



SlideBook™ 4.0

3D Deconvolution Module

for Windows 2000/XP

User's Manual

(Latest Revision 9.24.03)

Manual Outline

MANUAL OUTLINE	2
CONTACTING INTELLIGENT IMAGING INNOVATIONS, INC	3
MANUAL CONVENTIONS	4
TYPOGRAPHIC CONVENTIONS	4
OF PIXELS AND VOXELS	4
1 INTRODUCTION TO THE DECONVOLUTION MODULE	5
NEAREST NEIGHBORS DECONVOLUTION.....	5
CONSTRAINED ITERATIVE DECONVOLUTION.....	5
POINT SPREAD FUNCTION (PSF) GUIDE	5
2 PERFORMING DECONVOLUTION	6
INTRODUCTION TO DECONVOLUTION METHODS	6
<i>Nearest Neighbors</i>	6
<i>Constrained Iterative</i>	6
PERFORMING NEAREST NEIGHBORS DECONVOLUTION	7
PERFORMING CONSTRAINED ITERATIVE DECONVOLUTION.....	11
TROUBLESHOOTING DECONVOLUTION PROBLEMS.....	13
MEMORY REQUIREMENTS FOR DECONVOLUTION	13
ADVANCED DECONVOLUTION OPTIONS	15
<i>Interpolation</i>	15
<i>Edge Padding</i>	16
3 MEASURING POINT SPREAD FUNCTIONS (PSFS)	17
CAPTURING A 3D STACK OF FLUORESCENT BEADS USING SLIDEBOOK.....	17
GENERATING A PSF USING THE PSF GUIDE	17
VIEWING PSFS	20

Contacting Intelligent Imaging Innovations, Inc

If you have any questions or experience any problems with SlideBook™, please contact us by either phone or email. Our phone number is **(303) 607-9429** and our email address is **support@intelligent-imaging.com**. Our web site at **<http://www.intelligent-imaging.com>** also contains useful information about SlideBook™, frequently asked questions, and useful microscopy and hardware information.

SlideBook™ comes with one year of free technical support. We will answer questions as promptly as possible and in the event of any program errors, we will make every effort correct the problem as soon as possible.

Manual Conventions

Typographic Conventions

Menu commands are written in bold and follow the order of menu navigation. For instance, choosing “Open” from the “File” menu is written as **File > Open**.

Dialog fields and other interface items are written in bold, as in **Initial Offset**.

References to other sections of the manual are underlined, as in Image Capture.

New terms are *italicized*.

NOTES will be written in bold and small capitals. Notes that point out new and recently added features will be prefaced **SLIDEBOOK 3.0**. Notes that you should pay particular attention to are prefaced **CAUTION**.

Of Pixels and Voxels

In other contexts, *image* typically connotes a two-dimensional entity. SlideBook™, however, was specifically designed to operate on three-dimensional data as easily and intuitively as two-dimensional data. In light of this, a two-dimensional data set can be seen as a special case of a three-dimensional set where data has been collected for only one value of the z-axis. An image corresponds to *all* data collected during a particular exposure.

To simplify this manual, we always refer to a single data element as a *voxel* and a region as a *volume*, regardless of the depth of the image. These terms map directly to *pixel* and *area*, respectively, for a two-dimensional set.

1 Introduction to the Deconvolution Module

The Deconvolution Module adds advanced deconvolution tools to the SlideBook architecture. SlideBook's Deconvolution module is a powerful tool for removing out-of-focus information from a fluorescence image set. SlideBook can implement three deconvolution approaches: *no neighbors*, *nearest neighbors*, and *constrained iterative*, listed in order of increasing computational complexity and accuracy. The no neighbors algorithm is included in the SlideBook base package and is discussed in the SlideBook User's Manual. The Deconvolution module adds the following functionality:

Nearest Neighbors Deconvolution

This algorithm is more computationally intensive and accurate than the No Neighbors algorithm. An image must have at least three planes in order to be deconvolved with the Nearest Neighbors algorithm.

Constrained Iterative Deconvolution

The constrained iterative approach is the most computationally expensive form of deconvolution and, if performed properly, can result in the best final image. All statistical analysis should be performed on images that have been deconvolved with the constrained iterative algorithm.

Point Spread Function (PSF) Guide

All of the deconvolution processes use *Point Spread Functions (PSFs)* which are the descriptions of how an optical configuration blurs light. A different PSF must be used for each optical configuration (i.e. objective, magnification changer, filter combination). Constrained Iterative deconvolution allows the use of PSFs that are either measured directly using fluorescent beads or approximated computationally. The PSF Guide allows you to generate a database of measured point spread functions for each optical configuration. This allows you to refer to stored data automatically when performing Constrained Iterative deconvolution.

