ORIGINAL ARTICLE

Open Access

GraspJ: an open source, real-time analysis package for super-resolution imaging

Norman Brede[1,2] and Melike Lakadamyali[2*]

Abstract

We present an open source, real-time data analysis and rendering tool for super-resolution imaging techniques that are based on single molecule detection and localization (e.g. stochastic optical reconstruction microscopy - STORM and photoactivation localization microscopy – PALM). The recent availability of commercial STORM and PALM microscopes has made these techniques accessible to a wide range of researchers. However, the availability of high speed data analysis and rendering software lags behind, requiring researchers to develop their own analysis platforms or rely on commercial ones. We implemented GraspJ (GPU-Run Analysis for STORM and PALM), an ImageJ plug-in with a convenient user interface, that allows high accuracy localization of single molecules as well as processing and rendering of high resolution images in real-time. GraspJ includes several features such as drift correction, multi-color, 3D analysis/rendering, and is compatible with a large range of data acquisition software. In addition, it allows easy interfacing with other image processing tools available with ImageJ. Overall we believe that GraspJ will be a valuable tool for the super-resolution imaging field.

Keywords: STORM, PALM, Maximum likelihood, GPU

Background

In the past decade, a number of techniques have been developed that overcome the diffraction limit in fluorescence microscopy (Hell, 2007, Bates et al., 2008). Among such techniques, those that are based on single molecule photoactivation and localization (e.g. STORM, PALM, fPALM) depend heavily on data processing and analysis to produce high resolution images. These techniques break the diffraction limit by taking advantage of photoswitchable fluorophores (Fernandez-Suarez and Ting, 2008, Patterson et al., 2010, Dempsey et al., 2011) that can be "photo-activated" in sparse numbers. By only activating a small fraction of fluorophores that are spatially well separated, one can determine their mean positions precisely, even under dense labeling conditions. Repeated cycles of "photo-activation" followed by "imaging" and "deactivation" allows mapping of a large number of fluorophore-positions which can then be rendered into a high resolution image of a macromolecular structure, such as microtubules inside the cell (Bates et al., 2007).

A software package for the analysis of single molecule localization microscopy data needs to incorporate multiple features: *(i)* High accuracy localization of single molecule mean positions, *(ii)* sufficient data processing options (drift correction, high resolution rendering, multi-color and 3D analysis and display) and *(iii)* high speed for real-time analysis and rendering. Initial analysis programs for single molecule localization based super-resolution microscopy used a Gaussian model of the point spread function, which was fit to images of single molecules using a nonlinear least-squares method such as the Levenberg–Marquardt algorithm (Rust et al., 2006, Betzig et al., 2006). The computational demands of fitting and processing millions of molecules per image made these programs slow and several minutes to hours were needed for analyzing a single image. In the recent past, a number of algorithms have been developed to address the problem of speed in super-resolution data analysis (Smith et al., 2010, Wolter et al., 2010). Some of these algorithms have been further developed into open-source programs with multiple features for fast analysis and rendering of super-resolution images (Wolter et al.,

* Correspondence: melike.lakadamyali@icfo.es
[2]ICFO-Institut de Ciències Fotòniques, Mediterranean Technology Park, Castelldefels (Barcelona) 08860, Spain
Full list of author information is available at the end of the article

2010, Henriques et al., 2010, York et al., 2011). For example, QuickPALM is an ImageJ based plug-in that can process on the order of 2000–3000 molecules/second using an 8 CPU computer (Henriques et al., 2010). QuickPALM achieves this high speed by determining molecule positions via calculating the center-of-mass of single-molecule images instead of fitting (Gaussian) PSFs to them. However, the center-of-mass is an inaccurate method for determining molecule positions (Smith et al., 2010, Cheezum et al., 2001, Wolter et al., 2011) therefore is not the preferred algorithm for super-resolution imaging. In addition, the speed achieved by QuickPALM falls short of real-time analysis when high fluorophore densities and fast acquisition speeds are needed (e.g. 60 Hz or higher speeds used for live cell imaging (Jones et al., 2011)). Finally, when used for real-time analysis, QuickPALM is only compatible with one data acquisition software (µManager), which is a drawback for incorporating the data analysis with pre-existing acquisition software. rapidSTORM is another open source program that uses a more accurate algorithm (nonlinear least squares fitting of Gaussian PSFs) for determining molecule positions and that can achieve high speed analysis of super-resolution data (Wolter et al., 2010). However, in the real-time analysis mode, rapidSTORM only works with Andor cameras and is not compatible with other acquisition software, as it directly takes control over the camera. In addition, since rapidSTORM is a stand-alone program it cannot be easily modified or interfaced with powerful image processing tools offered by ImageJ. Therefore, there is still need for open source, easy to use, high speed and high accuracy software packages that include essential data processing features and are compatible with a wide range of data acquisition software. Such analysis tools will make super-resolution microscopy more accessible to researchers.

