



Published in final edited form as:

*Nat Methods*. 2010 May ; 7(5): 373–375. doi:10.1038/nmeth.1449.

## Fast, single-molecule localization that achieves theoretically minimum uncertainty

Carlos S. Smith<sup>1</sup>, Nikolai Joseph<sup>2</sup>, Bernd Rieger<sup>1</sup>, and Keith A. Lidke<sup>2</sup>

<sup>1</sup> Department of Imaging Science & Technology, Delft University of Technology, Delft, The Netherlands <sup>2</sup> Department of Physics & Astronomy, University of New Mexico, Albuquerque, NM , USA

We describe an iterative algorithm that converges to the maximum likelihood estimate of the position and intensity of a single fluorophore. Our technique efficiently computes and achieves the Cramér-Rao Lower Bound, an essential tool for parameter estimation. An implementation of the algorithm on graphics processing unit hardware achieves more than  $10^5$  combined fits and Cramér-Rao Lower Bound calculations per second, enabling real-time data analysis for super-resolution imaging and other applications.

In many single molecule fluorescence applications, it is often desired to find the position and intensity of a single fluorophore as well as to estimate the accuracy and precision of these parameters. Where accuracy is a measure of the systematic error or bias and precision is a measure of the statistical error of an estimator<sup>1</sup>. In recent work that uses single-molecule localization to generate super-resolution images<sup>2-6</sup>, single emitters are located and on the mosaic of their found positions a two-dimensional Gaussian profile is placed to generate the final super-resolution images. The width of the placed Gaussian blob,  $\sigma$ , is given by the precision of the fluorophore position localization  $\sigma = (\sigma_x^2 + \sigma_y^2)^{1/2}$  and in these super-resolution techniques it is therefore necessary to both find the parameters and estimate their precision. Reported values are in the range of 20-70 nm. In the application of super-resolution imaging, it may be required to find the positions of more than  $10^6$  fluorophores in order to generate one final image of a typical field-of-view of  $50 \times 50 \mu\text{m}$ . In many cases, background rates of light detection may vary across the field of view and the fluorophore emission rate of chemically identical fluorophores can vary due to effects such as uneven illumination profile, dipole orientation or different optical path lengths. In this work, we describe an iterative routine, implemented on a graphics processing unit (GPU) that calculates the Maximum Likelihood Estimate (MLE) for the  $xy(z)$ -position, the photon count of the fluorophore and the background fluorescence rate (Supplementary Note). We show that our approach achieves the Cramér-Rao Lower Bound (CRLB) over a wide range of

Users may view, print, copy, download and text and data- mine the content in such documents, for the purposes of academic research, subject always to the full Conditions of use: [http://www.nature.com/authors/editorial\\_policies/license.html#terms](http://www.nature.com/authors/editorial_policies/license.html#terms)

Correspondence should be addressed to K.A.L. (klidke@unm.edu).

### AUTHOR CONTRIBUTIONS

C.S.S. and K.A.L. worked out the algorithm and implementation, N.J. and K.A.L. performed the experiments, B.R. and K.A.L. designed the research and wrote the paper

parameters. The uncertainties of the fitted parameters are found by calculating their CRLBs<sup>1</sup>, and in this sense the estimated  $\sigma$  for building up the super-resolution image is optimal. We provide a software tool ([www.diplib.org/home22266](http://www.diplib.org/home22266)) that only requires an inexpensive graphic card in order for single molecule fitting speed to be sufficient for real-time data analysis.

