

Input Module Tutorial #2: Using the *Metadata* Module

This tutorial describes how to use the **Metadata** module for connecting information about the images (i.e., metadata) to your list of images for processing in CellProfiler. Since metadata is not necessary for every assay, **setting this module is optional**.

The **Metadata** module allows you to extract and associate metadata with your images. The metadata can be extracted from the image file itself, from a part of the file name or location, and/or from a text file you provide.

For the purposes of this tutorial, you can use a set of files from a translocation assay which are available from http://www.cellprofiler.org/linked_files/TranslocationActivity/TranslocationData.zip.

What exactly is “metadata,” anyway?

The term *metadata* generally refers to "data about data." For many assays, metadata is important in the context of tagging images with various attributes, which can include (but is not limited to) items such as the following:

- The row and column of the microtiter plate from which the image was acquired.
- The experimental treatment applied to the well from which the image was acquired.
- The number of timepoints or channels contained in the image file.
- The image type, i.e., RGB, indexed, or separate channels.
- Etc.

It can be helpful to inform CellProfiler about certain metadata in order to define a specific relationship between the images and the associated metadata. For instance:

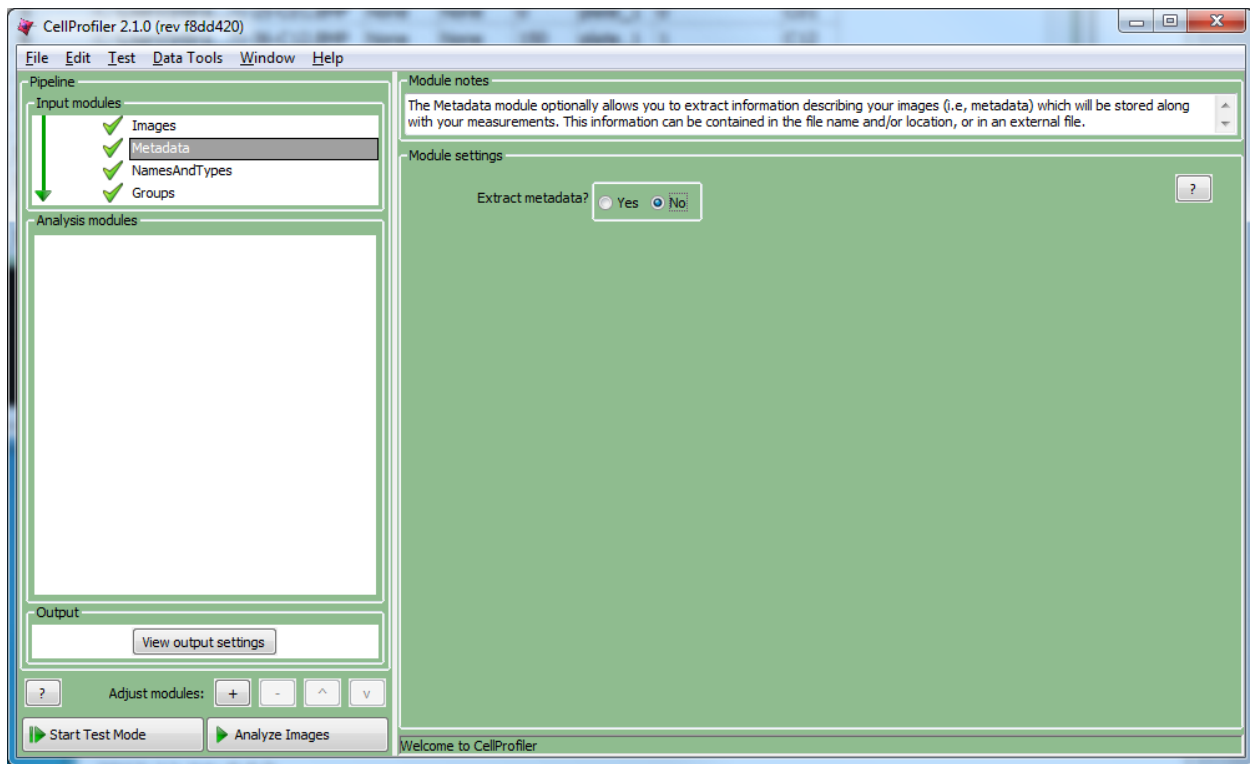
- You want images with a common tag to be matched together so they are processed together during the pipeline run. E.g., the filenames for fluorescent DAPI and GFP images contain different tags indicating the wavelength but share ‘_s1’ in the filename if they were acquired from site #1, ‘_s2’ from site #2, and so on.
- You want certain information attached to the output measurements and filenames for annotation or sample-tracking purposes. E.g., some images are to be identified as acquired from DMSO treated wells, whereas others were collected from wells treated with Compound 1, 2... and so forth.

