Computational Image Processing in Microscopy

PART 1. IMAGE PROCESSING

Biologists are facing an explosion in the amount of data generated, including large complex imaging data sets. Extracting reliable quantitative information from image data sets requires the creation of computational image processing pipelines. Image processing means using computational algorithms, sometimes with manual intervention, to extract information (features, measurements, patterns, etc.) from a digital image. Although the same principles apply to the processing of many types of images, here we focus on processing microscopy images. For example, the concentration of a fluorescent protein in each nucleus can be measured over time as the plant develops. Image processing detects each nucleus individually, allowing for measurement of size and quantification of the fluorescence, and tracks the nucleus over time. Thus, image processing is an important part of the computational morphodynamics approach, which combines experimental live imaging with image processing and computational modeling to understand the biology of the living, growing organism.

Typical Image Analysis Pipeline

Each type of image requires the development of its own computational image processing pipeline to detect and quantify the features of interest. Fortunately, many different software packages are already available and more are being developed as this is an active area of research. For example, the MorphoGraphX package is particularly suited for measuring cell growth and division over time on the curved surface of the tissue. The MARSALT package is particularly good for tracking cells over time in three dimensional samples. Many other plant image analysis tools are available (http://www.plant-image-analysis.org/). Matlab has an image processing tool box for coding new methods.

Here, we focus on Fiji, a versatile image analysis package for visualizing, manipulating, and analyzing scientific images. Within Fiji we will use the COSTANZA (COnfocal STack ANalyZer) plugin to identify nuclei in a 3D image and to extract quantitative data for the sizes and fluorescent intensities of the nuclei. The COSTANZA plugin includes the entire image analysis pipeline.

A typical computational image analysis pipeline starts with optimizing the imaging settings to make the initial images amenable for computational processing. Next the image is preprocessed to further optimize it for detecting specific features. Removing background noise which may obscure the signal with a denoising filter is a typical pre-processing step. Then the features of the image are automatically identified through segmentation. After segmentation, additional information about the features in the image can be used for post-processing. For example, the minimum size of nuclei can be used to exclude erroneous segmented features that are smaller than a nucleus. In the semi-automated approach, the human user can intervene at this step to correct errors in the segmentation. Finally, the data extracted from the post-processed segmentation is analyzed. For example, the number of cells can be counted, the size of nuclei can be measured, and the intensity of fluorescent proteins in each nucleus can be determined. As an example, throughout this lecture, we process an image of fluorescent nuclei in a developing Arabidopsis leaf with COSTANZA. The workshop portion at the end guides you step by step in the use of COSTANZA to process a sample image.

Step 1. Imaging

The optimization of an image for computational processing is often not the same as for visualization by a human. Many beautiful images are nearly impossible to process computationally. The best images for computational processing are simple images, in which the object of interest is in a single color of nearly uniform intensity on a featureless background. It is critical that a single feature is shown in a single color. For example, if the nuclei and the plasma membranes are the same color, it will be very difficult to accurately detect either one of these features alone. Likewise, a bright field image can be very difficult to segment when both the features and background are varying shades of colors and grays. The higher the contrast between the object and the background the better. Adjusting the image acquisition settings to have near saturation of the object and the absence of signal in the background can help achieve accurate detection of the borders of the object although the image may not be visually pleasing. For example, if you are trying to segment a cell, having some background signal inside the cell, or gaps in the plasma membranes, will cause incorrect segmentation and generally require some intervention by hand.

Standard Light Microscopy (Bright Field)

Light microscopes use visible light and magnifying lenses to visualize specimens. Generally, light comes from a bulb in the base of the microscope and is reflected up through the condenser lens, passes through the specimen on the stage, into the objective lens, and up to either the camera or the eyepieces. For standard light microscopes, light must transmit through the specimen, which often requires clearing or sectioning. Stains are often used to add contrast to the feature of interest. For example, the cell walls can be stained to visualize the cross sections of the cells. Images are recorded with a camera and the resolution of the microscope depends both on the quality of the camera (number of pixels recorded) and the optics of the microscope. Unfortunately, standard light microscopy images tend to be very difficult to use in computational image processing due to the complexity of colors and shadings.
**Fluorescence Microscopy**

Fluorescence microscopy is a type of light microscopy in which only fluorophores (usually fluorescent proteins and dyes) that the biologist has integrated into the specimen are imaged. Fluorescence microscopy is highly advantageous for image processing because of its great specificity—the biologist controls which type of fluorophore is excited, such that only the objects of interest labeled with that fluorophore (such as nuclei) can be imaged. Another major advantage is that living, developing specimens can be imaged. However, there can be challenges in fluorescence microscopy such as bleaching of the fluorophore so that it cannot be visualized, auto-fluorescence of the tissue obscuring the fluorophore added by the biologist, or multiple fluorophores may not be able to separated, and their signal may bleed through from one to another.