Here, we present a software package that can be used as an ImageJ plug-in or as a stand-alone program and that incorporates multiple features for analysis and rendering of super-resolution data. GraspJ includes peak finding, 2D and 3D maximum likelihood estimation (with Gaussian PSFs) of peak positions, drift correction and multi-color and 3D rendering of high resolution images. Most importantly, GraspJ runs at very high speeds both on the graphics card (>$1{*}10^6$ molecules/second processed on the GPU) and the central processing unit (~$6{*}10^4$ molecules/second processed on a 6-core CPU) and can be used to render 3D, drift corrected, high resolution images acquired at maximum frame rates attainable by modern CCD cameras in real-time. Moreover, due to the way in which real-time analysis is implemented, GraspJ can be compatible with almost any data acquisition software.

Materials and methods

Design and implementation of GraspJ

GraspJ is written in a highly abstracted, object oriented manner, making it easily extensible. Most parts are written in Java; while computationally expensive parts, like peak fitting, are implemented in OpenCL. It requires Java 7 or higher and OpenCL 1.1 or higher in order to run. All features of GraspJ are accessible and configurable from the graphical user interface. More information on how to run GraspJ, analyze the sample files and supported hardware can be found in the user manual provided here as Additional file 1. Fiji with GraspJ plugin, sample data for analysis and sample workflow can be downloaded from the GraspJ website (www.graspj.com).

Cell culture and immunostaining

BSC-1 cells were cultured inside chambered 8-well coverglasses (Labtek II, Nunc) in a 5% CO_2 incubator using Dulbecco's Modified Eagle Medium (DMEM) supplemented with 10% Fetal Bovine Serum (FBS). Cells were fixed prior to immunostaining with 3% paraformaldehyde (PFA) and 0.1% glutaraldehyde (GA) diluted in phosphate-buffered silane (PBS). Following fixation, cells were rinsed with PBS, and reduced with a fresh solution of 0.1% Sodium Borohydride in PBS. Cells were then rinsed once more with PBS and immunostained with primary and secondary antibodies as described previously (Bates et al., 2007). For tubulin and mitochondria labeling a rat-anti-tubulin primary antibody (Abcam, Cambridge, UK) and a rabbit-anti-Tom20 primary antibody (Santa Cruz Biotechnologies, Santa Cruz, CA) were used respectively. Secondary antibodies (Jackson Immuno Research, West Grove, PA) were custom labeled with activator and reporter dyes as described previously (Bates et al., 2007). For tubulin, a secondary antibody against rat labeled with Cy3 as an activator and A647 as a reporter dye was used. Similarly, for mitochondria a secondary antibody against rabbit labeled with A405 as an activator dye and Alexa 647 as a reporter dye was used.

For STORM imaging, cells were placed in a PBS imaging buffer containing 100mM cysteamine (Sigma Aldrich, St. Louis, MO) pH 8.5, 5% glucose (wt/vol), oxygen scavenging enzymes (0.5 mg ml^{-1} glucose oxidase) (Sigma Aldrich, St. Louis, MO), and 40 µg ml^{-1} catalase (Sigma-Aldrich, St. Louis, MO).

STORM imaging

Images were acquired either on a "custom-built" microscope fitted with a 100 × 1.4 NA oil immersion objective (Olympus, Center Valley, PA) as described previously (Bates 2010) or on a commercial Nikon STORM microscope (Nikon Instruments Europe B.V., Amstelveen, Netherlands) at ICFO's Super-resolution Light Nanoscopy

(SLN) facility. Multi-color images were acquired using a repetitive sequence of activation-imaging light as described previously (Bates et al., 2007). 532 nm light was used to activate the Cy3-A647 pair and 405 nm light was used to activate the A405-A647 pair. 647 nm light was used following each activation pulse in order to excite A647 and the fluorescence emission from A647 was recorded by an EMCCD camera (Andor Technology, Belfast, Northern Ireland) at a frame rate of 60 Hz. The localizations were later color coded based on whether the A647 emission was detected after the 405 nm, or 532 nm activation light. For 3D imaging, a cylindrical lens was used in the image path in order to introduce astigmatism to the single molecule point spread function as described previously (Huang et al., 2008a).

