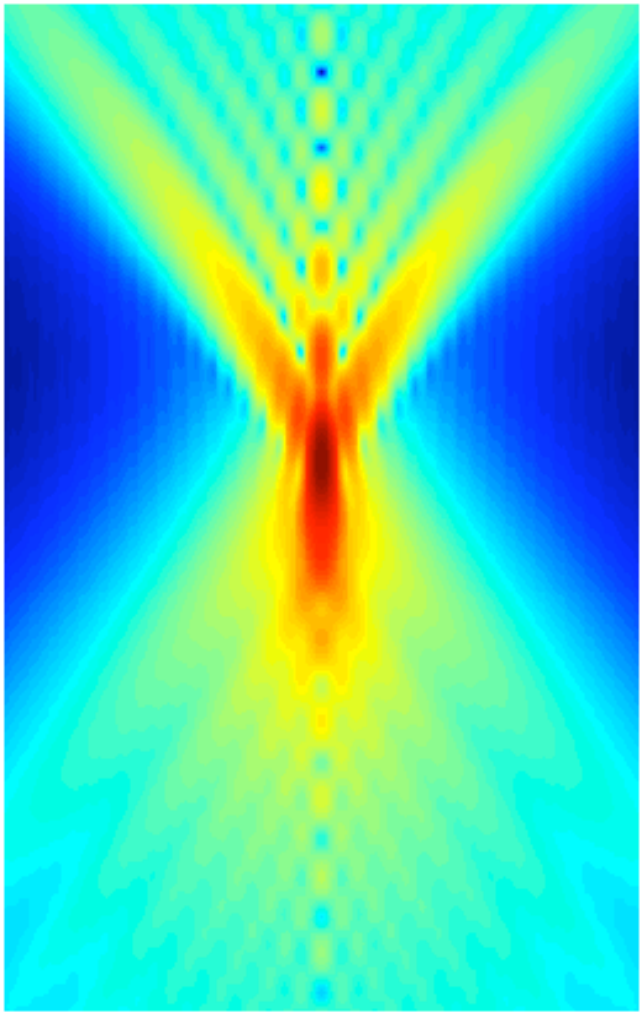
PSF Lab
Getting Started Manual

Program Version 3.0
Data Structure Version 1.6

Developed by Michael J. Nasse
with constributions by Jörg C. Woehl

The One Molecule Group
University of Wisconsin – Milwaukee



Conditions of Use

PSF Lab can be used **freely** for non-commercial research and educational purposes, **under the condition** that any publication or presentation that contains results derived from the use of this software

1. mentions the software by name (PSF Lab), and
2. properly cites the following journal article:

   M. J. Nasse and J. C. Woehl, "Realistic modeling of the illumination point spread function in confocal scanning optical microscopy", *J. Opt. Soc. Am. A* **27**, 295-302 (2010).

For inquiries concerning commercial use contact the UWM Research Foundation: `www.uwmresearchfoundation.org.`

1. About PSF Lab

PSF Lab is a software program that calculates the illumination point spread function (PSF) of a confocal microscope under various imaging conditions. It is based on theory described in the following, open-access journal article:

> M. J. Nasse and J. C. Woehl
> "Realistic modeling of the illumination point spread function in confocal scanning optical microscopy"
> *Journal of the Optical Society of America A* **27**, 295-302 (2010)

The Optical Society of America (OSA) selected this publication as the "Spotlight on Optics" article of the month of February 2010. A movie associated with this article was shown on the journal's main webpage. PSF Lab has been used to generate all figures in this paper.

PSF Lab is under active development. If you have any questions or suggestions for improving the program, please send us an email (`woehl@uwm.edu`).

2. Installation

Download the correct version of the PSF Lab package for your operating system from the One Molecule Group website and move the downloaded file to a new folder (*e.g.*, to *C:\Program Files\PSF Lab* under Windows, or to the Desktop on your Mac). The rest of the installation procedure differs depending on your operating system.

Windows

Double-clicking the package will unpack its contents, which consists of a batch file (*_install.bat*) and the following files:

1. *PSFLab.exe*
   The main program, which can also be downloaded separately from the One Molecule Group website. It won't run, however, if the correct bit version of the MATLAB Compiler Runtime version 7.14 is not installed on your system.

2. *MCRInstaller.exe*
   The installer program for the MATLAB Compiler Runtime (MCR), which is a standalone set of shared libraries that enable the execution of MATLAB code used in PSF Lab.

3. *readme.txt*
   A short text file containing basic program information, conditions of use, and legal disclaimer.

If an older version of any of these files already exist in this folder, you will need to confirm its replacement by entering **y** for yes.

After file extraction, the MCR installer is automatically launched, which will usually ask you to confirm the installation of the Microsoft Visual C++ redistributable (*VCREDIST_X64* for 64-bit or *VCREDIST_X86* for 32-bit).

Follow the on-screen instructions to install the MCR; once finished, PSF Lab is ready to run. You may want to create a shortcut to PSF Lab on your Desktop or to your Start Menu, and delete the downloaded package and MCR installer executables, which are no longer needed.

Mac OS

Double-clicking the package file will unpack its contents to a new folder, which contains the following files:

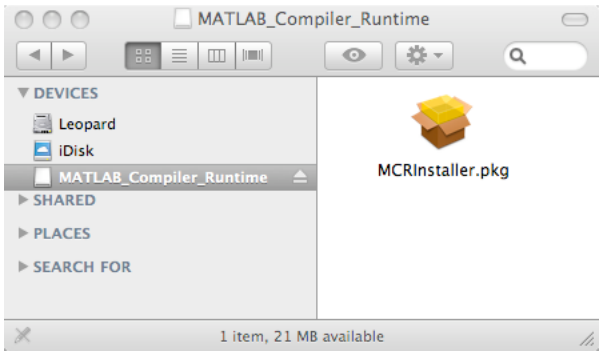
1. *PSFLab.app*
   The main program, which can also be downloaded separately from the One Molecule Group website. It won't run, however, if the correct bit version of the MATLAB Compiler Runtime version 7.14 is not installed on your system, or the environment variables aren't set up properly.

2. *MCRInstaller.dmg*
   The disk image for installing the MATLAB Compiler Runtime (MCR), which is a standalone set of shared libraries that enable the execution of MATLAB code used in PSF Lab.

3. *run_PSFLab.sh*
   A shell script needed to add the MCR path to the Mac OS environment variables.

4. *readme.txt*
   A short text file containing installation instructions, basic program information, conditions of use, and legal disclaimer.

Double-clicking on *MCRInstaller.dmg* mounts and opens the following volume:



Double-click on *MCRInstaller.pkg* to start the installation.

During the installation process, you will need to select the destination for the installation (typically your system drive) and provide your system password to allow the copying of the MCR files. Once the MCR installation has completed successfully, you will find a new *MATLAB* folder in your *Applications* folder. If you have accepted all installation defaults, the path to MCR version 7.14 is then given by

/Applications/MATLAB/MATLAB_Compiler_Runtime/v714/

Drag the *MATLAB_Compiler_Runtime* volume over the Trash icon to unmount the volume, which also closes the associated window.

Lastly, the environment variables for the operating system need to be set up so that PSF Lab can access the MCR library. To do this, open a Terminal window (found in *Applications – Utilites*) and change to the folder containing the shell script. Assuming that this folder is on your Desktop, you would type

```
cd /Users/Username/Desktop/PSFLab_mac64_pkg/
```

where "Username" is your login name. Run the script with

```
./run_PSFLab.sh /Applications/MATLAB/MATLAB_Compiler_Runtime/v714/
```

(or substitute the argument to the shell script with the correct path to *v714* on your system). Running the shell script will set the environment variables and automatically launch PSF Lab; it will also launch X11, the X Window System, which is required for running MATLAB code under Mac OS. From then on, PSF Lab can be simply launched by double-clicking on the PSF Lab icon.

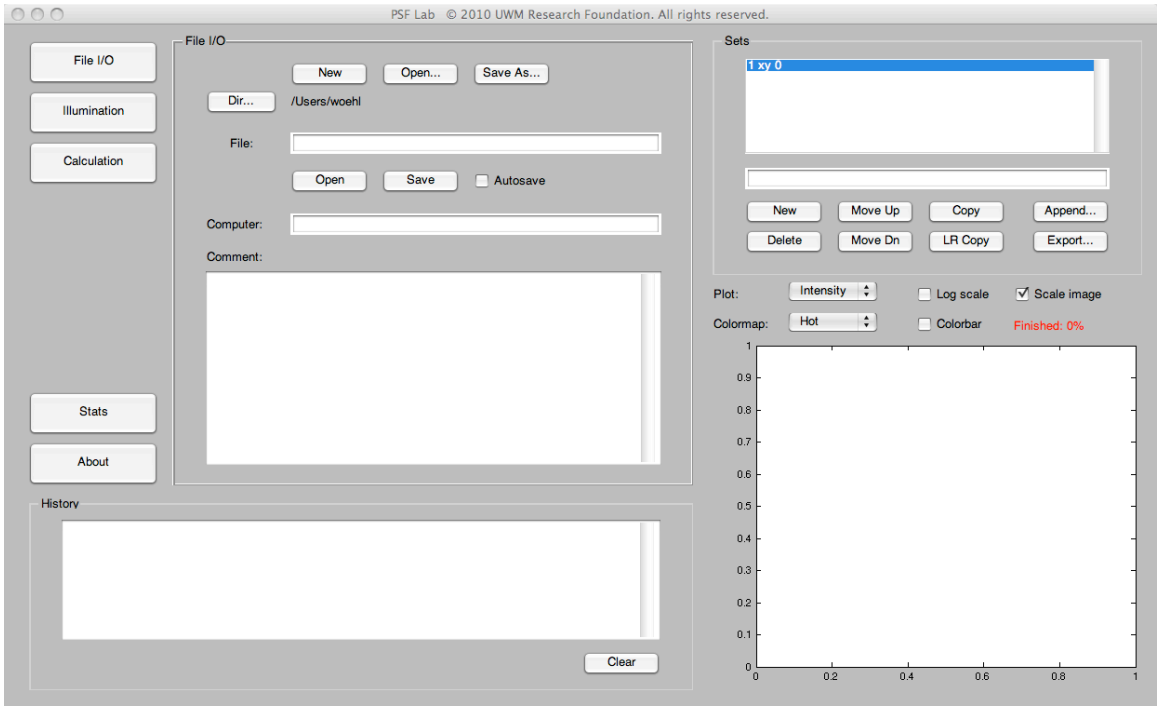
You may want to move PSF Lab into your Applications folder or to the Dock. The original package file and uncompressed folder can simply be deleted.

