewrite this constraint as a transfer function:

$$H_{\mathrm{aperture}}(f_x, f_y) = \begin{cases} 1 & \text{if } \sqrt{f_x^2 + f_y^2} \leq \dfrac{\mathrm{NA}}{\lambda n_s} \\ 0 & \text{otherwise} \end{cases} \tag{15}$$

We now know which part of the angular spectrum will pass the aperture. As an example, we apply this frequency limit to a pattern where all the frequencies have a constant intensity $G(g_x, g_y) = 1$ and are in phase with each other:

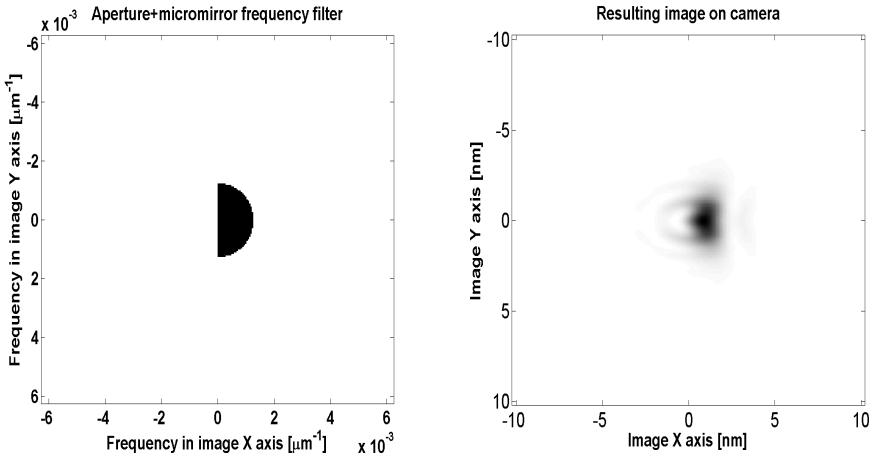


(a) Frequency-domain filter (b) Resulting PSF

Figure 32: An emitter at 2 micron away from the focal plane is imaged using an optical system with properties NA = 0.4, $M = 100$, pixelsize = $4\mu m$, $\lambda = 0.6$, and $n_s = 1.33$ (properties where selected such that the PSF was large enough to analyse). Only frequencies inside the circle (a) are seen on camera chip. This will produce a PSF as shown in (b).

4.3.5 (B) Frequency limits due to the micromirror

The micromirror blocks waves with certain angles (or frequencies in the image), which can be dealt with in a similar way as the numerical aperture above.



(a) Frequency-domain filter (b) Resulting PSF

Figure 33: The same emitter and optical system as Figure 32 are used, but some X axis frequencies are cut-off as they are in a micromirror setup. Only frequencies inside the shape (a) are seen on the camera chip. This will produce a PSF as shown in (b).

The calculation of the cut-off frequencies for the micromirror involve some trigonometry, illustrated using a very steep mirror surface in Figure 34.

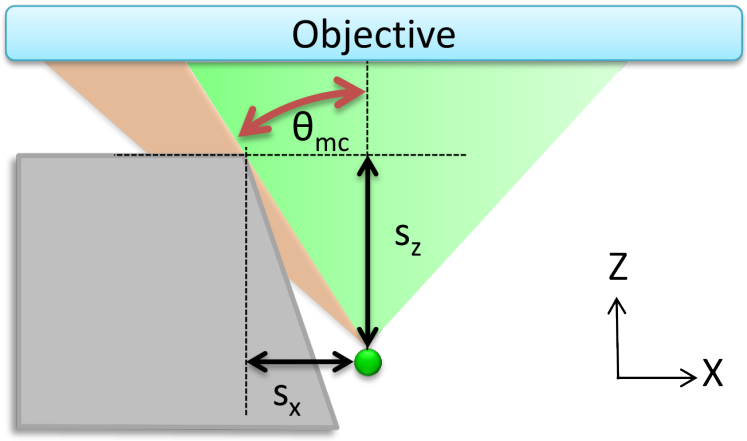


Figure 34: Light blocking by the micromirror, as already discussed in section 3.2.2. Calculating micromirror frequency cut-off is done using the distances s_x and s_z of the emitter (the green dot) towards the edge of the micromirror surface. θ_{mc} is the angle of micromirror cut-off

The sine of θ_{mc} is given by:

$$\sin\theta_{\text{mc}} = \frac{s_x}{\sqrt{s_x^2 + s_z^2}} \tag{16}$$

which can substitute the $\sin\theta_{max}$ from the numerical aperture, to get the spatial frequency at which the micromirror starts blocking. Equation 11 is used again to derive a maximal spatial frequency:

$$f_{x,max} = \frac{1}{\lambda}\frac{s_x}{\sqrt{s_x^2 + s_z^2}} \tag{17}$$

Note that, in a V-groove mirror, only the X axis frequency is subject to the micromirror cut-off.

Again, rewriting this as a transfer function:

$$H_{mc}(f_x, f_y) = \begin{cases} 1 & \text{if} \quad \lambda f_x \leq \dfrac{s_x}{\sqrt{s_x^2 + s_z^2}} \\ 0 & \text{otherwise} \end{cases} \tag{18}$$

4.3.6 (C) Contribution of the z position to the wavefield

To account for the z position of the emitter relative to the focal plane, wavefield propagation needs to be defined: we need to compute what happens with the wavefield when it propagates from the emitter to the focal plane.

One option would be to use the Rayleigh-Sommerfeld solution (see Voelz [38, chapter 4.4]) for computing the propagation of the wavefield over a distance z. It can be multiplied with the

wavefield at the emitter plane (the plane parallel to the focal plane, but with a different Z position) to simulate the wave propagation to the focal plane.

The Fresnel diffraction solution, which is an approximation of the Rayleigh-Sommerfeld solution, is used to test the localization algorithm with simulated data. A benefit of using the Fresnel solution is that it does not produce an imaginary phase shift for certain values of f_x and f_y, as Rayleigh-Sommerfeld does (see subsection 4.2 for the reasoning behind this).

The transfer function of the Fresnel diffraction solution is:

$$H(f_x, f_y) = e^{ikz} \exp\left(i\pi\lambda z(f_x^2 + f_y^2)\right) \tag{19}$$

As mentioned before, a physically correct approach would be to compute the optical path length according to an aberration model, and use this path length to compute a phase difference for each planar wave angle (such as done by Gibson and Lanni [9]). This aberration model requires many coefficients that have to be experimentally estimated, and this is assumed unnecessary for validation of the localization algorithm with simulated data.

4.3.7 (D) Contribution of the x and y position to the wavefield

Finally, we also need to correct for the position (p_x, p_y) of the emitter. For this, the shift theorem for Fourier transforms can be used (Goodman [10, chapter 2.1.3]), which says that if $\mathcal{F}\{g(x, y)\} = G(f_x, f_y)$ then

$$\mathcal{F}\{g(x - a, y - b)\} = G(f_x, f_y)e^{-2\pi i(f_x a + f_y b)} \tag{20}$$

In other words, moving the image by (a, b) is equivalent to a phase shift of $-2\pi(f_x a + f_y b)$ in the Fourier domain.

This contribution from the (x, y) coordinates can be defined as another transfer function H_xy:

$$H_{xy}(f_x, f_y) = \exp\left(-2\pi i(p_x f_x + p_y f_y)\right) \tag{21}$$

4.3.8 Combining all the contributions

Now, we can define the optical model by multiplying all the transfer functions defined in steps A to D. Function parameters (f_x, f_y) are omitted for clarity

$$G(f_x, f_y) = H_z H_{xy} H_{aperture} H_{mc,left} H_{mc,right} \tag{22}$$

The last step of is to move from the frequency-domain to the spatial domain using a Fourier transform:

$$U(x, y) = \iint_{-\infty}^{\infty} G(f_x, f_y) \exp\left(i2\pi(f_x x + f_y y)\right) df_x df_y \tag{23}$$

The CCD camera chip will not record the complex valued wavefield $U(x, y)$. Instead it will detect photons distributed according to the *image intensity* $I(x, y)$, which can be computed

from $U(x, y)$ by taking the square of its absolute value (see Goodman [10, chapter 4.1] for a physical explanation of the image intensity):

$$I(x, y) = |U(x, y)|^2 \tag{24}$$

The brightness of the spots on the image also needs to be modelled. It can be expected that brightness for reflections and direct images are different, since the amount of light blocked by the micromirror geometry differs between spots. Furthermore, the mirrors have a certain reflectivity (~ 90 for aluminium [1] and $\lambda = 0.6$ μm) that will lower the brightness of the reflected spot. The transmittance of light through the objective however, depends on the aberrations in the objective and the angle at which the light enters the objective. This will have to be considered when the aberration is modelled, as it requires experimental data. For the simulated performance tests, we will simplify spot brightness by simply keeping it constant relative to the total emitter brightness. The brightness of the reflected spot is scaled according to the reflectivity of the mirrors.

Thus, the top level simulation procedure is:

1. Given the fluorophore position, compute reflected positions.
2. Simulate the direct spot image.
3. Simulate the reflected spot images.
4. Sum the images, either before or after the Fourier transform. If the frequency-domain versions are summed (the $G(x, y)$ for each spot), then the reflected light interferes with the direct spot light, and if the spatial-domain images (the image intensities) are summed, then the reflected light does not interfere with the direct image. First experimental observations of fluorophores in the micromirror show that the last case is the most representative of reality (no interference between spots).
5. Normalize the image to match the emitter strength, and add the background constant.

We choose not to let the direct and reflected light interfere with each other, so the complete simulation expression will be $I_{combined}(x, y) = I_{direct}(x, y) + I_{lr}(x, y) + I_{rr}(x, y)$, where I_{lr} indicates the image intensity for the left reflection, and I_{rr} indicates the image intensity for the right reflection.

4.3.9 Computing reflected positions

The final simulated image of a fluorophore in a V-groove should contain one direct spot, and two reflected spots. This section defines some vector math in order to compute these reflected spot positions.

An infinite plane can be described by a normal vector ($\mathbf{n}_{plane}$) and a distance (d_{plane}) from the origin. The distance $d(\mathbf{v}, plane)$ of a point $\mathbf{v}$ to the plane can be calculated using

$$d(\mathbf{v}, plane) = \mathbf{n}_{plane} \cdot \mathbf{v} - d_{plane} \tag{25}$$

where d_{plane} is the distance along the plane normal vector $\mathbf{n}_{plane}$ to the origin. In this setup the origin always lies on the intersection of both mirror planes, so $d_{plane} = 0$.

The symbols are shown in Figure 35 and defined as:

- **p**: Particle position vector.
- **n**: Mirror surface normal vector. This is a unit vector which can be computed from the surface angle η: $\mathbf{n_x} = \sin\eta$, $\mathbf{n_y} = \cos\eta$. η is negative for the right mirror surface, and positive for the left.
- $\epsilon = \mathbf{p} \cdot \mathbf{n}$: Distance from emitter to plane can be computed using the inner product between the position and the normal.
- $\mathbf{p}_r = \mathbf{p} - 2\varepsilon\mathbf{n}$: Reflected particle position vector.
- **O**: Origin $(0, 0, 0)$.

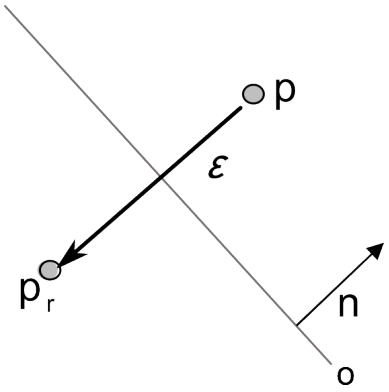


Figure 35: Reflected positions are computed using $\mathbf{p}_r = \mathbf{p} - 2\varepsilon\mathbf{n}$, where $\mathbf{p}_r$ is the reflected position, $\mathbf{p}$ is the particle position and $\mathbf{n}$ is the mirror surface normal.

4.4 Discretization

To numerically simulate the image formation the 2D inverse continuous Fourier transform (CFT) has to be translated into a 2D inverse discrete Fourier transform (DFT):

$$u(x, y) = \sum_{p=-W/2}^{W/2-1} \sum_{q=-W/2}^{W/2-1} g(p, q) \exp\left(\frac{2\pi i}{W}(px + qy)\right) \tag{26}$$

Here $u(x, y)$ is the spatial-domain image, $g(p, q)$ is the frequency-domain version of the image and W is both the width and the height of the image (it is square for simplicity). Converting to this inverse discrete Fourier transform is done according to Voelz [38, chapter 2.3].