Results

Peak finding and fitting

The general workflow for data analysis is summarized in Figure 1. First, single molecule peaks are identified based on a threshold. The threshold is set by the user and should correspond to the minimum intensity that will be identified as a single molecule (see User's Manual for more details on the choice of threshold). An initial estimate of the mean position is made based on the center-of-mass. This initial guess makes the program faster and more robust. The actual x, y and z positions are then determined by the maximum likelihood estimation (MLE) using a Gaussian point spread function and Poisson noise model as described previously (Smith et al., 2010). 3D analysis (and z-position determination) is based on astigmatism, a commonly used method for acquiring 3D super-resolution data (Huang et al., 2008a, Cella Zanacchi et al., 2011). The user must predetermine the calibration parameters for the z-positions by measuring the PSF widths (w_x and w_y) of fluorescent beads or single molecules at different z-planes using a piezo stage as previously described (Smith et al., 2010, Huang et al., 2008a). These calibration parameters can then be used as input parameters for 3D analysis in GraspJ (see the User's Manual for a more detailed description of 3D calibration). Smith et al. and others rigorously evaluated the accuracy of the MLE approach in comparison to Least Squares Minimization (LSM) using simulated data and found that MLE outperforms LSM in localization accuracy (Smith et al., 2010, Abraham et al., 2009). As a result, we adopted the same algorithm that was used by Smith et al. to achieve the same level of localization accuracy in peak fitting. For further improving localization accuracy, the same peak that appears in multiple frames is trailed together and the mean position is determined as the average of the mean positions in each frame, weighted by the number of photons collected in that frame as described previously (trail generation) (Bates
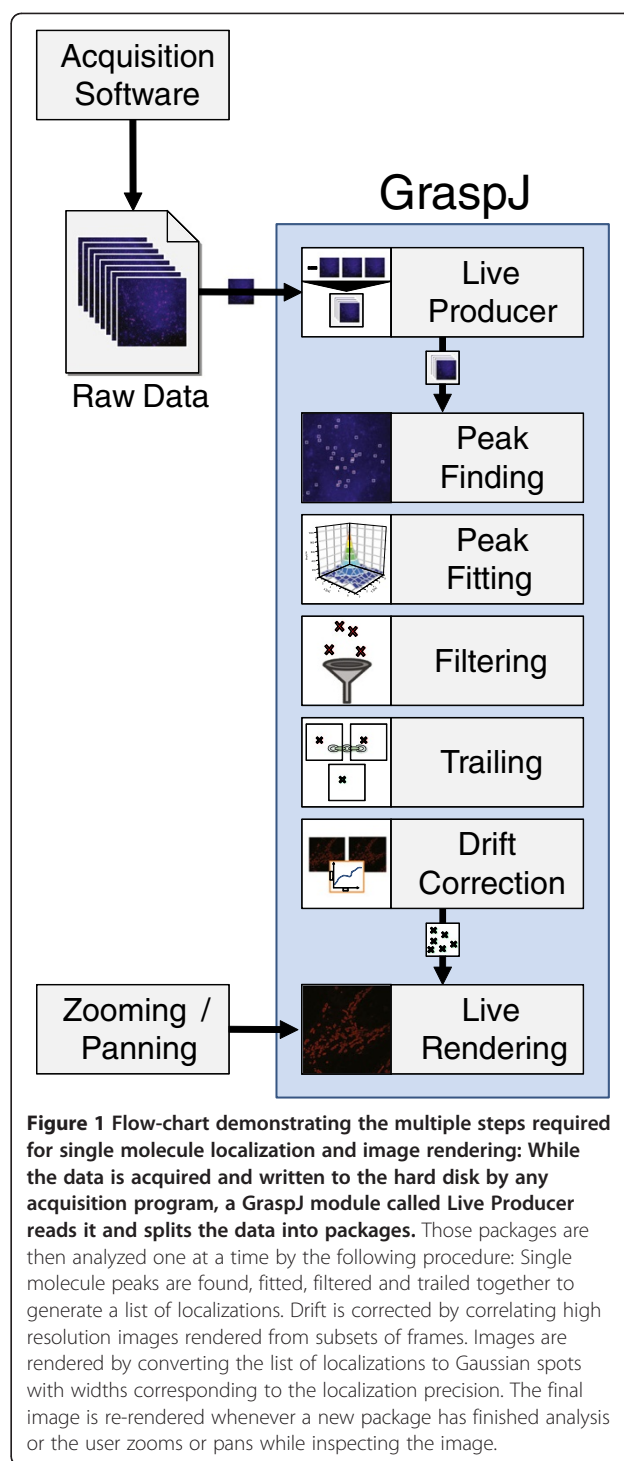
Figure 1 Flow-chart demonstrating the multiple steps required for single molecule localization and image rendering: While the data is acquired and written to the hard disk by any acquisition program, a GraspJ module called Live Producer reads it and splits the data into packages. Those packages are then analyzed one at a time by the following procedure: Single molecule peaks are found, fitted, filtered and trailed together to generate a list of localizations. Drift is corrected by correlating high resolution images rendered from subsets of frames. Images are rendered by converting the list of localizations to Gaussian spots with widths corresponding to the localization precision. The final image is re-rendered whenever a new package has finished analysis or the user zooms or pans while inspecting the image.