A fluorophore absorbs light at a shorter wavelength (higher energy) and emits light at a longer wavelength (lower energy). For example, green fluorescent protein (GFP) absorbs blue light and emits green light. There are a range of blue wavelengths that can be absorbed to excite GFP, which is usually depicted as an excitation curve. Typically, a 488-nm laser is used to excite GFP because the peak of the GFP excitation curve is near 488 nm. The range of green color wavelengths emitted by the excited GFP is shown as an emission curve. Each fluorophore has its own characteristic excitation and emission curves. As long as these do not overlap too much, multiple distinct fluorophores can be imaged within a sample.

In contrast to standard light microscopes where the light passes through the sample, fluorescence microscopes illuminate the sample from the top using a dichroic mirror that reflects light at one wavelength while transmitting other wavelengths of light. In a fluorescence microscope, the excitation light (e.g., blue) enters the top back of the microscope, hits a dichroic mirror and is reflected down through the objective to the specimen. The fluorophores in the specimen are excited and emit another wavelength of light (e.g., green). The emission light travels back up through the objective, passes through the dichroic mirror and is recorded by a camera or observed through the eye pieces. This is generally called widefield fluorescence microscopy because all of the sample is simultaneously excited and all of the fluorescence is simultaneously detected.

**Confocal Microscopy**

Confocal microscopy (short for confocal laser scanning microscopy) is a type of fluorescence microscopy that can generate an optical section of the sample. Similar to a thin section cut with a microtome, an optical section is an image of a single plane within the specimen. For example, an optical section can reveal the center of a pollen grain, whereas only the outside is visible with widefield fluorescence microscopy. The confocal microscope uses pinholes to create optical sections. In the confocal microscope, a laser is used as the excitation light source (e.g., 488 nm blue for GFP), which passes through a pinhole, and is reflected by a dichroic mirror down through the objective, to excite fluorophores at a single point in the specimen. The light emitted by the fluorophores (e.g., green from GFP) passes up through the objective lens, through the dichroic mirror and hits a pinhole. Only the emission light from the plane of focus can pass through the pinhole to the detector. All of the out of focus emission from fluorophores above or below the optical section plane, is blocked by the pinhole. The size of the pinhole is adjustable. Larger pinholes let in more light, but also make the visible section thicker. Note that the confocal is limited in its ability to image deep into tissues by the ability of the laser to penetrate into the sample and emission light to transmit through the sample back to the objective. This limitation is more severe in optically dense tissues, which plants tend to be, unless cleared. Multi-photon microscopes are able to image deeper into the sample.

In confocal microscopy, one point in the sample is detected at a time and the complete image is created through assembling all of the points generated by the laser scanning across the sample. When the laser excites the fluorophores at one point in the sample, the detector, which is often a photomultiplier tube, records the quantity of emission light detected from that point in the sample as a pixel in the image. Then the laser moves over to excite the next point in the image. The entire optical section image is built by scanning each pixel sequentially. Scanning can be relatively slow, which can be problematic for fast moving samples. Spinning disk microscopes image faster than traditional confocals because they use multiple pinholes on a spinning disk to image multiple points simultaneously. The laser light passes through the pinholes to illuminate points on the specimen. The emission light from the excited fluorophores passes back through the pinholes achieving the confocal effect of filtering out of focus light. The image is captured with a camera (image resolution is limited in part by the camera capabilities). One complete sweep of the pinholes across the sample as the disk spins generates one frame.

**Color Channels**

We often want to detect more than one object (e.g., nuclei and plasma membranes) in a fluorescent image simultaneously. We may want to co-localize a fluorescent protein with a subcellular structure (e.g., transmembrane protein with the plasma membrane). In this case, different fluorophores can be used to label these different objects as long as they are spectrally distinct. For example, blue and yellow or green and red fluorophores tend to be good pairs. In contrast, green and yellow can be hard to separate without more advanced techniques, such as spectral imaging with linear unmixing. A confocal image contains one channel for each fluorophore. A channel is essentially a grayscale image of the fluorescence recorded by one detector. The excitation laser wavelength and the emission wavelengths collected are optimized for one fluorophore and to not overlap with other fluorophores. A couple of channels can be collected simultaneously if the detectors, dichroic mirrors, and filters to do so are available on the microscope. If not, each channel can be collected sequentially. In viewing a composite image, generally each channel is assigned a different color. Although most microscopes default to green for channel 1 and red for channel 2, any colors can be chosen. For colorblind people, green and magenta or blue and yellow are better choices. For image processing, typically one channel is processed at a time, and composite images need to be first split into the color channels.
Z-Stack