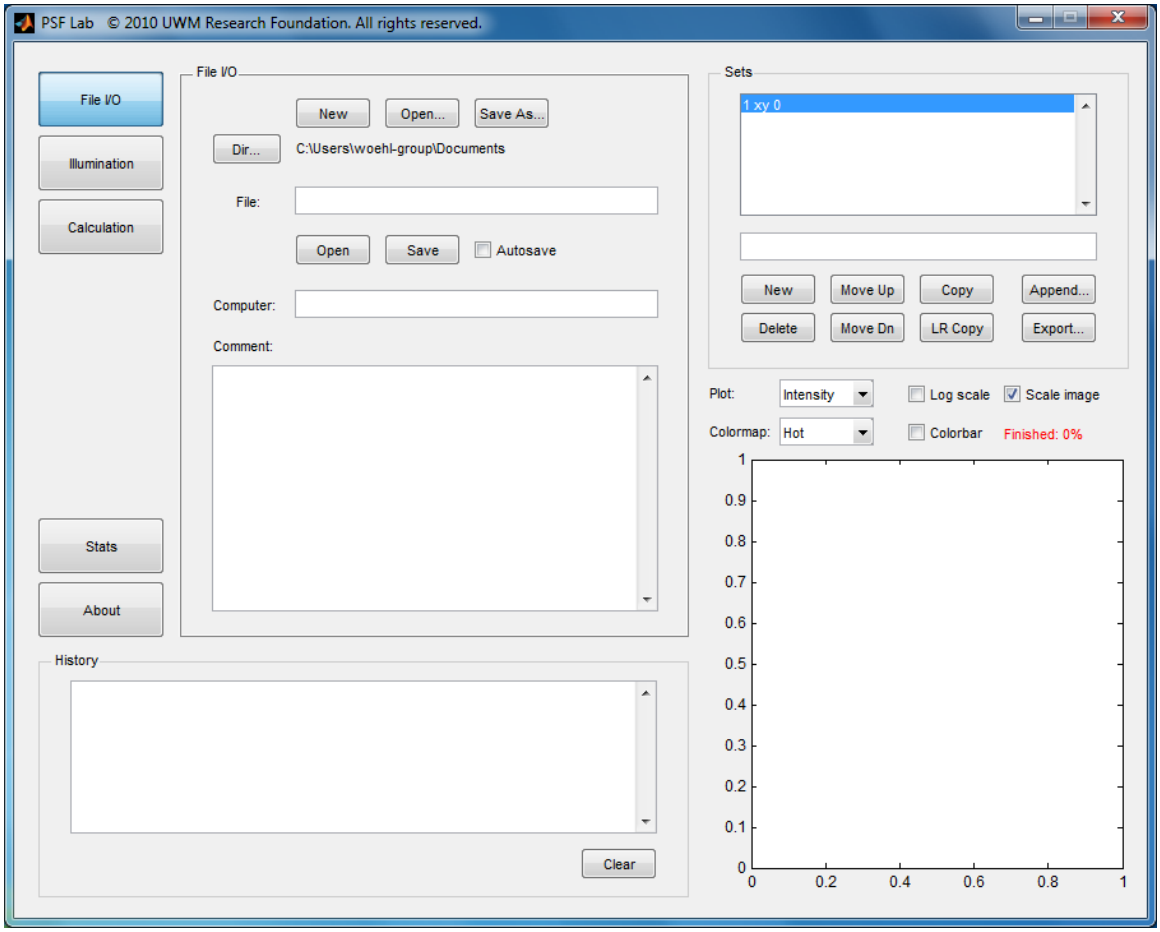
The Mac version of PSF Lab uses 1240×754 pixels on screen and should therefore run without problem all Mac desktops and notebooks. On the smallest MacBook screens, however, the lower part of the PSF Lab window may be obscured by the Dock; a workaround is to automatically hide and show the Dock (see *System Preferences – Dock*).

3. PSF Lab Quick Tour

Let us explore PSF Lab with a simple calculation in order to get a feel for the use and scope of the program. A more detailed list of all parameters and user interface elements as well as information about the internal structure of the program can be found in subsequent sections.

Upon program launch, PSF Lab performs a version check and automatically informs you when a new version is available. If you are already using the most current version or if the version check fails, the program opens directly with the main window. Shown below is the PSF Lab main window as it appears under Mac OS and under Windows.

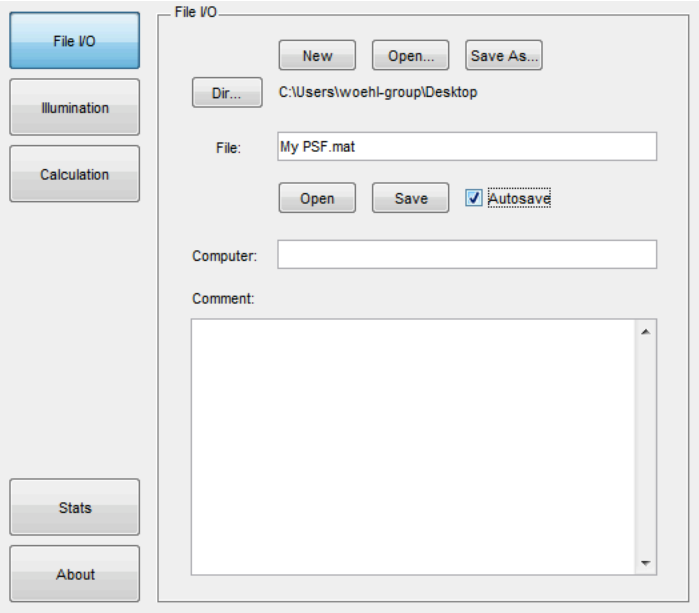

PSF Lab © 2010 UWM Research Foundation. All rights reserved.
File I/O
Illumination
Calculation
Stats
About
File I/O
New
Open...
Save As...
Dir...
/Users/woehl
File:
Open
Save
Autosave
Computer:
Comment:
History
Clear
Sets
1 xy 0
New
Move Up
Copy
Append...
Delete
Move Dn
LR Copy
Export...
Plot:
Intensity
Log scale
Scale image
Colormap:
Hot
Colorbar
Finished: 0%


PSF Lab © 2010 UWM Research Foundation. All rights reserved.
File I/O
Illumination
Calculation
Stats
About
File I/O
New
Open...
Save As...
Dir...
C:\Users\woehl-group\Documents
File:
Open
Save
Autosave
Computer:
Comment:
History
Clear
Sets
1 xy 0
New
Move Up
Copy
Append...
Delete
Move Dn
LR Copy
Export...
Plot:
Intensity
Log scale
Scale image
Colormap:
Hot
Colorbar
Finished: 0%

In the following, we will use the Windows user interface to illustrate the features of PSF Lab.

To the left of the main window are five buttons labeled **File I/O**, **Illumination**, **Calculation**, **Stats**, and **About**. Click on any of these buttons and observe how the content displayed in the panel next to it changes accordingly. Because of this behavior, we will refer to these interface elements as tabs. The three tabs on the top (**File I/O**, **Illumination**, **Calculation**) allow you to enter all the parameters needed to set up and perform a PSF calculation, while the bottom tabs (**Stats** and **About)** report information back to you.

Start with the **File I/O** tab. It is good practice to first choose a filename under which the results of your calculation will later be saved. The **Dir** field indicates the working directory; by default, it is preset to your home directory. You can change it by clicking on the **Dir…** button, which also allows you to create a new directory. For now, select your Desktop folder as your working directory.

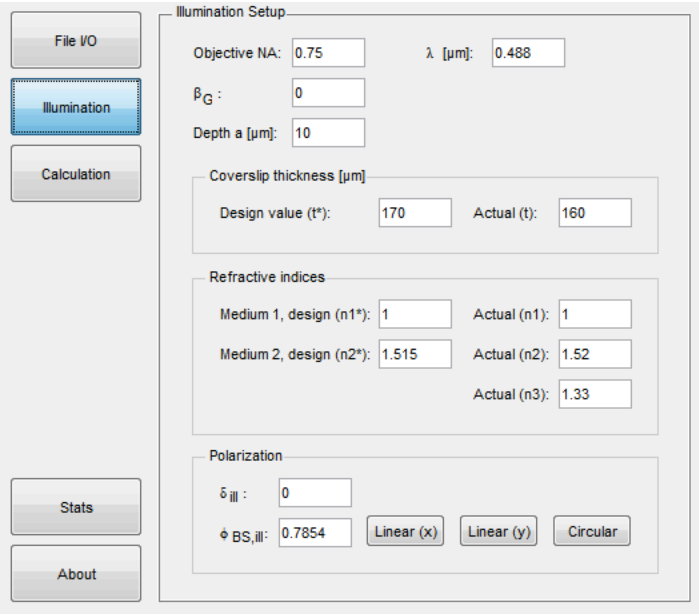


The **File** field contains the name of the PSF file under which all calculations will be saved; it is empty by default. Type in `My PSF.mat`, then click with your mouse outside the **File** field or press the <Return> key, which will enter the new value into PSF Lab (note that the file is not actually created until a save operation is performed). It is recommended that you add the extension *.mat* to all filenames to indicate the file type used for PSF Lab data (MATLAB binary file). Adding this extension is not strictly required, however; PSF Lab will read any valid PSF Lab data file even if it was saved using another extension (or no extension at all).

Finally, it is recommended that you check the **Autosave** checkbox, which will ensure that all data are automatically saved to the specified path once a calculation is finished. Enabling **Autosave** is especially useful when running multiple, successive calculations, because it keeps data loss at a minimum in case of a computer crash. Note that PSF Lab also saves data to a separate backup file at regular intervals during a calculation, but only for the calculation currently in progress.

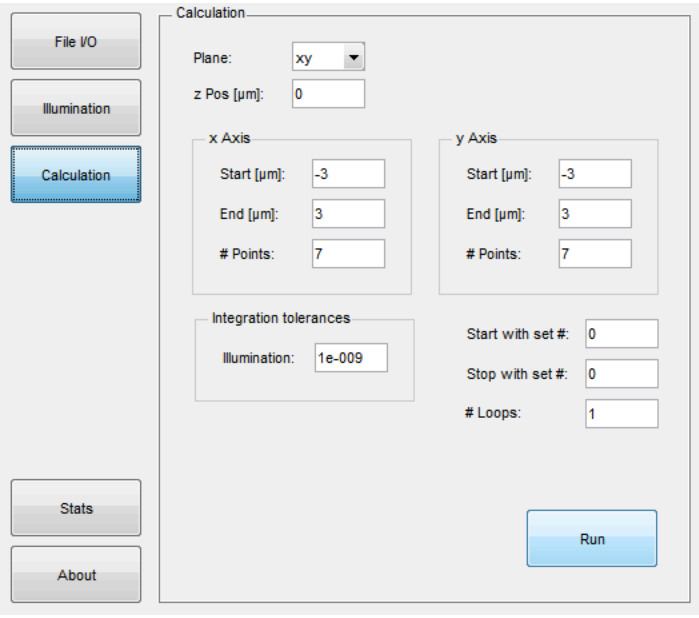
All other fields on the **File I/O** tab can for now be left empty.

Continue to the **Illumination** tab, where you enter information about the illumination optics. All fields are prefilled with example values, which you would normally need to change, but for now we'll leave them in place.



According to these parameters, the sample is illuminated with linearly polarized, 488 nm light using an air objective ($n_1^* = 1$) with a numerical aperture of 0.75. The back aperture of the microscope objective is illuminated by a collimated beam with constant intensity profile ($\beta_G = 0$). The goal of every objective manufacturer is to produce objectives that transform a (hypothetical) plane wave into a "near-perfect" truncated, spherical wave that propagates inside the coverslip towards the sample medium, assuming that a coverslip with a certain thickness and refractive index (RI) is used. This scenario, which we refer to as the design case, produces the optimal focus when the objective is positioned at a certain distance from the coverslip/sample interface. In our case, the design case calls for 170 µm thick coverslips with an RI of 1.515 at the illumination wavelength, but the coverslip actually used is 10 µm thinner and has a slightly higher RI of 1.52. Also, the microscope objective is moved 10 µm closer to the coverslip/sample interface (depth a) than in the design case. The sample medium is water ($n_3 = 1.33$).