Since the speed and precision of single particle localization has long played an important role in single particle tracking as well as in other single molecule biophysical techniques that rely on fluorescent reporters, others have also considered these issues. Several algorithms from the literature for finding particle positions are compared in<sup>7</sup>, but, without the context of a statistical framework. The theoretically best-possible estimation precision of a fluorophore position has been derived in<sup>8</sup> by using the well established statistical method of finding the CRLB in an unbiased parameter estimation problem. They considered many of the effects in a real system including background fluorescence, finite camera pixel size, and camera readout noise and they recently made a software tool available for estimation ([www4.utsouthwestern.edu/wardlab](http://www4.utsouthwestern.edu/wardlab)). Non-linear least-squares (NLLS) and MLE position estimates were compared to the CRLB in<sup>9</sup>, and it was found that, in general, MLE gives better precision than NLLS. The better precision of MLE is in concurrence with our results, however in<sup>9</sup> they were investigated with assumed known emission and background rates. Here we demonstrate a robust, iterative routine that finds the particle position, the intensity and the background count rate. We do not consider camera readout noise since for electron multiplying (EM)CCD cameras, which are generally used for the fast frame rates desired in super-resolution imaging, the readout noise is much less than 1 rms e<sup>-</sup> when using large EM gain. As described further in the Supplementary Note, Supplementary Data and Supplementary Figs. 1-4, the method presented is not restricted to 2D imaging with a symmetric point spread function (PSF), but can be extended to handle super-resolution techniques that encompass astigmatic imaging for  $z$  resolution as in<sup>10</sup>. In this case, the  $z$  position is also calculated directly (not from intermediate  $\sigma_x$ ,  $\sigma_y$  fits) and returned with CRLB based uncertainties. The results of the iterative algorithm compared to CRLB-based theoretical values are shown for a range of background rates and total collected photon counts of the PSF (Fig. 1). We show results for  $\sigma_{\text{PSF}} = 1$  with the size defined in unitless back-projected pixels. The diffraction limit for high NA visible light imaging is  $> 200$  nm and  $\sigma_{\text{PSF}} > 90$  nm<sup>11</sup>. The algorithm both achieves and correctly reports the CRLB uncertainties over a wide range of background and fluorophore intensities. Calculated precision remains within a few percent of the theoretically achievable value even for less than 100 collected photons. We find that in all conditions, when the reported CRLB is less than  $\sigma_{\text{PSF}} / 2$  (here 0.5), the reported CRLB matches the theoretical position, and the routine achieves the CRLB. In typical super-resolution applications this corresponds to  $< 50$  nm. Addition of camera readout noise has effectively the same negative influence on the parameter estimation as high background. Fortunately, this can be excluded for an EMCCD for the reasons mentioned above. Example images of single fluorophores with intensities and background rates near the  $\sigma_{\text{PSF}} / 2$  value are shown in Supplementary Fig. 5. The classical approach to solving the position fitting problem is via NNLS optimization (Fig. 1b). Here we chose a Levenberg-Marquardt (LM) optimization scheme with analytic and computed first derivatives with respect to the optimization parameters. Note that it is

common practice to use computed derivatives only. The NNLS optimization clearly performs worse in terms of precision than our iterative MLE approach due mainly to the incorrect Gaussian-noise model implicitly present in any least-squares based optimization scheme. We compare the predicted uncertainty of the fit as in<sup>12</sup> (Eq. 14) with the theoretical CRLB (Fig. 1c). Strikingly, they are identical for no background fluorescence but for any non-zero background and low light conditions the deviation is almost a factor of two. This means that in these cases the suggested uncertainty  $\sigma$  used to constitute the super-resolution image is estimated at nearly half the CLR based value (overly optimistic). The formula presented in<sup>12</sup> has the advantage that it can be readily calculated by hand from measurable quantities. However, for a precise estimate, the use of the reported CRLB from our iterative algorithm is preferred.

Our iterative update scheme is similar to that described in<sup>13</sup>. We show, however, that a Gaussian approximation for the two-dimensional fluorophore PSF<sup>11</sup> and following localization leads to a compact analytical expression that allows for computationally fast localization without compromising on the localization precision. Our approach achieves the CRLB after a few ( $\sim 10$ ) iterations. It should be noted that the CRLB predicts the correct precision only when the model function is correct, and the isotropic Gaussian model may not be appropriate when imaging fluorophores with a fixed dipole orientation<sup>14</sup>, leading to anisotropic emission. As described in the Supplementary Data, a ‘rule-of-thumb’ fitting region size of  $2 \times 3\sigma_{\text{PSF}} + 1$  is used. For  $z$  resolution imaging the Gaussian PSF model is less reliable due to optical aberrations and the simplified model itself<sup>11</sup>.