et al., 2007). An interesting extension of the trailing would be its application to single particle tracking, in which the positions of a moving particle can be linked based on proximity to determine its trajectory within space. The trails could be used as the tracked positions of a particle and a threshold can be set in terms of how far the particle

can move from one frame to the next to be still considered as the same particle. While GraspJ is internally able to calculate such trails, this function and the resulting information is not accessible through the user interface and therefore implementation of such features would be an interesting future improvement to the package.

The peak finding and fitting give robust results when the emitter densities do not exceed the range appropriate for single molecule fitting (i.e. as long as the emitters are not substantially overlapping in each frame, typically ~0.5-2 fluorophores per μm^2) (Huang et al., 2011, Zhu et al., 2012). Imaging at higher emitter densities requires the use of multi-peak fitting algorithms (Huang et al., 2011, Zhu et al., 2012, Babcock et al., 2012), which is currently not implemented in GraspJ but would be a useful future extension of its capabilities. Positions are finally filtered based on typical parameters that can be set by the user (e.g. uncertainty of the positions, minimum and maximum photon counts etc.). GraspJ also reports the average localization precision for a given image, which is one of the determining factors in the final image resolution.

Data processing and rendering
Drift correction
Data acquisition for super-resolution imaging typically lasts several minutes. During this time, there can be substantial stage drift that degrades resolution. Stage drift can be corrected in multiple ways. For example, fiduciary markers such as fluorescent beads can be tracked and their average motion subtracted from the mean positions (Rust et al., 2006, Betzig et al., 2006). Alternatively, the dataset can be divided into a number of equal length time segments for which drift is assumed to be small. Each time segment is then rendered as a high resolution image and the shift between these partial images is determined using an image registration method (intrinsic drift correction) (Bates et al., 2007, Huang et al., 2008a). As part of GraspJ, we implemented the latter approach in which we use a phase correlation method to compare rendered images from subsequent time segments. This "intrinsic" drift correction performs well if the image contains a well-defined structure, which can be correlated between high resolution images constructed from subsets of frames. However, in the case of images that lack a structure (for example small clusters of localizations) this approach can still be applied by adding fiduciary markers to the sample. If the fiduciary markers appear in a frame that does not contain fluorescence from the single molecules (i.e. an activation frame), then the positions of these markers alone can be used for "intrinsic" drift correction (see User Manual for more details).