In the DFT above m, n are integer pixel indices in the range $[-\frac{W}{2}, ..., \frac{W}{2} - 1]$. p and q are integer indices that index the frequency-domain image $g(p, q)$, and are also in the range $[-W/2, ..., W/2 - 1]$. In the Fourier domain, this same image can be defined with a grid of frequencies, where the frequencies on each axis are:

$$f_x = \left[-\frac{1}{2\Delta x}, ..., \frac{1}{2\Delta x} - \frac{1}{W}\right] \tag{27}$$

where Δx indicates the physical pixel size.

This can be explained by realising that $\frac{1}{2\Delta x}$ represents the highest frequency in the image (a wave period of 2 pixels which represents the maximum change between 2 neighbouring pixels). The interval between every frequency is: $\Delta f_x = \frac{1}{W}$.

The discrete frequency coordinates (p, q) and the continuous frequency coordinates (f_x, f_y) are related according to:

$$\begin{aligned} f_x &= \frac{p}{W\Delta x} \\ f_y &= \frac{q}{W\Delta y} \end{aligned} \tag{28}$$

As f_x and f_y are spatial frequencies in the sample space, Δx needs to be the pixel size w_{pixel} scaled by the magnification M:

$$\Delta x = \Delta y = \frac{w_{pixel}}{M}$$

The actual computation is done using a Fast Fourier Transform (FFT), which is an algorithm that computes a result equivalent to the above equation in a numerically efficient way. The speed of FFT's, and the broad availability of fast implementation of FFT is essential for a practical application of the localization method. We use an implementation that runs on fast consumer graphics hardware, see appendix B for details on this.

Another important factor in the runtime of the algorithm is the simulation resolution. The optical model defines a continuous image intensity $I(x, y)$, but the quantity of interest is actually the expected photon count on each pixel (which can be compared to a measured image). This expected photon count will be proportional to the image intensity $I(x, y)$, integrated over the complete pixel area. However, this integral cannot be computed directly, it can only be approximated by sampling $I(x, y)$ at a position in the pixel area. The proposed algorithm will simply use the image intensity at the center of the pixel. If higher accuracy is needed, an option would be to compute the inverse DFT at a high resolution, and then scale down the simulated image to the resolution of the actual camera chip.

4.5 Noise modelling

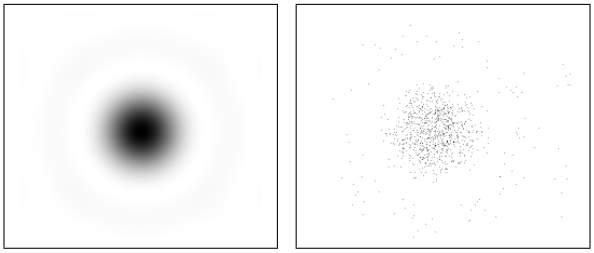
Application of fluorescence microscopy to biophysical or biological experiments limits the amount of light available for the camera, for three reasons:

- Fluorophores have a probability of bleaching every time a photon is absorbed.
- The excitation laser causes photodamage to the sample, which can interfere with the process that is being studied.
- When tracking moving objects, a short exposure time is important to take a snapshot of the situation, instead of a blur that cannot be localized.

Hence, it is a tradeoff between using a larger excitation power, producing more light for the camera, and having fewer of the negative effects described above.

The advanced Electron-Multiplying CCD (EMCCD) camera's have such low noise that the captured pixel values can be converted in photon counts [41, 42]. Because the light detected per pixel can be as low as a few photons, the distribution of the photon counts should be taken into account.

If the Electon-Multiplying capability of the camera is used, the majority of the noise encountered in photon detection is known as *shot noise*, which has a Poisson distribution and arises from the conversion of discrete energy quanta in light waves to single electrons. This distribution is defined as $p\left(x|\mu\right) = \frac{\mu^x e^{-\mu}}{x!}$, where x is the discrete variable and μ its expected value $E[x]$. The Poisson distribution models a stochastic process with events happening at a certain average rate, with the time of each event being independent of the other events.



(a) PSF (b) Poisson sampled

Figure 36: An arbitrary Airy-disk PSF (a) and its Poisson sampled version (b), for a total of 1000 photons.

The image I_k that the simulation generates defines the expected values ($\mu = I_k$) for each pixel. An example of this can be seen in Figure 36.

5 Localization

5.1 Defining the optimization problem

The optical model takes a set of parameters θ, and outputs an image of the intensity $I(x, y)$. Estimating the position of an emitter entails fitting this model to a measured image, and finding a set of parameters for the model such that the output image $I(x, y)$ matches the measurement with the lowest error possible (we will give a definition of this error below). According to 4.4, the image intensity $I(x, y)$ from the model is discretized as a set of pixel values $I_{1..n}$, where n is the total number of pixels in the simulated image (e.g. 65536 for a 256x256 image). Each I_k indicates the expected number of photons that are detected on pixel k.

Now, the localization problem can be formulated as a *numerical optimization* problem, where the goal is to choose parameters θ minimize an *objective function*. The objective function of an optimization problem is an arbitrary function that takes a set of parameters, and returns a score value computed from those parameters. A numerical optimization algorithm will search for the parameters that maximize or minimize the score computed by the objective function. In our case, the objective function can be defined as a sum of the errors over all pixels, comparing the measured image $v_{1..n}$ to the simulated image $I_{1..n}(\theta)$:

$$L(\theta) = \sum_{k=1}^{n} e\left(I_k(\theta), v_k\right) \tag{29}$$

Hence, the optimization goal is searching for the set of parameters θ that minimize $L(\theta)$. $e(I_k, v_k)$ is the *error metric* that defines the error between the pixel value v_k and the simulated pixel value I_k.

5.1.1 Error metrics

As an example of an error metric, a least squares optimization can be obtained by setting the error metric $e(a, b)$ to the squared difference of the two values, $e(a, b) = (a - b)^2$. If if $a = v_k$ and $I_k = b$, then this error metric results in the objective function L_{sqe}:

$$L_{sqe}(\theta) = \sum_{k=1}^{n} \left(x_k - I_k(\theta)\right)^2 \tag{30}$$

However, for an error metric that results in maximum-likelihood estimation (as discussed in subsection 2.6), we start with probability distribution of the photon counts that are derived from measured pixel values. As indicated in subsection 4.5, these photon counts can be modelled as independent Poisson processes, with a probability distribution:

$$P\left(x|\mu\right) = \frac{\mu^x e^{-\mu}}{x!} \tag{31}$$

Here x is the discrete random variable, and μ is the expected number of events.

Taking the logarithm of this probability $P(x|\mu)$, which is needed to write the joint probability on all pixels as a sum of log-probabilities.

$$\log P\left(x|\mu\right) = \log \frac{\mu^x e^{-\mu}}{x!} = x \log \mu - \mu - \log x! \tag{32}$$

The log-likelihood is negated in so it can be used as an error metric, $e(x, \mu) = \mu - x\log(\mu) + \log(x!)$. In this way, minimizing the error results in maximizing the likelihood. Finally, to apply this distribution to our localization problem, we set $x = v_k$ (number of photons measured on a pixel) and $\mu = I_k$ (the expected number of photons on a pixel) which results in:

$$L(\theta) = \sum_{k=1}^{n} v_k \log(I_k(\theta)) - I_k(\theta) - \log(v_k!) \tag{33}$$

where k is the pixel index, in the range $1..n$.

When searching for the optimal value of θ, the last term can be dropped because it does not depend on $I_k(\theta)$. In mathematical terms this can be expressed as:

$$\arg\max_{\theta} L(\theta) = \arg\max_{\theta} \left(\sum_{k=1}^{n} I_k(\theta) - x_k \log(I_k(\theta)) \right) \tag{34}$$

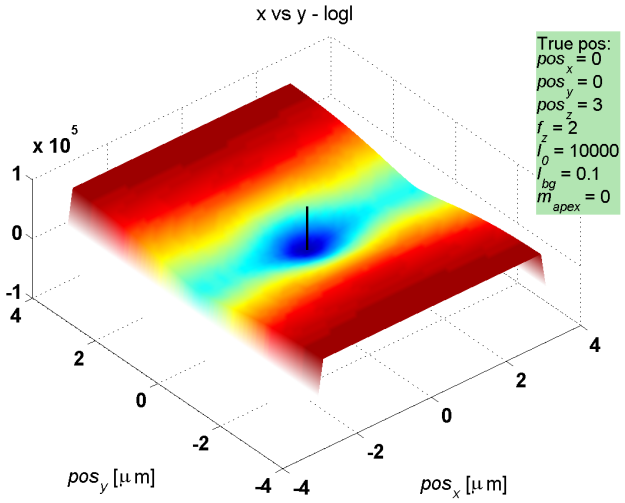


Figure 37: Log-likelihood optimization landscape for a range of pos_x and pos_y coordinates. Generating this graph involved the following steps:

- First, the parameters (indicated by the values in the green column) were used to compute the $I_{1..n}$ for the true position.
- A measurement was simulated. That is, values for $v_{1..n}$ were randomly generated, according to each pixels photon count distribution $p(v_k|I_k)$.
- The log-likelihood was evaluated for different points on the (pos_x, pos_y) plane, and used for plotting. The log-likelihood evaluation was done using the parameters in the purple column (except for variables pos_x and pos_y).

Table 1 shows the meaning of the symbols in the parameter column.

The search space can be visualized by plotting it for various parameter values. However, this is always a very limited view, since the search space has at least 4 dimensions or more, and the graph can only show a 2D landscape. If the search landscape is plotted around the true value,

Name	Symbol	Dimension
Emitter position	pos_x, pos_y, pos_z	μm
Focal depth	f_z	μm
Mirror apex X position	m_{apex}	μm
Emitter brightness (number of photons received from emitter)	I_0	$photons$
Background noise level	I_{bg}	$photons/pixel$

Table 1: List of parameters used in the search landscape plots, with symbols and their units.

as done in Figure 37, the black vertical bar is located in the global maximum. Of course, if other evaluation parameters are also modified, then the global maximum can shift away from this location in the graph.

While the pos_x-pos_y search landscape shows simple search landscape, varying other parameters can result in a complex multi-modal search problem, such as shown in Figure 38.

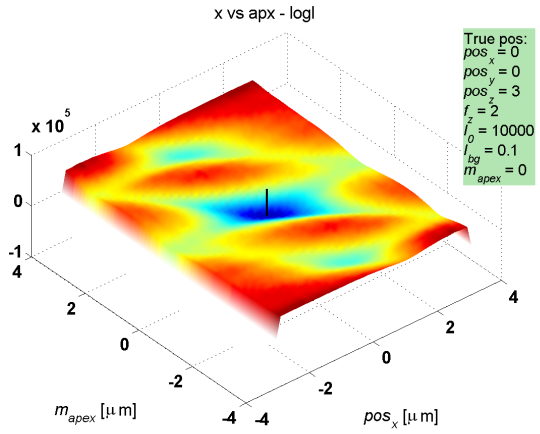
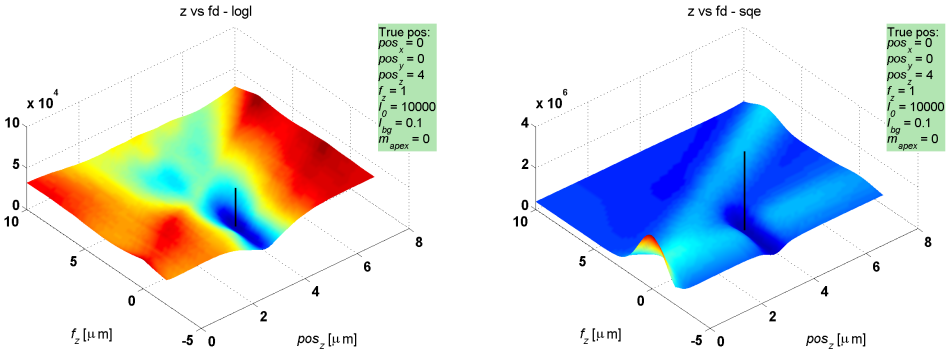


Figure 38: Log-likelihood optimization landscape of pos_x vs m_{axex}. The combination of the two parameters creates a landscape with many local minima. Table 1 shows the meaning of the symbols in the parameter column.