Confocal microscopes can generate a 3D image of the sample by taking a series of 2D optical sections called a z-stack. In the microscope software, the user specifies the first slice and the last slice and the interval (usually in μm) between slices. The total number of slices and the spacing between slices depend on the size of the sample. Generally, the first slice is taken immediately below the bottom of the sample and the last slice above the top. The smaller the interval between slices, the greater the resolution of the image, but also the longer it takes for the microscope to complete the image and the larger the file size of the image. Longer imaging times are more detrimental to living specimens. Generally, the microscope software will provide an optimal z-slice interval based on the optical properties of the microscope configuration, but this may not be the optimal for a given biological sample. If objects are to be detected in 3D in the image, having a small z-slice is critical and having a 1:1:1 ratio of x:y:z dimensions of the voxel may be advised. The optimum compromise is usually determined experimentally with consideration of the image processing pipeline.

Pixel and Voxel

Pixels are the 2D square elements of which an image is composed. Analogously, voxels are the 3D elements of which 3D images are composed. Each pixel or voxel has a location within the image and an intensity value for each color, e.g., red, green, and blue. For an 8-bit image, these intensity values range from 0 for black to 255 for white (or full color). A single pixel represents a defined size in the field of view based on the optical settings of the microscope. Analogously, the voxel represents a defined volume. The voxel is rarely a cube because often the interval of the z-slice is larger than the x and y dimensions. Most commercial confocal microscopes have their own proprietary image format which includes metadata about the imaging set up, including the scaling relationship between pixel and actual size. These formats are usually based on the TIFF image format. The scaling information can either be imported from the metadata or entered manually into image processing programs.

Resolution and Dots Per Inch (DPI)

The resolution of an image is the ability to resolve or distinguish features in the image, which generally depends on the size of the feature and the size of a pixel in the image. If two objects are closer together than one pixel, they cannot be resolved. Fundamentally, no matter how small the pixels, the resolution is also limited by the diffraction limit of the microscope, which depends on the wave-length of light. Generally, two objects closer than the wavelength cannot be resolved with a light microscope. However, recently super-resolution techniques have broken this barrier.

Print resolution is the number of dots (i.e., pixels) per unit distance. Dots per inch (DPI) is a commonly used measure of print resolution. Journals generally require at least 300 dpi for color images. On the microscope the user selects the number of pixels in the image. Common choices are $512 \times 512$, $1024 \times 1024$, and $2048 \times 2048$. Note that a $1024 \times 1024$ image at 72 dpi will be 14.2 × 14.2 inches. The same image at 300 DPI will be 3.4 × 3.4 inches. It is always better to take an image with more pixels because it is acceptable to down-sample an image so that you have fewer total pixels. It is not acceptable to resample an image to increase the number of pixels because this in essence generates data. Therefore, it is important when imaging to record enough pixels that the final image at 300 DPI will be large enough for the figure. However, the more pixels in the image, the larger the file size generated and the longer the acquisition time required for imaging (a strong limitation), so the user has to find a good compromise. Typically, 1024 X 1024 is sufficient.

Maximum Intensity Projection

A maximum intensity projection is a way to produce a single 2D image from a 3D Z-stack that appears to display the 3D image. For each pixel in the final image, the algorithm examines all the slices in the stack at that position and selects the brightest one for the final image. The brightest point tends to be the one on the outer surface of the object. Thus, the combined brightest points often show the surface of the object. Note that internal structures tend to be obscured through this method. Confocal images in publications are often maximum intensity projections; however, there may be times when a single optical section, or alternate visualization strategy is more informative. As an alternative to selecting the maximum intensity, other operations can be used to select the pixel from the stack for each position in the projection such as summing intensities of the pixels in the slices or selecting the average intensity or minimum intensity pixel. These functions are also available as options in Fiji under Image>Stacks>Z project.

Compression

Digital images are large files because the information for each pixel must be stored. Compression is used to reduce file size by filtering redundant information. For example, if a whole region is pure black, it can be stored in a more compact format. In lossless compression filters such as LZW in TIFF (Tagged Image File Format) images or Deflate used in PNG (Portable Network Graphics) images, no information is lost and the exact image can be recovered. In contrast, in lossy compression filters such as most JPEG (Joint Photographic Experts Group) settings, information is removed so that the original image cannot be recovered. The algorithms attempt to remove information that is not critical for the image. It is particularly important to avoid multiple rounds of lossy compression. Lossy compression can be problematic for image processing because it may erode or shift key features. Generally, the original data should be saved in lossless file formats such as TIFF or PNG, and lossy JPEG formats should be used only in final publications or presentations to reduce file size.

Step 2. Pre-processing

The accuracy of detecting objects in the image can often be improved by first pre-processing the image. Pre-processing often includes a denoising step. Noise is random fluctuations in the
intensity of the pixels of an image that obscure the true signal generated by the sample. Noise is always present in images due to low light conditions, collecting limited numbers of photons, and the electric circuitry of the microscope. Pre-processing can also take into account other properties of the sample to be detected, such as minimum intensity of the objects. The danger is that pre-processing could also remove real signal that is weak, thus altering the biological interpretation of an image.