GPU-based computation has the potential to increase floating point calculation speed by a factor of 10-100 as compared to a modern CPU if the problem is amenable to a parallel processing approach ([www.nvidia.com](http://www.nvidia.com)). A generic C-like language interface is available for simplified GPU programming (Nvidia CUDA), and a MATLAB (The Mathworks, USA) interface has been developed. Operating with a fixed number of iterations complements the GPU's single instruction multiple data strategy (SIMD). A GPU implementation of our iterative method can perform  $10^5$  combined MLE and CRLB calculations per second of the four parameter model needed to describe the emittance of a fluorophore (Fig. 2).

The CPU and GPU performance, characterized by the number of combined position fits and CRLB calculations performed per second, are shown in Table 1. The slowest GPU tested outperforms the CPU by more than one order of magnitude, with the fastest achieving  $2.6 \cdot 10^5$  fits per second on a  $7 \times 7$  fitting box size. We attribute this level of performance gain over the CPU to the fact that this estimation problem is almost ideal for the GPU SIMD architecture. Many iterations are performed on the same data, which are stored in local shared memory, and each thread is independent, eliminating synchronization delays. We also note that all CPU computation only ran on a single thread. In the right column of Table 1 the performance of a non-linear, least-squares fit (in C-code) on a CPU is shown for reference. It is twice as slow as our iterative algorithm on a CPU. Commonly used MATLAB LM optimization only computes about 5 fits/second. Given the readout rate of current high-end (EM)CCD cameras ( $\sim 10$  Mhz) our GPU implementation performs combined fits and uncertainty estimates at a speed sufficient for real-time analysis (see **Methods**).

To summarize our findings, we have derived an iterative approach for making a maximum likelihood estimate of the position and intensity of a single fluorophore as well as the background count rate using a two-dimensional Gaussian PSF model and a Poisson noise model. The iterative method achieves the minimal possible estimation uncertainty, as given by the Cramér-Rao Lower Bound, over a wide range of emission and background rates that could be found in single molecule experiments. Implementation of the iterative method on a modern graphics processing unit yields more than  $10^5$  combined fits and CRLB calculations per second, greatly facilitating the analysis of large data sets found in single molecule based super-resolution techniques and is suitable for real-time data analysis.

## METHODS

Methods and any associated references are available in the online version of the paper at <http://www.nature.com/naturemethods/>.

### Methods

#### GPU Implementation

The iterative method to solve the MLE problem described above is implemented on a GPU using NVIDIA's compute unified device architecture (Nvidia CUDA), a C based programming language that makes it possible to readily program parallelized algorithms that are executed on a GPU. High-end gaming and computing GPU's have a (card dependent) large region of global device memory, usually several hundred MBytes. Execution is performed on a number (card dependent) of multi-processors. Multi-processors contain eight sub-processors and have 16 KBytes local memory that is shared between sub-processors. Access to local shared memory is fast, whereas access to the device global memory incurs a large performance penalty, and should be avoided when possible. The programming model follows the GPU architecture in that parallelized execution is performed by breaking down computations into 'blocks' and 'threads'. Each block executes a set of threads using one multi-processor. Typically, performance is optimal when multiples of 32 threads (called a 'warp') are scheduled and executed using the 8 sub-processors. However, we found for all fit box sizes, the maximum fits/second occurred when the maximum fits/block were used, which is limited by the available 16KB shared memory per block. This is likely due to compiler optimization and the fact that each fit is independent of all others (no thread synchronization is required).

We map our iterative algorithm on this programming model in the following way. A data stack consisting of identically sized sub regions, which are centered single emitter images, are input to the function. This data is copied from host (CPU) memory to device (GPU) main memory. This data set is divided into blocks, which consist of the largest number of images that can fit into shared memory. The execution of a block begins by copying the data sub-set into local shared memory. Each thread then calculates a complete fit and CRLB calculation for one image. The Fisher information matrix is calculated using Supplementary Note Eqs. 9 and 11 and the CRLB is calculated using the analytical expression for the inverse of a  $4 \times 4$  matrix. The CPU performance was measured by replacing the block/thread architecture with nested loops which call the same sub-function, and was compiled

using Microsoft Visual Studio Express 2008. Images were loaded in MATLAB (The Mathworks, USA) as arrays and the C-Code and the CUDA GPU code were called via mex-files. The CPU was a AMD Phenom II X3 720 @2.8 GHz and only one core was used for the C-code.