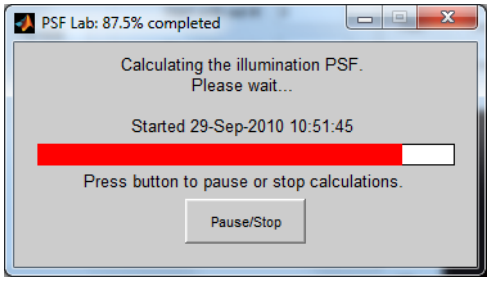
Next, open the **Calculation** tab in order to set up the calculation itself; again, all fields are prefilled with example values. PSF Lab only calculates two-dimensional slices of the illumination PSF (the three-dimensional PSF can be obtained by stacking a set of slices, but this functionality is not currently provided by PSF Lab). The z axis is the optical axis and points in the direction of light propagation, while x and y are defined by the polarization of the illumination light. The origin of this coordinate system



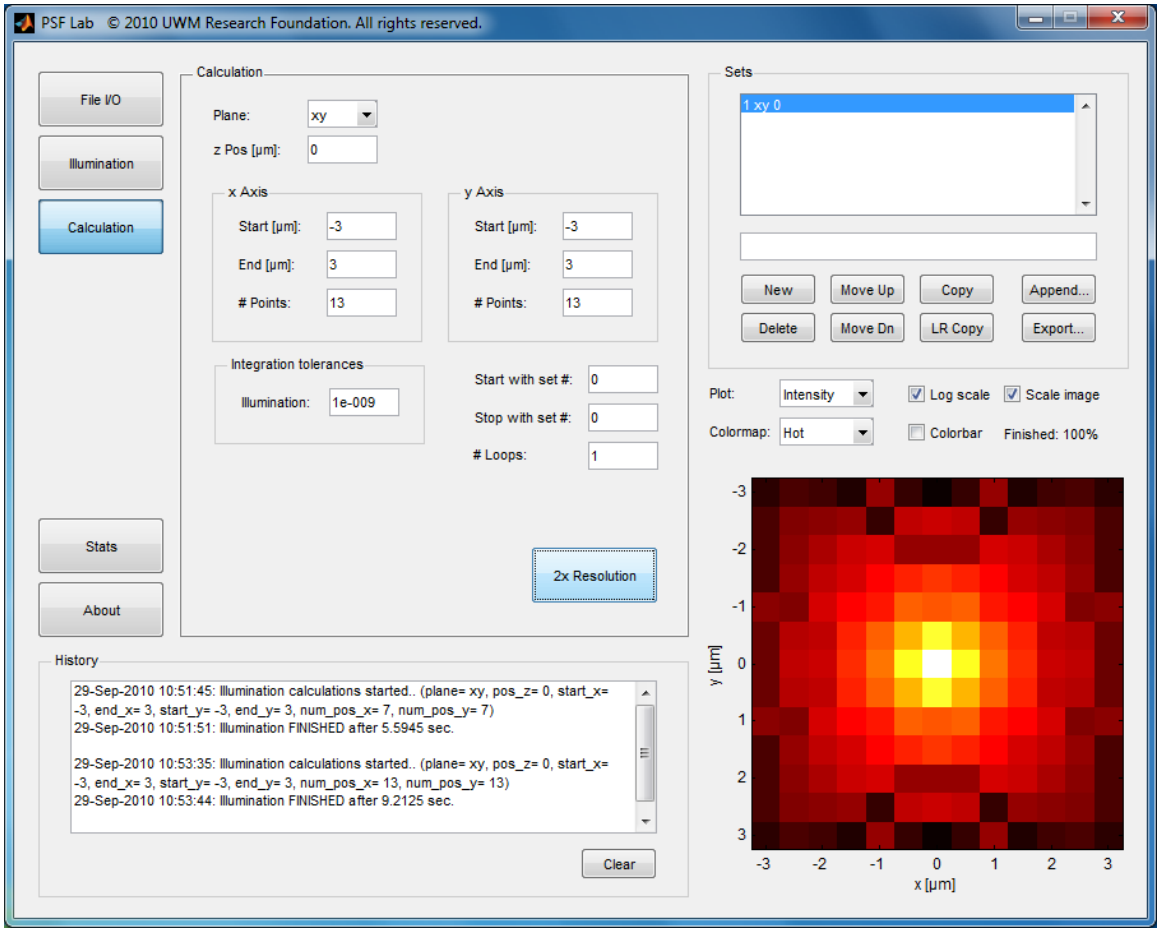
is the corrected Gaussian focus (see *J. Opt. Soc. Am. A* **27**, 295 (2010) for more details), which means that the sample/coverslip interface is always located at $z = -a$.

According to the example parameters, the PSF will be calculated in a plane perpendicular to the optical axis (*xy* plane) at $z = 0$, *i.e.*, 10 μm away from the coverslip/sample interface inside the sample medium. The calculation will be carried out for a 7×7 grid of points spanning the range of -3 to +3 μm along both the *x* and *y* axes. The integration tolerance is preset to a sensible value, which generally ensures the absence of numerical artifacts while keeping the computational time at a reasonable level. Leave the **Start with set #**, **Stop with set #**, and **# Loops** values as is, and click on **Run** to start the calculation.

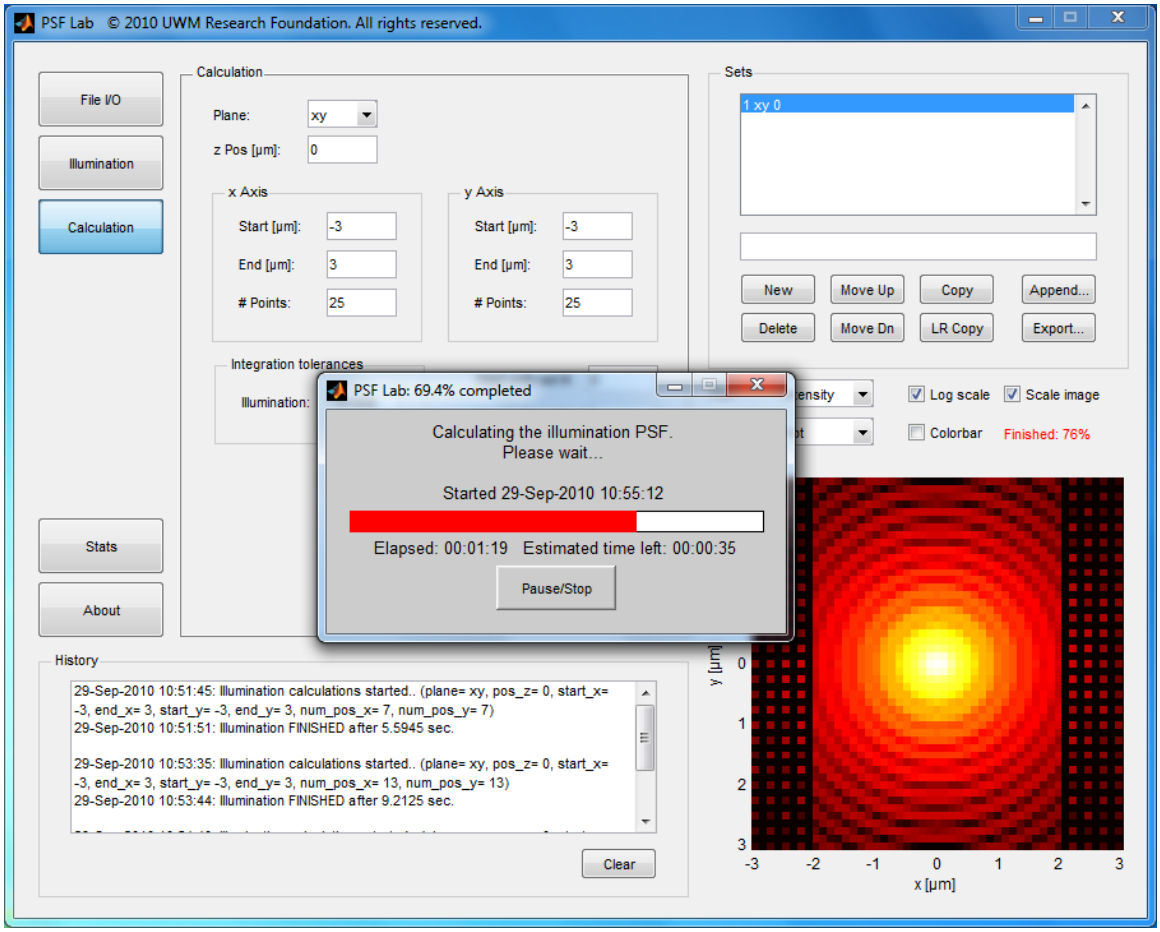
While the calculation is running, user interaction with PSF Lab is limited to a progress bar window, which has a **Pause/Stop** button and indicates the estimated amount of time remaining to finish the current calculation. The PSF image to the lower right of the screen is continually updated as the calculation proceeds. Once the calculation is finished, all PSF Lab data are saved to the file specified on the **File I/O** tab if **Autosave** is enabled, and the **History** panel, located below the tab panel, displays the start and stop times and some basic information about the calculation.



The resulting 7×7 image has a very high contrast, and more features can be seen when a logarithmic color scale is used (check the **Log scale** checkbox). However, it would be nice to obtain a higher resolution image, preferably without loosing the data that have already been calculated. This can easily be done in PSF Lab. Notice that the **Run** button has changed to **2x Resolution**. Click on it to launch a new calculation that doubles the number of grid lines. A dialog box appears to inform you that the existing *My PSF.mat* file will be overwritten without warning during the calculation because Autosave is enabled; click on **Continue** to confirm this choice (this will happen with every new calculation). The resulting 13×13 image is shown below:

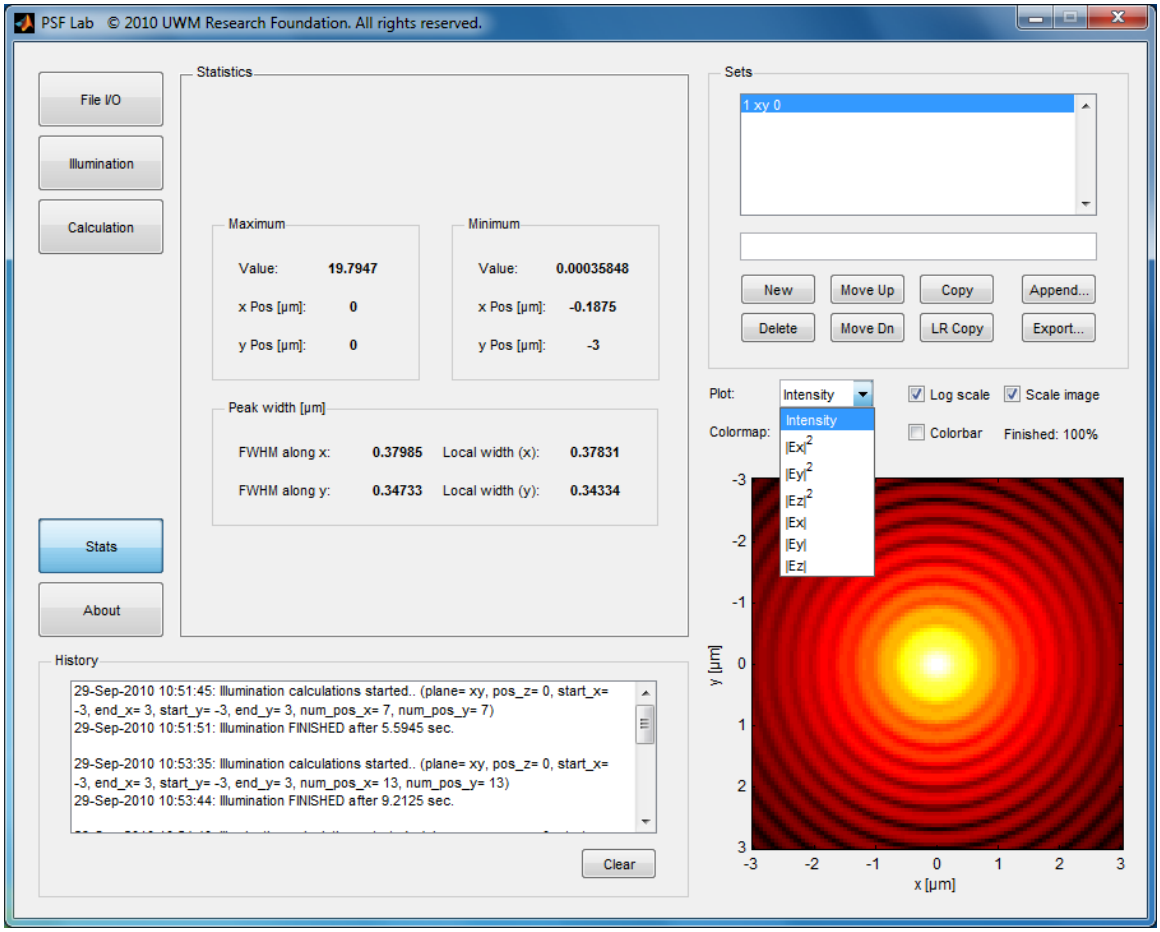


Instead of launching successive calculations manually, however, we can use automatic loops to make this process more convenient. For example, we can set up three successive calculations by entering **3** in the **# Loops** field. Click the **2x Resolution** button to launch the calculation. The first loop will double the resolution, using the existing 13×13 image to calculate a 25×25 image, the second loop will lead to a 49×49 image, and the last loop will produce the final 97×97 image (note that while the resolution doubles, the computational time approximately quadruples with each iteration). The **# Points** fields are automatically updated with the new values at the end of each loop. You may notice that the progress bar window indicates a *% complete* value that is different from what is displayed by the **Finished** field located just above the PSF image; the former only considers the new points that need to be calculated, while the latter indicates how much of the total image is finished (including points from previous calculations). In the case shown below, for example, 76% of the 49×49 image is completed, which includes the quarter of the image from the previous 25×25 calculation; more than half of the new points (69.4%) have already been calculated.