High resolution rendering of final images
Mean-position determination and drift-correction produces a list of molecular positions, typically in the range of 10^5-10^7 localizations depending on the dataset. In order to convert this list to a meaningful high resolution image, the fluorophore positions must be rendered. GraspJ uses a common rendering approach in which localizations are represented as Gaussian peaks of unit volume and width corresponding to the theoretical localization precision (Bates et al., 2007, Huang et al., 2008b). For single color data, all localizations are coded with the same color. Multi-color coding based on STORM imaging with multiple activator dyes is also possible (Bates et al., 2007). In this approach, localizations are color coded based on when they appear in a cycle consisting of "activation" and "imaging" frames (Bates et al., 2007). For example, for a sequence of one activation frame (405 nm laser), three imaging frames (647 nm laser) followed by one activation frame (560 nm laser) and three imaging frames (647 nm laser), the localizations appearing in the frame immediately after the 405 nm activation laser are coded in one color and those appearing in the frame immediately after the 560 nm laser are coded in a different color. All other localizations are coded as "non-specific", since they cannot be directly linked to the activation laser. Each color is displayed in a separate window, which can later be overlaid using ImageJ. The use of different activators linked to the same reporter dye is prone to color cross talk due to the non-specific or false activations as previously described (Bates et al., 2007, Dani et al., 2010, Bates et al., 2011, Lakadamyali et al., 2012). Statistical methods have also been developed to correct for color cross-talk in these multi-color images (Dani et al., 2010, Bates et al., 2011). Future implementation of these cross-talk correction algorithms would further enhance GraspJ's feature-set. For 3D data, localizations are color-coded based on their z-positions using a continuous color scale (Huang et al., 2008a, Huang et al., 2008b, Shtengel et al., 2009). The rendered 2D and 3D images can be visualized within GraspJ at multiple zoom levels and the image is re-rendered at high resolution for each zoom level. Overall GraspJ allows easy visualization of multi-color, 2D and 3D super-resolution images.

Speed
Data analysis is often a major bottleneck in super-resolution imaging techniques based on single molecule localization. Existing algorithms mostly rely on post-acquisition analysis, which dramatically slows down the progress of research. Decisions regarding the quality and information content of the images cannot be made until several hours after the image acquisition is completed. Therefore, typically several images are recorded only to

be discarded later due to low localization density, high drift or other factors that lead to low image quality.

Images are typically recorded at a frame rate of 50–60 Hz (256 × 256 pixels FOV), while live-cell imaging may require even faster frame rates (e.g. 500 Hz at 128 × 128 pixels FOV) (Jones et al., 2011, Shim et al., 2012). Taking into account the fastest available camera speeds and maximum density of localizations that can be tolerated for single molecule imaging, the processing speed needs to be on the order of $1.5*10^5$ molecules/second (given a box size of 7 × 7 pixels for fitting) to accommodate real-time analysis (Smith et al., 2010, Wolter et al., 2011, Huang et al., 2011). Figure 2 shows typical processing times required for different steps of the analysis using GraspJ. When run on a relatively inexpensive graphics card (such as ATI HD 6970), GraspJ can fit $1*10^6$ molecules/second. Combined with other processing times (e.g. drift correction, rendering etc.), the GPU speed comes very close to the maximum requirement for real-time analysis. The speed of GraspJ compares very favorably to other fast data analysis software. Using simulated data (single color, 2D data with 256×256 FOV, 20,000 frames and ~600,000 localizations) we determined the combined speed for "peak finding, peak fitting, filtering and rendering" steps to be ~9 seconds for GraspJ running on the GPU, ~16 seconds for GraspJ running on the CPU, ~13 seconds for rapidSTORM and ~75 seconds for QuickPALM. The speed advantage of GraspJ running on a GPU should become more pronounced as the complexity of the analysis increases (for example when working with 3D data or future implementations of multi-peak fitting algorithms). The comparison was performed on a computer with an Intel i5-2500K (3.6 GHz) CPU and an AMD Radeon HD 6850 GPU. The full test stack was pre-loaded into the RAM to avoid hard-disk speed from biasing the results. The "trail generation and drift correction" steps were not included since they were not supported by the other software tested. Figure 3 shows an actual example of experimental 3D, 2-color data as it is being analyzed by GraspJ (also see Movie1 provided as Additional file 2). The images of microtubules and mitochondria become clear over time as more localizations are added. While the speed achieved on the CPU is significantly slower (~$6*10^4$ molecules/second for fitting), it is still compatible with real-time analysis of super-resolution data acquired under most typical conditions.

Real-time analysis implementation

In order to make GraspJ compatible with real-time analysis of data acquired with a wide range of acquisition software, the data is read-out and analyzed in portions as it is being written to the hard disk. GraspJ loads a certain number of frames (referred to as one package) as they are recorded, performs the peak finding, mean position determination, intrinsic drift correction and renders the high resolution image in real-time. As more frames accumulate, the data is loaded and the rendered image is updated accordingly. This approach works well for images recorded in the form of raw data but is also compatible with most other data formats that can be loaded using the LOCI Bio-Formats library.