Denoising Filters

One of the typical pre-processing steps is to apply a filter to reduce noise in the image. A number of different filters are available, which are optimal for different kinds of noise and different objects within the images. Filters compare a pixel intensity value to all of the surrounding pixel values within a user defined area and change the pixel value based on its neighbors. The “mean filter” replaces the pixel value with the average of all the surrounding pixel values. Similarly, the “median filter” changes the pixel value to equal that of the median value of the neighbors. Both of these filters are available in COSTANZA. Notice that both filters produce nuclei that are more uniform in brightness against a background that is more uniformly dark. However, mean and median filters tend to disrupt linear objects such as membranes or cell walls in the image. The Gaussian blur filter better preserves boundaries and edges while denoising the image. The Gaussian blur filter replaces a pixel with the weighted average of the pixels in the surrounding neighborhood. The weight is assigned according to a Gaussian or normal distribution based to the distance from the pixel. Thus, the pixel itself has the highest weight and the weights of neighboring pixels decrease with their distance from the pixel. The Gaussian blur filter is included in the MorphoGraphX and MARS-ALT image processing packages.

Step 3. Segmentation

The objects (e.g., nuclei, cell membranes) in the image are detected and delineated during the segmentation step. Segmentation is the process of partitioning an image into regions of interest. These regions may be cells, nuclei, tissues types, etc. During segmentation, the pixels belonging to each object are defined. There are many segmentation algorithms and this is an active area of research. Trial of several segmentation methods may help identify the one that is best for a particular image data set.

Thresholding Segmentation

Possibly the simplest segmentation, thresholding divides the image into two parts, the objects with intensity above the user defined threshold and the background below the threshold. Setting the threshold is critical. If all the objects have similar intensity, thresholding can work well. Thresholding does not work well when objects have very different intensities. Fiji provides thresholding in the Image menu, under Adjust and then Threshold. COSTANZA uses thresholding as a pre-processing step called background extraction.

Gradient Ascent/Descent Segmentation

Gradient ascent/descent is a method for finding local maxima (ascent) and minima (descent). When used for segmentation, gradient ascent finds local maxima in the image intensity. For example, if the nuclei are bright in a black background, then this method would identify the brightest center of each nucleus. This method is particularly useful when not all nuclei have the same intensity, because the center will still be a local maximum. An image can be thought of as a landscape with peaks and valleys in intensity. The algorithm starts from each pixel in the image and moves to the neighboring pixel with the highest intensity and repeats the movement, until no neighboring pixel has a higher intensity. Thus, it climbs up the intensity hill until it reaches the local peak. All of the starting points from which the algorithm reaches the same peak are considered a Basin of Attraction (BOA). In the example of the nucleus, if the background has already been removed, then each BOA defines a separate nucleus. COSTANZA uses gradient ascent as its segmentation method.

Watershed Segmentation

Watershed is another commonly used segmentation method. Again, imagine the high intensity regions (e.g., cell walls) as peaks and the low intensity regions (inside cells) as valleys. Then image pouring water into the image and it slowly fills up the valleys until it runs over. The point just before the water runs over is the segmentation. The user initiates the process by manually placing a seed in each cell, or this process can be automated. The seed expands similar to the filling water. Watershed is used in the MorphoGraphX image analysis software for segmenting cells.

Edge Detection Segmentation

Edge detection finds edges of objects by looking for steep changes in intensity of the image. For example, at the edge of a nucleus, the intensity changes from nearly black to nearly white, a sharp change. Edge detection is built into Fiji with the Find edges command in the Process menu. Also, there are plugins for edge detection such as Canny edged detection (https://imagej.nih.gov/ij/plugins/canny/index.html).

Additional Segmentation Methods

This is far from a complete list of segmentation approaches, and additional methods are being actively developed. A number of other segmentation methods are available in Fiji under Plugins>Segmentation. Online tutorial videos are quite helpful in learning to use these other methods. One of these is Trainable Weka Segmentation, which is based on machine learning. The user hand segments some regions of the image to train the computer, which completes the segmentation. If the segmentation is not very accurate, the user can go back and add additional training segmentation regions. In the coming years, I expect to see increasing use of trainable segmentation method based on machine learning.
Step 4. Post-processing

Post-processing takes advantage of information known about the objects to refine the segmentation.