The underlying assumption in matching metadata values to image sets is that there is an exact pairing (i.e., a one-to-one match) for a given combination of metadata tags. A common example is that for a two-channel microtiter plate assay, the values of the plate, well, and site tags from one channel get matched uniquely to the plate, well, and site tag values from the other channel.

How do I start using this module?

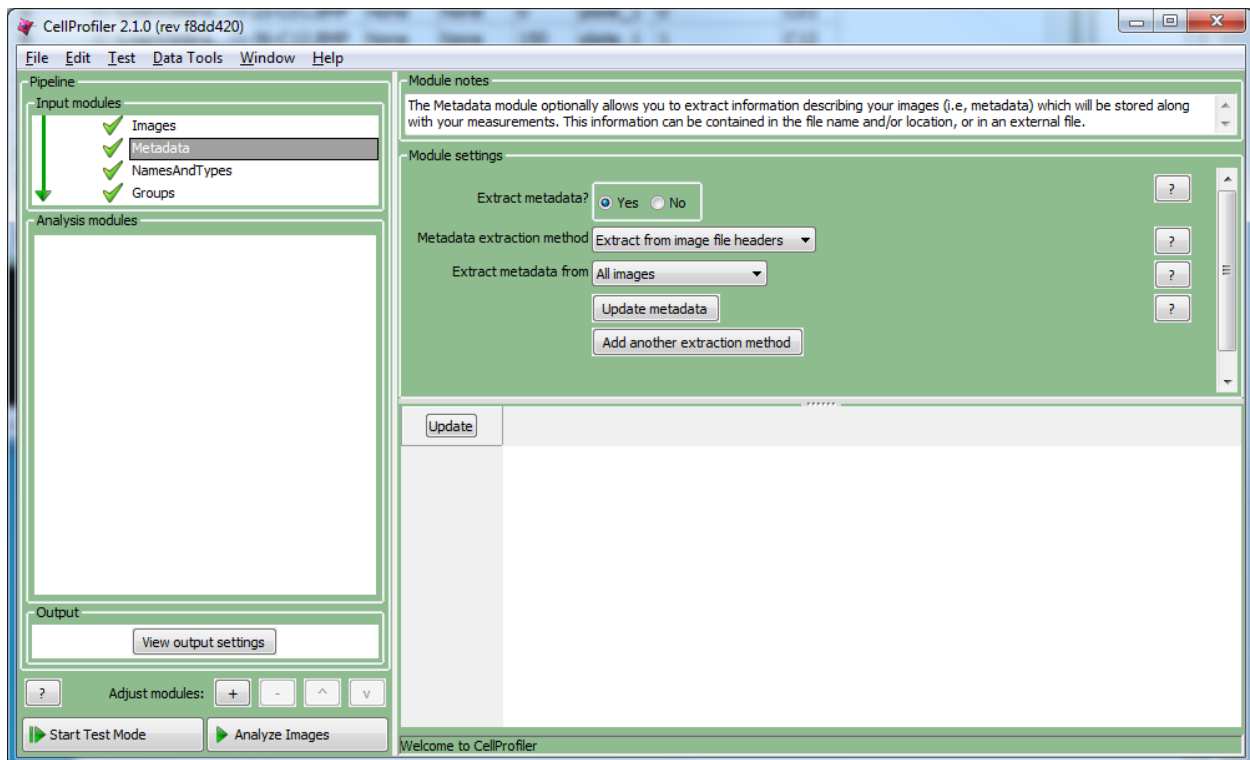
Using this module assumes that you have already opened CellProfiler and used the **Images** module to produce a list of images to analyze. If you have not done so, please refer to the [tutorial](#) for the **Images** module before progressing further here.