The provided additional movie file (Additional file 2) demonstrates real-time analysis and rendering of 2-color and 3D image of microtubules and mitochondria acquired at 60Hz. The first two windows display the
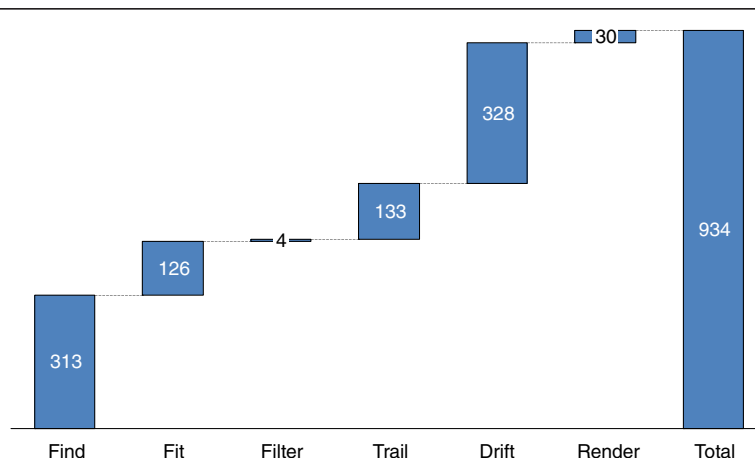
Figure 2 Typical processing times (in ms) for a 1024-frames package: During real-time analysis, a certain number of frames that constitutes one package is loaded and analyzed at a time. The chart shows the processing time for each individual step of the analysis for a typical package that contains 1024 frames. Individual processing times and their relative magnitude are highly dependent on the data set as well as configuration parameters like the chosen package size and the computer used for analysis. Therefore, these numbers can only give an impression about the approximate speed achieved by GraspJ. Typically, drift correction is the most time consuming part.
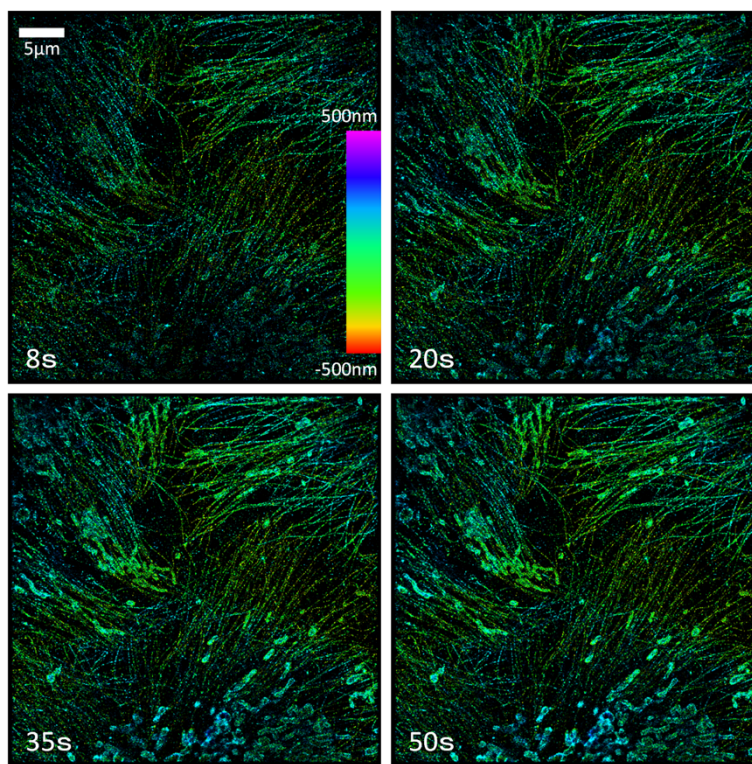
Figure 3 STORM data analysis with GraspJ: Snapshots that show the progress at different time points during the analysis of a 3D, 2-color STORM dataset. Each snapshot is a drift corrected and rendered image of microtubules and mitochondria color coded to indicate z-positions according to the color-scale bar. The microtubule and mitochondria become more apparent as more localizations are added to the image.

individual channels (mitochondria and microtubules, respectively) rendered in 2D and the final window displays the merged image color-coded in 3D. A drift corrected, high resolution image is displayed and updated after each package, which in this case was chosen to be 256 frames. The user can examine the image at multiple dynamic zoom levels and pan to different positions within the image as the image is building up. This ability to inspect images at multiple zoom levels in high resolution makes GraspJ a very powerful tool for super resolution imaging as important decisions regarding the sample and data can be made instantaneously.

