

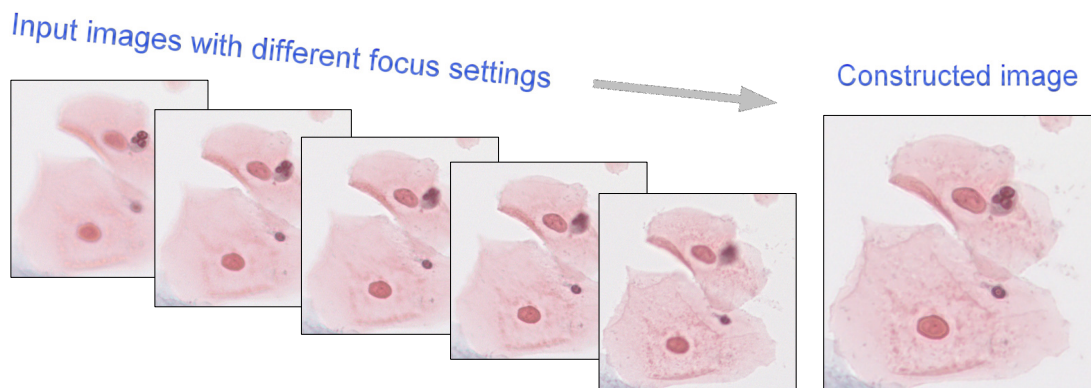
# MultiFocus

## GENERATE A COMPOUND, FOCUSED IMAGE FROM A SET OF IMAGES CAPTURED AT VARYING DEPTHS OF FIELD

Analyzing thick samples using an optical microscope is quite often a challenging process. As microscopes have a limited depth of field, it is almost impossible to acquire a fully focused image in a single acquisition frame using a regular camera. To overcome this limitation a set of images is captured either automatically using a confocal microscope to generate a Z stack of images, or by capturing a set of images at different microscope stage Z values. However, no one

of these images displays all objects of interest fully focused.

The Aphelion MultiFocus extension automatically constructs a single, focused image from multiple images acquired at different depths of field. All of the objects of interest are sharp and clear in the constructed image.



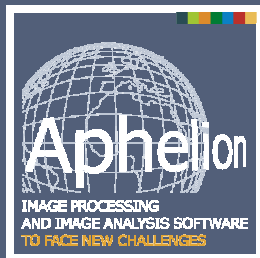
## MultiFocus Extension Architecture

There is no limit on the number of images in an input set, and a set may consist of either opaque or transparent, either color or black and white images.

The user is provided with two different methods for generating the constructed image. One method is based on regional texture analysis in each input image, and the other method is based on morphological residues. Once a method is chosen, generation of the constructed image is automatic.

Since MultiFocus is a plug-in for Aphelion Dev, one can concurrently display all input images using Dev and the constructed image can be further processed in Dev.

MultiFocus can work in concert with two other Dev plug-ins, Stage Controller and Virtual Image Capture, to provide a complete system for microscope image capture, processing, and management.



### Main benefits of MultiFocus:

- Construct focused image from either black & white or color image sets
- Optimize results by user selection of best method & fine-tuning a few parameters
- Easily integrate MultiFocus into a stand-alone application
- Increase productivity using confocal microscope Z-stack images