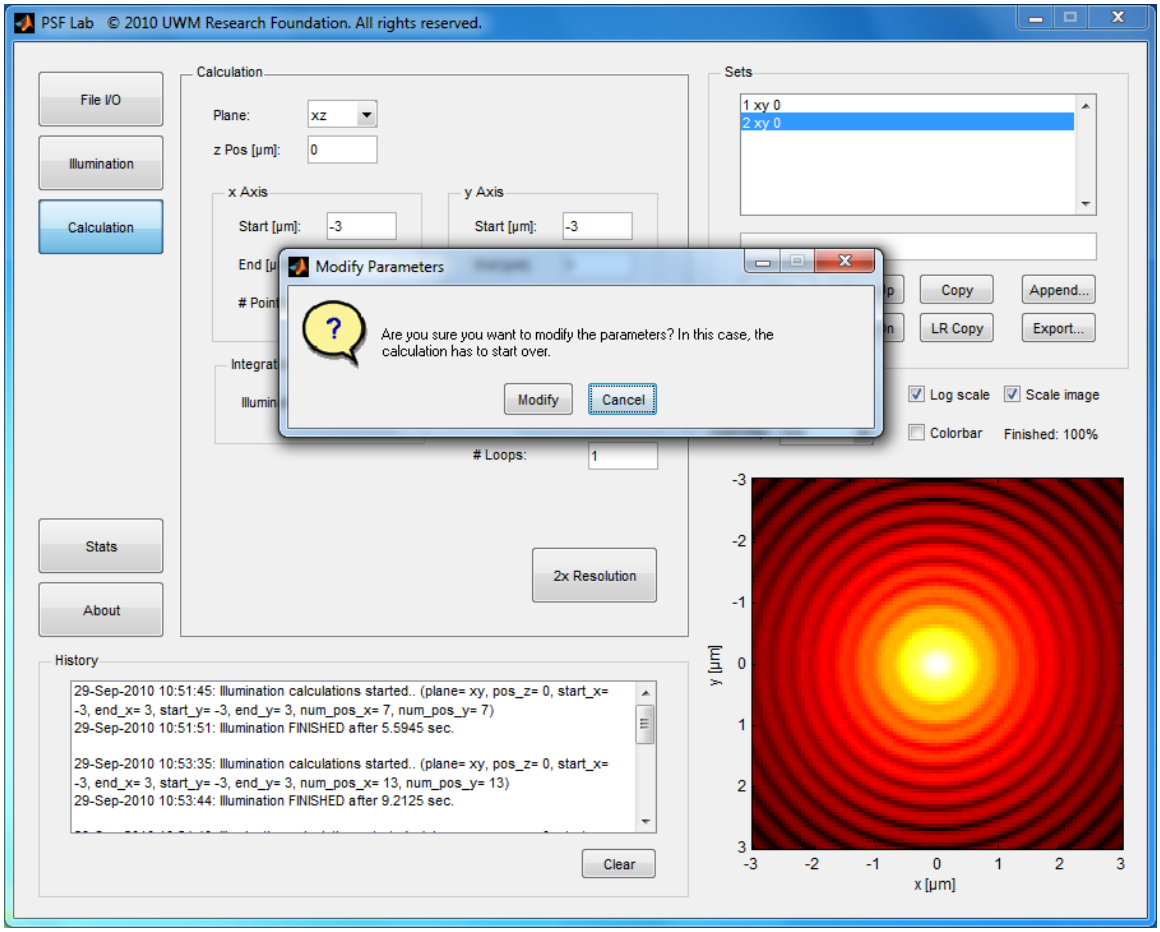


As the resolution of the PSF intensity plot increases, the diffraction rings become more and more pronounced. The **Stats** panel provides some basic statistical information about the PSF plot, such as global maximum and minimum as well as the width of the main peak along each axis. According to these data, the full width at half maximum (FWHM) of the illumination PSF along *x* and *y* is about 0.38 µm and 0.35 µm, respectively, which shows that linearly polarized light leads indeed to a small asymmetry in the PSF. The local width is taken at a value halfway between the global maximum and the first local minimum along each axis. Since the first local minimum has a nonzero intensity value, the local peak width is always smaller than the corresponding FWHM (although the difference is negligible here).

All statistical data depend on and change with the selected data type, which can be chosen using the **Plot** popup menu. The choices are: field intensity, magnitude of each field component, or squared field components.

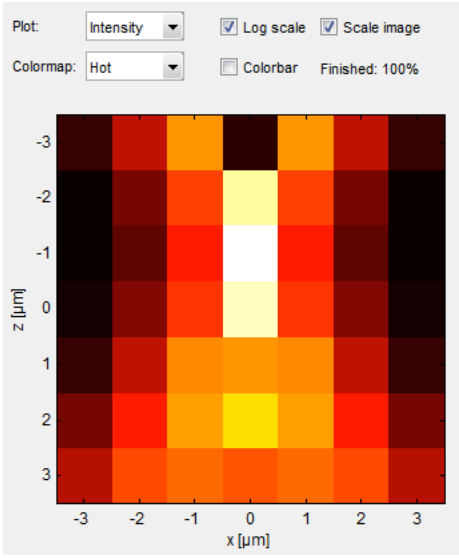


At last, we will perform a calculation along the optical axis in order to determine the extent of the focal spot in the *z* direction. PSF Lab provides a simple means of achieving this without having to reenter all parameters for the illumination optics: just click on **Copy** in the **Sets** panel (top right). This creates an exact duplicate of the first calculation, which will now be part of the same file, *MyPSF.mat*. Then, open the **Calculation** tab and select the *xz* plane for calculation. A dialog box asks you to confirm whether you really want to modify the parameters, because all calculated data (for this set) will need to be discarded. Confirm your choice by clicking on **Modify**.



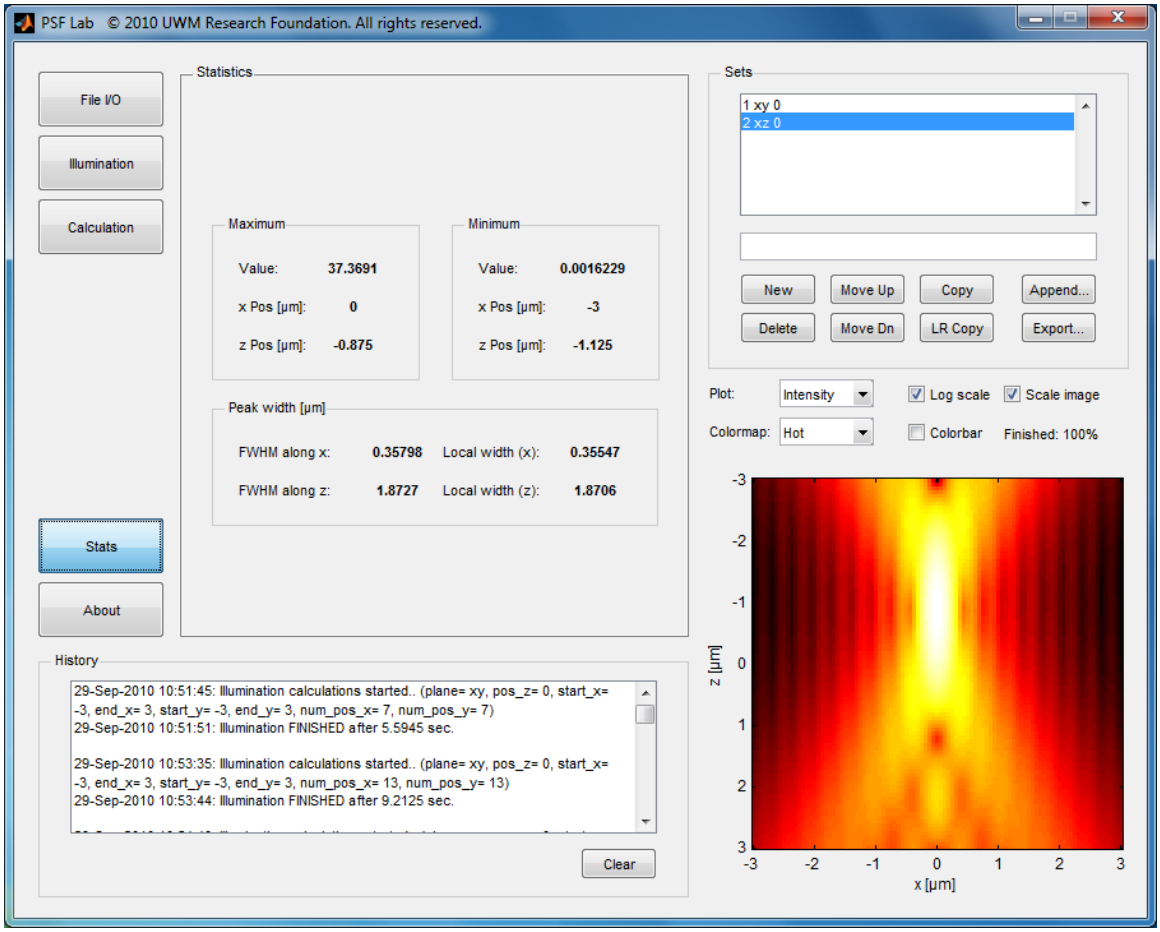
When you take a look at the **Sets** panel, you'll notice that the new set (#2) is now listed as a *xz* calculation at $y = 0$. Additional descriptive information can be added using the text entry field just below the sets list, which makes it easier to distinguish different sets in a longer list.

In order to get a quick overview of the location of the focus, it is recommended to decrease the number of points again to 7×7. Then click on **Run**. The resulting *xz* intensity plot reveals that the focus is located at about $z = -1$ µm, well within the selected *z* range. We can now increase the resolution by setting up *4* loops and clicking on **2x Resolution**. The resulting 97×97 intensity plot nicely shows the diffraction flares around the focal point (again, this is best seen if **Log scale** is selected).



The exact location and axial extension of the focal spot can be determined by opening the **Stats** tab. According to these data, the focal spot is located at

z = -0.875 µm (*i.e.*, 9.125 µm away from the sample/coverslip interface, which – as you will recall – is located at $z = -a$ = -10 µm). This shift of focus relative to the design case is due to a combination of effects. A 10 µm thinner coverslip, if used with an air objective, shifts the focal spot inside the coverslip medium, which is to a large extent counterbalanced by moving the objective 10 µm closer to the sample/coverslip interface – indeed, if the coverslip and sample medium RIs were equal to the coverslip's design value, these effects would exactly cancel out, positioning the focus at 10 µm from the sample/coverslip interface. The final location of the focus is therefore due to the difference between the actual values for coverslip/sample medium RIs and the coverslip design RI.