Semi-Automated Approach

Automated segmentation programs commonly make errors that are obvious to the scientist because the human eye and brain are still far better at detecting features and correctly filling gaps than computer algorithms. After optimizing the pipeline to reduce errors as much as possible, it is common for scientists to hand correct these errors in segmentation, which is generally a highly effective approach. Commonly in COSTANZA, the errors are in splitting one object into two. Therefore, during hand correction, the scientist edits the results table to fuse those two objects together again. However, hand correction or hand segmentation can be extremely time consuming, leading some scientists to look toward help from citizen scientists through crowd sourcing. A critique of this approach is that human intervention could create bias in the results.

Registration

In image analysis, registration is the process of aligning two images. Registration is commonly used for images in a time series because often the sample moves slightly between images. Registration precedes tracking and assists in the identification of corresponding features.

Tracking

Tracking is the identification of corresponding features in images from a time series. Commonly, cells or nuclei are tracked in time to determine growth rates, division rates, and changes in fluorescent protein expression. MorphoGraphX has a parent labels tool for hand tracking cells. ALT is an automated lineage tracking algorithm in the MARS-ALT software. Tracking is more difficult when cells have undergone major movements or divisions between time points.

Validation

It is important for the scientist to carefully compare the original image with the processed image to ensure the processing accurately reflects the original image. Many errors in processing pipelines can be identified this way. For example, too many rounds of smoothing can lead to shrinkage of the object being analyzed. The gold standard in validation is to compare automated segmentation with the hand segmentation by a scientist for the same image; however complete hand segmentations are time consuming and rarely available. Often validation is done in the semi-automated approach together with the hand correction of errors.

Summary

Computational image processing provides a powerful tool to extract quantitative information from a large image data set. Images should be optimized for future processing during image acquisition. Pre-processing can remove noise and take other properties of the image into account in preparation for automated segmentation which identifies and delineates the objects in the image. Different segmentation methods are good for different types of images. Post-processing can reduce errors in segmentation using known properties of the objects identified such as size and distance between objects. It is important to validate the results and small errors can be hand corrected.

PART 2. IMAGE MANIPULATION

Image manipulation is transforming, altering, or editing an image. Image processing involves many image manipulations designed to help accurately detect and quantify features in the image; however, these same manipulations are not acceptable when presenting the primary image for publication. For presenting the original image data, linear changes (e.g., brightness and contrast) applied to the whole image that do not obscure any aspect of the image are generally acceptable. Cropping is generally permitted as long as it is not designed to remove confounding evidence. Most other kinds of manipulations are not acceptable as they obscure, distort, or change the data. Such changes must be disclosed in the figure legends and/or methods. Do not erase or clean background because what you think is not important may turn out to be. Do not splice parts of different images together without leaving a clear line indicating the splicing. Keep in mind that your goal is to clearly present the actual data, not what you think the data should be. Most importantly, be transparent about what you have done to the image with detailed methods and figure legends.

How Do We Reconcile Image Processing with Image Manipulation?

We have just discussed image processing which has the goal of extracting information from images by applying computational algorithms. Most of these algorithms make changes that violate the image manipulation rules. For example, denoising applies non-linear changes to the image that remove background. Yet image processing is still highly scientifically valuable because it can provide quantitative information such as the size of nuclei or cells in the image, the number of cells in the image, etc. To extract this information as reliably as possible, the image processing pipeline purposely creates major changes in the images including multiplying background through denoising filters, hand editing incorrect segmentation results, and sometimes warping images to register them. How can we reconcile these major changes with the image manipulation guidelines? The most important step is to be very clear in both the methods and the figure legends exactly how the images have been processed. Detail which programs you have used, in what order, and with which parameters. Process all images in the data set the same way. It can be useful to include a figure or illustration describing the image analysis pipeline. It is often appropriate to publish both the original image and the processed image in the figure, so that the reader can evaluate the accuracy of the processing. It is critical to always keep the original image and journals or referees may request to see the original images. Considering making the original images available in a public data repository so that they can be used in further analysis.
by others. Increasingly, journals are requiring publication of original images or access to the original images during review.

PART 3. COMPUTATIONAL IMAGE PROCESSING
WORKSHOP

The best way to learn about image processing is with hands-on experience. Here are some exercises to introduce you to Fiji as an image analysis platform and COSTANZA as a specific processing plugin. This activity guides you through exploring a confocal stack image in Fiji, hand measuring a few nuclei, and using COSTANZA to detect and measure all of the nuclei (Workshop Slide 39). Questions to think about are indicated throughout.

