M Squared Aurora GUI User Manual

Software version: Deconvolution 0.5

Authors	Version	Date
Pedro Almada and Valentina Lo Schiavo	2.0	28-11-2019
Pedro Almada	2.1	19-12-2019
Pedro Almada	2.2	03-02-2020
Pedro Almada	2.3	27-02-2020
Pedro Almada, Valentina Lo Schiavo, Tom Michell	2.4	25-03-2020
Pedro Almada	2.5	30-04-2020
Pedro Almada	2.6	12-03-2021

1 Contents

M Squared Life The Surrey Technology Centre, 40 Occam Road, Guildford, Surrey, GU2 7YG, United Kingdom
M Squared Life Limited Venture Building, West of Scotland Science Park, Glasgow, G20 0SP, United Kingdom
Phone +44 (0) 141 945 0500 | **Fax** +44 (0) 141 945 0505 | **Email** life@m2lasers.com | **Web** m2lasers.com
VAT Registration Number GB 304 3396 25 | **Registered in Scotland Number** SC522938

M Squared Life The Surrey Technology Centre, 40 Occam Road, Guildford, Surrey, GU2 7YG, United Kingdom
M Squared Life Limited Venture Building, West of Scotland Science Park, Glasgow, G20 0SP, United Kingdom
Phone +44 (0) 141 945 0500 | **Fax** +44 (0) 141 945 0505 | **Email** life@m2lasers.com | **Web** m2lasers.com
VAT Registration Number GB 304 3396 25 | **Registered in Scotland Number** SC522938

M Squared Life The Surrey Technology Centre, 40 Occam Road, Guildford, Surrey, GU2 7YG, United Kingdom
M Squared Life Limited Venture Building, West of Scotland Science Park, Glasgow, G20 0SP, United Kingdom
Phone +44 (0) 141 945 0500 | **Fax** +44 (0) 141 945 0505 | **Email** life@m2lasers.com | **Web** m2lasers.com
VAT Registration Number GB 304 3396 25 | **Registered in Scotland Number** SC522938

2 Introduction

2.1 Scope

This user guide for the Aurora Deconvolution software includes sections describing bead calibration and user data deconvolution using software version 0.5.

2.2 Airy beam light sheet

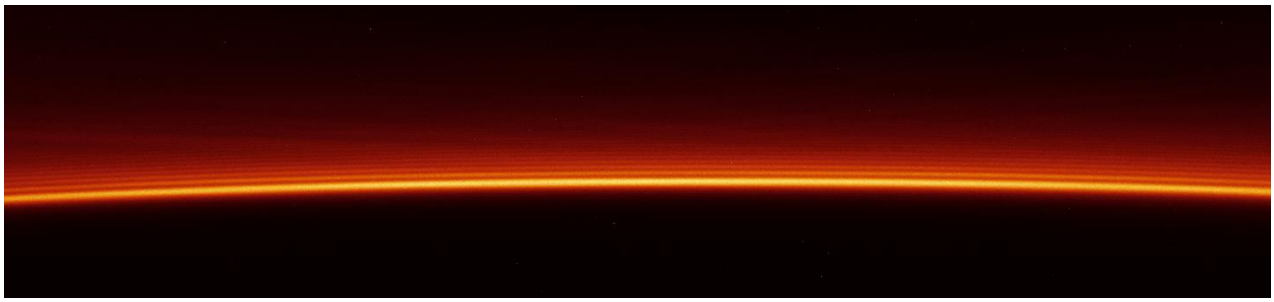


Figure 2.1 - The Airy beam illumination profile that defines the Aurora Light Sheet system.

The M Squared Aurora microscope is an Airy beam light sheet microscope. Instead of using the Gaussian beam illumination profile that is common to most microscopes, it uses an Airy beam illumination profile (Figure 2.1). Because of the optical properties of the Airy beam, the Aurora microscope, in comparison with other light sheet microscopes:

- Maintains image contrast throughput the entire field of view of the camera, instead of having good contrast only at the centre of the field of view.
- Has a much-improved axial resolution.
- Does not suffer as much from the striping artefacts that are common in normal light sheet.

The trade-off is that the Airy beam shape needs to be deconvolved (see 2.3). This user guide will guide you through the M Squared software designed to perform this process.

2.3 Deconvolution and the Point Spread Function

When a microscope images the light emission of very small point in space, the resulting images will not be a perfect representation of the original point. This is because the emitted light is spread by the optics of the microscope, meaning the image will be blurred by the optics of the microscope. However, this spreading of the light is consistent for a given microscope, and can be mathematically described by a ***Point Spread Function***, or PSF.

When imaging biological samples, all images acquired with a microscope will be blurred in the same way by the microscope's PSF. Mathematically, this is called a ***convolution*** process, where it is said the original light of the sample was ***convolved*** with the microscope's PSF.

A ***deconvolution*** algorithm takes either a simulated or measured PSF and attempts to reverse the convolution that the image suffered through the microscope's optics. This will

result in an image that is less blurry and with higher resolution. When deconvolving an image that was convolved with an ***Airy beam***, because of the shape of the Airy beam, the resolution of the resulting image along the Z axis (or the axial resolution) will improve by a much larger factor than when deconvolving a sample illuminated with a Gaussian beam.

This User guide will show you how to obtain the PSF of your microscope and how to deconvolve your data using that PSF.

2.4 Developmental stage

The 0.x.x versions of the software are for the developmental systems developed by M Squared Life. This means the software may have bugs and other issues present. It is also is in constant development, and each version may have some interface changes.

2.5 Checking your version

Each user guide is matched to a specific version. Please make sure that the version of the software that is installed on your system matches this guide's version number. Generally speaking, the shortcut to the software will include the version number, or the folder where the software was installed to will include the version number.

2.6 Addressing bugs

Any issues that are found can be reported on the website:

https://www.msquaredcubes.com.

To access the website, you need a username and password which should have been assigned to you upon training. Otherwise, please speak with the person in charge of your system. If you do not get a response quickly enough, please email:

pedro.almada@m2lasers.com

2.7 Numbering scheme

- *MAJOR number (v**0**.1.2):*
  A new public release that is not backwards compatible. For the Aurora development program, this will remain at 0 until the public release of a production system.
- *MINOR number (v0.**1**.2):*
  Addition of any new major features to the software that have gone through regression testing. For example, GPU processing or Variable PSF.
- *BUGFIX number (v0.1.**2**):*
  Small bug fixes, usually addressing specific issues a client has faced. Small interface changes may occur to address some issues.

3 First Time Installation Steps

3.1 Download and extract

If installing for the first time, you'll need to download two zip files (explained below) from www.msquaredcubes.com. This is a private website that requires your own account to access. Please contact the M Squared Team for login details.

You'll need to download two files:

- The "Base Deconvolution Installation" package
- And a specific deconvolution package (e.g. "Deconvolution 0.4.0").

Once the files have been downloaded, extract the base installation package to its destination folder (we have used "C:\Applications" in the past and use "C:\M Squared\Deconvolution" more recently). Once that has been done, you can extract the specific update Package to the same folder as the Base Deconvolution Package and replace any files that may be there.