2 Performing Deconvolution

In this chapter you will learn how to do the following:

- Perform Nearest Neighbors Deconvolution
- Perform Constrained Iterative Deconvolution
- Troubleshoot Deconvolution Problems
- Determine the Image Size Limit for your Computer
- Use Advanced Deconvolution Options

Introduction to Deconvolution Methods

The following methods are supported in the Deconvolution Module.

Nearest Neighbors

Nearest neighbors deconvolution is a deblurring technique that uses a simplifying assumption that most of the out-of-focus blur in an image is due to the planes directly neighboring the plane of interest (i.e. directly above and below). To perform nearest neighbors deconvolution, one must start with a data set that is at least three planes deep. The algorithm computes how much blur the neighboring planes contribute to the captured image and subtracts the blur out. Since the algorithm is purely subtractive, it tends to delete very low intensity features. However, it is considerably less sensitive than constrained iterative deconvolution to low intensity image defects such as dust in the optical path. The largest single factor that affects the result of nearest neighbor deconvolution is the distance between neighboring planes. A smaller distance between neighboring planes will cause the algorithm to subtract more from the plane of interest.

Constrained Iterative

The constrained iterative approach is the most computationally expensive form of deconvolution and, if performed properly, can result in the best final image. All statistical analysis of intensities should be performed on images that have been deconvolved with the constrained iterative algorithm. Essentially, the algorithm makes sequential guesses about the true distribution of fluorescence in an unblurred three dimensional data set. It then compares its computationally blurred guess with the original data and modifies its next guess appropriately. Sequential guesses should converge on a unique solution. The user must determine how many iterations or guesses the algorithm makes.

Because constrained iterative deconvolution considers all of the data to perform its guesses, the approach is very sensitive to the condition of the original data. If performed improperly, the constrained iterative approach will result in a highly artifactual data set. To ensure good results with this approach the following conditions should be met:

- The original data set should span the sources of fluorescence in the z dimension. If there are significant sources of fluorescence that reside above or below the original data set, the algorithm will not be able to properly assign their source within its next guess.

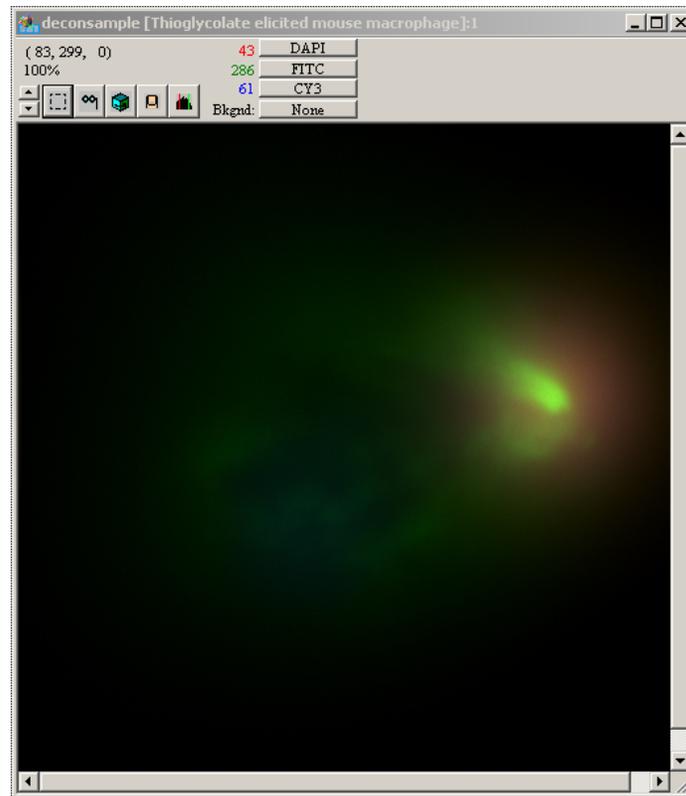
- The data should be flat field corrected. If there are any constant artifacts within each plane of the original data (e.g. dust specks), the algorithm must guess 3D structures which will account for them. This usually results in fictitious tube structures in the final image set.
- The data, as always, should reside in the linear range of the camera (not saturated) and should be well above the noise floor of the camera. Very low intensity data sets have a poor signal to noise ratio. This will result in a “scratchy looking” final guess.

Constrained iterative deconvolution can be very time consuming as well as memory intensive. Most users find no neighbors or nearest neighbors desirable for every day deblurring. However, there are certain cases where only the constrained iterative approach yields a reasonable result.