The extension of the focal spot in the z direction is about 1.9 µm, more than five times larger than its lateral extension (along x) of about 0.36 µm. Note that the value for the width along x is somewhat smaller than the 0.38 µm that we had calculated for the xy image. At closer inspection this is not surprising, however, since the xy calculation was carried out at z = 0, which is 0.875 µm away from the true focal spot.

A sample file with PSF data for different optical setups can be found on the One Molecule Group website. We will use screenshots from this data file in the next sections,

where more detailed information about all parameters and user interface elements will be provided.

4. User interface elements

PSF Lab parses all *numerical* entry fields according to normal MATLAB rules. This means that you can use mathematical expressions in these fields, so instead of entering the approximate value of **`1.5708`**, you can also use **`pi/2`**. More complicated expressions are also allowed, such as **`atan(pi/2)*4`** etc.

Many parameters can only take on positive values. If negative values are entered in such a field, the sign is simply ignored.

Program Start

During startup, PSF Lab performs a version check using in Internet connection and notifies the user via dialog box if a newer version is available; no information whatsoever is transmitted from your computer to the server during that process. For this to work, outgoing traffic must be allowed by your firewall (which is the default setting of most firewalls). If you are already using the most current version, or if the version check fails for any reason, the program opens directly with the main window.

The **New PSF Lab version** dialog has a **Do not show this dialog again** option that can be checked to prevent the dialog box from appearing during every program launch. Selecting this silencing option creates a file named *PSF_Lab_options.txt* on the system; simply deleting this file will restore the default behavior.

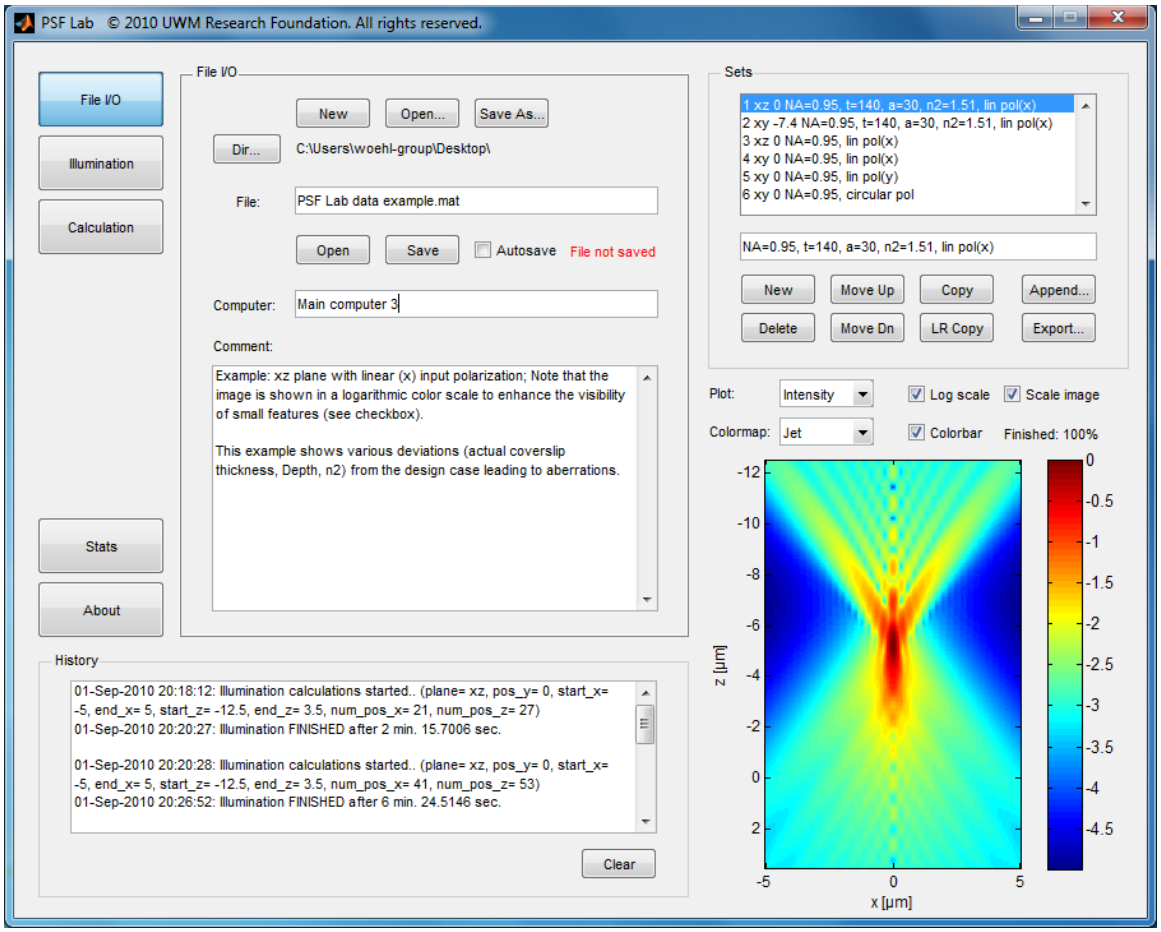
File I/O tab

A PSF Lab data file consists of one or more *sets* displayed in the **Sets** panel; each set corresponds to a single PSF calculation with its own associated parameters, such as refractive indices, cover slip thickness, etc. Thus, several calculations can be conveniently kept together in one file, for example in order to create a movie consisting of several image frames with a single varying parameter. This also makes it easy to quickly compare several PSF images without opening another file.

Note that as soon as PSF Lab detects a modification to any set in a previously saved file, a red indicator text “File not saved” appears next to the **Autosave** checkbox.

New: clears all data from memory and opens a new PSF Lab data file. All parameters except for the working directory are reset to their default values (those present after program launch). If any unsaved data exist, the user is given the option to save the current PSF Lab data file before proceeding.

Open…: opens a dialog box for reading a PSF Lab data file. By default, only files with the extension *.mat* can be selected, but this can be changed by setting the file format/type to **All Files**.



Save As…: opens a dialog box for saving the current PSF Lab data file. By default, the extension *.mat* will be added, but this can be changed by selecting **All Files**.
PSF Lab natively stores its data in binary form in a structure array named "data", with each set stored in a sub-structure array. PSF Lab data files are compatible with MATLAB 7.0 (R14) and higher. If you have MATLAB installed on your system, you can open these files directly, but it may be easier to work with a *.mat* file created by the **Export** function (see **Sets** panel for more details).

Dir…: the working directory for the current PSF Lab session. All dialog boxes use this directory as the default directory. After program launch, the working directory is preset to the user's home directory, which can be changed by clicking on the **Dir…** button. The resulting dialog box also allows the creation of a new directory. The **Open…** and **Save As…** commands will change the working directory according to the selected file.

File: the filename of the PSF Lab data file in the working directory (not that specifying the filename in this field will not automatically save it). Saving your data to a new file is a simple as typing in the new filename (or changing the existing filename, *e.g.*, from **`My PSF data file 1.mat`** to **`My PSF data file 2.mat`**) and clicking on **Save**. Note that it is recommended (but not strictly required) to add the extension *.mat* to all filenames.

Open: opens the PSF Lab data file specified by **File** in the working directory. If the file doesn't exist, an Open dialog box will appear.

Save: saves all PSF Lab data sets to **File** in the working directory. If the file does not yet exist, a new file is created; otherwise, the user is asked if the existing file can be replaced.

Autosave: if enabled, all PSF Lab data sets are automatically saved to **File** in the working directory after a calculation is finished; any existing file will be overwritten without warning (the user is asked to acknowledge this in a dialog box before the calculation is actually started).
Enabling Autosave is especially useful when running calculations involving multiple sets, because it keeps data loss at a minimum in case of a computer crash. PSF Lab also saves data (for the currently calculating set *only*) at regular intervals *during* a calculation to a separate backup file called *temp_ill.mat*, located in the working directory. This set can be loaded into PSF Lab, just like any regular PSF Lab data file.

Computer: a user-editable field (saved with each set) that can be used to help distinguish between several computers on which PSF calculations are run; it does not influence the function of the program in any way.

Comment: a user-editable field for various notes that are saved with each set; it does not influence the function of the program in any way. In older versions of PSF Lab (which use data structure version 1.5 and earlier), two comment fields were stored with each set. For backwards compatibility, loading such a file leads to concatenation of the two comment fields, separated by the single line
`---------- Comment 2: ----------`

Illumination tab

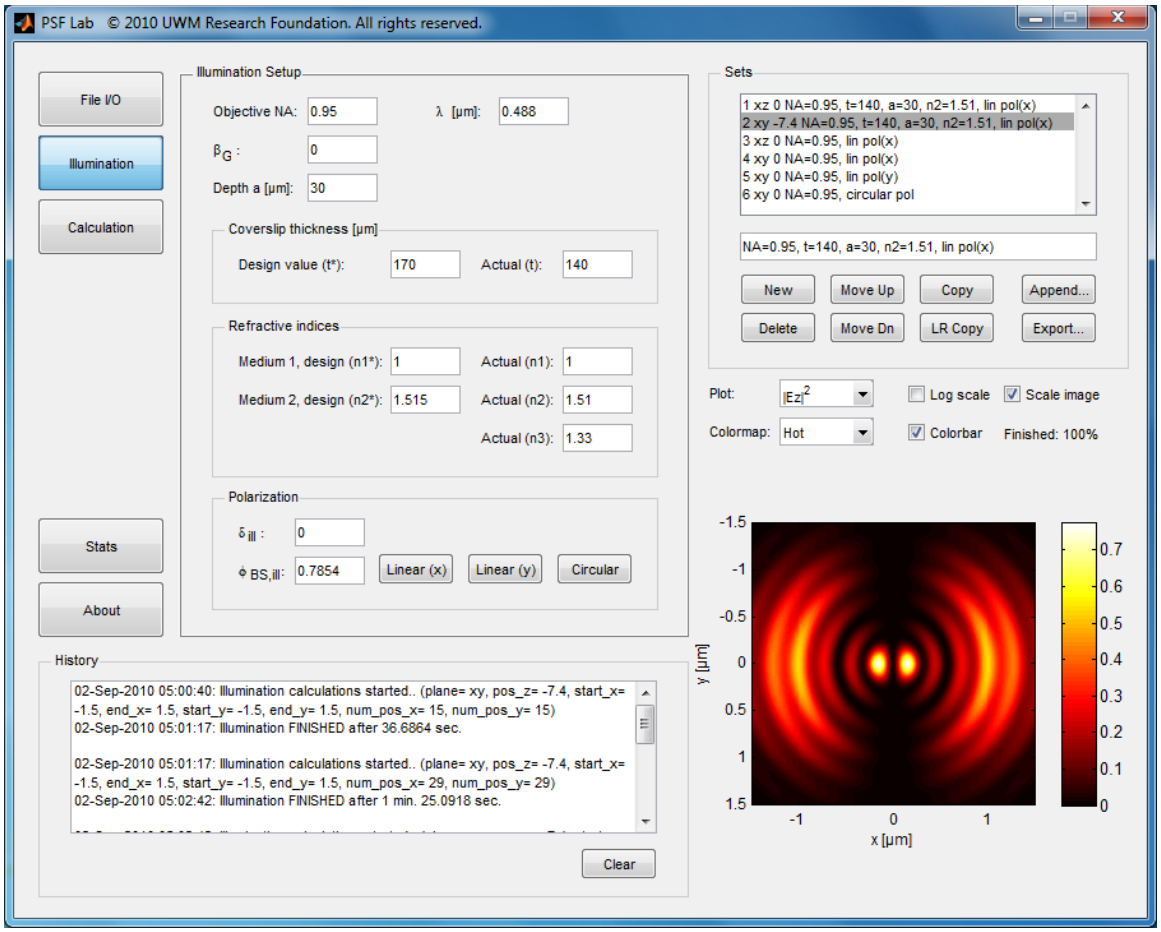
This panel contains all the parameters that are needed to specify the illumination chain.