### Synthetic Data Generation and Analysis

The CUDA routine described above operates on a data set consisting of identically sized images that contain images of (potential) single molecules. For the analysis of synthetic data, a stack is generated using the same finite pixel approximation, including background as described in the Supplementary Note. The center coordinate of each simulated emitter is randomly shifted within the central pixel to prevent a biased result. After the generation of the data stack, the images are corrupted with Poisson noise. This data stack is analyzed by the routine, which returns the estimated position, intensity, and background and the CRLB calculation for each 2D image in the stack. No camera read noise is added. Camera read noise for electron multiplying (EM) CCD cameras, which are generally used for the fast frame rates desired in super-resolution imaging, is much less than 1 rms  $e^-$  with large EM gain.

### Levenberg-Marquardt Non-Linear Least-Squares Fitting

The standard way of fitting a Gaussian to data is via least-squares fitting. We used 1) an existing implementation of MATLAB via the optimization toolbox (lsqcurvefit) where we enabled the Levenberg-Marquardt option as minimization scheme and 2) an implementation from Numerical Recipes in C<sup>15</sup>. We ran the test on fits with computed Jacobian only and on fits where we supplied analytic first derivatives. We used the following limits in the stopping criterion: tolerance on the parameters  $10^{-4}$ , tolerance on the function  $10^{-15}$  and maximal  $10^5$  function evaluations. The stopping criterion was in all cases determined by the accuracy put on the parameters. The MATLAB routine is a lot slower (about two orders of magnitude) than the C implementation although it makes automatic use of all available cores on the CPU if multi-threading is enabled.

### Single Molecule Imaging

Single molecule imaging experiments were performed in an epi-fluorescence microscope setup consisting of an inverted microscope (IX71, Olympus America Inc.), 1.45 NA objective (U-APO 150x NA 1.45, Olympus America Inc.), 635 nm diode laser (Radius 635, Coherent Inc.), and an electron multiplying CCD camera (Luca DL6581-TIL, Andor Technologies PLC.). The pixel size is 10  $\mu\text{m}$ . The epi-fluorescence filter setup consisted of a dichroic mirror (650 nm, Semrock) and an emission filter (692/40, Semrock). Individual Cy5 molecules were immobilized on an amino-silane ((3-Aminopropyl)triethoxysilane, Sigma-Aldrich) treated 8-well chambered cover slips (Lab-Tek II, Nunc) via an NHS-ester linkage attached to the Cy5 (Cy5 Mono-reactive dye pack, GE Healthcare). An oxygen scavenging system<sup>16</sup> was used to extend fluorophore lifetimes and quench fluorophore triplet states. This was necessary to perform repeated measurements of the same single emitter for several frames while acquiring sufficient photons in order to address localization accuracy. This is not necessary in a dedicated experiment.

Data was recorded by the CCD camera at either 10 or 20 frames per second. All data was post-processed by 1) subtracting a pixel-dependent camera-offset, which was created by averaging 300 dark frames, and 2) multiplying the resulting image by a gain factor to restore correct Poisson statistics, as done in<sup>17</sup>. Single molecules candidates were identified in each time frame as regions where the 2-D Gaussian filtered image ( $\sigma = \sigma_{\text{PSF}}$ ) was greater than one standard deviation of this image. Note that due to the speed of the GPU implementation, a simple but fast method for identifying candidates is preferred as well as one that errs on the side of including regions that do not contain single molecules. Square regions of a specified number of pixels that included all identified regions in the time series were collected into one stack and input to the GPU routine. The resulting found coordinates were used in building trajectories only if they passed the following criterion: 1) Reported localization accuracy was less than one fifth  $\sigma_{\text{PSF}}$  in each dimension, and 2)  $1/N \sum_k \ln(L(x_k | \theta)) > -1$ , where  $N$  is the number of pixels, which essentially performs a shape test and can rule out obvious cases of two proximate emitters. The remaining coordinates were connected into ‘trajectories’ using an existing single particle tracking routine<sup>18</sup>. Only ‘trajectories’ that showed little triplet state blinking were used in the final analysis, with a cut-off criterion that  $\text{var}(I(t)) < 2 * \text{mean}(I(t))$  where  $I(t)$  is the sum over all pixels in the analyzed region in frame  $t$ . ‘Trajectories’ were adjusted to compensate for microscope stage drift by subtracting a linear regression line from each single particle trajectory.