5.1.2 Comparing error metrics



(a) Log-likelihood

(b) Squared error

Figure 39: Error landscape comparison for the (pos_z, f_z) plane, showing the log-likelihood search landscape (a) versus the squared-error search landscape (b). Although both metrics have some local minima, it can be seen that the log-likelihood landscape keeps a higher curvature than the squared error landscape, when the distance from the true position increases. A higher curvature is likely beneficial during optimization, as it forces the search algorithm towards the true position.

The graph of the error metric was plotted in various ways. A 2D surface plot of the squared-error on the (pos_z, f_z) plane is shown in 39b, alongside a log-likelihood plot in 39a of the same parameter space.

5.2 Choice of parameters

The measured images contain a limited amount of information, which also limits the number of parameters one can accurately fit. Fitting too many parameters is known as *overfitting*: it decreases the accuracy of the estimate of each individual parameter. If on the other hand we assume the value of many inputs to be known, the resulting estimates of the remaining parameters will be precise (high resolution), but may not be an accurate description of reality, because of invalid assumptions. The computational aspect is also important: every unknown parameter increases the search space of the optimization problem, which can make the algorithm impractical to use. For every model parameter, we can choose between several options:

- It is a parameter that we want to determine, such as the fluorophore coordinates.
- We assume the input is a known constant, such as the speed of light.
- We assume the input depends on microscope setup properties, but stays constant during a single experiment. This means it could be fitted prior to the experiment itself. For example, the numerical aperture of the objective.
- We assume that the input depends on sample properties, so it can be fitted prior to the localization measurements. For example, the refractive index of the medium, or the position of the micromirror V-groove.

5.3 Optimization algoritms for micromirror localization

We now have a model and error metrics to compare the measurements to the model. The next step is to find the set of parameters for the model that minimizes the error (or maximizes the likelihood).

As it is a continuous search space with many dimensions, it is impossible to evaluate the error at every possible location. The combination of a computationally highly complex objective function $L(\theta)$ and a large search space (many parameters) demands an efficient numerical optimization method. Numerical optimization is a research area by itself; Nocedal and Wright [25] give a good description of the many optimization algorithms that exist. Out of the many optimization methods to choose from, a few are selected that seem suitable for the micromirror localization problem.

The localization problem is split up in two phases:

- In the *alignment* phase, the first frame of a fluorophore movie is processed in order to estimate the mirror apex position ($apex$), the focus depth (fd) and optionally, the background noise level I_{bg}, if we assume that it will not change.
- In the *tracking* phase, we know the approximate position of the emitter (from the previous frame or the alignment phase) and the other parameters are assumed fixed and known. Thus, the problem is reduced to estimating the emitter position (pos_x, pos_y, pos_z).

Three different optimization methods are applied to both of these problems, in order to find the most suitable optimization method for each phase. These three methods are **1)** the MATLAB implementation of the Nelder-Mead simplex method (the MATLAB `fminsearch` function), **2)** a *gradient descent* method that walks to the minimum along the loglikelihood gradient using fixed size steps, and **3)**, the Particle-Swarm-Optimization (PSO), which is a global optimization method suitable for solving optimization problems with multiple minima.

is tested as a method primarily to solve the alignment case, when the search landscape has more local minima. Compared to the other optimization methods, the PSO is better suited to solving problems with many local minima, if given enough particles and iterations. It should be noted that without specific knowledge of the search space, no optimization method is guaranteed to find the global minimum in a continuous search space with multiple minima. However, some optimization methods are more suited to particular situations than others, which has to be determined by reason and experimentation.

5.3.1 Nelder-Mead Simplex method

The MATLAB `fminsearch` methods is an implementation of the Nelder-Mead simplex algorithm (Nelder and Mead [24]). This algorithm is a *simplex*-based search method. A simplex is a convex geometrical shape with $n + 1$ vertices, where n is the number of dimensions in the space. For example, in 2D space a simplex is a triangle, and in 3D space a tetrahedron (Figure 40). Each iteration, the Nelder-Mead algorithm evaluates the objective function at the simplex vertices, and decides how it should modify the simplex for

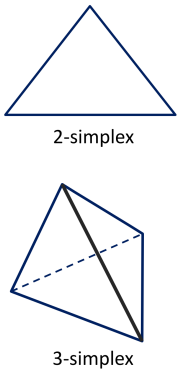


Figure 40: A simplex in 2D space is a triangle (top), and a simplex in 3D space is a tetrahedron (bottom)

the next iteration (such as shrinking the simplex or moving a vertex). The overall properties of the Nelder-Mead algorithm are [19]:

- No gradient approximation is performed.
- It is designed for a unimodal problem (no local minima).
- It is known to work efficiently in practice, but theoretical analysis of the algorithm and its convergence is limited to very simple cases.

5.3.2 Gradient descent

The gradient descent method has been implemented in the following way:

1. Several random start positions are generated uniformly in the search space.
2. For each of the positions, the gradient vector is calculated using numerical approximation*.
3. The computed gradient vector is scaled to a constant length, and added to the position. By scaling the gradient vector, we make sure that the algorithm still moves at an acceptable speed, even if computed gradient is very small (that is, when the landscape is nearly flat). Furthermore, incorrect jumps due to any discontinuities in the landscape (caused by numerical errors or noise) are limited to small distances.
4. Step 2 and 3 are repeated until the error metric does not significantly decrease anymore. At that point a minimum has been found.
5. The parameter set θ with the smallest error of all the visited values for θ is returned as the final estimated θ

Considering the search landscape, the gradient descent method is most suitable for the tracking phase, where a recent emitter position is known and the search distance is limited to a few microns. In that case, it is likely that the nearest minimum is also the global minimum.

5.3.3 Particle Swarm Optimization

The Particle-Swarm-Optimization (PSO) algorithm is a global optimization algorithm that was inspired by bird flocks searching for corn [18]. Since its introduction, the method has been further analysed and widely applied for different applications [28]. In a PSO, a number of particles move through the optimization search space, and communicate with each other to find the optimal position in that search space. Each of the particles has a position $\mathbf{p}$, a velocity $\mathbf{v}$. Furthermore, each particle keeps track of the position with the best score it has visited so far, denoted by $\mathbf{p}_{best}$

Each particle is connected to other particles by a graph (Figure 41), and each particle knows the $\mathbf{p}_{best}$ of its neighbouring particles in this graph, which influences the particles defined by a set of equations defined below. In this way, different topologies of the graph cause different behaviours in the movement of the swarm of particles. A particle's *neighbourhood* is defined as the particle and the set of particles directly connected to it.

*The derivatives in the gradient vector are approximated by using the forward difference quotient, i.e. $\frac{df(x)}{dx} \approx \frac{f(x+\Delta x)-f(x)}{\Delta x}$.

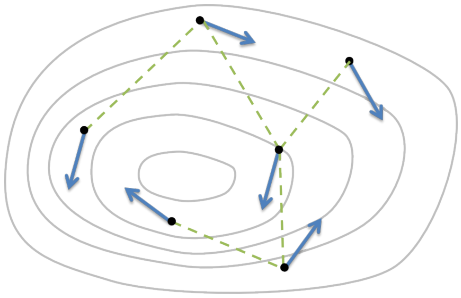


Figure 41: The particle swarm moves through the search space, shown as a contour plot of a 2D function. Each particle is connected to a number of other particles by a graph, indicated with the green dashed lines. The blue arrows indicate velocity vectors.

Many modifications to the original PSO algorithm have been proposed [28], but for simplicity the original algorithm is used. The PSO algorithm starts by assigning a random valid* position $\mathbf{p}$ to each particle, and a random vector to each particle velocity $\mathbf{v}$. Parameters do not all have the same order of magnitude, so in order to have the same searching behaviour for each parameter, the components of the random initial velocities should also be scaled appropriately.

Then, the algorithm loops until a stopping criterion is met. A fixed number of iterations is used in our localization tests. Each loop a number of steps is executed:

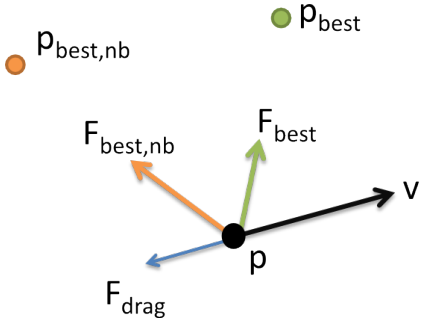


Figure 42: Each particle is pulled in different directions by different influences, which can be visualized as forces acting on it. $\mathbf{p}_{best}$ is the best location visited by the particle so far, and $\mathbf{p}_{best,nb}$ is the neighbourhood best position, which is the best position known to the particle $\mathbf{p}$ and its direct neighbours in the graph.

1. For each particle, evaluate the objective function (score $= L(\theta)$, with $\theta = \mathbf{p}$).
2. For each particle, if the score is better than the previous best score, replace $\mathbf{p}_{best}$ with the current $\mathbf{p}$.
3. For each particle, find the $\mathbf{p}_{best,nb}$, which is the best known position of both the particle and its direct neighbours in the graph.
4. Update the particle velocity using the 'forces' shown in Figure 42.

$$\mathbf{v}_{new} = \mathbf{v} + \mathbf{F}_{drag} + \mathbf{F}_{best,nb} + \mathbf{F}_{best} \tag{35}$$

*By *valid* position, a position is meant that lies within the physical constraints of the setup, for example the micromirror dimensions or a reasonable limit for the numerical aperture.

In this equation, $\mathbf{F}_{drag} = -(1-\omega)\mathbf{v}$, $\mathbf{F}_{best} = U(0,\phi_1)(\mathbf{p}-\mathbf{p}_{best})$ and $\mathbf{F}_{best,nb} = U(0,\phi_2)(\mathbf{p}-\mathbf{p}_{best,nb})$. $U(a,b)$ is the uniform random distribution between a and b. Every iteration new samples are generated from this distribution. ϕ_1, ϕ_2 are constants defining the influence of the best position and neighbourhood best position. ω is the *inertia weight*, that defines the fraction of velocity that remains from the previous value after every iteration.

5. Update the particle position by adding the velocity to the current position.

$$\mathbf{p}_{new} = \mathbf{p} + \mathbf{v}_{new} \tag{36}$$

In practice, the particle swarm will swirl around the known minima. Because not every particle knows the global minimum, local minima (that might be close to an even better global minimum) will also be explored further. Hence, the PSO is specifically useful for optimizing problem with multiple local minima.

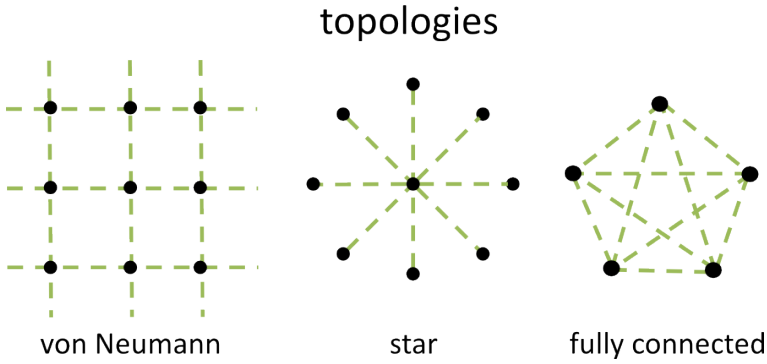


Figure 43: Different graph topologies for the particle swarm. This graph defines the neighbourhood of the particle, which is used to calculate $\mathbf{F}_{best,nb}$. The *von Neumann* topology is a 2D grid in which the edges wrap around to the opposite side (north connects to south, and east connects to west). The *star* topology has 1 center particle with all others connected to it, and the *fully connected* topology connects all to all.

According to Poli et al. [28], the graph topology the optimal graph topology for the PSO depends primarily on the search space. Many different topologies have been tried: static topologies where the graph is fixed, topologies that depend on the distance within the search space (for example a particle is connected if it has less than a certain Euclidean distance to another particle), and dynamic topologies that change during the optimization. Three static topologies are shown in Figure 43. One of the key conclusions of research on PSO graph topology [28] is that a more densely connected graph will focus more *exploitation* of the current minima, whereas a less densely connected graph will focus more on *exploration* of the search landscape. In this sense, it can be a good idea to have a sparsely connected graph for the micromirror calibration case and a densely connected graph for the tracking case. Among the static topologies, the von Neumann topology has one of the best overall performance [22]. We therefore use the von Neumann topology in the evaluation of the localization performance, to keep the implementation simple.