Objective NA: numerical aperture of the lens or microscope objective.

λ [µm]: wavelength of the (monochromatic) illumination light, expressed in units of µm.

β_G: a floating point number indicating the fill factor of the Gaussian illumination beam, *i.e.*, the ratio of objective aperture radius to the beam waist radius. A fill factor of **`0`** corresponds to a flat (constant) beam profile.

Depth a [µm]: value by which the distance between objective and coverslip/sample interface is reduced relative to the design case (illumination light focused on coverslip/sample interface through an immersion medium of RI n_1^* and coverslip with RI n_2^* and thickness t^*). Note that "depth" is not a very adequate description of this parameter, as it is in general not exactly equal to the depth at which the location of the focus is found inside the sample medium (only in the design case, and only when the sample medium RI is the same as the immersion medium RI will the focus be at a distance *a* inside the sample medium).



Coverslip thickness [µm]: design value, t^*, and actual (= real) thickness, t, of the coverslip that supports the sample medium. Units: µm.

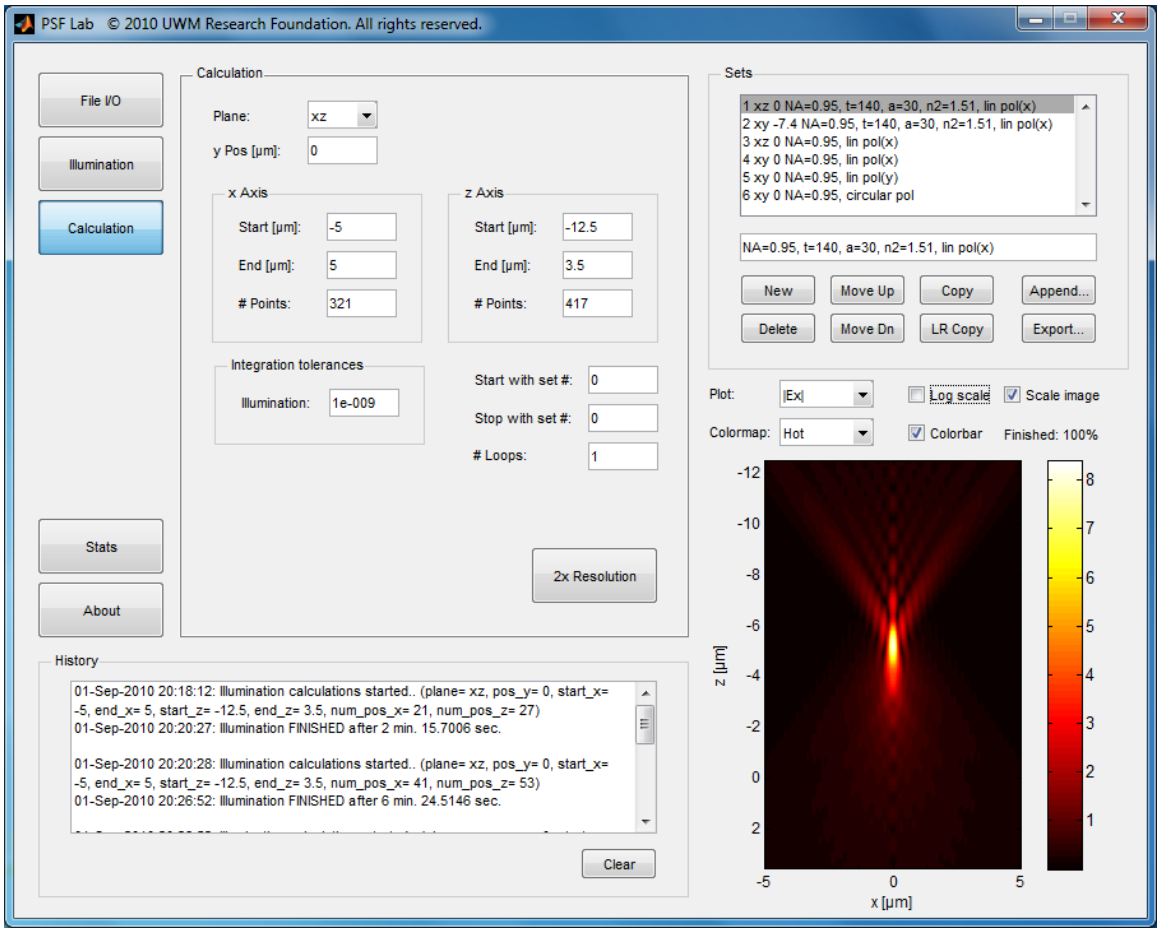
Refractive indices: design values and actual (or real) values for the RIs at the illumination wavelength of the different stratified media that the illumination light traverses from the objective to the sample.
Medium 1: immersion medium between microscope objective and coverslip (RIs n_1^* and n_1, respectively).
Medium 2: coverslip (RIs n_2^* and n_2, respectively)
Medium 3: sample (actual value n_3; a design value for n_3 does not exist because the focus is exactly at the coverslip/sample interface in the design case, and sample medium properties do therefore not affect its position).

Polarization: parameters defining the polarization state (linear, circular, or elliptical) of the illumination light. It is imagined that the polarization is changed by a Babinet-Soleil compensator with angles δ_{ill} and $\phi_{BS,ill}$ (both in radians), which acts as a variable waveplate. For convenience, the correct angles that produce linearly polarized light along the x or y axis as well as circularly polarized light can be easily preset using the **Linear (x)**, **Linear (y)**, and **Circular** pushbuttons.
Note that PSF Lab's parsing feature, described in the beginning of this section, is especially useful here since it allows the use of expressions like `pi/3` etc.

Calculation tab

PSF Lab uses a right-handed Cartesian coordinate system where the *z* axis is aligned with the optical axis, pointing in the direction of light propagation. The origin of this coordinate system is the corrected Gaussian focus, which has a fixed physical distance from the microscope objective, independent of any other parameters such as actual thickness or position of the coverslip (see *J. Opt. Soc. Am. A* **27**, 295 (2010) for more details). The *x* and *y* axes are defined by the polarization of the illumination light and are equivalent to each other if circularly polarized light is used.

At this point in time, PSF Lab is only a single threaded application, but several instances of PSF Lab can be run simultaneously on a multi-core machine to decrease the computational time. Under Windows, this is easily achieved by double-clicking the PSF Lab executable multiple times, whereas under Mac OS, several copies of PSF Lab need to be created first, which can then be launched one after another. It is recommended to use separate working directories in order to avoid that the PSF Lab instances overwrite each other's backup files (*temp_ill.mat*). Using different filenames is also recommended for identifying them more easily. After all calculations are finished, the **Append** button in the **Sets** panel can be used to merge the different calculations into a single file if this is desired.

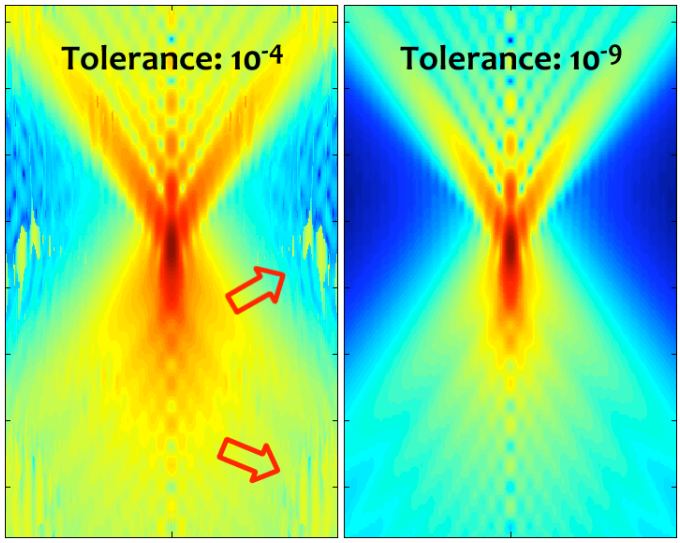


Plane: the plane (*xy*, *xz*, or *yz*) in which the electric field is to be calculated. PSF Lab only calculates two-dimensional slices of the illumination PSF; the three-dimensional PSF can be obtained by stacking a set of slices, but this functionality is currently not provided by PSF Lab.

***x*/*y*/*z* Pos:** the position (in µm) of the plane along the third axis.

***x*/*y*/*z* Axis:** these panels define the exact in-plane coordinates of the grid points at which the electric field is to be calculated (PSF Lab uses a rectangular grid). **Start** and **End** define the position (in µm) of the first and last point, respectively, along each of the two axes, whereas **# Points** defines the number of equally spaced points between these limits (including the limits). **# Points** is always rounded to the next integer value; the smallest allowed number is 1.
For example, if **Start** is -5, **End** is 5, and **# Points** is 5, the electric field will be calculated at the positions -5, -2.5, 0, 2.5, and 5 µm.
Note that for *x* and *y*, the **Start** and **End** values have to be symmetric with respect to the origin; if not, the calculation will be carried out using the greater (absolute) value of the two (the fields are updated to reflect this after the calculation is finished). In addition, for the *x* and *y* direction the **# Points** field must contain an odd number because the central grid point is always located at *x*/*y* = 0; if an even number is entered, it will be reset to the next smaller odd number. The **Start** and **End** values defining the *z* range do not need to be symmetric and can have be even or odd.

Integration tolerances: This allows the user to set the (absolute) error tolerance level of MATLAB's "quad" command for numerical integration (adaptive Simpson quadrature), which is at the heart of the calculations. A tighter tolerance (smaller value) leads to more precise numerical integrations, but at the cost of longer calculation times. The default value in PSF Lab is now 10^{-9}, which can be entered manually as `1e-9` or using the text `default`. For backwards compatibility with previous versions of PSF Lab, entering `0` defaults to a tolerance of 10^{-6} (the default value used by MATLAB's "quad" command). The figure on the right shows what happens if the tolerance is not tight enough. The difference is not perceptible when a linear color scale is used, but shows up with a logarithmic color scale, which enhances very small differences. It can be seen that a loose tolerance level leads to image artifacts, most notably an artificial graininess of the image (especially visibile in the areas marked by arrows) and decreased image contrast, which can be avoided by using the