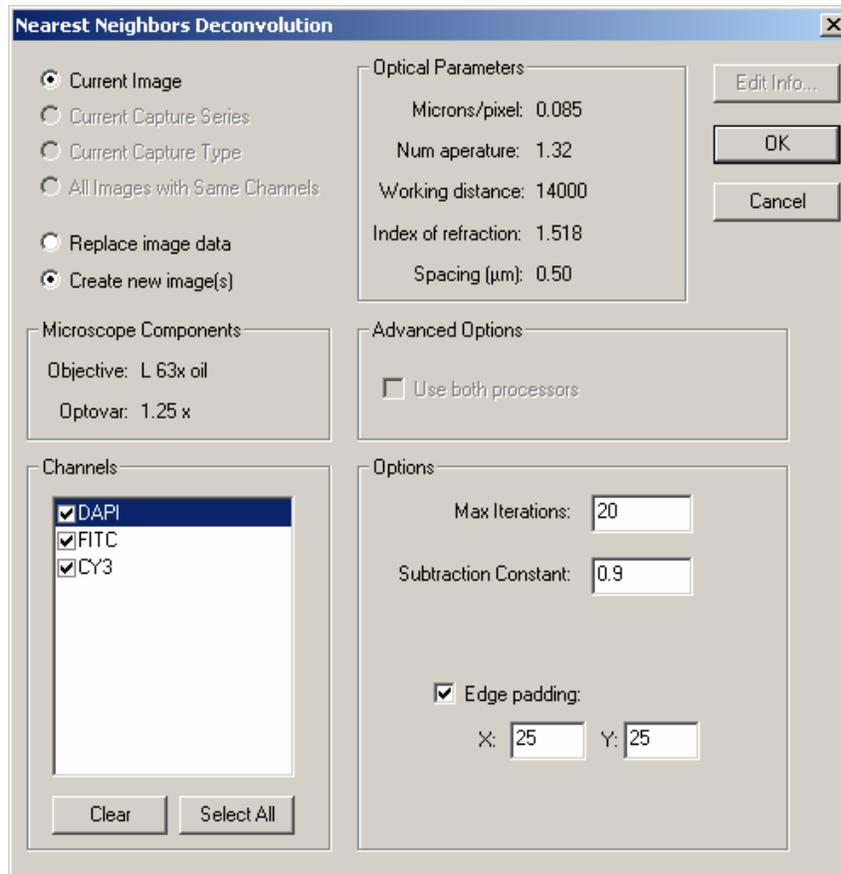
Performing Nearest Neighbors Deconvolution

We will now explore deconvolution using the sample slide titled “**deconsample.sld**” that accompanies this manual. To begin this demonstration:

1. Open SlideBook and select **File>Open Slide**.
2. Navigate to **deconsample.sld** and select **Open**.
3. Create a main view of the **Thioglycolate elicited mouse macrophage** image. You can do this by double clicking on the thumbnail of the image in the slide window or single clicking on the thumbnail in the slide window and selecting **View > New Main View**.



4. Select **Image>Deconvolve > Nearest Neighbors**. This will bring up the Nearest Neighbors Deconvolution dialog box.



The dialog box has the following sections (starting from the upper left):

- **Scope of operation** – allows you to choose to perform deconvolution on the current image, current capture series, current capture type, or all images with the same channels.
- **Deconvolved image destination** - allows you to choose the destination of the deconvolved image
- **Microscope Components** – displays the optical components that were used to take the image.
- **Channels** – allows you to select the channels to be deconvolved. **Clear** removes all checks, while **Select All** checks all channels
- **Optical Parameters** - displays the optical parameters, which may be altered if incorrect.
- **Advanced Options** – allows you to take advantage of multiple processors, if present
- **Parameters** – allows you to alter parameters that will affect the extent of de-blurring.