6 Algorithm evaluation on simulated measurements

As discussed in the localization chapter (5.3), we divide the optimization into 3 different phases. The performance of the different optimizers (fminsearch, gradient descent and the PSO) is tested separately for these phases, so we can see which optimizer is most suitable for each particular phase. Furthermore, we measure the accuracy on simulated data in the tracking phase, which is of interest for the eventual experimental use of the algorithm. The definition of accuracy and precision needs to be clarified if we discuss experimental situations. In simulation, the accuracy and the precision are the same quantity, because the sample image is based directly on the same optical model that is used to localize. However, in an experimental application, the two terms differ:

- Assuming that the physical state is the same, the localization *precision* relates to the variance of the estimated position, or the repeatability of the estimation.
- The localization *accuracy* relates to a constant bias on the estimated position. If the optical model does not represent the physics properly, it can result in a low accuracy measurement. In other words, it is incorrect, but reproducible (high precision).

Without comparison to experimental data, we cannot estimate the accuracy. However, we can estimate the localization precision for various situations, such as varying background noise or the number of emitted photons.

If not specified otherwise, every performance test described here uses the following parameters:

Parameter	**Value**	**Alignment**	**Tracking**
Position: $pos_{x,y,z}$	Anywhere in the mirror where $pos_z < 7$ μm	*	*
Background noise level: I_{bg}	5 photons/pixel	*	+
Emitter brightness (photons detected from emitter): I_0	10^5	*	+
Focal depth: f_z	4 μm	*	
V-groove apex position on X-axis: m_{apex}	0 μm	*	
V-groove depth: m_h	10 μm		
Mirror reflectivity: m_r	0.9		
Mirror left edge position: m_{left}	$-m_h/\tan(45°) = -10$ μm		
Mirror right edge position: m_{right}	$m_h/\tan(45°) = 10$ μm		
Numerical aperture: NA	1.2		
Pixel size:	$w_{pixel} = 14$ μm		
Refractive index of sample:	n_s 1.33		
Wavelength of light: λ	0.6 μm		
Magnification: M	150		

Table 2: The default parameter values are listed. The 2 right columns indicate for each phase which parameters are part of the search space. * indicates a parameter that is always part of the search space, + indicates a potential candidate parameter for the search space, if experimental conditions require it. For example, in live cell measurements the fluorophores such as GFP tend to have varying brightness, and the background level can fluctuate.

6.1 Performance in the alignment phase

6.1.1 Comparison of optimization methods for the alignment phase

The optimization runs produce a lot of data that can be shown in different ways. To demonstrate the ability to find the global maximum, we plotted for each optimizer, the final distance to the true position against the starting distance (distance between starting guess and true sampling position). This allows the reader to see whether the algorithm moved in the direction of the global minimum, or remained stuck in a local minimum.

A first attempt is done to compare the three optimization algorithms in their ability to converge to the global minimum. This is shown in Figure 44. The optimization algorithms were configured according to Table 3. We include the following list of parameters in the search space: $pos_x, pos_y, pos_z, f_z, m_{apex}, I_0, I_{bg}$, using random values around the defaults given in Table 2.

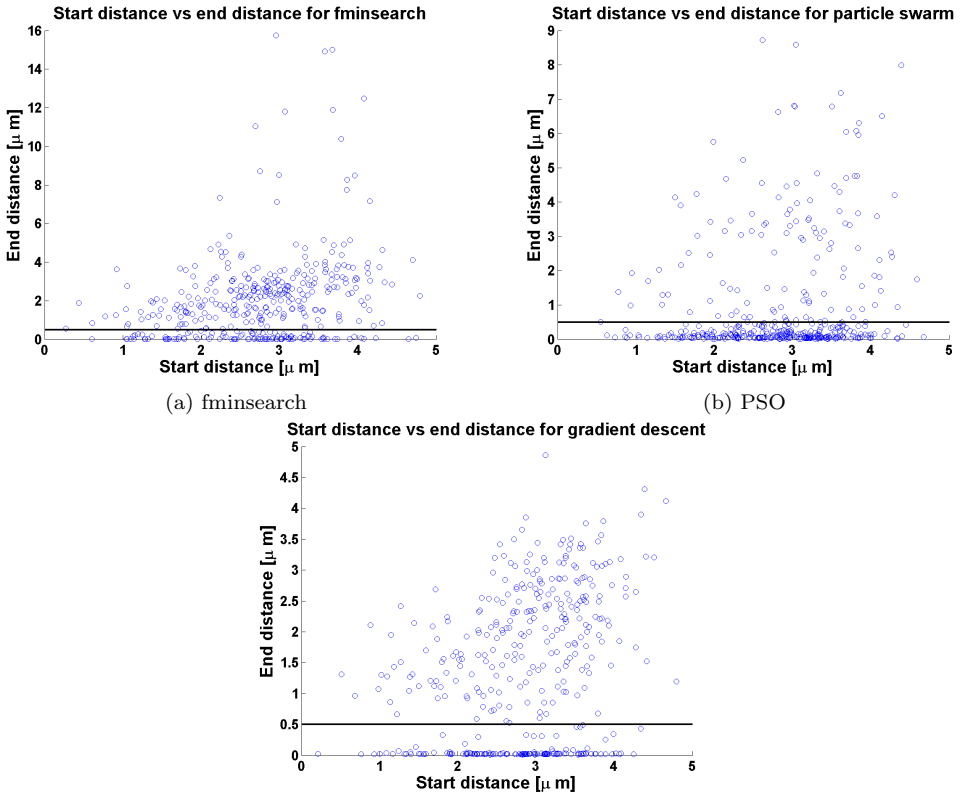
Method	**Settings**
fminsearch	MATLAB defaults: The algorithm stops iterations after 1400 function evaluations, or when loglikelihood changes less than 10^{-4} per iteration.
PSO	50 iterations with 25 particles, $\omega = 0.7$, $\phi_1 = 1$, $\phi_2 = 1.4$. Topology: von Neumann
gradient descent	Using a *stepsize* = 10nm, stopping when the likelihood has not increased for 20 iterations.

Table 3: If not otherwise specified, these settings are used for the optimization algorithms throughout this chapter.

It is clear that with this optimization configuration, many of the localization runs do not end near the global minimum. If we count all end distances (Euclidian) below a threshold of 0.5 micron as a successful run, we can compute a *success rate* for each method. In this way, we can compare the methods quantitatively, as shown in Table 4.

Optimizer	**Success rate**
fminsearch	0.28
PSO	0.68
gradient descent	0.35

Table 4: The success rate convergence of each optimization method on the align search space. Every run that ends in a Euclidian distance closer than 0.5 μm to the true position is counted as a run with successful convergence.



(a) fminsearch

(b) PSO

(c) gradient descent

Figure 44: The optimization method's ability to convergence can be compared by plotting the start distance versus the end point distance to the true position. All distances are Euclidian, computed from the $pos_{x,y,z}$ coordinates. The black line indicates the threshold value at an end distance of 0.5 μm. This threshold is used in computing the success rate (see Table 4).

Figure 45 shows the problematic convergence to the global minimum in a clear way, by using a dual log scale on the axes. It plots m_{apex} and f_z accuracy for localization on a very low noise image ($I_0 = 10^6$), ruling out the possibility of low accuracy due to image noise. The resulting image clearly shows two clusters, one in which the optimization remained in a local minima, and one where the global minima was reached and high accuracy was achieved.

As the PSO is better suited to optimization in a multi-modal search landscape, the succes rate of 65% of the PSO can be expected, compared to fminsearch and the gradient descent. However, an estimation method that succeeds 68% of the time is still not very useful, so it will need further adjustment. This difficulty in estimation can have three different root causes.

- The landscape can be complex, such as having many dimensions and a large number of local minima.
- It can be an optimization landscape difficulty (there could be some challenging locations in the landscape) such as various attractors to local minima that guide the optimization

away from the true optimum.

- It could also be the randomness of the optimization methods or the gradient approximation, in which case it could be interesting to have multiple runs on the same sample or multiple samples from the same parameters. In other words, the random starting point of the run influences the success of the estimation.

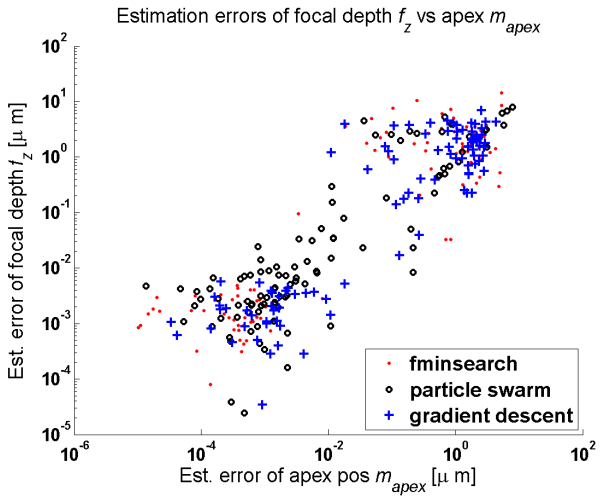


Figure 45: The estimation accuracy of the mirror apex position m_{apex} (a), and the focal depth f_z (b) are plotted against each other, for 400 runs of each different optimizer. To rule out the possibility of noisy measurements, the number of photons received by the emitter was set to $I_0 = 10^6$. It can be seen that the points are again clustered in two regimes, one where a reasonable estimation is done, and one where the estimation essentially fails to achieve a good estimate.

6.1.2 Increasing PSO particle count

The PSO is so far the most effective optimizer for the alignment phase, so its behaviour is further investigated. We will try improving the PSO performance, first by increasing the number of particles, and secondly, by running the PSO on the same sample multiple times, and selecting the estimate with the high likelihood from these runs.

Figure 46 shows the improved convergence to the global minima, when the PSO particle count is increased.

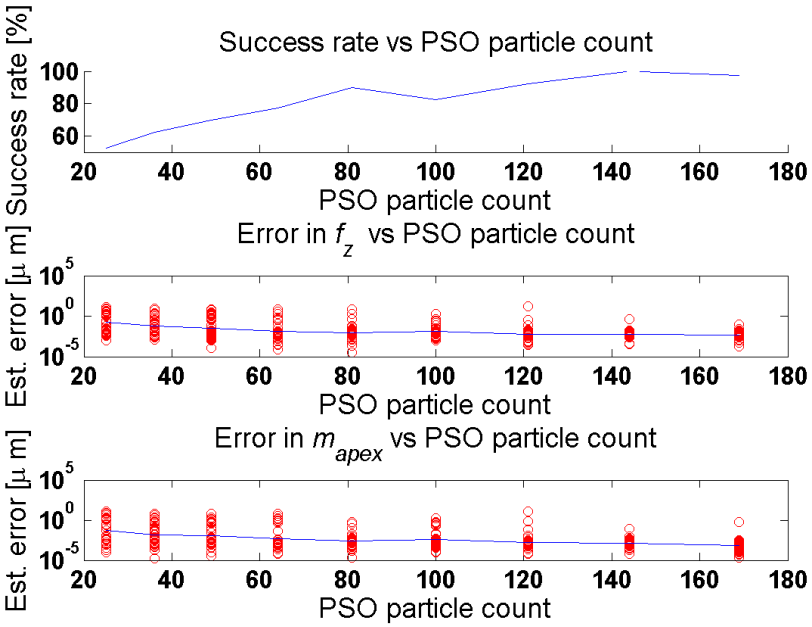
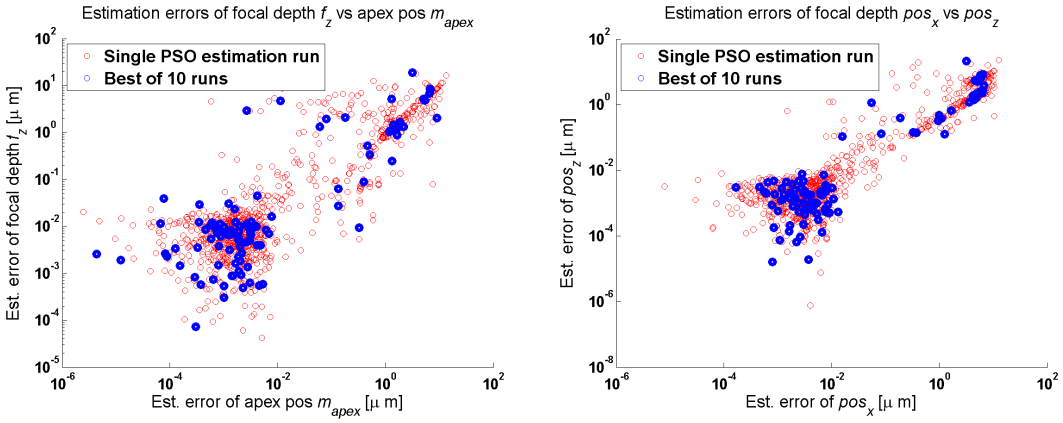


Figure 46: The top plot shows the alignment success rate plotted versus the PSO particle count. In this case, a success was defined as an estimated error of pos_x lower than 0.06 μm. Except for some minor random fluctuation, a steady increase of the success rate can be seen for higher particle counts, which is as expected. The middle and bottom plots show the average estimation error of f_z and m_{apex}, where the estimation error is defined as the absolute difference between the true and found values. The blue line indicates the logarithmic mean of the estimation errors.