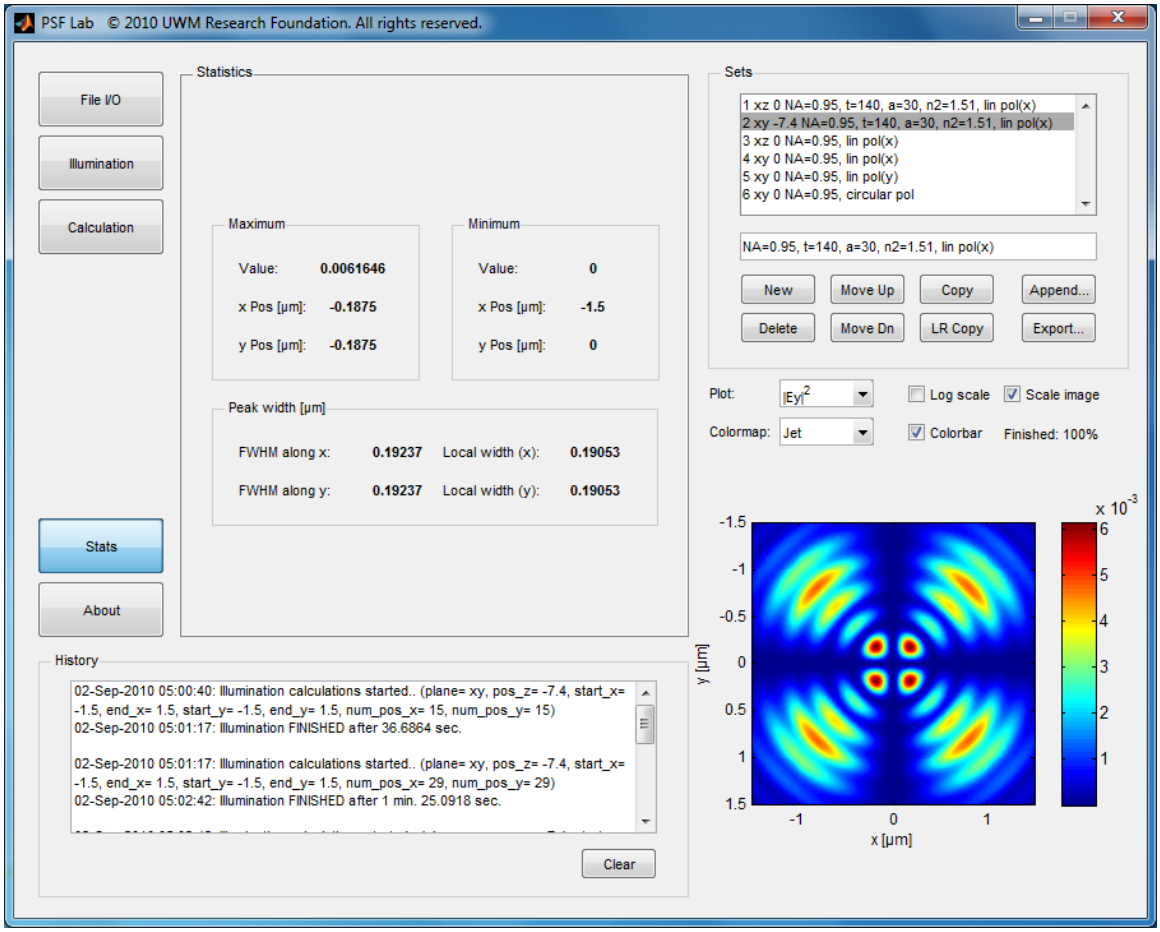
default tolerance level of 10^{-9}. Note that that this mostly affects areas farther away from the central axis (where the field intensities are small), so that smaller tolerances may have to be chosen if these areas are of interest. However, it appears that the "squad" function becomes unstable if exceedingly small tolerances (such as 10^{-15}) are used.

Start with set #/Stop with set #/# Loops: may be used to set up automatic loops in order to facilitate longer or successive calculations. In this mode, PSF Lab starts with the set specified by **Start with set #** and successively doubles the image resolution in a certain number of calculations (given by **# Loops**). Once this is achieved, the same happens with the subsequent set, and so on until all calculation for the last set (given by **Stop with set #**) are finished. Since the image is regularly updated during each of these calculations, the user can get an impression of how the image improves as the resolution is increased. **Start with set #** and **Stop with set #** values of **`0`** calculate the current set only. The value of **# Loops** has to be at least **`1`**.

Run/Resume/2x Resolution/Run Sets: clicking on this button starts the PSF calculation using the current parameters. While the calculation is running, user interaction with PSF Lab is limited to a progress bar window, which has a **Pause/Stop** button and indicates the estimated amount of time remaining to finish the current calculation (not including any subsequent loops) and the percentage of new points that are already calculated. Clicking on **Pause/Stop** allows the user to temporarily pause or definitely interrupt the calculation (this does not discard any points already calculated).

Depending on the context, the **Run** button changes its label to better indicate the action that is going to be performed. For new calculations on a single set, this button is simply labeled **Run**, but changes to **2x Resolution** once the calculation is entirely finished. If the user interrupts a calculation in progress, the label changes to **Resume**, because the incomplete image will first be completed when this button is pressed; only subsequent calculations will then increase the resolution. If a range of sets is specified, the button is labeled **Run Sets**.

Note: for the automatic loop feature, resuming the calculation of an incomplete image counts as one loop.



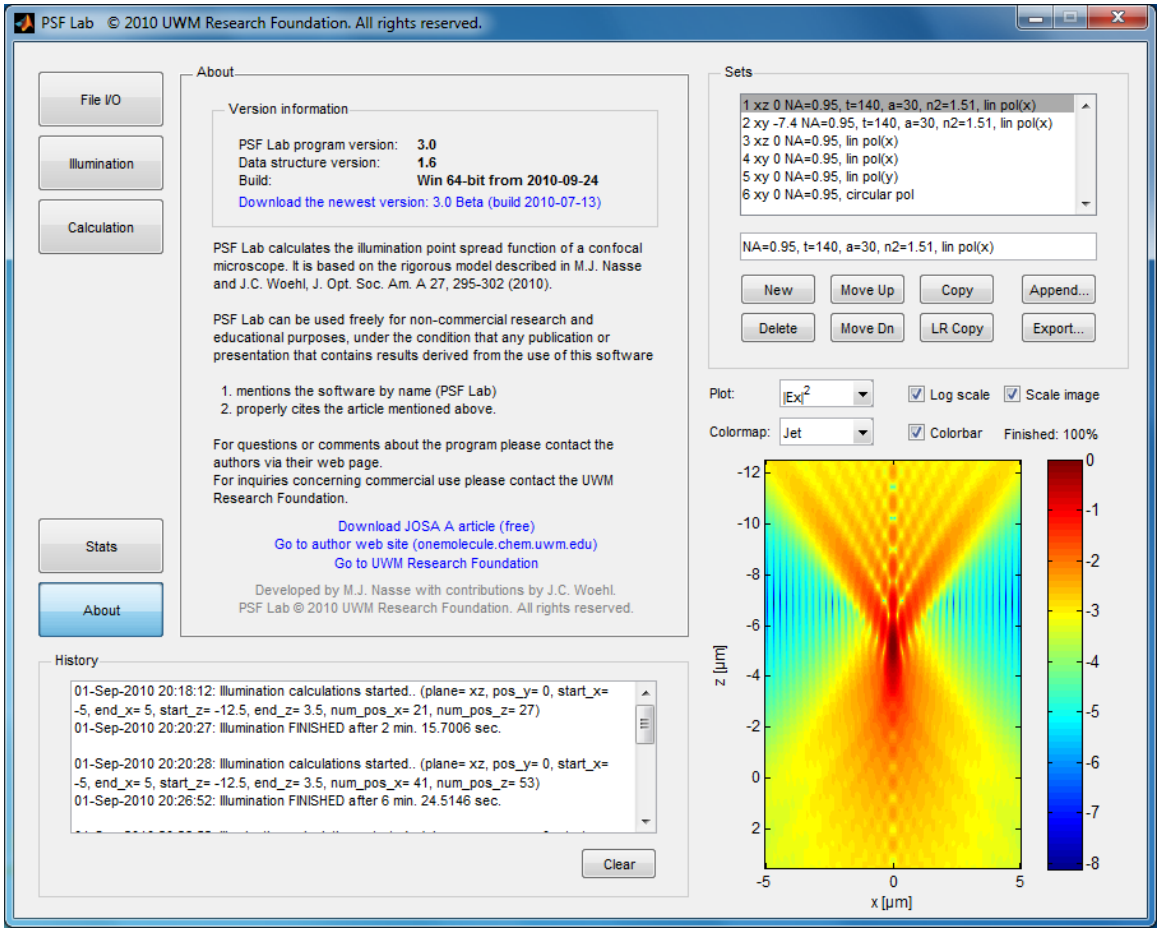
Stats tab

The **Stats** panel (see previous page) displays various statistical properties of the currently displayed PSF image. Note that the displayed information also depends on the type of data that is displayed (which can be selected using the **Plot** popup menu; see "Plot area" section).

Maximum: information about the value and in-plane position (coordinates in µm) of the global maximum of the displayed data type. Only points within the image area are considered; the maximum always falls on a grid point (*i.e.*, no interpolation occurs).

Minimum: information about the value and in-plane position (coordinates in µm) of the global minimum of the displayed data type. Only points within the image area are considered; the minimum always falls on a grid point (*i.e.*, no interpolation occurs).

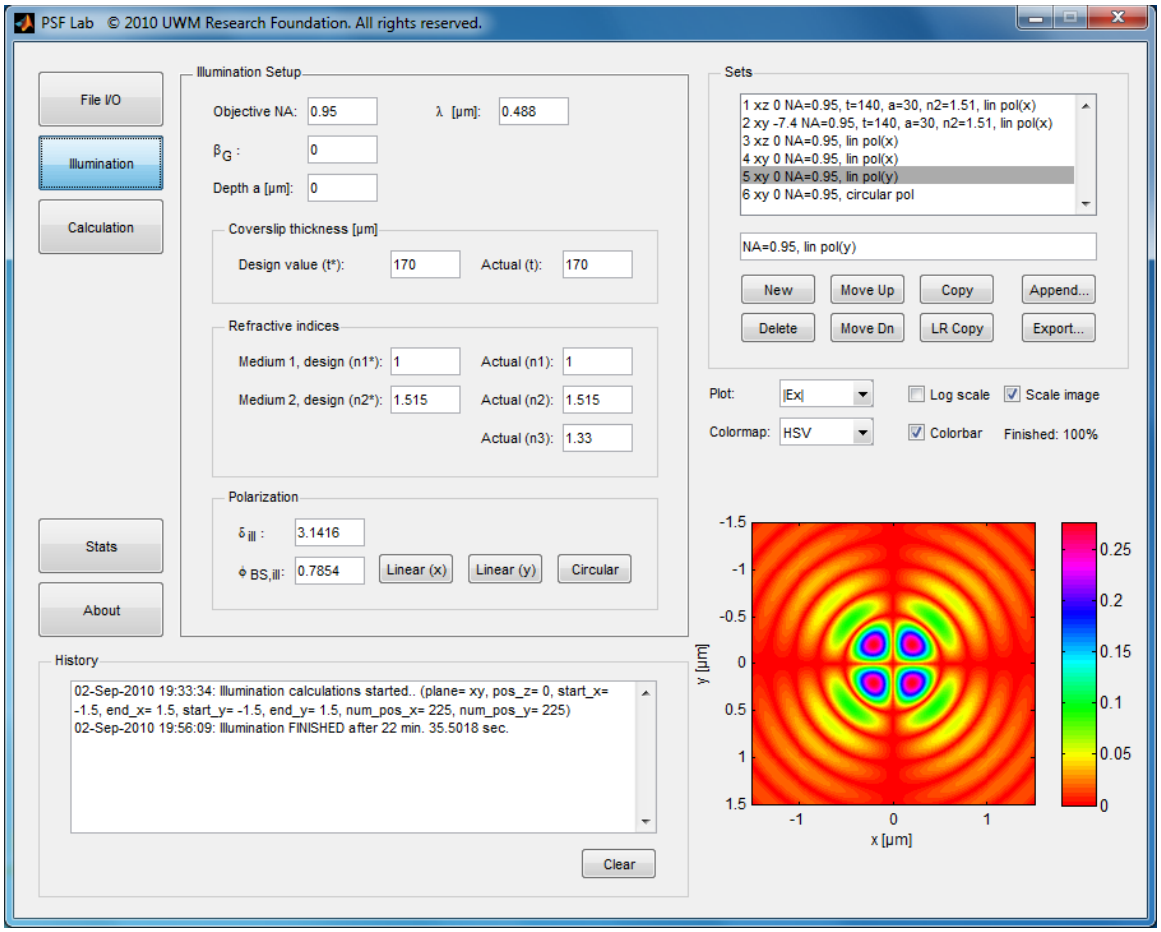
Peak width [µm]: width of the main peak (global maximum) along each of the axes spanning the plane of calculation. The full width at half maximum (FWHM) along each axis is taken at a value of half the global maximum, while the corresponding local width is taken at a value halfway between the global maximum and the first local minimum along that axis.