3.2 Updating drivers

The deconvolution software can make use of NVIDIA GPUs to accelerate the deconvolution. In order to do that, please make sure you have the latest NVIDIA drivers installed.

3.3 Create a desktop shortcut

To create a desktop shortcut, navigate to the M Squared Cubes software folder and locate the "Aurora.exe" file, then Right click>Create shortcut.

4 Updating the software

If you already have the Base Deconvolution package extracted in a folder, in most cases you should only need a specific deconvolution package (e.g. “Deconvolution 0.5.0”).

Occasionally a specific deconvolution package will request you to upgrade the base deconvolution package to a specific version. In those cases, you’ll need to download the base deconvolution package as well.

Extract the downloaded .zip(s) to the root directory where you have installed Aurora Deconvolution software previously.

The shortcut you have created previously should work without needing any changes.

M Squared Life The Surrey Technology Centre, 40 Occam Road, Guildford, Surrey, GU2 7YG, United Kingdom
M Squared Life Limited Venture Building, West of Scotland Science Park, Glasgow, G20 0SP, United Kingdom
Phone +44 (0) 141 945 0500 | **Fax** +44 (0) 141 945 0505 | **Email** life@m2lasers.com | **Web** m2lasers.com
VAT Registration Number GB 304 3396 25 | **Registered in Scotland Number** SC522938

5 Starting the software

Double-click the desktop shortcut to open the Aurora Deconvolution software. The software GUI will appear (Figure 5.1).

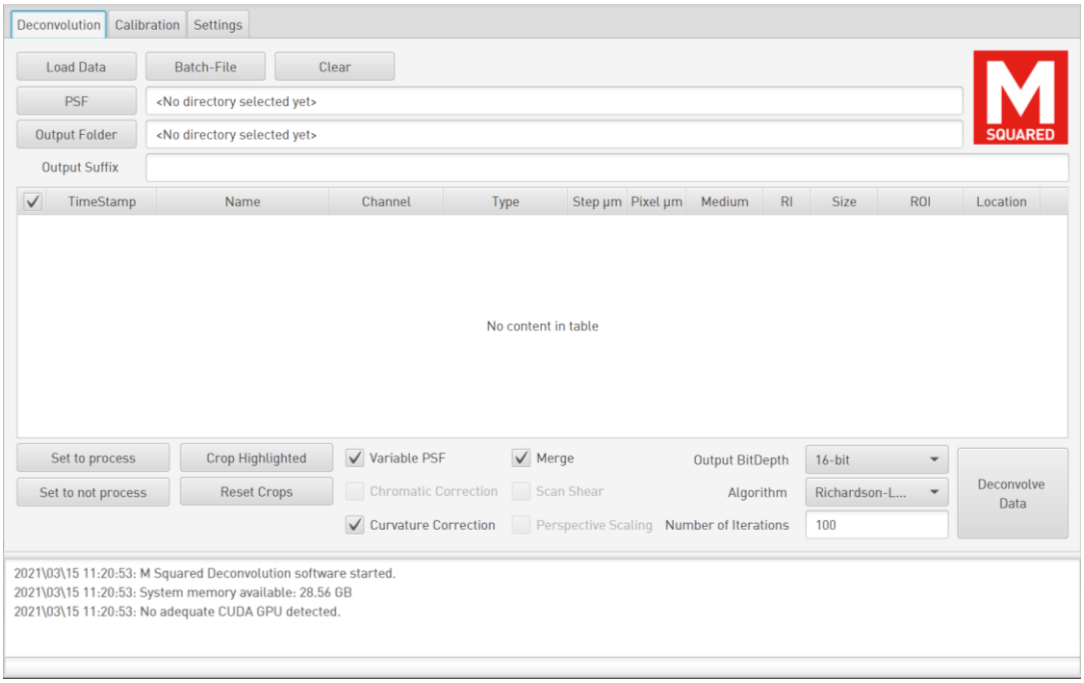


Figure 5.1 - Deconvolution software as it appears on startup

The software GUI has three tabs located in the top left that allow the user to switch between Calibration and Deconvolution routines, as well as Settings. Currently, the only setting that can be altered is the directory where the temporary files created during the deconvolution are saved (Figure 5.2). Please make sure it is on a fast drive, to increase processing speed.

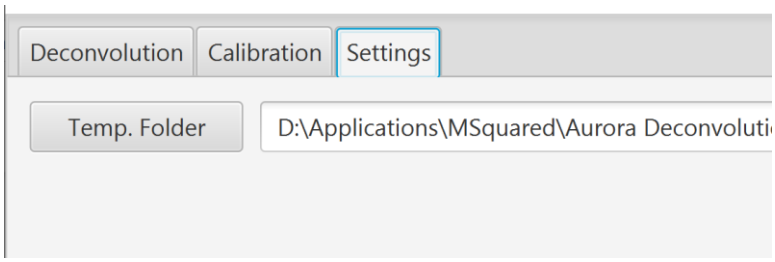


Figure 5.2 - Settings tab showing the temporary folder location

On start-up, the GUI shows the deconvolution tab.

6 Calibration

6.1 Overview

To deconvolve data successfully, the system's PSF (see 2.3 for an explanation) needs to be extracted from sub-resolution bead images. The Airy beam is also curved (see Figure 2.1 on page 4), which means that the images will need to be corrected for this curvature. Finally, some older developmental systems may have two types of aberrations (Perspective scaling and Scan shear) that also need to be accounted for.

The Aurora Deconvolution software contains an automated calibration routine that performs all of these measurements and outputs a cube.tiff file containing the PSF data and calibration information for each channel.

6.2 Data for calibration

To perform a calibration, the calibration routine needs image stacks of point-like bead samples that are sub-resolution (<0.5 microns). The beads need to be homogenously distributed and without any clumps, as this will affect the extracted PSF.

For most systems, we also need to acquire images of the static beam (see the acquisition manual). If the system's Airy beam curvature is negligible, you may safely skip this step.

It should be noted that one image stack and beam data is needed for each channel present in the hardware. These should be acquired in one multi-channel acquisition, to allow for the software to calibrate all channels.

6.3 The calibration procedure

6.3.1 Inputting data

Make sure you have the calibration tab selected. From there, select the bead data by pressing "Bead Directory" and, optionally, the beam data by selecting "Beam directory". The bead and beam data should have been acquired as a multi-colour acquisition and therefore all channels should be saved under the same folder (Figure 6.1).

However, if for some reason each channel was acquired separately, the software can also accept bead and/or beam data from different acquisitions. You only need to ensure that they are on the same folder and that there is one stack/image for each channel.