Scientific Question (Workshop Slide 38)
The transcription factor ATML1 has a well-established role in specifying the identity of epidermal cells during embryogenesis in Arabidopsis thaliana. ATML1 is expressed in every epidermal cell of the developing plant. ATML1 also has a second role in specifying the formation of giant cells within the epidermis. Giant cells are enlarged highly endoreduplicated cells in the leaf and sepal epidermis in Arabidopsis. Giant cells affect the curvature of the sepal which is important for both the sepal to protect the bud and for the flower to open when it blooms. This raises the question of how ATML1 specifies some cells to become giant cells when it is expressed in every epidermal cell. We hypothesized that the amount or concentration of ATML1 might differ between cells and influence whether they became giant or small cells. The goal of this workshop is to use COSTANZA to quantitatively determine whether ATML1 expression varies between cells in the sepal epidermis. In the data supplement, we have provided an image of pATML1:mCitrine-ATML1, in which the ATML1 protein is translationally fused to the yellow fluorescent protein mCitrine (Channel 2 in green). In the image, the cells are outlined in with a red fluorescent plasma membrane marker (Channel 1 in red, pML1:mCherry-RCI2A).

Step 1. Setting Up Fiji and COSTANZA (Workshop Slides 40 to 44)
ImageJ is a free open source software package that was initially developed at the NIH for visualizing, manipulating, and analyzing scientific images (Workshop slide 40). ImageJ is written in Java so that it can run on any computer. ImageJ itself contains many basic features for analyzing scientific images such as splitting color channels, measuring objects, and working with 3D image stacks. The image processing capabilities of ImageJ can be extended by adding plugins that researchers have developed (Workshop slide 42). A plugin is a software component that adds function to an existing computer program. ImageJ provides the graphical user interface (GUI) for these plugins making them easy for the biological community to use. In general, to install a plugin, drag it to the plugin folder and restart the program (detailed instructions below for installing the COSTANZA plugin). Fiji is a distribution of ImageJ in which many of the common plugins have already been incorporated (https://fiji.sc). Many of the plugins available are listed at http://rsbweb.nih.gov/ij/plugins/index.html. To process many images in Fiji, it is easy to write a macro using the record function (Plugins->Macros->Record).

We will use the COSTANZA (COnfocal STack ANalyZer) plugin to identify nuclei in a 3D image (stack) and to extract quantitative fluorescent intensities of mCitrine-ATML1 (Workshop Slide 43). The COSTANZA plugin includes the entire image analysis pipeline (http://home.thep.lu.se/~henrik/Costanza/). For advanced users, COSTANZA is also available on Gitlab (https://gitlab.com/sclu/teamJiCOSTANZA). The Gitlab repository contains the code and a python version.

1. Install Fiji (Workshop Slide 41)
2. Download and install the appropriate version of Fiji for your computer http://imagej.net/Fiji/Downloads - Fiji
3. Install COSTANZA plugin (Workshop Slides 44)
4. Download the COSTANZA plugin http://home.thep.lu.se/~henrik/Costanza/
5. Install COSTANZA. Unzip and drag the folder into the plugins file in Fiji. If the plugins file is not immediately obvious within Fiji, right click (control click on mac) on Fiji and select Show Package Contents. When you relaunch Fiji you should now see COSTANZA listed under the Plugins menu. (Hint: Make sure you drag a single COSTANZA folder containing two .jar files into the plugins file in Fiji, not an individual file and not an extra folder.)
6. Download the research image mCitrine-ATML1_mCherry-RCI2A.tif Use this image as an example for Steps 2 to 4.

Step 2. Exploring a Confocal Stack (Workshop Slides 45 to 46)

1. Open mCitrine-ATML1_mCherry-RCI2A.tif in Fiji.
2. The image has two channels as shown by the channel slider (c slider). Channel 1 (initially red) shows mCherry-RCI2A; RCI2A is a plasma membrane localized protein and it is fused with mCherry, a red fluorescent protein. Channel 2 (initially green) shows mCitrine-ATML1; ATML1 is a transcription factor that is nuclear localized and is fused to the yellow fluorescent protein mCitrine.
3. The z slider shows each slice of the confocal stack. Z=1 is at the bottom of the image.
4. Try out both sliders.

Question: How many z slices are in the image? (Hint look at the information at the top.)
Question: Pick a large nucleus (c slider =2). For how many consecutive z-slices is the nucleus visible (i.e., from the bottom to the top)?
5. Open the channels tool under Image-Color-Channels tool. View both channels simultaneously by changing the toggle from Color to Composite. You can change the color of each channel under More. Move the channel slider to select the channel to change. (See above for more information about color choices and the color blind).
Question: When you are reading a research paper, how do you know what the colors in a confocal image mean?
6. Alternatively, pseudo-color the image channel using a lookup table (LUT) that may better emphasize a feature in the image. A lookup table is used to assign a grayscale pixel intensity to a particular color. For example, the Fire LUT in Fiji can be used to make differences in intensity more readily apparent to the viewer by assigning bright pixels to white through yellow and darker pixels to blue through purple.
7. Apply different LUTs in Image->Lookup tables. If you apply a LUT the associated calibration bar should also be shown so that the viewer can interpret the colors and the LUT should be mentioned in the figure legend. The calibration bar can be added with Analyzer->Tools->Calibration Bar.