The width of the PSF used in the fitting routine was found by minimizing the mean square error between the finite pixel model and the summed projection over a 100 frame time series. The  $\sigma_{\text{PSF}}$  used in further analysis is the average found from analyzing the summed projection of 5 different single emitters.

### Astigmatic Imaging

The 3D astigmatic imaging was calibrated by imaging 100 nm red (690 nm emission) beads (FluoSphere, Invitrogen) bound to the bottom of an 8 well cover slip chamber (Lab-Tek II, Nunc). The filter setup used was the same as that used for single molecule Cy5 imaging. We imaged using a 60x 1.2 NA water objective. A 500 mm focal length cylindrical lens was inserted in the emission beam path just after the first lens of a two color beam splitter (OptoSplit II, Cairn Research, UK). A piezoelectric z-stage (Nano-LPS, Mad City Labs) translated the focal plane in steps of 50 nm from  $-0.5 \mu\text{m}$  to  $0.5 \mu\text{m}$ . At each focal plane, 20 images of a bead were captured. The fit box size used is again calculated by  $2 \times 3 \times \sigma_{\text{PSF}} + 1$ , but here  $\sigma_{\text{PSF}}$  is taken as the maximum value of either  $\sigma_{\text{PSF}_x}$  or  $\sigma_{\text{PSF}_y}$ ; in this case giving a fit box size of  $13 \times 13$  pixels.

After gain and background correction, the sum of all images from each focal plane were used to find  $\sigma_x(z)$  and  $\sigma_y(z)$ , which were then fit to model of Supplementary Note, Eq. 15. The fit is shown in Supplementary Fig. 1. From the calibration the following values for the parameters of Supplementary Note, Eq. 15 are found  $\sigma_{0_x} = 1.08$ ,  $\sigma_{0_y} = 1.01$ ,  $A_x = -0.708$ ,  $A_y = 0.164$ ,  $B_x = -0.073$ ,  $B_y = 0.0417$ ,  $d = 0.531$ ,  $\gamma = 0.389$ . The depth of field for a high NA imaging system is given by  $DOF = \lambda / (4n(1 - (1 - NA^2/n^2)^{1/2})) \approx 230 \text{ nm}^{22}$  but here is included as a fit parameter.

## Real-time data analysis

Initially, the bottleneck for the build-up of a super resolution image was the switching cycle speed for activating only a very small number of particles per image, which resulted in imaging times of many hours<sup>19</sup>. Subsequent speed-ups were achieved by optimizing the fluorophores for activation based super resolution, protocol improvements or reduction in the number of time frames<sup>20,22</sup>.

The fundamental relationships between error and acquisition rate (number of activation cycles) were investigated in a theoretical study<sup>24</sup>. The findings are relevant to specific chosen or given activation probability. However, to assess the required speed for real-time data analysis, we address the problem somewhat differently. We use the field-of-view  $V$ , the size of the footprint of the PSF  $P$ , the frame rate  $F$  and the fill factor  $f$  of the single emitters distribution on the field-of-view. The required fits/second for real-time data analysis are then