The **Metadata** module is the second Input module, shown in the “Input modules” panel on the upper left. Selecting this module will display a panel, allowing you to select whether you want the extract metadata or not.



If you do not have metadata that is relevant to your analysis, you can leave this module in the default setting, and continue on to the **NamesAndTypes** module (Input module #3; see [tutorial](#)).

If you do have relevant metadata, select the button for extracting metadata; a new group of settings will appear below:



The **Metadata** module receives the file list produced by the **Images** module. It then associates information to each file in the File list, which can be obtained from several sources, available from the “Metadata extraction method” setting:

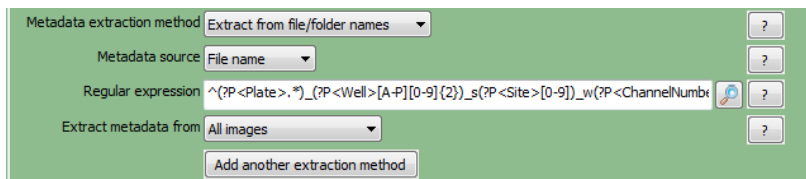
- From the image file name or location (e.g., as assigned by a microscope). In this case, you will provide the text search pattern to obtain this information (“Extract from file/folder names”).


- In a text file created and filled out by you or a laboratory information management system. In this case, you will point the module to the location of this file (“Import from file”).
- In the image file itself (“Extract from image file headers”).

Specifics on the metadata extraction options are described below. Any or all of these options may be used at time; press the “Add another extraction method” button to add more.

1. Extract from file/folder names

This approach retrieves information based on the file nomenclature and/or location. A special syntax called “regular expressions” is used to match text patterns in the file name or path, and then assign this text as metadata for the images you specify. The tag for each metadata is assigned a name that is meaningful to you, e.g., “Plate” between the ‘<’ and ‘>’ signs in the figure below.

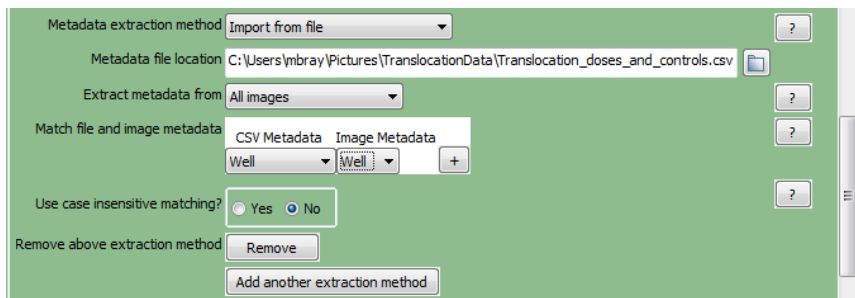


Regular expressions can be daunting at first, but quite powerful for this use; we encourage the use of the regular expression editor (available via the  icon) to experiment and verify your regular expression with a test file name or path of your choice. The editor also includes a “Guess” button to try to fill in the appropriate regular expression if the images come from a commercial automated microscope. You have the option of applying your regular expression to all the image files coming in from the **Images** module, or using filtering rules to select only a subset of images. There is more information on the syntax for regular expressions in the Help for this module.

When would you want to use this option? If you want to take advantage of the fact that acquisition software often automatically assigns a regular nomenclature to the filenames or the containing folders. Alternately, the researcher acquiring the images may also have a specific nomenclature they adhere to for bookkeeping purposes.

2. Import from file

This option retrieves metadata from a comma-delimited file (known as a CSV file, for comma-separated values) of information; you will be prompted to specify the location of the CSV file. You can create such a file using a spreadsheet program such as Microsoft Excel.



The CSV file needs to conform to the following format:

- Each column describes one type of metadata.
- Each row describes the metadata for one image site.
- The column headers are uniquely named. You can optionally prepend “Metadata_” to the header name in order to insure that it is interpreted correctly.