Question: Using the Fire LUT and the calibration bar, what is the approximate range of intensities of the pixels in the nuclei (c =2)?
8. To visualize a 2D image of the stack, make a maximum intensity projection using Image>Stacks>Z Project. Use all of the slices (which should come up as the default). Select Max Intensity under projection type. For each pixel in the final image, the algorithm examines all the slices in the stack at that position and selects the brightest intensity for the final image. Other operations such as minimum, average, or sum can be used to determine the final pixel intensity from the slices. Compare the maximum intensity with other projection types. Maximum intensity is the most commonly used. (If you turn on composite in the Channel window you will see both channels in the projection; Workshop Slide 45)

Question: What can you see in the projection that was not clear from the stack? What information is included in the stack is lost in the projection?

9. To visualize the cross sections of the stack, create orthogonal views using Image>Stacks>Orthogonal Views while the original image is selected. Drag the yellow lines to see cross sections of different parts of the image. (Workshop Slide 46)

Question: Is the mCitrine-ATM1 signal in the epidermis? Mesophyll?

10. Close the orthogonal windows, to return to the normal view.

Step 3. Measuring Objects in Images by Hand (Workshop Slides 47 to 48)

ImageJ has built in tools for measuring objects in images, which is often needed when quantifying scientific images.

1. In Analyze>Set Scale, you can set the correspondence between pixels in the image and distance. You can usually get this information from the microscope software. (Workshop Slide 47)

2. Set: Distance in pixels: 0.9635

3. In Analyze>Set Measurements, you can choose the properties you want measured (Workshop Slide 47). For example, if you choose Area, Centroid, and Add to overlay when you mark a region with the cursor and press measure it will give you a center point window with the area of the region, the location of the center point of that region, and the intensity of fluorescence within the region. Also a region will be marked on the overlay (Control-t on PC / Command-t on mac will bring up the ROI manager).

4. Next, measure the area and signal density of several nuclei. First, make the sum slices projection (Image>Stacks>Z project>sum slices) and toggle to channel 2 so the nuclei are visible. Next, use the magnifying glass to increase the magnification of the image several times. Pull the bottom right corner of the image window until it fits the larger size. Trace the edge of the nucleus with the freehand selection tool (it looks like a heart). Press Analyze>measure (or control-m on PC / command-m on mac). The results of the measurement should appear in the results window. The results window will show Area, X, Y which are the coordinates of the centroid, Integrated Density (mean pixel intensity times area) and Raw Integrated density (sum of pixel intensities in the area). Both Integrated Density and Raw Integrated Density quantify the fluorescence in the region outlined.

5. Measure the cross-sectional area of 5 nuclei. After clicking on the results window, click on Results>Summarize to show the mean, sd, min and max for each value. (Workshop Slide 48)

Question: What is the average nuclear area? What is the maximum that you measured? Are all of the nuclei about the same size? Question: The integrated density gives the total fluorescence in the nucleus. Do all the nuclei have the same integrated density?

6. To clear the results, press Analyze>Clear Results. Close the image without saving it.

Step 4. Segmentation: Detecting Objects in Images in COSTANZA (Workshop Slides 49 to 54)

The goal of segmentation is to detect objects in the image and mark their locations. We will use the COSTANZA plugin to ImageJ. COSTANZA is designed to detect fluorescent nuclei in confocal stacks and was created by Henrik Jönsson and his lab for detecting nuclei in Arabidopsis meristem images. The segmentation algorithm in COSTANZA is gradient ascent. The algorithm starts at each point in the image and moves to the neighboring point with a higher intensity. When the program moves to the point with the highest intensity, it stops. This point is likely to be the center of the nucleus. The process is repeated from each point in the image until all nuclei have been identified. (Workshop Slide 49)

1. If you have installed COSTANZA correctly, you should be able to launch it from the Plugins menu Plugins>COSTANZA>COSTANZA Plugin. If this does not work, check to make sure that the COSTANZA Plugin has been installed in Fiji (Workshop Slide 44).

2. Reopen the stack image to make sure you have the original image as the starting point. Re-enter the scale as above in Analyze>Set scale.

3. COSTANZA works on grayscale images, not colors. Bring up the channels tool under Image>Color>Channels tool. With channel 2 containing the nuclei selected, choose grayscale from the toggle menu. The image should lose its color.

4. Separate the channels into new stacks using Image>Color>Split Channels. Close the plasma membrane channel 1 leaving only the nuclear channel stack open.

5. Set the parameters: COSTANZA contains 4 submenus at the top.

COSTANZA General (Workshop Slide 50)

The COSTANZA parameters relating to gradient ascent segmentation are in the general menu.