$$\text{fits/second} = VFf/P$$

We consider two common cases of emCCD cameras for the maximal fill factor of  $f=1$  and a PSF of  $P=7 \times 7$  pixels: i)  $V=128 \times 128$  pixels,  $F=500$  frames/second and ii)  $V=512 \times 512$  pixels,  $F=30$  frames/second. For the first case  $\sim 1.7 \times 10^5$  fits/second are required, and for the second  $\sim 1.6 \times 10^5$  fits/second respectively; these values are about equal as the total readout rate ( $\sim 10$  Mhz) is the limiting factor and is about equal. The PSF footprint can vary for different physical CCD camera pixel sizes and magnifications. In any case, the fastest GPU ( $2.6 \times 10^5$  fits/second) tested already full fills this requirement for current fluorophores and CCD cameras! Of course, a fill factor of  $f=1$  is optimistic; more realistic values are 0.1-0.5, but dependent on the experimental conditions and can be chosen according to<sup>23</sup>. This means that also the slower (and cheaper) cards already are sufficient in current practice for real-time fitting of positions. The significance of the GPU fitting in the context of the entire process of segmentation (identifying regions of interest for single molecule fits), organizing ROIs, single molecule fitting, and reconstruction, is shown in Supplementary Table 1. Segmentation and reconstruction are performed as described in<sup>25</sup> with the segmentation performed on the GPU. The results show that with even with  $10^6$  total fits, corresponding to 100 fits per frame, the overall processing could exceed the maximum possible frame rate of 500 Hz of available EMCCDs.

## Supplementary Material

Refer to Web version on PubMed Central for supplementary material.