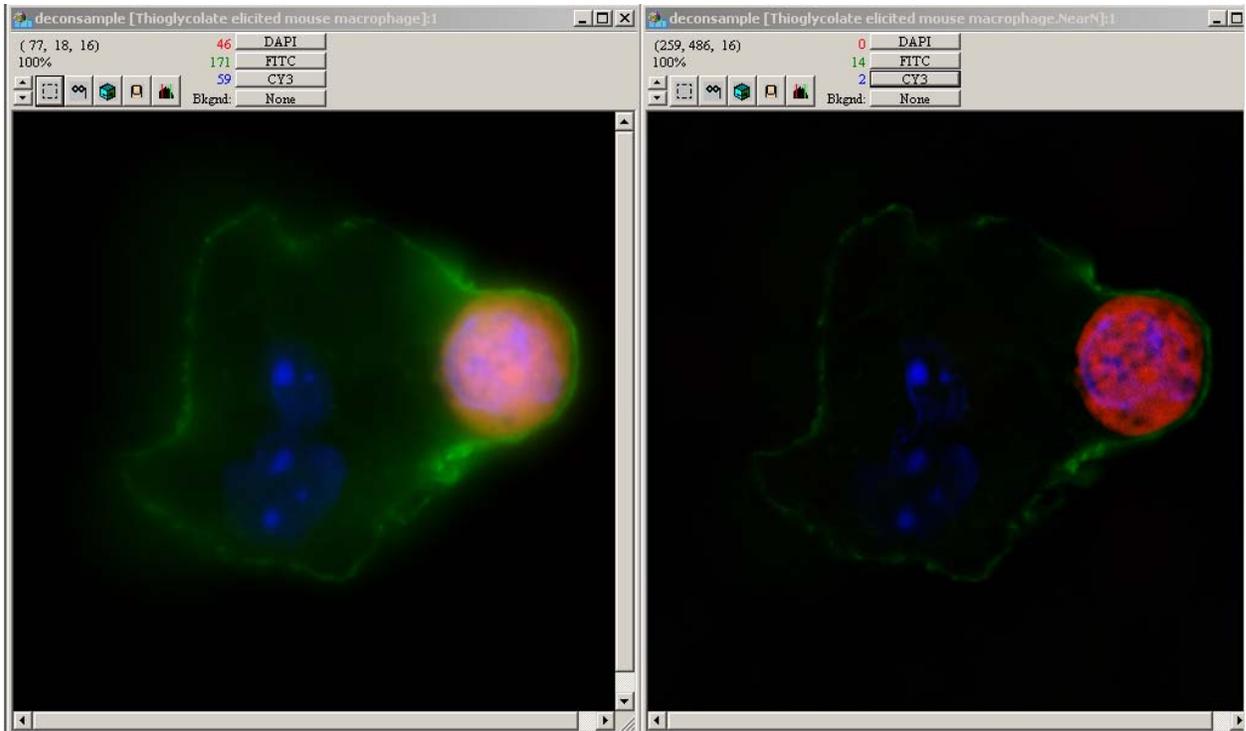
- **Iterations** - The default is 20, but the deconvolution will stop earlier if the algorithm reaches the optimal solution. The deconvolution algorithm operates by generating a guess, blurring the guess with the point spread function, and then comparing the blurred guess with original image. If the comparison determines that the image is not improving, the algorithm is stopped. Increasing this number can generate a better set of guesses: but often, the comparison condition will stop the algorithm before completing the iterations. Therefore, you should only increase this parameter if the previous deconvolution completed all of the specified iterations.
- **Subtraction** – Increasing the Subtraction constant will increase the amount of deblurring. The subtraction parameter adjusts the amount of light subtracted.
- **Edge Padding** – allows you to add pixels to the x and y dimension using mirror padding. This is particularly useful when fluorescence is present at the edges of the image. This is more fully described in the [Advanced Deconvolution Options](#) section on page

5. Note that **Create new image** is selected, and then click on **OK**.

This will begin the Nearest Neighbors deconvolution on the image stack. This process will take several minutes and can be canceled if desired. When completed, a new main view will appear with the deconvolved data, and a new thumbnail will appear in the slide view. The file name will carry the extension *.NearN.

6. Click on the renormalize button in the main view of the deconvolved image. (Please see the SlideBook User's Manual Chapter 8 if you are unfamiliar with Renormalization.)
7. Enter the following values for the high and low thresholds, being sure to select **Apply** before proceeding to the next channel, then select **OK** to register the changes:
- CY3 – 100 and 470
 - FITC – 100 and 500
 - DAPI – 100 and 1000
8. Click on the thumbnail button in the main view of the deconvolved image to save the settings.
9. Click on the newly created image. Scroll to the 16th plane of the image by selecting **View>Go to Plane....** Enter "16" in the data entry field and click **OK**.