6.1.3 Multiple runs per sample

To see whether an increase of iterations and particles would help the align phase estimation, another series of experiments was done, in which the estimation procedure was repeated 10 times for each of the sample images. The position with the highest likelihood was selected as the final estimate for that sample. A PSO with 36 particles and allowed to run for 150 iterations. The results are shown in Figure 47. We computed the success rate for all the estimations, and compared it with the success rate of the set of best-of-10 estimations, using 0.5 μm as a threshold value. Surprisingly, best-of-10 results in a success rate of 74%, and the complete set of runs had a success rate of 70%. This minor difference appeared to be random after running the test several times. This leads to the conclusion that selecting the best estimate out of multiple estimates does not contribute to the estimation performance. A possible explanation for this could be that the true parameters for which estimation failed were at particular difficult positions in the search landscape, for example a very small and deep minimum surrounded by many wider local minima. As a result, the probability of finding the true minimum is small enough for 10 estimation runs to fail at the task.



(a) (b)

Figure 47: The estimation errors of f_z vs m_{apex} are plotted on logarithmic scales (a). (b) shows the same estimation experiment, but estimation errors of pos_x and pos_z are used instead. Again, estimation error is defined as the absolute difference between the true and estimated parameter values. The blue circles indicate the best-of-10 estimates, determined by selecting the estimate with the highest likelihood value. A surprising result is that selecting the estimate with the best likelihood does not actually improve the success rate. This can also be seen in these graphs, as the blue circles are spread throughout the entire set of points, and do not have a higher concentration in the low error clusters.

6.1.4 Neighbourhood topologies in the PSO algorithm

As noted in section 5.3.3 the topology used in the PSO algorithm can have a large effect on the optimization efficiency. Ignoring generally inferior topologies such as fully connected or star formations, we compare 2 different topologies: von Neumann, and something we named *hub* formation, which is essentially a series of connected star formations intended to focus most on exploration of the search space. Figure 48 shows the success rates of the two topologies, compared to different PSO particle counts. The hub topology appears to be slightly superior compared to the von Neumann topology, but the difference are very small. It should also be noted that the effect of the topology is mainly a difference in exploration (finding different minima), versus exploitation (improving the current minima). This balance is also highly dependent on the inertia parameter ω in the PSO, so a different ω may cause the von Neumann topology to be superior.

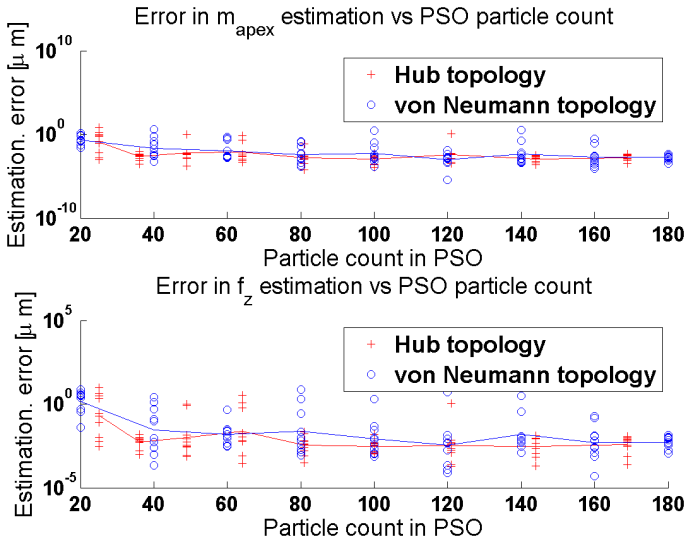
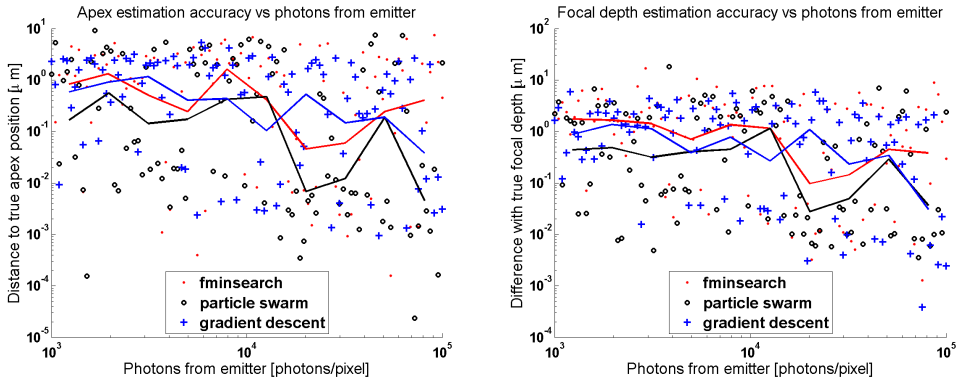


Figure 48: The estimation error (absolute difference) is plotted for a range of different PSO particle counts. The hub (red) topology appears to have a slightly lower error on average, although this might be caused by random sampling (each PSO particle count has 10 samples).

6.1.5 Effect of emitter brightness on align phase performance

The primary goal in the align phase is to estimate the focal depth f_z and the mirror X position m_{apex}. A range of different emitter photon count I_0 were used, but no correlation was found between the number of photons (which to some extend quantifies how much information is available for estimation), and the accuracy of the estimated parameters, as shown in Figure 49.



(a) (b)

Figure 49: The estimation accuracy of the mirror apex position m_{apex} (a), and the focal depth f_z (b) are plotted against the number of photons received from the emitter. A clear difference can be seen between the estimated positions that are the result of succesfull convergence to the global minimum, and the positions that are the result of failure to find the global minimum. As can be expected, the averages (indicated by the lines) separate these two regimes. Additionally, no significant difference can be seen between the 3 optimization algorithms.

6.1.6 Conclusion

The alignment phase optimization works well in simulation, but only with bright enough emitters (see section 6.1.5), and optimization It remains to be seen if this alignment approach can be applied on single frame measurements in experimental settings. If not, estimating an accurate mirror position and focal depth may still be possible by taking many frames into account (see appendix A).

6.2 Performance in the tracking phase

Tracking is the endpoint phase for the micromirror localization, to which the other phases build up to. We will calculate and compare several aspects of the tracking, for each of the three optimization algorithms. We will plot the precision for simulated measurements (the Euclidian distance between true and found position), for a variety of situations. The precision which gives a good indication of the localization precision obtainable on the experimental data. In all following performance tests of tracking, we only include the pos_x, pos_y and pos_z parameter, and assume that all other parameters are known. The optimizers are configured as followed: `fminsearch` is used with its default settings. Gradient descent is ran with a stepsize of $3nm$ until it has not found a better likelihood for 40 iterations in a row. The PSO is given 100 iterations to find the minimum using 9 particles.

6.2.1 Influence of photon count on precision

The number of photons required to do accurate localization will determine a large part of the usability of the micromirror method. Figure 50 shows a plot of the tracking precision on simulated data. The outliers visible in this plot show that even for very bright emitters, there is still a chance that the optimizer will get stuck on a local minima in the landscape. However, on average, the localization in the tracking phase is precise, resulting in single nanometer precision for photon counts larger than 10^5, which are typical for fluorescent bead tracking.

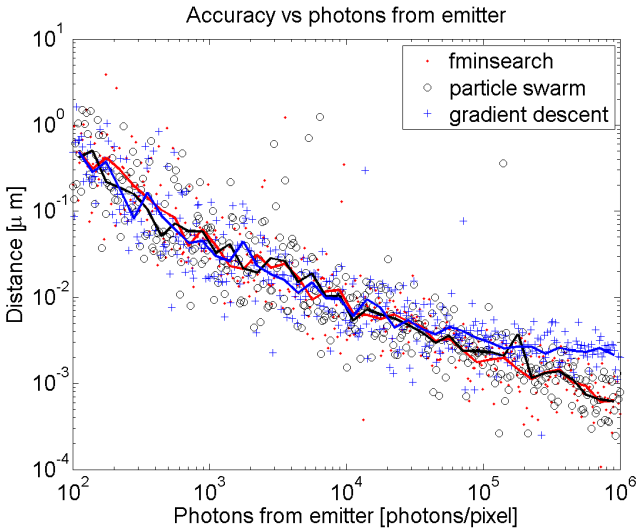
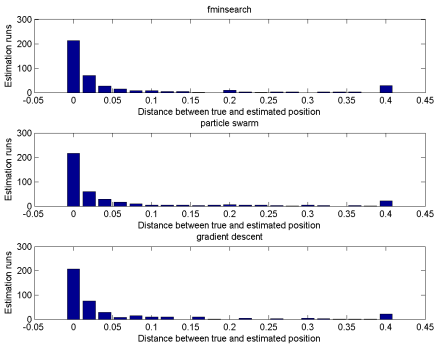


Figure 50: The tracking precision, measured as the distance to the true position (in the X, Y and Z axes), is plotted against the number of photons received from the emitter. The result is an expected exponential increase of the precision (or decrease of the distance). Some outliers are visible, even for high photon counts. It can also be noticed that the gradient descent stepsize is limited to 3 nm, as its precision stabilizes around that value when the other optimizers continue to increase in precision.



(a)

Figure 51: This plot shows all the localization runs from Figure 50 binned in a histogram, according to the distance of estimated position to the true position. No significant differences between the optimizers are found, which suggests that selecting a optimizer for tracking can be done based on other aspects, such as the processing speed.

6.2.2 Influence of background noise on precision

Experimental measurements, such from live cell imaging, have considerable amounts of background noise due to autofluorescence (fluorescence of unspecific cell molecules), especially relative to the brightness of typical live cell fluorophores such as GFP. For this reason, it is important to estimate the performance of the localization algorithm if used on noisy image data. The precision is estimated for a range of different background noise values I_{bg}, shown in Figure 52. Figure 53 shows the resulting precision. Although on average an reasonable precision is achieved (Approximately $40nm$ precision for low noise), there are still many outliers. Some could be caused by an unfortunate sampling of the simulation, others may be caused by a the localization algorithm getting stuck in a local minima on their way to the global minima, as they were in alignment phase. If local minima were a problem however, Figure 54 would likely show values that are further away than the estimation start position (such as the case for 44a).

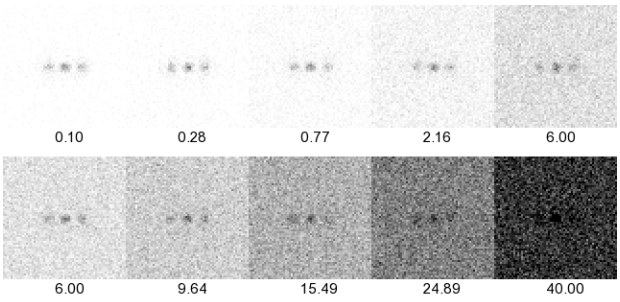


Figure 52: A range of samples generated with different values for the background noise I_{bg}, matching the range used to generate Figure 53. $I_0 = 1000$, which means that the camera receives 1000 photons from the emitter. It can be seen that at 40 photons/pixel, the emitter is barely discernible from the background noise.

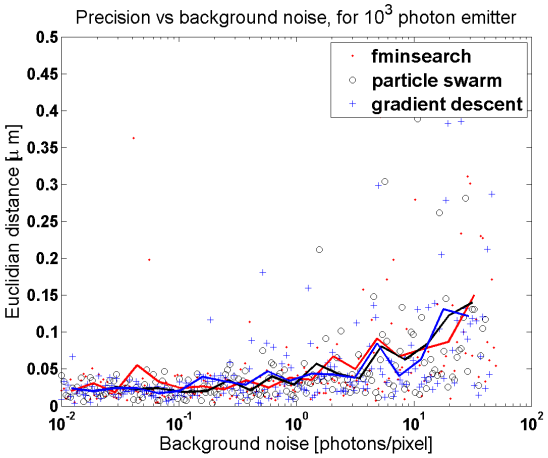


Figure 53: In this benchmark, the localization precision was estimated on a range of background noise values. All three optimizers were tested, but show no significant differences (as expected for the tracking phase).