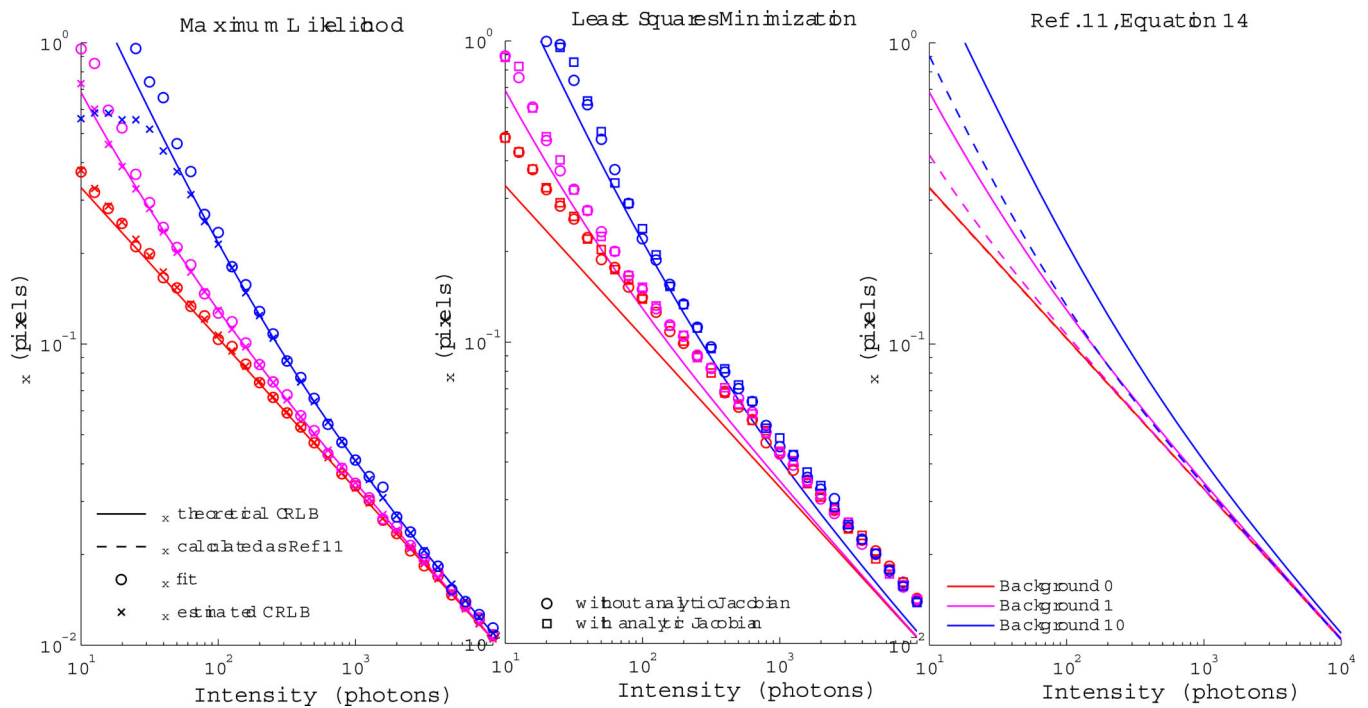
## ACKNOWLEDGEMENTS

This work was supported by American Cancer Society grant #IRG-92-024 and National Institutes of Health grant #1P50GM085273-01A1. We thank M. Malik and I.T. Young for reading the manuscript. The authors declare that they have no competing financial interests.

## REFERENCES

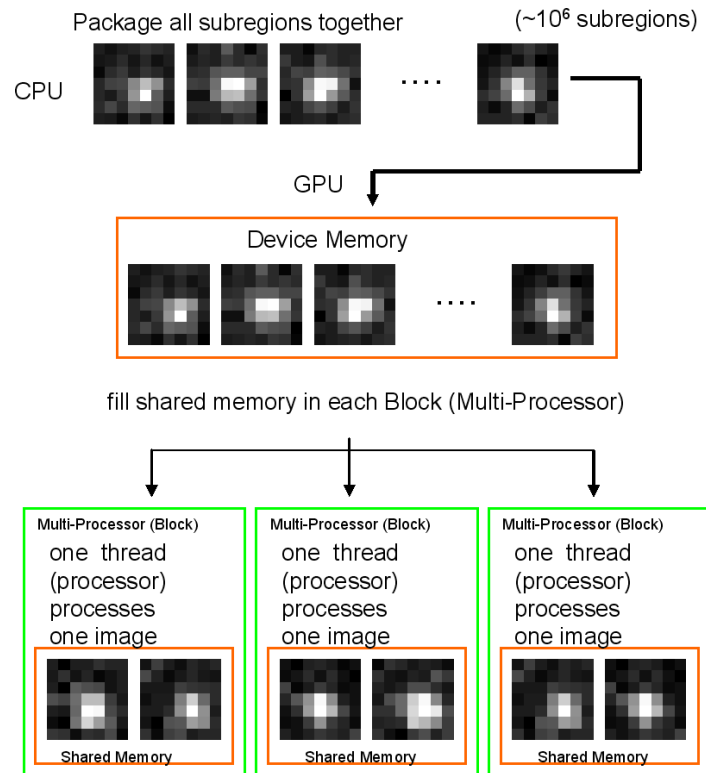
1. van den Bos, A. *Parameter Estimation for Scientists and Engineers*. Wiley & Sons; New Jersey: 2007.
2. Betzig E, et al. *Science*. 2006; 313:1642–1645. [PubMed: 16902090]
3. Bates M, Huang B, Dempsey GT, Zhuang X. *Science*. 2007; 317:1749–53. [PubMed: 17702910]
4. Egner A, et al. *Biophysical Journal*. 2007; 93:3285–3290. [PubMed: 17660318]
5. Hess ST, Girirajan TPK, Mason MD. *Biophysical Journal*. 2006; 91:4258–4272. [PubMed: 16980368]
6. Lidke KA, Rieger B, Jovin TM, Heintzmann R. *Optics Express*. 2005; 13:7052–7062. [PubMed: 19498727]
7. Cheezum MK, Walker WF, Guilford WH. *Biophysical Journal*. 2001; 81:2378–2388. [PubMed: 11566807]
8. Ober RJ, Ram S, Ward ES. *Biophysical Journal*. 2004; 86:1185–1200. [PubMed: 14747353]
9. Abraham AV, et al. *Optics Express*. 2009; 17:23352–23373. [PubMed: 20052043]
10. Huang B, Wang W, Bates M, Zhuang X. *Science*. 2008; 319:810–813. [PubMed: 18174397]
11. Zhang B, Zerubia J, Olivo-Marin JC. *Applied Optics*. 2007; 46:1819–1829. [PubMed: 17356626]
12. Thompson RE, Larson DR, Webb WW. *Biophysical Journal*. 2002; 82:2775–2783. [PubMed: 11964263]
13. Aguet F, Van De Ville D, Unser M. *Optics Express*. 2005; 13:10503–10522. [PubMed: 19503266]
14. Enderlein J, Toprak E, Selvin P. *Optics Express*. 2006; 14:8112–8120.
15. Press, W.; Teukolsky, S.; Vetterling, W.; Flannery, B. *Numerical Recipes in C: The Art of Scientific Computing*. 2 edn.. Cambridge University Press; Cambridge: 1992.
16. Rasnik I, McKinney SA, Ha T. *Nature Methods*. 2006; 3:891–893. [PubMed: 17013382]
17. Lidke KA, Rieger B, Lidke DS, Jovin TM. *IEEE Transactions on Image Processing*. 2005; 14:1237–1245. [PubMed: 16190460]
18. Andrews NL, et al. *Nature Cell Biology*. 2008; 10:955–963. [PubMed: 18641640]
19. Betzig E, et al. *Science*. 2006; 313:1642–1645. [PubMed: 16902090]
20. Rust MJ, Bates M, Zhuang XW. *Nature Methods*. 2006; 3:793–795. [PubMed: 16896339]
21. Bates M, Huang B, Dempsey GT, Zhuang X. *Science*. 2007; 317:1749–53. [PubMed: 17702910]
22. Egner A, et al. *Biophysical Journal*. 2007; 93:3285–3290. [PubMed: 17660318]
23. Young, I., et al. Depth-of-focus in microscopy.. In: Høgdal, KA.; Braathen, B.; Heia, K., editors. SCIA'93, Proceedings of the 8th Scandinavian Conference on Image Analysis; Norwegian Society for Image Processing and Pattern Recognition, Tromsø Norway. 1993; p. 493-498.
24. Small A. *Biophysical Journal - Biophysical Letters*. 2009; 92:L16–L18.
25. Wolter S, et al. *Journal of Microscopy*. 2010; 237:12–22. [PubMed: 20055915]