Discussion

A number of open source programs are available for super-resolution imaging (Wolter et al., 2010, Henriques et al., 2010, York et al., 2011); however, the combination of high speed, high accuracy, compatibility with a range of acquisition software and interfacing ability with ImageJ makes GraspJ unique. The maximum likelihood estimator used for determining molecule positions achieves theoretically maximum accuracy (Abraham et al., 2009). The speed realized on the GPU surpasses the maximum required speed for real-time analysis with currently used frame rates and molecule densities. Recently, multiple-

peak-fitting algorithms have been developed to increase the tolerable density of molecules in each frame (Huang et al., 2011, Zhu et al., 2012, Babcock et al., 2012). Future implementation of these algorithms would dramatically decrease the speed. However, since GraspJ is implemented on the GPU, the speeds achieved would still be compatible with real-time analysis while other open source programs that only run on the CPU may not achieve real-time analysis speeds with these algorithms (Huang et al., 2011, Zhu et al., 2012, Babcock et al., 2012). Overall, GraspJ should prove useful for researchers interested in super-resolution imaging.

Conclusions

In summary we developed a fully functional software analysis package that not only provides the list of molecular positions obtained from single molecule photoactivation and localization data but also renders these positions as a high resolution, drift-corrected image in real-time. The easy to use interface provides visualization options of the rendered high resolution images in multi-color and 3D. Since GraspJ can be used as an ImageJ plug-in, it can be easily interfaced with all other image processing tools available through ImageJ. In addition, users can easily extend the capabilities of

GraspJ by creating and incorporating additional modules (such as different fitting algorithms, rendering types or additional analysis methods) in Java, without having to modify the remaining source code.

Additional files

Additional file 1: User manual for the GraspJ software.

Additional file 2: Real-time analysis demonstration: The movie shows a high resolution, multi-color and 3D image of microtubules and mitochondria building up in real-time during a simulated STORM image acquisition in which a previously acquired 256 X 256 pixel dataset was re-written onto the hard disk at a speed of 60 Hz. The analysis was performed using a 256-frame analysis package. The first two windows show the separate channels (mitochondria and microtubules respectively) and the final window shows the combined channels with z-color-coding. Initially, the movie starts at a high zoom level in which the image of an individual mitochondrion can be seen building up in the first window. The zoom level is dynamically and synchronously changed during analysis in order to inspect different regions of the image.

Authors' contributions

NB and ML conceived of the study, and participated in its design and coordination. NB wrote the software and carried out the data analysis. NB and ML wrote the paper. All authors read and approved the final manuscript.

Author details

[1]Department of Physics and Astronomy, University of Heidelberg, Heidelberg, Germany. [2]ICFO-Institut de Ciències Fotòniques, Mediterranean Technology Park, Castelldefels (Barcelona) 08860, Spain.

References

Hell SW (2007) Far-field optical nanoscopy. Science 316:1153

Bates M, Huang B, Zhuang X (2008) Super-resolution microscopy by nanoscale localization of photo-switchable fluorescent probes. Curr Opin Chem Biol 12:505

Fernandez-Suarez M, Ting AY (2008) Fluorescent probes for super-resolution imaging in living cells. Nat Rev Mol Cell Biol 9:929

Patterson G, Davidson M, Manley S, Lippincott-Schwartz J (2010) Superresolution imaging using single-molecule localization. Annu Rev Phys Chem 61:345

Dempsey GT, Vaughan JC, Chen KH, Bates M, Zhuang X (2011) Evaluation of fluorophores for optimal performance in localization-based super-resolution imaging. Nat Methods 8:1027

Bates M, Huang B, Dempsey GT, Zhuang X (2007) Multicolor super-resolution imaging with photo-switchable fluorescent probes. Science 317:1749