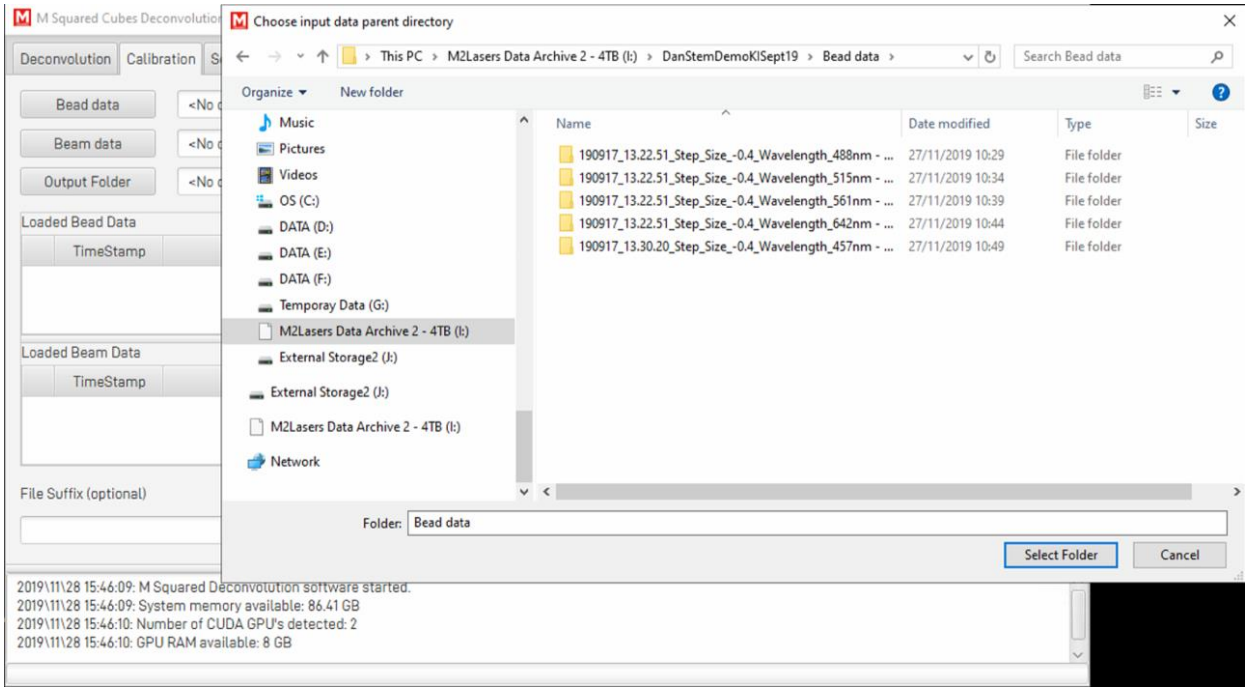


Figure 6.1 - Calibration tab selected and bead folder displayed

Once loaded, every channel folder within the parent folder will be loaded and displayed in the "Bead Directory" table in the GUI (Figure 6.2). The GUI will also check the size of each file, so this may take a moment.

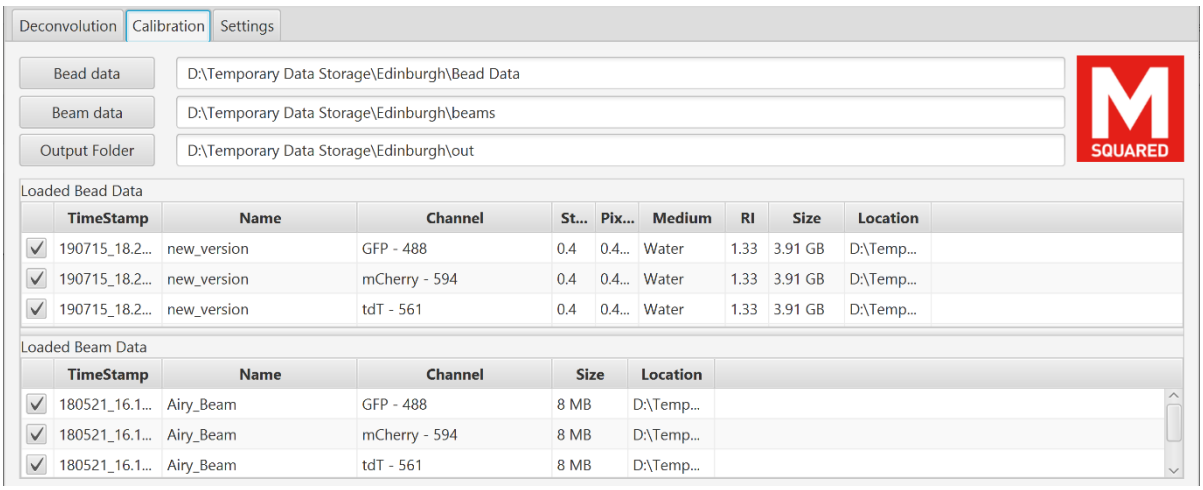


Figure 6.2 - Calibration pane after loading beads and beam data

Next, select the appropriate folder where the calibration file is going to be saved (Figure 6.3).

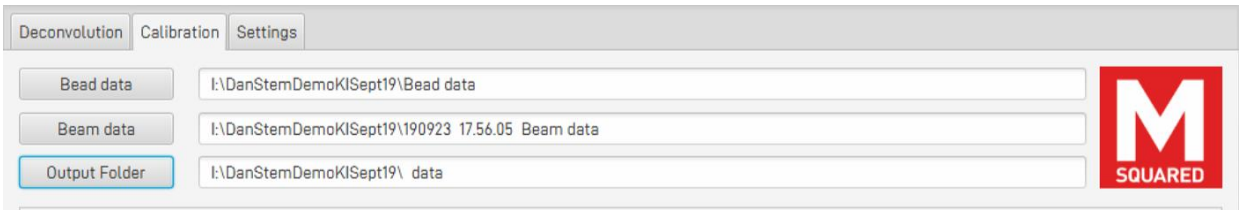


Figure 6.3 - Highlighted button to set output folder for the deconvolved data

6.3.2 PSF dimensions

Next, ensure that the dimensions for the extracted PSF are large enough to contain all the PSF information (Figure 6.4). This will vary from system to system and the M Squared team will most likely have shared some values for you to remember. "PSF XY" sets how large the saved PSF will be in the XY dimensions, and "PSF Z" how long the PSF will be in the Z dimension.

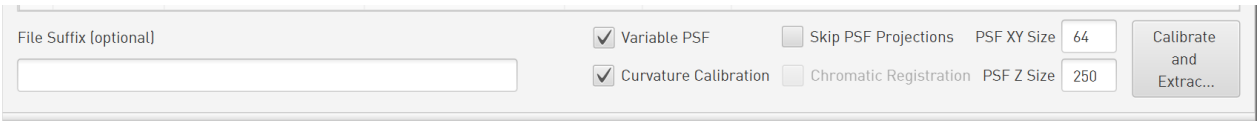


Figure 6.4 - Part of the interface where the calibration options are set.

6.4 Optional calibration inputs

6.4.1 File suffix (Optional)

If required, a custom suffix can be specified that is appended to the calibration output using the "File Suffix" text box.