- The CSV must be plain text, i.e., without hidden file encoding information. If using Excel on a Mac to edit the file, choose to save the file as “Windows CSV” or “Windows Comma Separated”.

This option is a convenient way for you to add metadata from your own sources to the output generated by CellProfiler. The most common usage is to use a CSV in conjunction with the filename/path metadata matching, such that both options are capturing metadata in common. For example, you might be extracting the well tag from the image filename while your CSV contains treatment dosage information paired with each well. Therefore, you would want to let CellProfiler know that the well tag extracted from the image filename and the well tag noted in the CSV are in fact the one and the same.

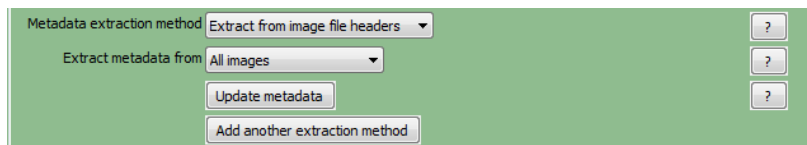
To expedite this process, you can set the following options:

- Set the drop-downs to pair the metadata tags of the images and the CSV, such that each row contains the corresponding tags. This can be done for as many metadata correspondences as you may have for each source; press to add more rows.
- Set whether you want case insensitive metadata matching. For example, if you want well ‘A01’ extracted from the image filenames to correspond to well ‘a01’ extracted from the CSV, enable this option.

When would you want to use this option? You have information curated in software that allows for export to a spreadsheet. This is commonly the case for laboratories that use computerized data management systems that track samples and acquisition.

3. Extract from image file headers

This option retrieves information from the internal structure of the file format itself. Typically, image metadata is embedded in the image file as header information; this information includes the dimensions and color depth among other things. If you select this method, press the "Update metadata" button to extract the metadata. Note that this extraction process can take a while for assays with lots of images since each one needs to be read for extraction.



Since the metadata is often image-format specific, this option will extract information that is common to most image types:

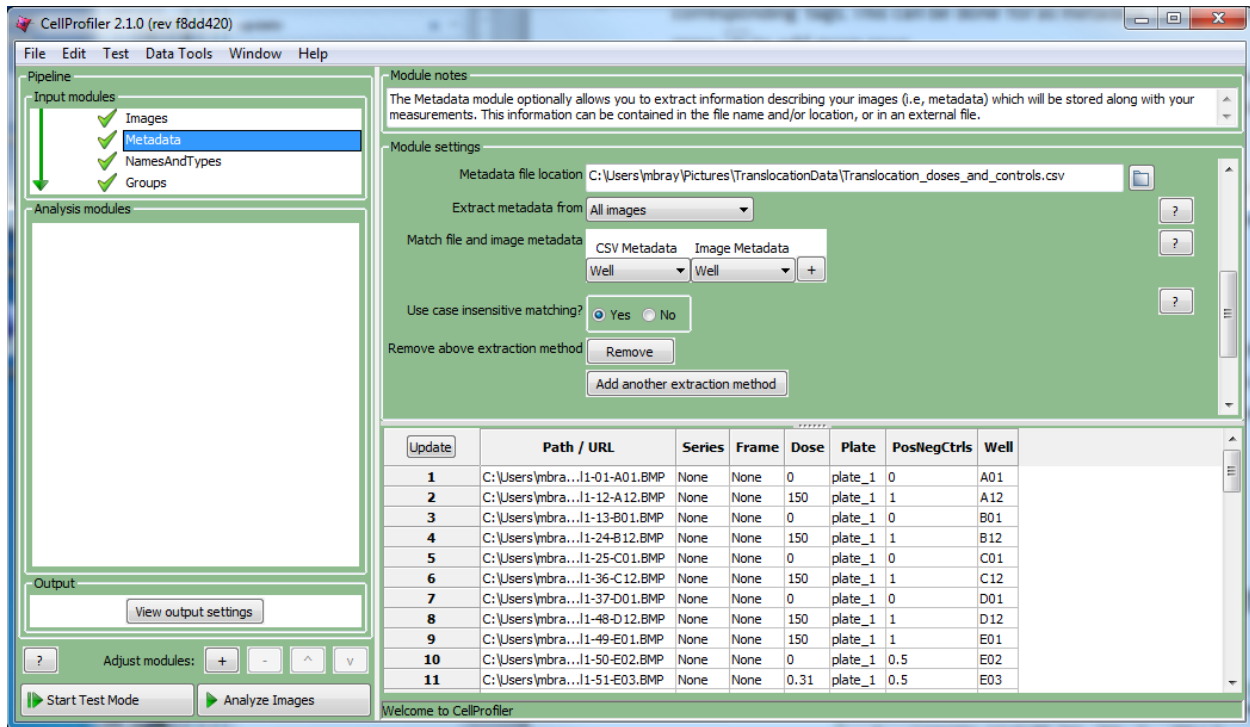
- *Series*: The series index of the image. This value is set to “None” if not applicable. Some image formats can store more than one stack in a single file; for those, the “Series” value for each stack in the file will be different.
- *Frame*: The frame index of the image. This value is set to “None” if not applicable. For stack frames and movies, this is the frame number for an individual 2-D image slice.
- *ColorFormat*: This value is set to “Monochrome” for grayscale images, “RGB” for color.
- *SizeZ*: The number of image slices, or Z-stack size. This value is typically > 1 for confocal stacks and the like.
- *SizeT*: The number of image frames, or timestamps. This value is typically > 1 for movies.
- *SizeC*: The number of color channels. This value is typically > 1 for non-grayscale images and for confocal stacks containing channel images acquired using different filters and illumination sources.