10. Repeat steps 7 and 8 for the undeconvolved image and compare the images side-by-side.



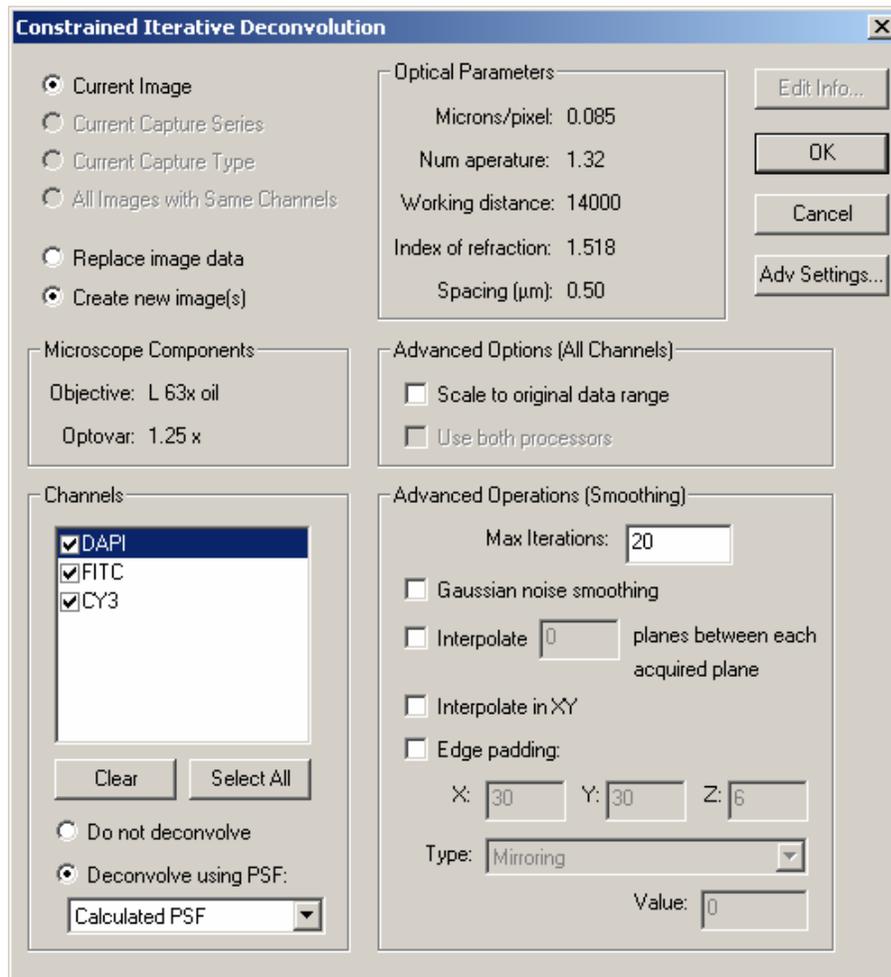
Note that the new data set (shown on the right, above) is two planes shorter than the original. This is because nearest neighbors requires data from both above and below each plane, so the first and last planes are truncated in the deconvolved set.

Try exploring the **Thioglycolate elicited mouse macrophage** and **Thioglycolate elicited mouse macrophage.NearN** data sets with tile views and three views as described above. Note the improved resolution in all dimensions resulting from deconvolution.

Performing Constrained Iterative Deconvolution

We will now deconvolve the T cell image using constrained iterative deconvolution and compare to the nearest neighbors result.

1. Click on the main view of the undeconvolved **Thioglycolate elicited mouse macrophage** image.
2. Select **Image>Deconvolve>Constrained Iterative**. The following dialog box will appear.



Notice that the left half of the dialog box is the same as the Nearest Neighbors dialog box, with the exception of the added option to choose to deconvolve using a **Measured PSF** or **Calculated PSF**. If there is not a measured PSF for the given optical configuration in the PSF database, the default will be **Calculated PSF**. See the next chapter for a discussion on measuring PSFs. You may also choose to perform advanced processing techniques, which are discussed in [Advanced Deconvolution Options](#) on page 15. These options are:

- **Gaussian noise smoothing** – minimizes artifacts due to noise
- **Interpolation** – adds a user-specified number of planes between each acquired plane
- **Edge padding** – creates a space around the border of the image in order to resolve fluorescent point sources that exist outside of the image.

3. Note that the default selection is set to **Create New Image**, and all channels will be deconvolved. Choose **OK** to begin deconvolution.

A **Deconvolution Status** dialog box will appear, displaying the channel that is being deconvolved, the current iteration, and degree of convergence. This operation may take several minutes depending on the speed of your computer.

Once the deconvolution is finished, the image will appear in a window titled: **Thioglycolate elicited mouse macrophage.CI**.

4. Click on the newly created image. Scroll to the 16th plane of the image by selecting **View>Go to Slice...** and entering “16” in the data entry field.
5. Click on the Renormalization button and experiment with various thresholds on each channel. Note that by first displaying a plane with high channel intensities, you may easily see the effect of renormalization.
6. Enter the following values for the high and low thresholds, being sure to select **Apply** before proceeding to the next channel, then select **OK** to register the changes:
 - CY3 – 100 and 1000
 - FITC – 10 and 500
 - DAPI – 100 and 1000
7. Compare the CI and Nearest Neighbors images directly, displaying the same z plane in each image. Note the increased detail of the CI deconvolved image.