Similarly as in the calibration phase, the distance between the estimated position and the true position was plotted against the starting distance to the true position, to check whether the optimizers hit a local minimum on the way to the global minimum.

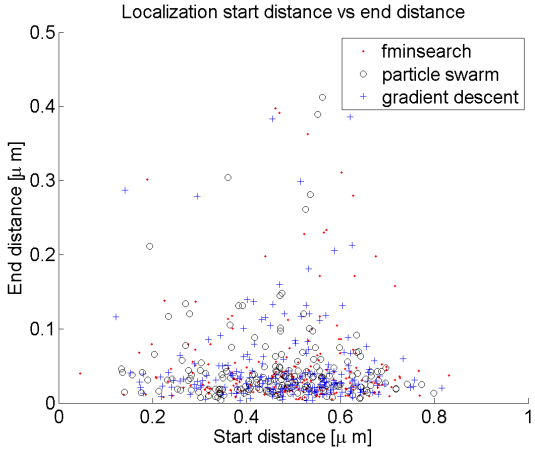


Figure 54: The localization results from Figure 53 shown as a scatter plot. It can be seen that there is no correlation between the starting distance and the endpoint distance to the true position, which suggests that the optimizers do not systematically get stuck on local minima in the tracking search space.

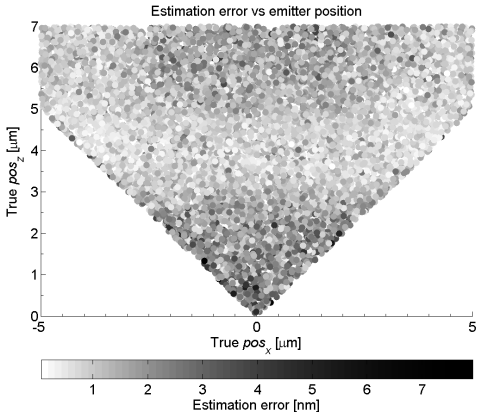


Figure 55: This plot shows the localization precision as a function of the emitter position in the micromirror. To generate this plot, emitter positions where randomly generated over a 5x5x5 μm space in the micromirror. These positions were used as parameter for simulating a measurement based on the optical simulation. The initial position for the PSO was randomly chosen in a 400nm box around the true position. One can see a faint band around $z = 4\mu$m, which corresponds with the focal plane at $z = 4$. It is interesting to see that the band of higher precision near the focal plane is also mirrored.

6.3 Influence of particle position in mirror on precision

Depending on the position of the particle, the image can look very different: In some cases reflections and direct images can be overlapping, and in other cases one of the spots might be significantly more out of focus than the other spots. To investigate the influence of these differences on localization precision, localization was tested on a large area of the micro mirror space. This is shown in Figure 56 (focal depth of 1) and Figure 55. The particle swarm optimizer was used, and the starting position was chosen less than 400nm to the true position. These choices ensured that the optimizer would not get stuck in a local minima. The most straightforward explanation of patterns seen in these two plots, is that as a spot (direct or reflection) gets closer to the focal plane, the accuracy increases. It follows that optimal accuracy of the micromirror is achieved when at least one of the spots in located at the focal plane.

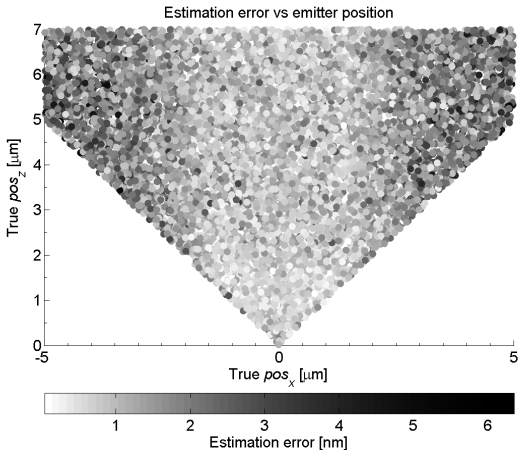


Figure 56: Localization precision plot similar to Figure 55, but with the focal plane set to $z = 1$. The band of higher accuracy is again visible, and just like for $z = 4$, it is reflected in the mirror.

7 Conclusion and discussion

7.1 Discussion

Although a theoretical proof of concept, the current model is simplified in various ways, making it currently less effective for direct application on experimental data:

- Currently, the brightness of each of the three spots cannot change relative to each other. In practice, the spot brightness is probably determined by how much of the light is blocked by the mirror for each spot, a combination of properties of the objective (i.e., defining the light transmittance through the objective for a particular angle), and ofcourse the reflectivity of the mirror itself. Currently, we only include a constant reflectivity factor in the model.
- Accurately computing the PSF will require taking into account the optical path differences within the sample. That is, the light path from an out-of-focus light source through the medium and coverslip causes a phase difference that should be taken into account, using methods such as described by Gibson and Lanni [9]. Currently, the refractive indices of the immersion fluid of the objective and that of the sample medium (n_s) are assumed to be equal, which is not likely in an experimental setting.
- As mentioned in the optical model description, it is not clear whether there should be interference between the reflected light, and the light coming from the emitter directly. Does it only happen when the emitter is close to a mirror surface, and if so, at which distance would that become noticable?
- The work by Hajjoul et al. [12] on single-facet micromirrors (see section 3.2.1) is primarily an effort to reduce the effect of secondary and tertiary reflections (light of the reflection being reflected again etc..) manifesting as background noise. Our model only includes light being reflected once.

On the subject of numerical optimization we have only tried a very limited subset of possible implementations. The particle swarm optimization for example, can be tuned and optimized in many various ways: different topologies may work better, or different rules that tune convergence and divergence of the swarm (so the search landscape is explored more effectively).

In any live cell measurement, there will be fluorescent molecules at close proximity to each other (within the diffraction limited range). In conventional 2D tracking, separating these already is a difficult problem. In fact, separation and tracking the fluorophores is a research area on its own. However, in the case of micromirror this problem could be worsened, as emitters can never be completely in focus (in a micromirror, either a reflection is out-of-focus, or the direct spot is out-of-focus).

7.2 Conclusion

In this thesis, we have described why the use of micromirrors has potential for 3D localization in fluorescence microscopy, compared to other 3D localization methods, as it allows one to capture more photons from a fluorescent molecule, while simultaneously supporting the estimation of the Z position without sacrificing accuracy on the X and Y axes (chapter 3). An optical model was developed to analyse the optical diffraction effects (such as an asymmetric aperture) caused by the micromirror geometry (chapter 4).

We have developed a 3D localization method, intended to process microscopy images from a micromirror experiment. By testing our localization method on simulated data, we have demonstrated that the optical effects that distort the PSF in micromirror imaging can be accounted for by fitting an appropriate optical model to the image data, using various optimization methods. In this way, our localization method is able to correctly deal with the defocussed and perturbed PSF of spots in a micromirror.

The theoretical accuracy of micromirror localization was estimated for a variety of conditions, such as different emitter brightness or different amounts of background noise (chapter 6.2). The resulting localization error decreases as expected when increasing the emitter brightness, and we achieve single nanometer resolution when 10^5 photons are received from the emitter.

In order to work, the localization method requires accurate knowledge of the imaging parameters (such as focal depth), and micromirror properties (such as the location of the micromirror). Estimating these parameters from an image, termed the *alignment* phase in this thesis, is shown to be very computationally intensive, as the optimization landscape that arises from this set of parameters has various local minima. We successfully applied a particle swarm optimization algorithm to this problem, which is able to find the local minima when given enough iterations and particles.

On the software development side, we have implemented the simulation of the optical model as a C++ MATLAB plugin that uses fast graphics hardware. As such, this implementation is a good framework to build upon, such as implementing other Fourier-optical estimation methods.

8 Outlook

Now that our micromirror localization method has been shown to work in theory, we can look at what it can be used for, and how this would have to be accomplished. For direct application to biophysical research, two applications come to mind:

- We can apply the method to typical biophysical in-vitro research where the mechanics of proteins and other molecules are studied, for example by attaching them to DNA strands. These molecules can be attached to fluorescent beads by strands of DNA/RNA, and the micromirror localization method would allow following them in 3D. This application is expected to be fairly straightforward: the concentration of fluorescent beads in these experiments is limited, so overlapping of the PSF's between different beads can be minimized. Furthermore, the brightness of fluorescent beads will help a lot in achieving a useful localization accuracy.
- A potentially more challenging application is the use of micromirror localization in in-vivo experiments, where the concentration of fluorophores such as GFP molecules cannot be directly controlled. As a result, a cell most often has many fluorophores in its system, which are likely to overlap (considering a cell size of 1 micron for *E. coli*, or several microns for a eukaryote). The micromirror localization algorithm will have to be modified to seperate the emitters signals from eachother, which is outlined in the recommendations section 9.0.3.

The micromirror localization algorithm pioneers the idea that a Fourier-optical model can be used efficiently during localization to estimate emitter positions. While Fourier-optical estimation is presumably overkill for some typical localization problems, such as 2D tracking of on-focus fluorophores without any micromirrors, it has potential to be combined with other methods for 3D localization. For example, the ability to use an accurate aberrration model in the Fourier-optical method could be used to track out-of-focus emitters with unprecedented accuracy. Another idea is to combine it with other existing 3D localization methods, for example the double-helix 3D localization[30], which uses a spatial light modulator to shape the PSF in a way that it depends on Z (see Figure 14). In their method, a least-squares Gaussian fit was applied to extract the Z position, but one can imagine that with fast Fourier-optical estimation in mind, different considerations can lead to a higher accuracy localization.

Concluding, we see a variety of opportunities where the micromirror localization, or the use of the Fourier-optical estimation framework, can make a useful contribution.

9 Recommendations for future work

9.0.1 Measuring aberrations

The theoretical study of the Center-Of-Mass localization errors in a micromirror, as done by Berglund et al. [4], involves an aberration model by Gibson and Lanni [9]. In this model, a phase difference is computed for each light wave direction from the aberration model, which is characterized by a set of aberration coefficients. However, instead of modelling the aberration with a model that contains just a few parameters (and is therefore likely to be underfitted), it would be better to measure the aberration directly. For this purpose, a class of algorithms exist known as *phase retrieval* algorithms, that estimate the complex valued wavefield from sets of microscopy images. These methods, such as described in Pankajakshan et al. [27] and Fienup [8] can be used to measure phase differences for each spatial frequency, for example those caused by aberration in the microscope. This could be applied to measurements of fluorescent bead specimens (without any micromirrors involved) to accurately characterize the aberration as an image, instead of just a few constants. Once this is done, we can use the measured phase difference image and apply it to the Fourier-optics simulation. As well as measuring the aberration, it can also be used to confirm that the model matches reality. For example, if the transmitted spatial frequencies can be measured accurately, it could be possible to rule out effects such as diffraction around the micromirror edges (see Figure 27).

9.0.2 Improving the speed of the algorithm

The efficiency or speed of a computational method is often given a low priority in development. For good reason, as a fast method that does not work properly is worth less than a method that is slow, but correct. However, if we look at the practical use of this localization method, a fast calibration and alignment procedure would benefit the user a lot, as it make it possible to test a wider array of experimental settings without being a burden on the researchers time. Furthermore, if we achieve a reduction in the running time of the algorithm, this can usually be spend somewhere else on achieving a higher accuracy. For example, by increasing the resolution of the optical simulation.

We can think of several ways to significantly speed up the localization algorithm:

Using a Gaussian PSF model to speed up the first iterations of calibration and alignment In the alignment phase, a first guess of the parameters can be done using a Gaussian PSF model, instead of the Fourier-optical model. The Gaussian model should be able to approximate out-of-focus spots. The 3D localization method by Toprak et al. [36] defines an example of a such a Gaussian model to fit an out-of-focus spot, by adding a ring shape to the PSF, which radius depends on the Z coordinate. After convergence of the method using the Gaussian model, the Fourier-optical model can be used to further refine the parameters. Additionally, we could precompute a map of the errors made by a localizer with a Gaussian PSF, similar to the map of COM errors done by Berglund et al. [4]. If, for every position estimated using the Gaussian PSF model, an error vector is known that brings us to the correct location in the micromirror, the estimated might already be accurate enough for certain applications.