6.4.2 Variable PSF

With this checkbox selected, the PSF extraction procedure will split the bead data into sections and extract the PSF for each section. This is to account for any variations that the PSF might have across the FOV and will improve the deconvolution result at the edges of the FOV.

This process will take significantly longer than extracting a single PSF and result in a much larger file, as the whole process is repeated ~85 times.

If the checkbox is not selected, the software will only extract a PSF from a 512x512 region in the centre of the FOV. While this is fine in most cased, the edges of the FOV might not have the Airy side-lobes fully removed.

6.4.3 Skip projections

If “Variable PSF” is selected, the software may generate a series of XZ projections of the PSF data cross the different sections. This is to allow the user to inspect the quality of the generated PSF’s and to evaluate whether the extraction was successful. This checkbox will tell the software to

6.4.4 Curvature calibration

Tuns on or off the beam curvature calibration procedure.

6.4.5 Chromatic correction

This option is a work in progress and is currently disabled.

6.5 Running the calibration

Click "Calibrate and extract PSF" to run the Calibration routine. Once finished, the loading bar shows the value 100% and the “PSF extraction finished” will be displayed on the GUI (Figure 6.5).

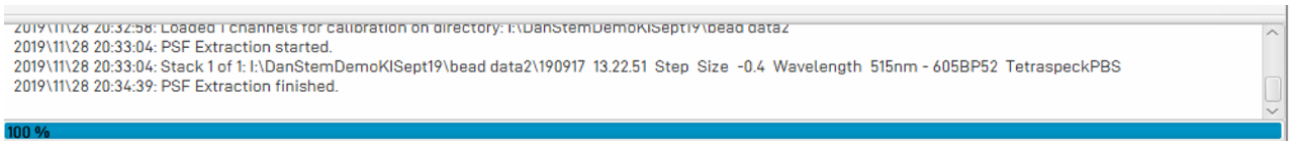


Figure 6.5 - Calibration process has finished

The PSFs that are extracted for each channel will be stored as a single “cube.tiff” file on the previously selected output folder.

7 Deconvolution

7.1 Loading data

First, make sure the "Deconvolution" tab is selected.

Press the “Load Data” button to choose a data directory to load. The directory can be a top-level directory containing many acquisitions inside. Any non-Aurora file under the directory will be ignored. The GUI will find all Aurora acquisition data in the directory (Figure 7.1) and process all of them in a batch once deconvolution has started. You can also tick and untick the “Active” checkbox to choose which files to process. More directories can be

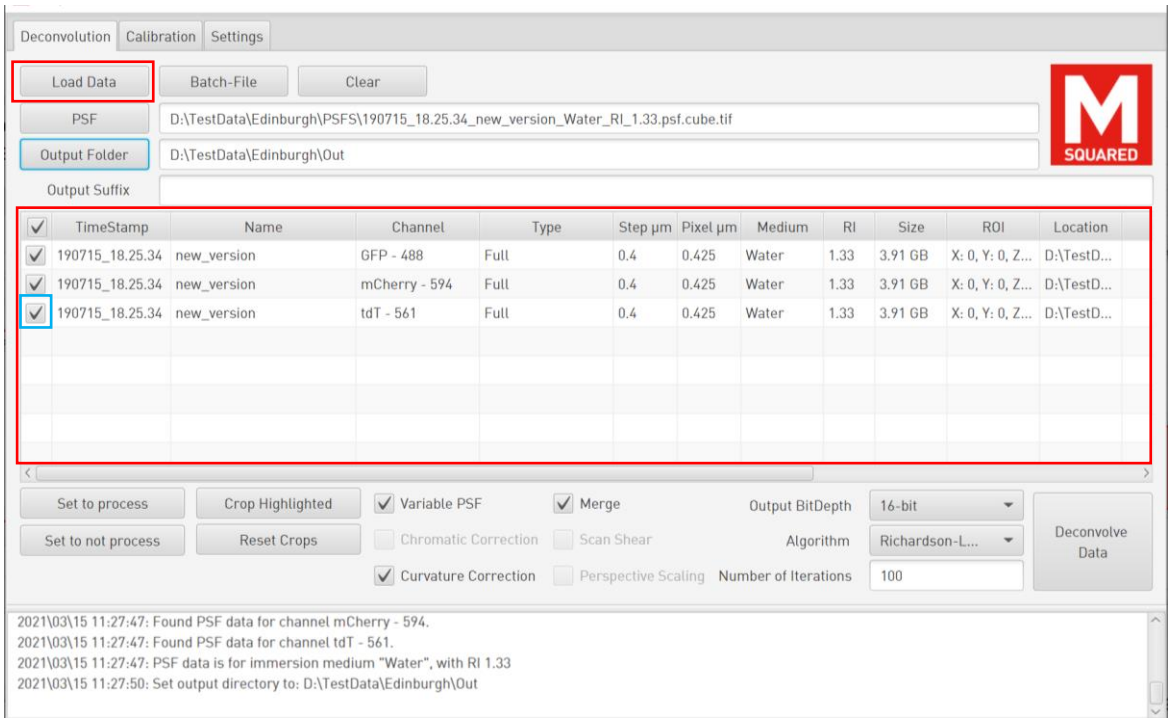


Figure 7.1 - Deconvolution tab after loading data. Red highlighted areas show the button to load data, and the table with the loaded acquisitions. The blue box highlights an example of a checkbox to select which data to process.

loaded by pressing the “Load Data” button again.

Click the "PSF" button to select the PSF file. The deconvolution can accept “**cube.tif**” files generated with the newer versions of the software or “**calibrated.json**” files from older version of the software.

7.1.1 Dual-Camera / Split-View data

Dual-camera data and Split-view data will show as the corresponding individual channels, if the acquisition software was configured correctly. For example, a “Dual-GFP-RFP” channel will show as two separate stacks in the deconvolution list of loaded data, named as “Single-GFP” and “Single-RFP”, or whatever the single-colour channels were named

as. A channel that is both dual and split-view will show up as 4 single-colour stacks. The type of data (Full, cropped, or split-FOV) should also be displayed in the “Type” column.

After the deconvolution, a separate result file will be created for each channel.