Troubleshooting Deconvolution Problems

There are several keys to ensuring that your deconvolution runs smoothly.

- Make sure that the image information is correct as described in Getting and Editing Information after Capture in the SlideBook User’s Manual. You must have the emission wavelengths and all objective information correct.
- If at all possible crop the image before performing constrained iterative deconvolution. This will increase the speed of the operation and prevent your operating system from crashing.
- If you are deconvolving a large image, be sure that no other images or programs are open.

Memory Requirements for Deconvolution

Deconvolution is a memory-intensive process and is limited by the size of the image that is being deconvolved. On the PC, any given program can use up to 2 GB of memory. This results in the following size limits for deconvolution.

- **No Neighbors Deconvolution:** 800 MB

- **Nearest Neighbors Deconvolution:** 800 MB
- **Constrained Iterative Deconvolution:** 180 MB

It is recommended that, for large images, only a single slide containing a single image is open. For smaller images that don't push these memory limits, this is not necessary. However, you should always be aware of the limits. If those limits are hit, then an error should be returned to that effect, as well as a location in the deconvolution algorithm where the error occurred.

The error should be: "**CIDecon_MemoryAllocationFailure.**" If you get this error, you are probably reaching the limits of deconvolution on the system (see section above regarding troubleshooting deconvolution).

If you have less than 2 GB of memory, you may calculate the size limit as follows. Each channel of the image occupies an amount of memory, called M. This number is directly related to the number of pixels in the image. For instance, a 1280 by 1024 image with 100 planes and 3 channels has $M = 1280 \times 1024 \times 100 \times 2 \times 3 = 786$ megabytes (each pixel is 2 bytes).

For No Neighbors and Nearest Neighbors, when an image is deconvolved, the amount of memory necessary is equal to the old image, plus the new image, plus some small scratch space. Thus, M can be about 40% of the RAM on the PC (being conservative, to preserve space in memory for the operating system and the program).

For Constrained Iterative, the answer is more involved. The formula for determining the amount of memory used for a single channel image is:

$$M \text{ (orig image)} + 2M \text{ (conversion into fourier space)} + 2M \text{ (PSF)} + 6M \text{ (scratch space)} + M \text{ (resulting image)} = 12M$$

In order to deal with this large memory requirement, the algorithm can place the point spread function on the disk and remove it from memory, reducing the amount of necessary memory from 12M to 10M.

The speed of the algorithm depends on how much RAM is available. Placing the point spread function on the disk means that the disk must be accessed in order to complete the algorithm, and it is always slower to access a disk than to access RAM. Therefore, if the full 12M will fit in available memory, then it will be placed into memory, which can limit other operations in the program.

It is recommended that, for large images, only that image is open. For smaller images that don't push these memory limits, it is not necessary to be overly concerned, but it is a good idea to be aware of the limits. If those limits are hit, then an error should be returned to that effect, as well as a location in the deconvolution algorithm where the error occurred.

If deconvolution will be used often, it is imperative to have as much RAM as possible—2 GB on the PC.

Advanced Deconvolution Options

SlideBook's Constrained Iterative deconvolution algorithm offers several advanced options.

Interpolation

If the image that you would like to deconvolve was captured at a z spacing that is larger than that of the image that was used to generate the PSF, or if the z resolution is quite a bit larger than the xy resolution, this option helps take advantage of prior knowledge of a PSF whose pixels are nearly cubic. For instance, if you have a PSF that has an inter-plane spacing of $0.1\ \mu\text{m}$ and pixels sizes of $0.1\ \mu\text{m}$, the information in the PSF may be used to add additional planes to an image with an inter-plane spacing of $0.4\ \mu\text{m}$.

When using this option, it is important to either take the default value (which represents the closest integer ratio to having cubic pixels) or some integer divisor of that number. This condition ensures that the interpolated image aligns itself correctly with a measured point spread function.

As an example: Assume you have a PSF sampled at $0.1\mu\text{m}$, and an image sampled at $0.4\ \mu\text{m}$.

Without Interpolation: every fourth plane in the PSF will be used when performing deconvolution.

PSF planes:	0	1	2	3	4
Image planes:	0				1

Ratio=2: every second plane in the PSF will be used. Planes 0 and 4 will match image planes, and plane 2 of the PSF will match the interpolated plane.