Using a 3D lookup table to store the spot PSF images A large speed gain could be achieved if we can precompute the PSF in some way, in order to avoid discrete Fourier transforms at every evaluation of the optimization function. If we assume that no aberration is present in the system, the PSF images of single spots can be generated in advance, and directly accessed from a lookup table during tracking. This can be realized by noting that there are only 3 variables that really determine the shape of the PSF of a single spot (reflected or direct). These variables are **1,2)** the frequencies at which the mirror blocks (left and right side of mirror, named a, and b as shown in Figure 57), and **4)** the distance to the focal plane of the spot. Given that we have such a lookup table, what would the procedure look like? We can compute the simulated image $I(x, y)$ by looking up the appropriate PSF shapes for the 2 reflected spots and the direct spot, shifting these according to their position and scaling their pixel values according to the brightness of the emitter. As mentioned above, the lookup table would have 3D indices: the a and b coordinates, and the distance to the focal plane dz. Suppose we allow 64 * 64 * 64 possible positions in the lookup table (a, b and dz could each have 64 different values). If each of those positions have a 64x64 pixel PSF image of 16 bits per pixel, we can have a lookup table of 2 GB, which can be located entirely in video memory and accessed with very high speed. The speed gain of such a method could easily be several orders of magnitude.

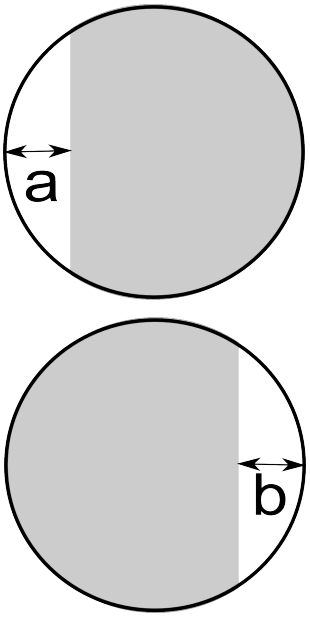


Figure 57: We can define aperture cut-off coordinates a and b (in the range $[0 - 1]$). These coordinates define at which spatial frequencies the micromirror blocks the light. a and b could be computed from the distance to the micromirror edges (see s_x from chapter 4.3.5).

9.0.3 Separating spots from different emitters

Effective separating the signals from different emitters will be important in the practical application of micromirror localization. Ideally, one would know for all detected photons, which photon is emitted by which emitter. If that is known, then we can simply generate a new measurement image for each separate emitter, and run the localization algorithm on this image.

One idea to approach this, would be to implement an Expectation-Maximization algorithm (EM). A typical example of an EM algorithm is the fitting of a set of Gaussian distributions to a dataset (this problem is known as a *mixture-of-gaussians*, or MoG). The algorithm will start with a first guess set of Gaussian distributions, then compute for each point in the dataset which Gaussian distribution it most likely belongs to. Secondly, this knowledge of membership for each point is used to compute new Gaussian distributions, and these steps are repeated for a number of iterations. Now in order to apply such a method to the spot separation problem, we would replace the Gaussian distribution with the distribution computed by the simulation ($I(x, y)$) and iterate until convergence, when the spot positions are no longer changing.

10 Acknowledgements

Working on my master research project at Bionanoscience was a great and fun experience! I learned many new things, both science itself and how to approach it. I tend to be stubborn and do everything myself, but the difficulties that arose from this (such as learning the Fourier optics) really learned me a lesson, of how important and useful it is to work together with others. I made many new friends at BN, and found the environment very stimulating to work in. First of all, I would like to thank David Dulin for allowing me to work on his project, and for always being a positive force in the lab. He gave me a lot of freedom on the project, but simultaneously helped me focus on things. This was very inspiring, I could not have found a better daily supervisor! I'd like to thank Zhuangxiong Huang for starting the micromirror project, together with David. I'd like to thank Nynke Dekker and Dick de Ridder for pushing me to get the best results, without their effective critiques I would never have learned to actually write a readable scientific document. Thanks for guiding my through my master project, and spending a lot of time on all my draft versions! Furthermore, I would like to thank Jacob Kerssemakers, for having a lot of inspiring discussions with me especially in the first few months.

Finally, thanks to the whole group of BN! I had a lot of fun playing foosball, talking science, and generally making friends here, and it was a great time!

A Multiple-image calibration and alignment

Estimating all parameters in the calibration- and align phase from a single image can be problematic, as the image only contains a limited amount of information. If the resulting estimated parameters are less accurate that those that a user could estimate by manual calibration, then there is no benefit in doing automatic calibration. This section proposes a potential solution to this problem, by defining a single estimation problem on a whole series of images, instead of just one. The problem of simultaneously localizing emitters of on a series of images (frames), is shown in Figure 58. For each of the N frames, parameters are added for the emitter position ($pos_{x,1..N}$, $pos_{y,1..N}$ and $pos_{z,1..N}$). The alignment parameters and calibration parameters are assumed identical for each frame.

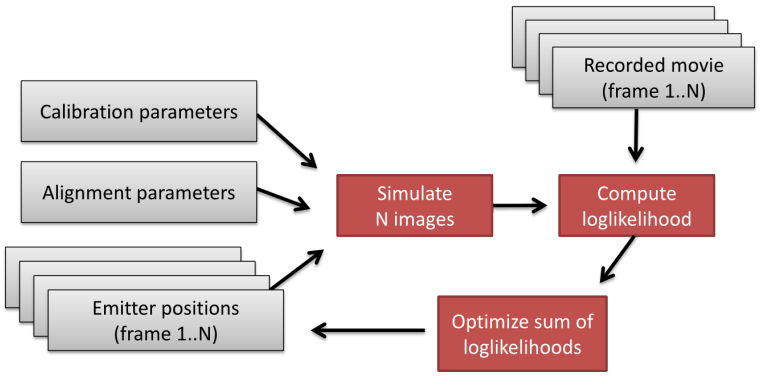


Figure 58: The calibration and alignment parameters can be estimated more accurately, if the optimization is done using several frames at the same time. It is then assumed that the alignment and calibration stays the same. Because the emitter can move between frames, each of the frames will have a seperate set of parameters that describe the emitter position.

Referring back to the localization chapter 5, we define a new optimization problem in which the objective is to minimize the error sum for all the frames. We split the parameter set θ into a set θ' that is shared between frames, and a $\theta_{1..N}$ that indicates the position of each emitter per-frame ($\theta_k = \{pos_{x,k}, pos_{y,k}, pos_{z,k}\}$).

$$\theta_{ML} = \underset{\{\theta', \theta_{1..N}\}}{\arg\min} \sum_{k=1}^{N} L(\theta', \theta_k) \tag{37}$$

L is the sum of loglikelihood for each pixel over the measured image, defined in 5.1.1.

B Implementation

The computational complexity of our optical model required a fast implementation, as it needs to perform several discrete Fourier transform in every iteration of the optimization algorithm. For this reason, the optical simulation was implemented using CUDA, which is a C++ language extension that allows the developer to define methods that run on the graphics hardware itself. CUDA also provides a Fast-Fourier-Transform implementation, called CUFFT [**?**]. CUFFT takes support both double and single precision FFT operations on complex valued images. Due to the internal implementation of the FFT algorithm, CUFFT only accepts images with sizes that can be factored as $2^a * 3^b * 5^c * 7^d$ (with $a, b, c, d \in \mathbb{N}$). Because sizes that are multiples of 2 have the fastest implementation, we have restricted the output and input image sizes for the localizer to squared images, such as 256x256 or 512x512.

B.1 Software package

An opensource software package has been developed, called MIMIL (Short for MIcroMIrror Localization). The source code is freely available, and written in C++ ,CUDA 4.2 and MATLAB. It requires a CUDA capable GPU. It has been build and tested on Windows XP and 7, 32 bit running MATLAB versions ranging from 2007 to 2011. Interaction with the software can be done either by directly linking the library into a C++ application, or by making use of the MATLAB MEX plugin, demonstrated below.

Code and binaries can be found at
`http://code.google.com/p/micromirror-tracker/`

B.2 MATLAB Plugin

Although the simulation code is written in C++ and CUDA, the optimization algorithms such as the particle swarm optimization or the gradient descent have primarily been developed within the MATLAB environment. To do this, a MATLAB plugin library was created in order to interface with the C++ and CUDA code of the simulator. A simple example of using the MIMIL plugin is shown below:

```
function mimil_example()
    m = mimil(); % load the plugin and create interface object
    m.init(512); % initialize the simulator to a buffer size of 512x512 pixels
    pos = m.defaultparam; % set the parameter vector to the default set
    % the pos vector be converted to a structure with informative field names
    pos_struct = m.paramstruct(pos)
    pos_struct.x = 2;
    pos_struct.z = 4; % set emitter Z position to 4
% run the simulator, producing an image based on the default parameters in pos
    img = m.psf(pos_struct);
    img = normalize(img);  % make sure the image values range from 0 to 1
    imshow(img.^0.5);   % adjust the contrast and display the image
    imwrite(1-img.^0.5, 'mimil_example.png');
end
function n=normalize(u)
    a=max(max(u));
    b=min(min(u));
    n=(u-b)/(a-b);
end
```

The output of this code is:

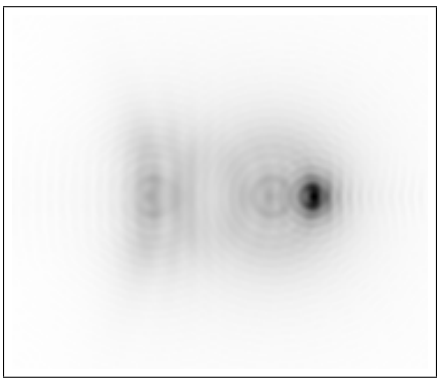


Figure 59: The output image of the code shown above. The image was cropped, and contrast is adjusted for better visualization.

```
Using mimil version: 0.2 - release - sp
CUDA device: GeForce 9800 GTX/9800 GTX+, processors: 16,
    global memory: 512 MB, max total threads: 12288

pos_struct =

         x: 0
         y: 0
         z: 2
        fd: 1
       sig: 1000
        bg: 0.1000
        NA: 1.2000
         M: 150
        ps: 14
    lambda: 0.6000
        ns: 1.3300
      refl: 0.9000
       apx: 0
```

References

[1] J. Bartl and M. Baranek. Emissivity of aluminium and its importance for radiometric measurement. *Science*, 4(2):31–36, 2004. URL `http://www.measurement.sk/2004/S3/p3.html`.

[2] M. Bates, B. Huang, G. T. Dempsey, and X. Zhuang. Multicolor super-resolution imaging with photo-switchable fluorescent probes. *Science (New York, N.Y.)*, 317(5845):1749–53, Sept. 2007. ISSN 1095-9203. doi: 10.1126/science.1146598. URL `http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=2633025&tool=pmcentrez&rendertype=abstract`.

[3] A. J. Berglund, M. D. McMahon, J. J. McClelland, and J. A. Liddle. Fast, bias-free algorithm for tracking single particles with variable size and shape. *Optics express*, 16(18):14064–75, Sept. 2008. ISSN 1094-4087. URL `http://www.ncbi.nlm.nih.gov/pubmed/18773017`.

[4] A. J. Berglund, M. D. McMahon, J. J. McClelland, and J. A. Liddle. Theoretical model of errors in micromirror-based three-dimensional particle tracking. *Optics Letters*, 35(11): 1905–1907, 2010. doi: 10.1364/OL.35.001905.

[5] E. Betzig, G. H. Patterson, R. Sougrat, O. W. Lindwasser, S. Olenych, J. S. Bonifacino, M. W. Davidson, J. Lippincott-Schwartz, and H. F. Hess. Imaging intracellular fluorescent proteins at nanometer resolution. *Science (New York, N.Y.)*, 313(5793):1642–5, Sept. 2006. ISSN 1095-9203. doi: 10.1126/science.1127344. URL `http://www.ncbi.nlm.nih.gov/pubmed/16902090`.

[6] D. T. Burnette, P. Sengupta, Y. Dai, J. Lippincott-Schwartz, and B. Kachar. Bleaching/blinking assisted localization microscopy for superresolution imaging using standard fluorescent molecules. *Proceedings of the National Academy of Sciences of the United States of America*, 108(52):21081–6, Dec. 2011. ISSN 1091-6490. doi: 10.1073/pnas.1117430109. URL `http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=3248526&tool=pmcentrez&rendertype=abstract`.