7.2 Output options

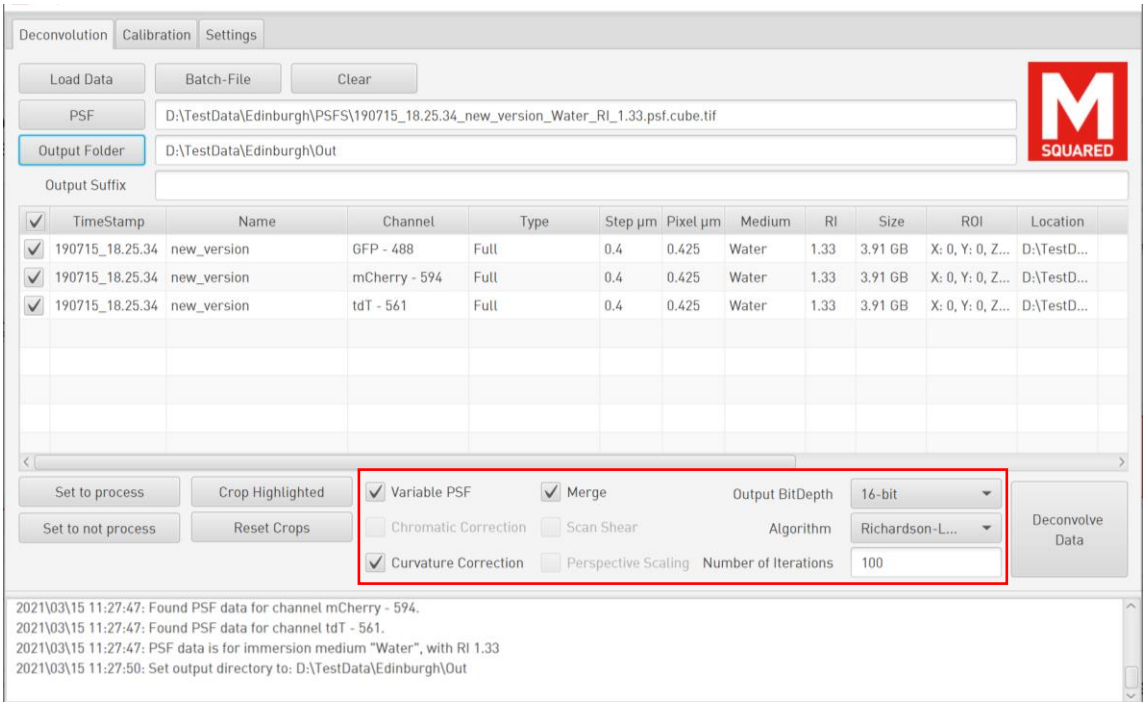


Figure 7.2 - User interface with deconvolution options highlighted

7.2.1 Output directory

By default, deconvolved files will be saved in the parent folder where the acquisition data is stored. For example, data saved under:

```
D:\Bead Data\190715_18.25.34_Step_Size_+0.4_Wavelength_GFP_tetras
```

Will have the deconvolved output saved as:

```
D:\Bead Data\190715_18.25.34_GFP_tetras.cube.tif
```

If users wish to save all deconvolved files to a single output directory, they can click "Output Directory" and choose their folder of choice. If the “Merge” option is used but no output directory is selected, the merged files will be saved in the parent folder of the acquisition folder. For example, two channels saved under:

```
D:\Bead Data\190715_18.25.34_Step_Size_+0.4_Wavelength_GFP_tetras
D:\Bead Data\190715_18.25.34_Step_Size_+0.4_Wavelength_Cy5_tetras
```

Will have the deconvolved output saved as one file:

```
D:\Bead Data\190715_18.25.34_tetras
```

7.2.2 Output BitDepth

The data can be saved either as 16-bit or 32-bit. The data is processed at 32-bit precision and saving as 32-bit will save the result as-is. However, it will also double the size of the data on disk when compared with the original acquired data size. Saving at 16-bit will reduce the size on the disk to be the same as the original data, but the intensity of the resulting data might need to be rescaled to fit the 16-bit range. This means that different positions or time-points in a time-lapse acquisition can have drastically different intensities between each other. To solve this after opening the images, the user needs to read the scaling value in the metadata (please see section 7 for more details), convert the image in memory to 32bits and scale each tile/time-point by its respective scaling value.

In short:

- *16-bits are usually advised, unless the data to be deconvolved are tile/grid positions or time-lapse acquisitions.*

7.2.3 Merge

If the merge option is ticked, deconvolved multi-colour and timelapse acquisitions will be merged into a single “cube.tif” file for each acquisition that has been loaded. If it is unchecked, each channel and time-point will be saved as a separate “cube.tif” file. If no output directory is selected, the merged files will be saved in the parent of the acquisition folder. See 7.2.1 for an example.

7.3 Deconvolution options

7.3.1 Variable PSF

Should be **ON**, as it will use the most adequate PSF for each portion of the FOV. The variable PSF has no trade-off in deconvolution speed. If no variable PSF data is present on the PSF file, this option will be ignored once deconvolution is started.

7.3.2 Curvature correction

Should be **ON**, as it will correct the data for the curvature introduced by the Airy beam. If no correction data is present on the PSF, this option will be ignored once deconvolution is started.

7.3.3 Data selection

Once the datasets are loaded, it is possible to decide which of them needs to be deconvolved. For each stack, there is a checkbox to manually select whether it will be processed or not. At the top of the stack list there is a checkbox on the header row which will also allow you to either select or de-select all stacks to be processed in bulk.

Additionally, individual stacks can be highlighted by drag-selecting and there are two buttons blow which allow you to either set or unset the highlighted stacks to be selected for deconvolution.

- “Set sel. to process” Button – Sets the highlighted datasets to be deconvolved
- “Set sel. to not process” Button – Sets the highlighted datasets to NOT be deconvolved

7.3.4 Region of interest

The datasets can be set to be cropped to process only a certain region of interest by using the “Crop Highlighted” button. First, highlight the stacks you’d like to process ((Figure 7.3, left), and then press the “Crop Highlighted” button. A dialog window will appear with the parameters for cropping (Figure 7.3, right). Once the ROI has been defined, the values chosen will appear on the software GUI under “ROI”. It is possible to remove all ROI crop information from the stacks by clicking on “Reset Crops”.

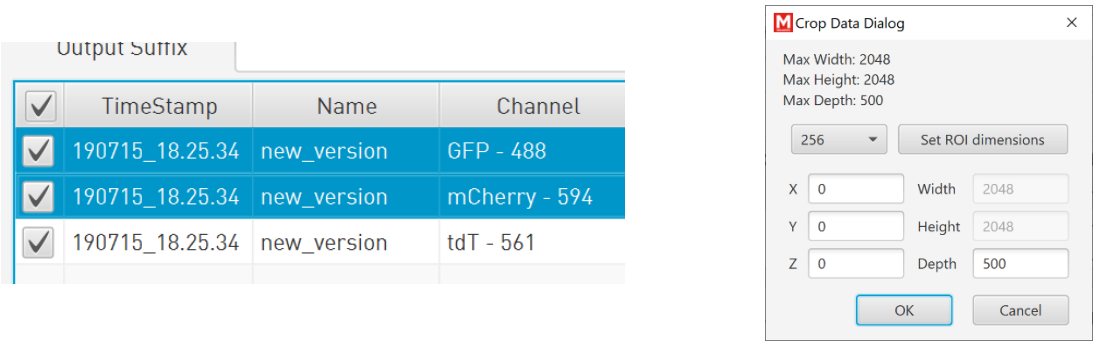


Figure 7.3 – Left – The stack list showing the first two stacks highlighted. Right - Crop dialog that opens when the “Crop Highlighted” button is pressed.

7.3.5 Algorithm

Three algorithms are currently available.

7.3.5.1 Richardson-Lucy

This is the standard algorithm used and is the recommended algorithm for deconvolution as it provides the best results.