About tab

This panel (see previous page) displays information about the PSF Lab program version, the data structure version, and the build version (platform and date). If a newer program version is available, a clickable link to the download web page appears underneath the version information. If no update is available, or if for any reason new version information could not be obtained (*e.g.*, because no Internet connection was available or because of any other error during version check), the link will be substituted by this information.

The version information box is followed by the conditions of use of PSF Lab and general information concerning the program, followed by some useful links to program-related websites.



Sets panel

The listbox in the **Sets** panel shows all the sets in the current file; the current set is highlighted. By default, each set is referenced by its running number (indicating its position in the list), the plane of calculation, and the position of the plane along the third axis. Other descriptive text can be freely added using the text box just below the list. For example, **`2 xy -7.4 NA=0.95, n2=1.5`** refers to set #2, which is a calculation of the electric field in the *xy* plane at z = -7.4 µm; the descriptive text "NA=0.95, n2=1.5" was added by the user.

Selecting a different set in the **Sets** panel updates all parameters in PSF Lab accordingly (tab panels, **History** panel, and **Plot** panel).

New: creates a new set at the end of the list. Default values are used for all parameters.

Delete: deletes the selected set from the list. The deleted set is not removed from the PSF Lab data file until the file is actually saved.

Move Up/Dn: moves the selected set up/down in the list. Rearranging the order of sets can be useful for setting up successive calculations on multiple sets with the automatic loop feature (see **Calculations** tab).

Copy: creates an exact duplicate of the selected set at the end of the list. This is convenient for setting up multiple calculations with slightly varying parameters.

LR Copy: creates a duplicate of the selected set, but at about half the resolution (LR = lower resolution). This feature can be used when the user has interrupted a PSF calculation and wants to revert back to the original data set (instead of displaying a partially finished image). Note that a LR Copy can only be performed if the current number of points along *x*/*y* can be expressed as $2n+1$ (n = 2, 3, 4, …) because the central grid point is always located at *x*/*y* = 0; the current number of points along *z* needs to be of the form $n+1$ (n = 2, 3, 4, …).

Append…: loads all sets from another PSF Lab data file and adds them to the end of the current list. This allows the user to merge different PSF Lab data files (including the backup file *temp_ill.mat*, which contains the last set that was saved to disk during a calculation).
The **Append** functionality does not work with PSF Lab data files that use an older data structure version. A workaround is to simply open the old file using **Open…**, save the data to a new file (which automatically uses the current data structure version), and then use **Append** with this new file.

Export…: exports the current set either as an image file or as a data file. Supported file formats are:

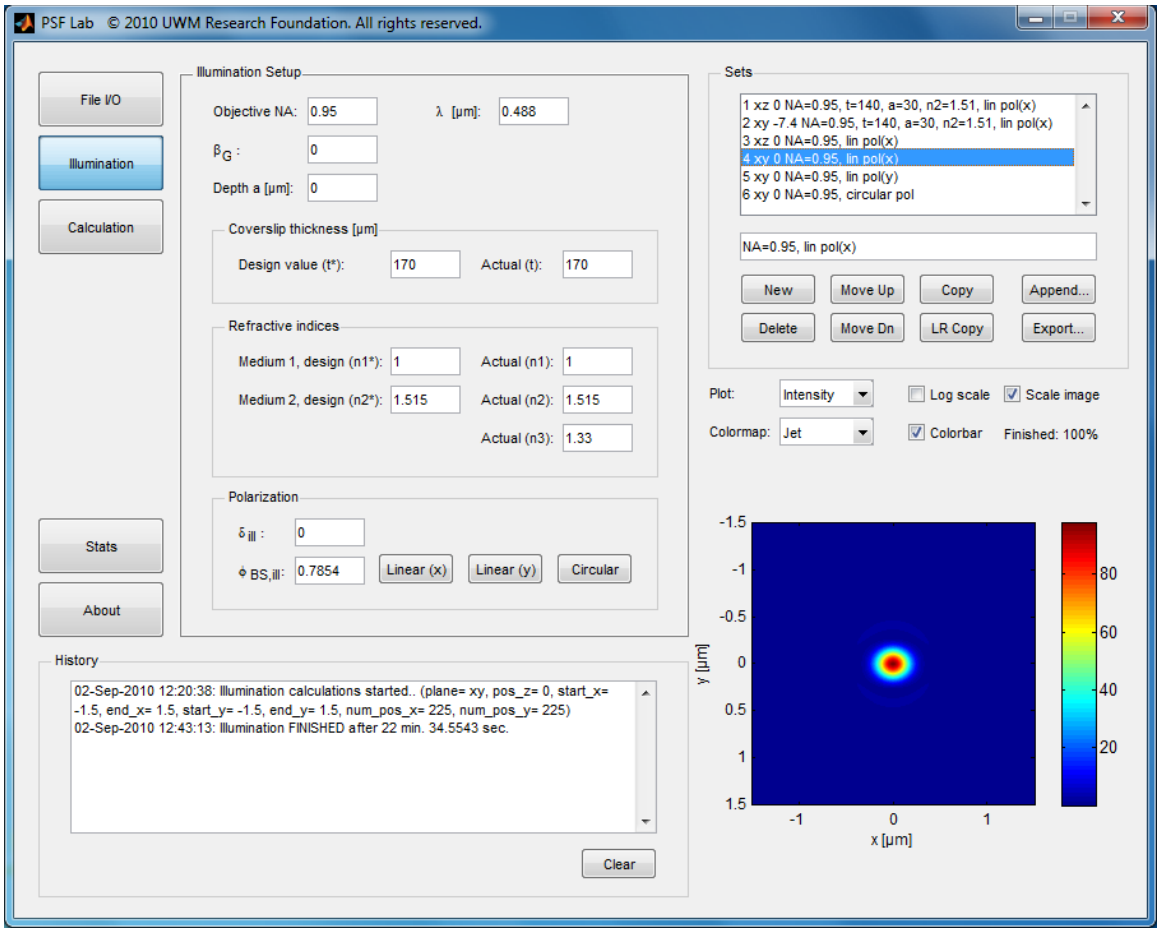
Image files:

- *.fig*: native MATLAB figure (v7, R14)
- *.ai*: Adobe Illustrator
- *.bmp*: Microsoft Bitmap
- *.emf*: Enhanced Metafile
- *.eps*: Encapsulated Postscript
- *.jpg*: JPEG
- *.pcx*: Paintbrush
- *.pdf*: Adobe Portable Document
- *.png*: Portable Network Graphics
- *.tif*: Compressed TIFF

Data files

- *.mat*: binary MATLAB (v7, R14)
- *.h5*: HDF5 (v 1.8.1)
- *.txt*: ASCII text (double floating point numbers)

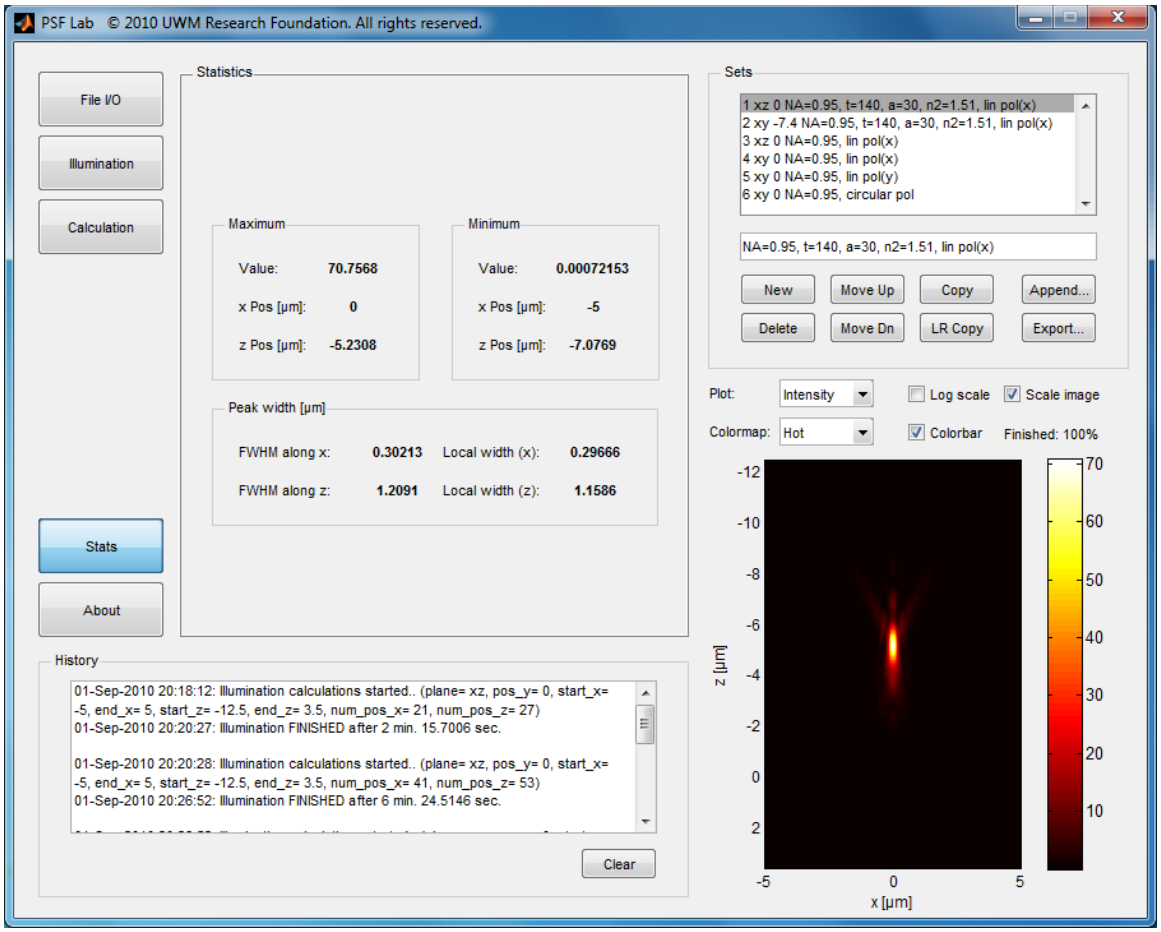
While image files are saved *exactly* as shown in the **Plot** panel (using the same aspect ratio, linear or logarithmic color scale, and colorbar), data files are always saved in linear scale. Note that PSF Lab cannot read exported MATLAB data files as they do not contain metadata (like refractive indices, coverslip thickness, etc.); they only contain the full matrix of numerical values corresponding to the displayed PSF image.



History panel

This non-editable field shows a log of date, time, computational time and some key parameters for each calculation that is performed.

Clear: clears the history log. The history log is not removed from the PSF Lab data file until the file is actually saved.



Plot area

The area in the lower right shows the calculated PSF image. If the calculation was interrupted, the percentage of calculated pixels is shown in red on the top right corner of the plot area; otherwise, "100%" is displayed in black.

The display settings listed below are stored with each set, so that different sets can be displayed with different settings when selected. Exported images will also reflect these display settings.

Plot: popup menu to choose the type of PSF data to be displayed. The choices are: field intensity, magnitude of each field component along *x*, *y*, or *z*, or squared field components along these same axes. The choice of displayed data type will affect the values displayed on the **Stats** panel.

Colormap: popup menu to select one of the built-in MATLAB colormaps. The default colormap is "hot".

Log scale: if checked, the display uses a logarithmic (base 10) color scale instead of the default linear scale. This can be very useful for enhancing subtle features in high-contrast PSF images. Note that checking this option has no effect on the underlying data.

Colorbar: if selected, a colorbar is displayed to the right of the PSF image. The colorbar, if displayed, will be exported as part of the image.

Scale image: if checked, the aspect ratio of the plot is set so that the horizontal and vertical axes use the same physical length per displayed μm. Otherwise, the image fills the plot area.