**Figure 1.**

Performance comparison on simulated data. a) The localization precision of our iterative method is compared to that given by the CRLB. Also shown are the mean uncertainties reported from CRLB calculations for every image using the found intensity and background rates (constant offset). Calculations are made using a square fitting region of size  $2 \times 3 \sigma_{\text{PSF}} + 1$  and 10 iterations. b) Fits are performed using non-linear least squares Levenberg-Marquardt with and without an analytical Jacobian. c) The theoretical uncertainty calculated from the four-parameter fit CRLB is compared to the commonly used formula of Ref <sup>12</sup>, Eq. 14 for estimating localization precision. It underestimates the true uncertainty by nearly a factor of two for low light conditions and any background rate.



**Figure 2.**

Basic Concept of Single Molecule Localization via GPU Implementation. The input consists of multiple (up to millions) pre-selected ROIs arranged in a 3D data set. Smaller data sets are arranged and processed in chunks that fill the multi-processor shared memory. Each image is analyzed with the same iterative algorithm. The hundreds of sub processors available on the GPU give a speed increase due to massive parallel processing. See supplementary information for more details.

Author Manuscript

Author Manuscript

Author Manuscript

Author Manuscript

**Table 1**

Computational performance. CPU and GPU implementations of the iterative MLE and a LM non-linear least-squares fit.

| Box Size [pixel] | iterative MLE method                       |  |  |   | LM (numeric Jacobian)                      |  |
|------------------|--|--|--|---|--|--|
|                  | AMD phenomII (10 <sup>2</sup> fits/second) | Nvidia 8600GTS (10 <sup>2</sup> fits/second) | Nvidia 8800GTS (10 <sup>2</sup> fits/second) | Nvidia GTX285 (10 <sup>2</sup> fits/second) | AMD phenomII (10 <sup>2</sup> fits/second) |  |
| 7 × 7            | 31   | 430  | 880  | 2600  | 15   |  |
| 13 × 13          | 9.4  | 45   | 100  | 950   | 5.2  |  |
| 15 × 15          | 4.3  | 10   | 22   | 330   | 2.3  |  |