7.3.5.2 Inverse filter

This is the simples deconvolution algorithm available. It provides a quick result but will not greatly improve resolution and is liable to introduce artefacts. It is useful to get a quick result when quality and resolution are not as important.

7.3.5.3 Wiener-Butterworth

This is a new, experimental algorithm that is similar to Richardson-Lucy. However, it should result in a result that is roughly similar in only 7-10 iterations. Early results are promising but we've observed that the resolution obtained is not as great and some artefacts can be introduced. Some features also seem not to recover as well. However, it is a good middle-ground result between Richardson-lucy and the inverse filter.

7.3.6 Number of iterations / Regularization factor

Depending on the algorithm selected, we can either set the number of iterations (Richardson-Lucy and Wiener-Butterworth) or a regularization factor (Inverse Filter).

7.3.6.1 Richardson-Lucy and Wiener-Butterworth

Richardson-Lucy and Wiener-Butterworth are both iterative algorithms, where they will try to continuously improve the result over several iterations. You can input the desired number of iterations (Figure 7.2, "Number of iterations") according to the following guidelines:

- 1-5 to test the deconvolution pipeline and see if all the required datasets are loaded;
- 10-30 for a "quick" deconvolution run;
- 100 for a recommended "full" deconvolution.

The larger the number of iterations, the longer the deconvolution run will take. Quality should improve as well, but artefacts might appear. If artefacts appear, try reducing the number of iterations.

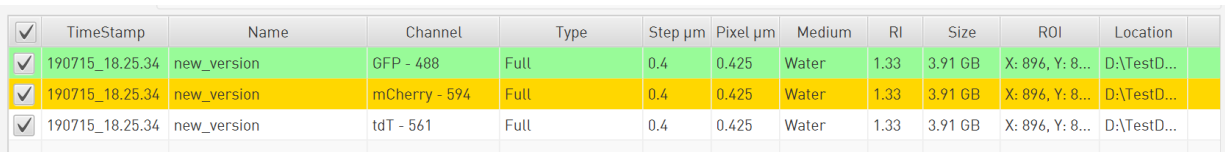
7.3.6.2 Inverse Filter

The Inverse Filter algorithm is a single step process but uses a regularization term that controls how it handles noise and low signal. If your results look incorrect, try and change the regularization factor in powers of ten, i. e, decrease from 0.0001 to 0.001.

7.4 Start the deconvolution

Finally, click "Deconvolve Data" to begin the deconvolution routine. Once the deconvolution starts, each stack will be highlighted according to a colour code:

- Yellow – Stack is currently being processed.
- Green – Stack finished processing successfully.
- Red – Stack stopped processing due to an error.



TimeStamp	Name	Channel	Type	Step µm	Pixel µm	Medium	RI	Size	ROI	Location
190715_18.25.34	new_version	GFP - 488	Full	0.4	0.425	Water	1.33	3.91 GB	X: 896, Y: 8...	D:\TestD...
190715_18.25.34	new_version	mCherry - 594	Full	0.4	0.425	Water	1.33	3.91 GB	X: 896, Y: 8...	D:\TestD...
190715_18.25.34	new_version	tdT - 561	Full	0.4	0.425	Water	1.33	3.91 GB	X: 896, Y: 8...	D:\TestD...

Figure 7.4 - User interface during deconvolution demonstrating the "in progress" and "finished" colour states for 2 of the 3 stacks currently processing.

7.5 Saving settings

The Deconvolution settings can be saved to disk as batch-files and loaded again later. This can be done by pressing the “Batch” button on the Deconvolution tab. When the button is pressed, a menu appears with a few options (Figure 7.4).

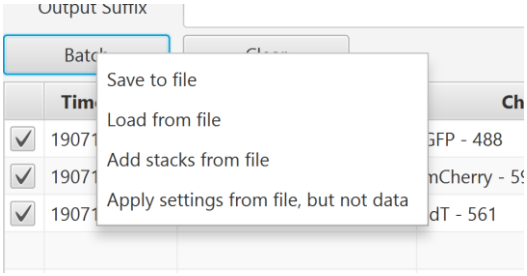


Figure 7.5 – Menu that appears when the Batch button is pressed.

7.5.1 Save to file

This will save the current settings on the GUI as a “.batch.json” batch-file to disk.

7.5.2 Load from file

This will load a “.batch.json” batch-file from disk and apply the settings on that file to the GUI.

7.5.3 Add stacks from file

This will load the stacks defined on the batch-file but not settings like number of iterations, etc…

7.5.4 Apply settings from file, but not data

This will load settings like the number of iterations, etc… but ignore the input directory and any manually specified stacks.

7.6 Clearing settings

The Deconvolution settings can be cleared by pressing the “Clear” button on the Deconvolution tab (Figure 7.5).

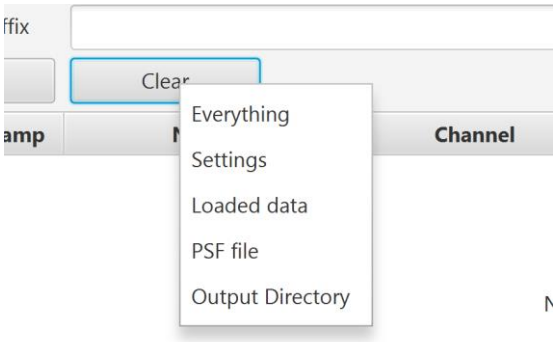


Figure 7.6 – Menu that appears when the “Clear” button is pressed.

7.6.1 Everything

Will reset all interface settings to their defaults and remove any loaded data.

7.6.2 Settings

Will reset all interface settings to their defaults but keep any loaded data.

7.6.3 Loaded data

Will remove any loaded data from the list, but keep any user settings like number of iterations, etc…

7.6.4 PSF File

Will clear the currently selected PSF file.

7.6.5 Output Directory

Will clear the currently selected output directory.

8 Opening deconvolved data saved as cube.tif

The deconvolution routine output is saved as a cube.tif image stack (Figure 8.1).