PSF planes:	0	1	2	3	4	
Image planes:	0		1*		1	(* - interpolated)

Ratio=3: three of four planes in the PSF will be used. The image planes and PSF planes do not line up.

PSF planes:	0	1	2	3	4	
Image planes:	0		1*	1*	1	(* - interpolated)

In this case, the division of 0.3 into 0.4 did not yield an integer result (1.33).

Ratio=4: every plane in the PSF will be used, and of those planes, one will correspond to actual data while the other three will correspond to interpolated planes.

PSF planes:	0	1	2	3	4	
Image planes:	0	1*	1*	1*	1	(* - interpolated)

Therefore, when choosing the number of planes to interpolate, make sure that it is an integer that falls between the PSF and image sampling rate, and that it does so on an integer multiple of the spacing.

Note that the interpolation option will increase the time that is required for the algorithm to finish. The larger the number of interpolated planes, the more time the algorithm will require. Specifically, the time will increase by a factor of $((\text{number of planes}-1)*\text{ratio} + 1)$ —in other words, a larger ratio is a multiplication of the amount of time and space that is used by the algorithm.

Edge Padding

In deconvolution microscopy, containing all the source of fluorescence in the image results in optimal deconvolution results. If any blur that is present in the image is caused by point sources within the image, then resolving those point sources can proceed. However, if point sources exist outside the range of the image that still affect the image (such as imaging tissue sections, where fluorescence will extend to the edge of the field), then the algorithm has no space in which to resolve that blur. Padding creates a space outside the border of the image that can resolve those outside point sources.

Always, if possible, capture enough planes to account for the blur in the z direction both above and below the object to be imaged. Point Spread Functions (PSFs) vary in depth, depending on the system, so this is up to the imager; in general, once there is no fluorescence detected in the imaged plane, you have captured a sufficient range.

When selecting the size of the padding space, do not pad larger than the twice radius of the PSF (in XY or Z).

The possible padding options:

- **Zero** - The lowest number in the image (called the DC offset) is automatically subtracted from the image; generally this will be the most common pixel value. The padding values are set to zero.
- **User-determined constant** - When the most common pixel is either not relevant or not the lowest amount, you can try another padding value.
- **Mirroring** – the most rigorous option, this method takes advantage of the nature of Fast Fourier Transforms (FFTs) by generating mirror images of the image in the padded space, helping to resolve the original image space.

3 Measuring Point Spread Functions (PSFs)

SlideBook stores point spread functions (PSFs) in a data file. The PSF data file is accessed during deconvolution that specifies the use of measured PSFs. PSFs are computed using 3D images of fluorescent beads. In order to measure a PSF, the following steps are carried out: 1) prepare a slide containing fluorescent beads (0.2 μm), 2) capture a 3D image stack of the beads, and 3) perform measurements on that stack using the PSF guide. This sequence should then be repeated for each combination of objective, optovar, and fluorescence channel. You may also store more than one PSF for each optical configuration. This is useful for storing PSFs captured under different sample media conditions, different users, or different brightfield optic settings, for example.

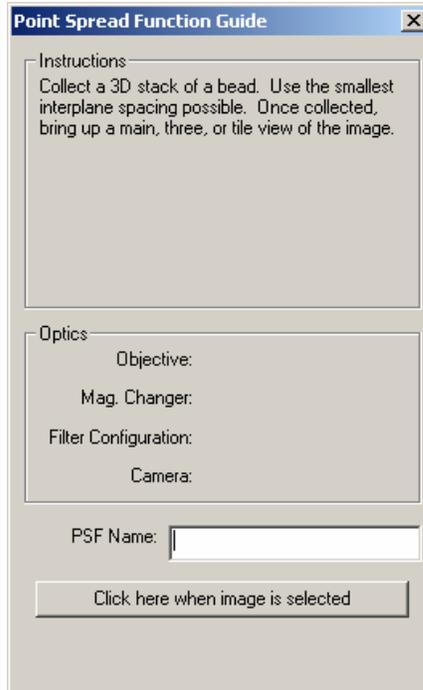
Capturing a 3D Stack of Fluorescent Beads Using SlideBook

The 3D bead image may either be captured using SlideBook or imported from another program. To perform 3D capture, see Chapter 7 of the SlideBook User's Manual. Generally, a total depth of 14 μm taken at 0.1 μm intervals is sufficient for capturing a 0.2 μm bead. In order to reduce the amount of memory required for capture, you may wish to capture a single bead. Be sure to allow ample room around the single bead (at least fifty pixels on any side) so that all of the light from the bead is captured. Additionally, the image used to generate a PSF should be flat field corrected (see Chapter 9 of the SlideBook User's Manual). To import an image, please see Chapter 6 of the SlideBook User's Manual. Make sure that the image information is correct before proceeding.