When would you want to use this option? If you want to analyze images that are contained as file stacks, i.e., the images are related to each other in some way, such as by time (temporal), space (spatial), or color (spectral).

How do I know whether I’m setting everything correctly?

As you are extracting metadata from your various sources, you can click the “Update” button below the divider to display a table of results using the current settings. Each row corresponds to an image file from the **Images** module, and the columns display

the metadata obtained for each tag specified. You can press this button as many times as needed to display the most current metadata obtained.



In addition, you can set whether the metadata is stored as either a text or numeric value. The default is text, but there are instances where you would want to choose the data type separately for each metadata entry.

- An example of when this approach would be necessary would be if a whole filename is captured as metadata but the file name is numeric, e.g., "0001101". In this situation, if the file name needs to be used for an arithmetic calculation or index, the name would need to be converted to a number and you would select "Integer" as the data type.
- On the other hand, if it important that the leading zeroes be retained, setting it to an integer would remove them upon conversion to a number. In this case, storing the metadata values as "Text" would be more appropriate.

What should I have when I'm done with this module?

The final product of the **Metadata** module is a list of files from the **Images** module, accompanied by the associated metadata retrieved from the source(s) provided and matched to the desired images.

Some downstream use cases for metadata include the following:

- If the metadata establishes how channels are related to one another, you can use them in the **NamesAndTypes** module to aid in creating an image set.
- If the images need to be further sub-divided into groups of images that share a common metadata value, the **Groups** module can be used to specify which metadata is needed for this purpose.
- You can also use metadata to reference their values in later modules. Since the metadata is stored as an image measurement and can be assigned as an integer or floating-point number, any module which allows measurements as input can make use of it. For example:
 - The **CalculateMath** module can perform arithmetic operations using metadata values.
 - The **CalculateStatistics** module uses metadata to identify the images representing your controls and assign treatment dosages.
- Several modules are also capable of using metadata for more specific purposes. Refer to the module setting help for additional information on how to use them in the context of the specific module.

- **CreateWebPage:** If a zip file is used to contain the full-size images, this file can be named according to metadata values.
- **SaveImages:** The filename and/or the path of the saved image can be named according to the metadata value that is currently being processed during the analysis run. For example, you can assign plate-specific names to saved images during an analysis run which covers multiple plates.
- **ExportToDatabase:** The output files can be saved to folders which are named according to metadata values.
- **ExportToSpreadsheet:** Metadata tags can be used to name spreadsheets and output folders.
- **StraightenWorms:** The location of the training set file can be specified in a folder named with metadata tags.