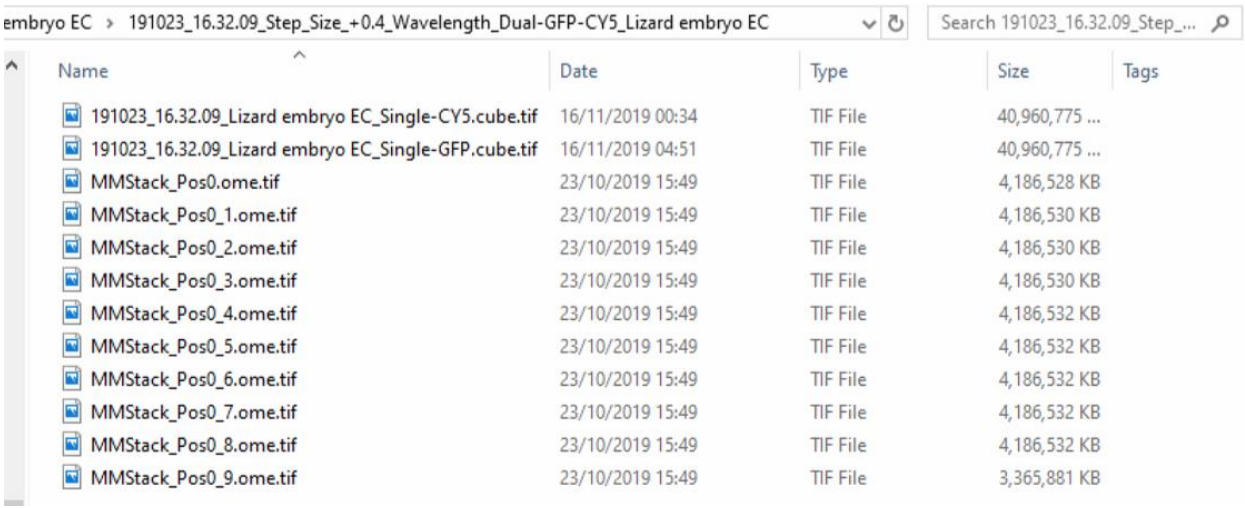


Figure 8.1 - Example output folder containing the original TIF and output cube.tif

The cube.tif files are OME-TIFF files that have minor changes to allow fast saving/opening and for the M Squared metadata to be saved with them. The OME-TIFF format is universally compatible with any modern image processing software.

8.1 Tested software

So far, the below software was shown to open cube.tiff files out of the box, and to immediately interpret the special metadata (pixel size and step size) correctly..

- FIJI / ImageJ
- Arivis Vision4D
- Bitplane Imaris

8.1.1 M Squared FIJI plugin

To accelerate opening full cube.tiff files and to easily view the metadata in FIJI, we provide a plugin. This can be easily added by following our update website:

https://fileformat.msquaredcubes.com

If you don’t know how to add this, please follow the instructions on this link:

https://imagej.net/Following_an_update_site

To Add the following update site address in the URL field::

Once the update site has been added, cube.tiff files can be dragged and dropped on to FIJI to open more quickly than with the standard FIJI opener, and the metadata can be viewed by using the “Image > Show Info…” menu option.

8.1.2 More loading options in FIJI

If the above update site is not available, the full metadata of the files can still be viewed by using the BioFormats Importer. This can be found under “Plugins>Bio-Formats>Bio-Formats Importer”. The bio-formats importer also allows for partial loading of the file, which is helpful for very large files.

Once this option is chosen, the importer will ask you for an input file. Choose a cube.tiff file and the below window will be displayed.

- To see the file metadata, choose Show OME-XML metadata.
- To only load a cropped XY area of the full file, tick “Crop on import”
- To only load a portion of the image stack, tick “Specify range for each series”.

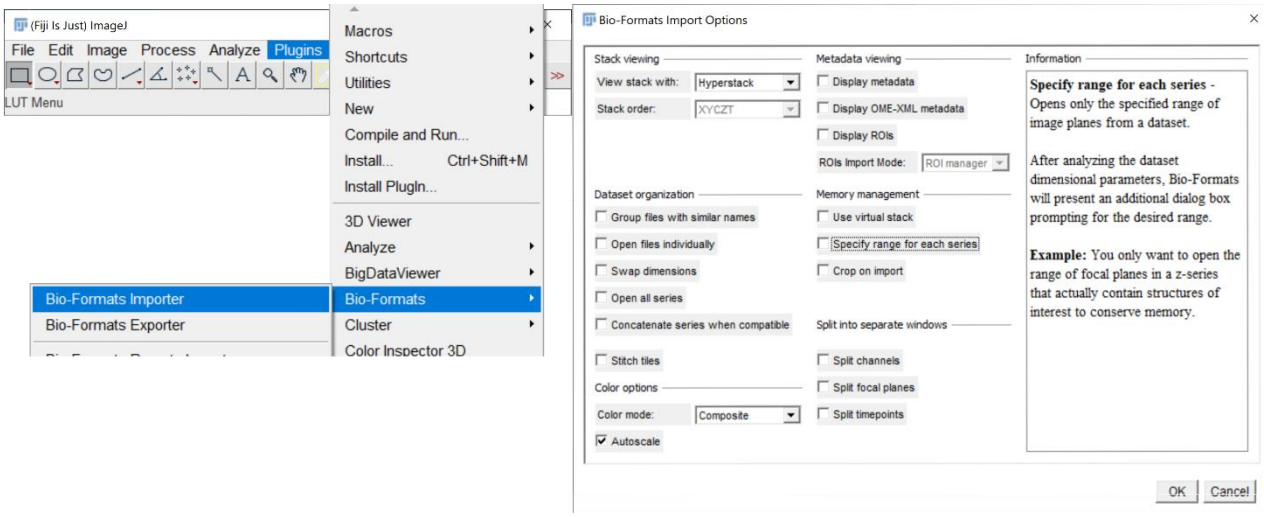


Figure 8.2 - Bioformats Import plugin

9 Command-line and batch-file control

The Aurora Deconvolution software can have its settings manipulated from the command-line. These settings can be provided either as command-line arguments or by providing a batch-file in the JSON format.

Command-line arguments are passed directly to the `AuroraDeconvolution.exe` binary executable. Batch-files are passed by using the `batch_file` argument.

9.1 Providing arguments

To start using a command-line argument, you first need to open a terminal and browse to the Aurora Deconvolution software install directory.

1. On Windows, the easiest way to do this is to first open the directory where the software is installed.
    a. If you don't know where this is, you can right click the desktop short-cut and choose "Open file location".
2. On the explorer window that opens, click on "File" on the top right corner of the window.
3. Click "Open windows PowerShell". This will start a windows command-line terminal with the current directory set to the Aurora Deconvolution software directory.

On Windows there are two types of terminals:

- The Windows PowerShell, which is the recommended terminal. The above instructions show you how to start the PowerShell.
- The older Command Prompt. This is still in wide use by many users and you should feel free to use it. But the below instructions assume the use of PowerShell.

Starting the software on the PowerShell requires "`./`" to be added before the "`AuroraDeconvolution`" command but the Command Prompt does not. This guide will show examples for PowerShell as that is the recommended terminal. If you are using the command prompt, please ignore the "`./`" that may be present on some examples.

To start the Aurora Deconvolution software, type the name of the executable "`./AuroraDeconvolution`" (which has the "`.exe`" extension on Windows), followed by any number of command-line arguments, separated by spaces. Once you press enter, the software will immediately start with the currently provided command-line arguments.

More information is provided below about the different types of arguments and how to provide them to the executable.

9.1.1 Flag-type arguments

Some arguments serve as flags to the software to change its default behaviour. For example, the below command:

```
./AuroraDeconvolution.exe no_curv_corr
```

will start the software with the curvature correction option turned off (whereas it is turned on by default).