[7] M. K. Cheezum, W. F. Walker, and W. H. Guilford. Quantitative comparison of algorithms for tracking single fluorescent particles. *Biophysical Journal*, 81(4):2378–2388, 2001. URL `http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=1301708&tool=pmcentrez&rendertype=abstract`.

[8] J. R. Fienup. Phase retrieval for undersampled broadband images. *Journal of the Optical Society of America A*, 16(7):1831, 1999. ISSN 10847529. doi: 10.1364/JOSAA.16.001831. URL `http://www.opticsinfobase.org/abstract.cfm?URI=JOSAA-16-7-1831`.

[9] S. F. Gibson and F. Lanni. Experimental test of an analytical model of aberration in an oil-immersion objective lens used in three-dimensional light microscopy. *Journal of the Optical Society of America A: Optic*, 8(10):1601–1613, 1991. URL `http://www.opticsinfobase.org/abstract.cfm?URI=josaa-8-10-1601`.

[10] J. W. Goodman. *Introduction to Fourier Optics*, volume Second. McGraw-Hill, 1996. ISBN 0071142576. URL `http://books.google.com/books?id=ow5xs_Rtt9AC&pgis=1`.

[11] H. Hajjoul, S. Kocanova, I. Lassadi, K. Bystricky, and A. Bancaud. Lab-on-Chip for fast 3D particle tracking in living cells. *Lab on a Chip*, 9(21):3054–3058, 2009. URL `http://www.ncbi.nlm.nih.gov/pubmed/19823719`.

[12] H. Hajjoul, J. Mathon, Y. Viero, and A. Bancaud. Optimized micromirrors for three-dimensional single-particle tracking in living cells. *Applied Physics Letters*, 98(24):243701, 2011. ISSN 00036951. doi: 10.1063/1.3599586. URL http://link.aip.org/link/APPLAB/v98/i24/p243701/s1&Agg=doi.

[13] E. Hecht. *Optics (4th Edition)*, volume 1. Addison Wesley, 2001. ISBN 0805385665. URL http://www.amazon.ca/exec/obidos/redirect?tag=citeulike09-20&path=ASIN/0805385665.

[14] S. W. Hell, R. Schmidt, and A. Egner. Diffraction-unlimited three-dimensional optical nanoscopy with opposing lenses. *Nature Photonics*, 3(7):381–387, July 2009. ISSN 1749-4885. doi: 10.1038/nphoton.2009.112. URL http://www.nature.com/doifinder/10.1038/nphoton.2009.112.

[15] L. Holtzer, T. Meckel, and T. Schmidt. Nanometric three-dimensional tracking of individual quantum dots in cells. *Applied Physics Letters*, 90(5):053902, 2007. ISSN 00036951. doi: 10.1063/1.2437066. URL http://link.aip.org/link/APPLAB/v90/i5/p053902/s1&Agg=doi.

[16] B. Huang, W. Wang, M. Bates, and X. Zhuang. Three-dimensional super-resolution imaging by stochastic optical reconstruction microscopy. *Science*, 319(5864):810–813, 2008. URL http://www.ncbi.nlm.nih.gov/pubmed/18174397.

[17] S. M. Kay. Fundamentals of Statistical Signal Processing: Estimation Theory. *Technometrics*, 37(4):465, 1993. ISSN 00401706. doi: 10.2307/1269750. URL http://www.jstor.org/stable/1269750?origin=crossref.

[18] J. Kennedy and R. Eberhart. Particle swarm optimization. In *Neural Networks, 1995. Proceedings., IEEE International Conference on*, volume 4, pages 1942 –1948 vol.4, nov/dec 1995. doi: 10.1109/ICNN.1995.488968.

[19] J. C. Lagarias, J. A. Reeds, M. H. Wright, and P. E. Wright. Convergence Properties of the Nelder-Mead Simplex Method in Low Dimensions. *SIAM Journal on Optimization*, 9 (1):112–147, 1998. ISSN 10526234. doi: 10.1137/S1052623496303470. URL http://link.aip.org/link/SJOPE8/v9/i1/p112/s1&Agg=doi.

[20] B. R. Masters. The Development of Fluorescence Microscopy. *Life Sciences*, pages 1–9, 2001. doi: 10.1002/9780470015902.a0022093. URL http://dx.doi.org/10.1002/9780470015902.a0022093.

[21] M. D. McMahon, A. J. Berglund, P. Carmichael, J. J. McClelland, and J. A. Liddle. 3D particle trajectories observed by orthogonal tracking microscopy. *ACS nano*, 3(3):609–614, 2009. URL http://www.ncbi.nlm.nih.gov/pubmed/19309171.

[22] R. Mendes. *Population Topologies and Their Influence in Particle Swarm Performance*. PhD thesis, Universidade de Minho, 2004. URL http://www.di.uminho.pt/~rcm/publications/RuiPhD.pdf.

[23] K. I. Mortensen, L. S. Churchman, J. A. Spudich, and H. Flyvbjerg. Optimized localization analysis for single-molecule tracking and super-resolution microscopy. *Nature Methods*, 7 (5):377–381, 2010. URL http://www.ncbi.nlm.nih.gov/pubmed/20364147.

[24] J. A. Nelder and R. Mead. A simplex method for function minimization. *The Computer Journal*, 7(4):308–313, 1965. ISSN 00104620. doi: 10.1093/comjnl/7.4.308. URL http://comjnl.oxfordjournals.org/cgi/content/abstract/7/4/308.

[25] J. Nocedal and S. Wright. *Numerical Optimization*. Springer Series in Operations Research. Springer, 1999. ISBN 9780387987934. URL http://books.google.nl/books?id=epc5fX0lqRIC.

[26] R. J. Ober, S. Ram, and E. S. Ward. Localization Accuracy in Single-Molecule Microscopy. *Biophysical Journal*, 86(2):1185–1200, 2004. URL http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=1303911&tool=pmcentrez&rendertype=abstract.

[27] P. Pankajakshan, L. Blanc-Feraud, Z. Kam, and J. Zerubia. Point-Spread Function retrieval for fluorescence microscopy. *2009 IEEE International Symposium on Biomedical Imaging From Nano to Macro*, pages 1095–1098, 2009. doi: 10.1109/ISBI.2009.5193247. URL http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=5193247.

[28] R. Poli, J. Kennedy, and T. Blackwell. Particle swarm optimization An overview. *Information Processing Letters*, 1(1):33–57, 2007. doi: 10.1007/s11721-007-0002-0. URL http://www.sciencedirect.com/science/article/B6V0F-4PMT31J-7/2/298057b92c348dc943012fe32d1a89bd.

[29] S. Quirin, S. R. P. Pavani, and R. Piestun. Optimal 3D single-molecule localization for superresolution microscopy with aberrations and engineered point spread functions. *Proceedings of the National Academy of Sciences of the United States of America*, 109(3):675–9, Jan. 2012. ISSN 1091-6490. doi: 10.1073/pnas.1109011108. URL http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=3271897&tool=pmcentrez&rendertype=abstract.

[30] S. Rama, P. Pavani, M. A. Thompson, J. S. Biteen, S. J. Lord, N. Liu, R. J. Twieg, R. Piestun, and W. E. Moerner. Three-dimensional, single-molecule fluorescence imaging beyond the diffraction limit by using a double-helix point spread function. *PNAS*, 106(9), 2009.

[31] C. S. Smith, N. Joseph, B. Rieger, and K. a. Lidke. Fast, single-molecule localization that achieves theoretically minimum uncertainty. *Nature methods*, 7(5), Apr. 2010. ISSN 1548-7105. doi: 10.1038/nmeth.1449. URL http://www.ncbi.nlm.nih.gov/pubmed/20364146.

[32] S. Stallinga and B. Rieger. Accuracy of the gaussian point spread function model in 2D localization microscopy. *Optics Express*, 18(24):24461–24476, 2010. doi: 10.1364/OE.18.024461. URL http://www.ncbi.nlm.nih.gov/pubmed/21164793.

[33] Y. Sun, J. D. McKenna, J. M. Murray, E. M. Ostap, and Y. E. Goldman. Parallax: high accuracy three-dimensional single molecule tracking using split images. *Nano letters*, 9(7):2676–82, July 2009. ISSN 1530-6992. doi: 10.1021/nl901129j. URL http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=2728077&tool=pmcentrez&rendertype=abstract.

[34] J. Tang, J. Akerboom, A. Vaziri, L. L. Looger, and C. V. Shank. Near-isotropic 3D optical nanoscopy with photon-limited chromophores. *PNAS*, 2010(22):1–6, 2010. ISSN 10916490. doi: 10.1073/pnas.1004899107/-/DCSupplemental.www.pnas.org/cgi/. URL http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=2890476&tool=pmcentrez&rendertype=abstract.

[35] R. E. Thompson, D. R. Larson, and W. W. Webb. Precise nanometer localization analysis for individual fluorescent probes. *Biophysical Journal*, 82 (5):2775–2783, 2002. ISSN 00063495. doi: 10.1016/S0006-3495(02)75618-X. URL http://apps.isiknowledge.com/full_record.do?product=UA&search_mode=GeneralSearch&qid=35&SID=R1j9HLn2I7PPFO6GmIc&page=1&doc=1&colname=WOS.

[36] E. Toprak, H. Balci, B. H. Blehm, and P. R. Selvin. Three-dimensional particle tracking via bifocal imaging. *Nano letters*, 7(7):2043–5, July 2007. ISSN 1530-6984. doi: 10.1021/nl0709120. URL http://www.ncbi.nlm.nih.gov/pubmed/17583964.

[37] R. Y. Tsien. The green fluorescent protein. *Annual Review of Biochemistry*, 67(1):509–544, 1998. URL http://www.ncbi.nlm.nih.gov/pubmed/9759496.

[38] D. Voelz. *Computational Fourier Optics: A MATLAB Tutorial - Voelz — Publications: SPIE*. SPIE Press, 2011. URL http://spie.org/x648.html?product_id=858456.

Web links

[39] http://seasweetie.files.wordpress.com/2011/08/cute-kittenweee.jpg. [Online; accessed 20-3-2012].

[40] AMMRF. http://www.ammrf.org.au/myscope/sem/background/. [Online; accessed 2-4-2012].

[41] Andor. Andor Technical Note - Count Convert - Quantifying Data in Electrons and Photons. http://www.andor.com/pdfs/learning/ixon/Andor%20iXon%20-%20Count%20Convert.pdf, . [Online; accessed 3-5-2012].

[42] Andor. Andor Technical Note - Making Sense of Sensitivity. http://www.andor.com/pdfs/learning/ixon/Andor%20iXon%20-%20Sensitivity.pdf, . [Online; accessed 3-5-2012].

[43] B. Barbieri. A short history of fluorescence. http://www.fluorescence-foundation.org/lectures/madrid2010/lecture1.pdf, 2010. [Online; accessed 20-3-2012].

[44] M. W. Davidson. Specialized Microscopy Techniques. http://micro.magnet.fsu.edu/primer/techniques/index.html, 1998-2012. [Online; accessed 13-3-2012].

[45] S. D. Kohlwein. http://www.leica-microsystems.com/science-lab/obese-and-slim-yeast-cells-microscopic-insights-into-cellular-lipid-metabolism/, 2008.

[46] U. f. B. W. Molekulare Pflanzenphysiologie. http://www.dagz.boku.ac.at/14606.html. [Online; accessed 2-4-2012].

[47] M. W. D. R. N. Day. http://zeiss-campus.magnet.fsu.edu/articles/probes/jellyfishfps.html. [Online; accessed 20-3-2012].

[48] R. Y. Tsien. http://www.nobelprize.org/nobel_prizes/chemistry/laureates/2008/tsien-slides.pdf, 2008. [Online; accessed 20-3-2012].

[49] Wikipedia. http://en.wikipedia.org/wiki/Airy_disk#Approximation_using_a_Gaussian_profile, . [Online; accessed 20-3-2012].

[50] Wikipedia. http://en.wikipedia.org/wiki/Chromatic_aberration, . [Online; accessed 20-3-2012].

[51] Wikipedia. http://commons.wikimedia.org/wiki/File:Delft_Voldersgracht.jpg, . [Online; accessed 20-3-2012].

[52] Wikipedia. http://www.scienceinyoureyes.com/index.php?id=79, . [Online; accessed 20-3-2012].

[53] Wikipedia. http://en.wikipedia.org/wiki/Green_fluorescent_protein, . [Online; accessed 20-3-2012].

[54] wikipedia. http://en.wikipedia.org/wiki/File:FluorescenceFilters_2008-09-28.svg. [Online; accessed 3-4-2012].