Rust MJ, Bates M, Zhuang X (2006) Sub-diffraction-limit imaging by stochastic optical reconstruction microscopy (STORM). Nat Methods 3:793

Betzig E, Patterson GH, Sougrat R, Lindwasser OW, Olenych S, Bonifacino JS, Davidson MW, Lippincott-Schwartz J, Hess HF (2006) Imaging intracellular fluorescent proteins at nanometer resolution. Science 313:1642

Smith CS, Joseph N, Rieger B, Lidke KA (2010) Fast, single-molecule localization that achieves theoretically minimum uncertainty. Nat Methods 7:373

Wolter S, Schuttpelz M, Tscherepanow M, Van de Linde S, Heilemann M, Sauer M (2010) Real-time computation of subdiffraction-resolution fluorescence images. J Microsc 237:12

Henriques R, Lelek M, Fornasiero EF, Valtorta F, Zimmer C, Mhlanga MM (2010) QuickPALM: 3D real-time photoactivation nanoscopy image processing in ImageJ. Nat Methods 7:339

York AG, Ghitani A, Vaziri A, Davidson MW, Shroff H (2011) Confined activation and subdiffractive localization enables whole-cell PALM with genetically expressed probes. Nat Methods 8:327

Cheezum MK, Walker WF, Guilford WH (2001) Quantitative comparison of algorithms for tracking single fluorescent particles. Biophys J 81:2378

Wolter S, Endesfelder U, van de Linde S, Heilemann M, Sauer M (2011) Measuring localization performance of super-resolution algorithms on very active samples. Opt Express 19:7020

Jones SA, Shim SH, He J, Zhuang X (2011) Fast, three-dimensional super-resolution imaging of live cells. Nat Methods 8:499

Bates M (2010) Jones S A, Zhuang X. In: Yuste R (ed) Imaging: A Laboratory Manual. New York, Cold Spring Harbor Laboratory Press

Huang B, Wang W, Bates M, Zhuang X (2008a) Three-dimensional super-resolution imaging by stochastic optical reconstruction microscopy. Science 319:810

Cella Zanacchi F, Lavagnino Z, Perrone Donnorso M, Del Bue A, Furia L, Faretta M, Diaspro A (2011) Live-cell 3D super-resolution imaging in thick biological samples. Nat Methods 8:1047

Abraham AV, Ram S, Chao J, Ward ES, Ober RJ (2009) Quantitative study of single molecule location estimation techniques. Opt Express 17:23352

Huang F, Schwartz SL, Byars JM, Lidke KA (2011) Simultaneous multiple-emitter fitting for single molecule super-resolution imaging. Biomed Opt Express 2:1377

Zhu L, Zhang W, Elnatan D, Huang B (2012) Faster STORM using compressed sensing. Nat Methods 9:721

Babcock H, Sigal YM, Zhuang X (2012) A high-density 3D localization algorithm for stochastic optical reconstruction microscopy. Optical Nanoscopy 1:6

Huang B, Jones SA, Brandenburg B, Zhuang X (2008b) Whole-cell 3D STORM reveals interactions between cellular structures with nanometer-scale resolution. Nat Methods 5:1047

Dani A, Huang B, Bergan J, Dulac C, Zhuang X (2010) Superresolution imaging of chemical synapses in the brain. Neuron 68:843

Bates M, Dempsey GT, Chen KH, Zhuang X (2011) Multicolor super-resolution fluorescence imaging via multi-parameter fluorophore detection. Chemphyschem 13:99

Lakadamyali M, Babcock H, Bates M, Zhuang X, Lichtman J (2012) 3D Multicolor Super-Resolution Imaging Offers Improved Accuracy in Neuron Tracing. PLoS One 7:e30826

Shtengel G, Galbraith JA, Galbraith CG, Lippincott-Schwartz J, Gillette JM, Manley S, Sougrat R, Waterman CM, Kanchanawong P, Davidson MW, Fetter RD, Hess HF (2009) Interferometric fluorescent super-resolution microscopy resolves 3D cellular ultrastructure. Proc Natl Acad Sci U S A 106:3125

Shim SH, Xia C, Zhong G, Babcock HP, Vaughan JC, Huang B, Wang X, Xu C, Bi GQ, Zhuang X (2012) Super-resolution fluorescence imaging of organelles in live cells with photoswitchable membrane probes. Proc Natl Acad Sci U S A 109:13978