9.1.2 Input type arguments

Some arguments are provided in the form of:

```
arg=input
```

or

```
arg="input"
```

Where `arg` is the argument and `input` is the input that the argument is meant to provide the software. For example, the argument `psf` requires the location of a PSF file. To use it, you'd pass the argument as:

```
psf=G:\Beads\190715_18.25.34_Water.psf.cube.tif
```

However, if the file path has a space in it, you'll have to use double-quotes. For example:

```
psf="G:\Bead Data\190715_18.25.34_Water.psf.cube.tif"
```

9.1.3 Multiple arguments

The Deconvolution software is based on the Java programming language. When providing multiple arguments, these are separated by a space character. For example, starting the software will the below command:

```
./AuroraDeconvolution.exe cpu_only no_curv_corr
```

will start the software with the GPU's disabled and the curvature correction option will be turned off by default.

It should also be noted that the order of the arguments does not matter.

9.2 Standard command-line arguments

The below table summarizes arguments that will be useful to most users.

Argument	Type	Default	Description
`psf`	Input	-	Tells the software to load the given PSF file on start-up.
`in`	Input	-	Tells the software to load the stacks in the given Input Directory file on start-up.
`out`	Input	-	Sets the given directory as the output directory for the deconvolution.

suffix	Input	-	Sets the output suffix for the deconvolved data.
split	Flag	-	The software will start with the "Merge" option turned off. Each stack will be saved as a separate ".cube.tif" file.
no_curv_corr	Flag	-	The software will start with the "Curvature correction" option turned off.
no_chrom_corr	Flag	-	The software will start with the "Chromatic correction" option turned off.
no_variable	Flag		The software will start with the "Variable PSF" option turned off.
32bits	Flag	-	The software will start with 32-bits set as the output bit-depth for the deconvolved data.
iter	Input	100	Gives the number of iterations the software to use for deconvolution.
start	Flag	-	If this flag is present, the software will immediately start a deconvolution with the provided settings. For the deconvolution to start successfully, at least a psf file and an input directory need to be provided. Alternatively, a batch-file can be used (see more in 9.4).

9.3 Development and testing command-line arguments

The below table summarizes arguments that are mostly used for development and testing.

Argument	Type	Default	Description
debug	Flag	-	Starts the software in debug mode. This means all debug output is displayed on a terminal along-side the GUI.
cpu_only	Flag	-	Disables the GPU(s).
alg	Input	rl	Sets the Algorithm. Currently accepted values are rl, inverse are bw.
reg	Input	0.0001	If using alg=inverse, this sets the regularization factor for the algorithm.

9.4 Batch-file control

Instead of directly providing command-line arguments, a batch-file can be used. This allows the same set of settings to be provided each time. They also enable more advanced usage.

9.4.1 Command-line arguments related to batch-file usage

The table below summarizes command-line arguments that are specific to batch-file usage.

Argument	Type	Description
`batch_file`	Input	Starts the software with the settings provided in the batch-file.
`generate_batch`	Flag	Meant to be used in conjunction with other command-line arguments. Tells the software to generate a batch-file with the given command-line arguments and finish execution immediately. The file will be saved to the same directory as `AuroraDeconvolution.exe`.
`explicit`	Flag	To be used with `generate_batch`. This will tell the software to include all possible batch-file settings when generating the batch-file.
`save_to`	Input	To be used with `generate_batch`. This allows the user to specify the name and location of the generated batch file.

9.4.2 Generating and editing batch-files

There are two ways of generating a batch-file.

1. Start the user interface and edit the settings you wish to save. You can then use the “Batch” button and the “Save to file” option to save the settings to a batch-file.
2. Using the `generate_batch` command-line argument in conjunction with other command-line arguments. This tells the software to generate a batch-file with the given command-line arguments on the same directory as `AuroraDeconvolution.exe`, or wherever the `save_to` argument specifies. Please note that the software will finish execution immediately after generating the batch-file.

Once a batch-file has been generated, it can be edited with any text-editing program. Batch-files follow the JSON specification. Therefore, care must be taken while editing to ensure the JSON specification is adhered to. To generate a batch-file that contains all possible settings in the correct JSON specification, use the command-line arguments `generate_batch explicit`.

9.4.3 Batch-file settings

While the settings in a batch-file can control the same settings as the above command-line arguments, they are not named in the same way. This is to make batch-files more readable than command-line arguments, which are designed to be short and easy to type.

The below table lists all possible batch-file settings and their command-line argument equivalent (CLA column).

Setting	Type	CLA	Description
`Auto start`	Boolean	`start`	–
`Algorithm`	String	`alg`	Valid inputs are `"Richardson-Lucy"` or `"Inverse Filter"`
`Regularization factor`	Float	`reg`	-
`Disable GPU`	Boolean	`cpu_only`	-
`Load directory`	String	`in`	-
`Output directory`	String	`out`	-
`PSF file`	String	`psf`	-
`Merge data`	Boolean	`split`	-
`Do curvature correction`	Boolean	`no_curv_corr`	-
`Do chromatic correction`	Boolean	`no_chrom_corr`	-
`Use variable psf`	Boolean	`no_variable`	
`Bit Depth`	Integer	`32bits`	Valid inputs are `16` or `32`. All other values will be read as `16`
`Number of Iterations`	Integer	`iter`	-
`Output suffix`	String	`suffix`	-
`Stacks`	JSON	–	This is a special setting. See section 9.4.4for more information.

9.4.4 Stacks - Individual stack entries

The batch-file specification allows for individual stacks to be specified, instead of just a single input directory. The `Stacks` setting is a JSON formatted field that lists individual stacks to load. Each stack to load also includes a region of interest (ROI) crop setting.

> *Please note: If the `Input directory` setting is also set or the `in` command-line argument is provided, the `Stacks` setting will override either of those settings.*

Below is an example `Stacks` entry which will load two stacks located in different directories:

```
"Stacks": [
    {
      "ROI": {
        "depth": 100,
        "x": 768,
        "width": 512,
        "y": 768,
        "z": 200,
        "height": 512
      },
      "Directory": "D:\\Temporary Data Storage\\Edinburgh\\Bead
Data\\190715_18.25.34_Step_Size_+0.4_Wavelength_GFP - 488_new_version"
    },
    {
      "ROI": {
        "depth": 500,
        "x": 0,
        "width": 2048,
        "y": 0,
        "z": 0,
        "height": 2048
      },
      "Directory": "D:\\Temporary Data
Storage\\Karolinska\\Organoid\\200120_18.25.34_Step_Size_+0.4_Wavelength_GFP_or
ganoid"
    }
  ]
```

To create such an entry, it's best to start by generating a template batch file. A batch file can be created by:

- Using the command-line to generate a batch-file. You can get more information on how to do this on section 9.4.2.
- Using the GUI. You'll need to load some data, optionally set crop settings and then use the "Batch" button to save a new batch-file to disk with the relevant `Stacks` entry.

Once a batch file has been generated, it will have a `Stacks` entry with whatever stacks were loaded using the command-line or the GUI. These entries can be manually edited using any text editor to:

- Manually add or remove stacks to process
- Change ROI settings for each stack

Future development of the software will allow each stack to have its own properties like number of iterations, etc…