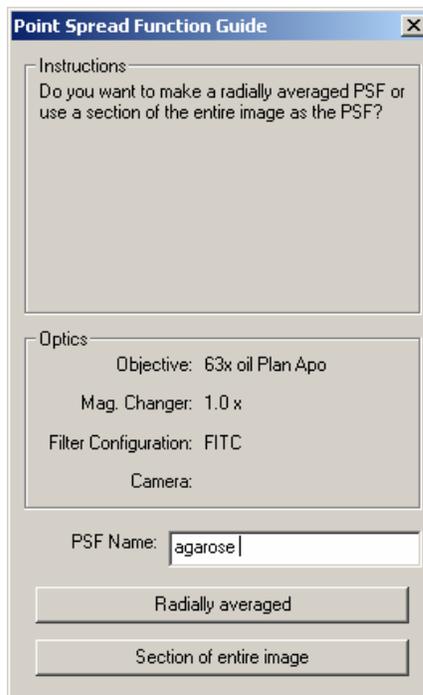
Generating a PSF using the PSF Guide

Once the 3D image stack has been collected, a PSF entry can be created. To do so:

1. Select **Edit>Setup Guides>PSF Guide**. The following dialog box will appear.



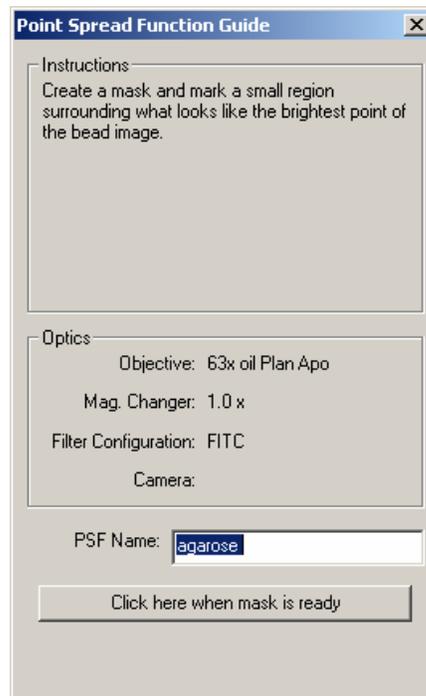
2. As directed by the guide, open a main view of the image stack and then click on the button in the dialog box. Enter a unique name for your PSF (this is the name you will select from the drop-down menu in the Deconvolution dialog boxes). The guide will advance to the next dialog box.



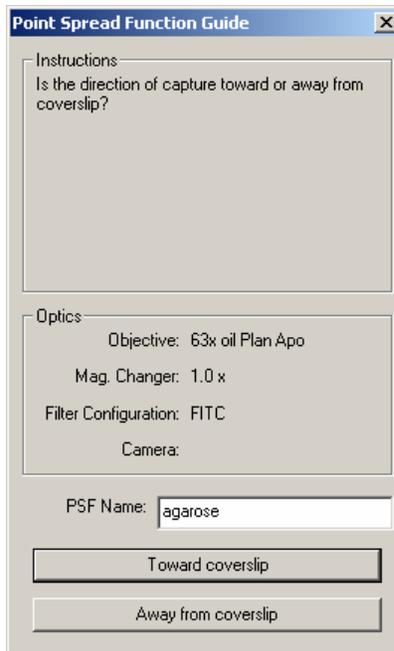
3. Select either **Radially averaged** or **Section of entire image**.

- **Radially averaged** – uses one half of a PSF that is radially averaged. This is good for noisy PSF data.
- **Section of entire image** – uses a 101x101x101 section of the image. This is best for images with good signal. Accounts for more aberrations than a radially averaged PSF

When you have made your selection, the following dialog box will appear.



4. Create a new mask by selecting **Mask>Create** and give it a name.
5. Select the pencil tool from the tool bar of the main view, and mark the center of the bead at what is approximately the brightest region.
6. Advance to the next dialog box by clicking the button at the bottom of the dialog box, and then choose the direction of objective movement.



7. In the final dialog box, check that the desired objective, optovar, and fluorescence channel are selected, and then choose **Add PSF**. The measured PSF will be added to the PSF database.

Viewing PSFs

The PSFs that are stored in SlideBook can be viewed by selecting **Edit>Setup Guides>Display PSFs**. A table of objectives, optovars and channels will be displayed. To view the PSF for a particular combination, click on the PSF icon for the desired optical combination, then choose **Display**.