Select the “use extended (box) neighborhood” option, which determines the size of the region around the pixel that is searched for a higher intensity pixel during gradient ascent. If this option is not checked, the algorithm searches 6 neighboring pixels. If this option is checked, the algorithm searches 26 neighboring pixels.

Select the “mark intensity plateau with a single maximum” option, which allows you to confirm that if you have multiple equal maximum pixels neighboring each other, then the program should mark this as a single object, not divide it into multiple objects. For example, when several pixels in the center of the nucleus are equally bright, marking it as a single nucleus would be advantageous.

Select the “mark cell centers” option, which allows you to mark the object maxima identified by the algorithm. Set the pixel radius to 3.

Select “Display basins of attraction (BOAs)”, which allows you to mark the entire object. The algorithm colors each object with different colors.

Select “Display basins of attractions according to measured intensity”, which measures the intensity of each BOA and displays...
a heat map of these intensities. This can be useful if you are looking at
the expression levels of a fluorescent protein in different nuclei or cells.

For this exercise, do not select “use secondary stack to
measure intensity levels”, which allows COSTANZA to perform
the segmentation on one stack and measure the intensity of each
BOA on another stack. This option could be useful in the case
where a ubiquitous nuclear marker is used to identify the outline
of the nucleus and a second color channel is used to quantify the
fluorescent protein in each nucleus. The ubiquitous nuclear
marker could also be used to normalize the signal.

Also, do not select “Display internal working stack”, which al-
lo ws the user to observe the effects of the pre-processing steps on
the image.

Recap: For this image, we recommend you set:

Pre-processor:
加工强度：15.0
Mean filter (denoising)
Radius: 1.1
Number of times: 6

COSTANZA Post-processor (Workshop Slide 52)
Select the post-processor tab. COSTANZA provides two post-
processing functions. First, the BOA remover allows the user to
set minimum size and intensity thresholds under which the object
(BOA) will be removed. This is useful if you know the object
should be at least a certain size or brightness. Second, the BOA merger
allows the user to merge objects that are less than the user
specified radius apart. COSTANZA commonly over-segments
objects dividing them into two or more pieces because the object
has more than one maximum intensity peak within it. If you know
that the objects should be spaced apart, you can use this
function to correct the over-segmentation by fusing BOAs
together.

For this image, we recommend you set:
Post-processor
Processor queue:
BOA remover (removes segmented objects under size or intensity)
Size threshold 10
Intensity threshold 10
BOA merger (merges together peaks that are nearby)
Radius 4

COSTANZA Scaling (Workshop Slide 53)
COSTANZA allows the user to either select “use ImageJ stack
calibration” which uses the metadata of the image or to manually
type the errors. The background extraction intensity

Pre-processing in COSTANZA (Workshop Slide 51)
Select the pre-processor tab. COSTANZA offers four possible
pre-processing steps. The preprocessing steps are executed in
the order they are listed in the processor queue. It may be nec-
necessary to remove a step and replace it after adding another step to
get the steps in the right order.

Invert Image is the first option available. COSTANZA segments
bright regions in a dark background. If your image is dark regions
on a white background you will need to invert it first. Likewise, if
you would like to segment the body of the cell which is dark,
covered by a bright plasma membrane, invert the image so that
the body of the cell is bright and the membranes dark. Invert image
should be run first in the queue.

COSTANZA offers background extraction. This is a thresh-
holding operation, which we discussed in the segmentation
section. The user specifies an intensity threshold value below
which everything is considered background. This parameter is
critical and should be carefully optimized for each image
data set.

Next there are two denoising filter options: mean filter and median
filter. Generally, one of these is selected. The radius defines the
circle of neighboring pixels considered. The number of times is how
many times this filter is repeated.

In the COSTANZA general menu, display internal working stack
allows you to examine the results of pre-processing.

For this image, we recommend you set:
Pre-processor:
Processor queue:
Background extraction (thresholding to remove background)
Intensity threshold: 15.0
Mean filter (denoising)
Radius: 1.1
Number of times: 6

Question: Is each nucleus correctly identified? Are there obvious
mistakes? Are nuclei missing, or divided in two?
Question: How many nuclei are detected? (Note – check Results
table)
Question: Looking at the results, are all nuclei the same size (BOA
volume)?

2. Try running COSTANZA on a max intensity projection of the nuclei
instead of a stack.

Question: How many nuclei are detected in the projection? Which is
more accurate in terms of nuclei detected?
Note that measuring the fluorescent intensity in the 3D image will be more
accurate because of the loss of information in the z-dimension of the
max-intensity projection.

3. Try changing the parameters to see what they do and see if you
get better or worse results. The background extraction intensity

Question: How many nuclei are detected in the projection? Which is
more accurate in terms of nuclei detected?
Note that measuring the fluorescent intensity in the 3D image will be more
accurate because of the loss of information in the z-dimension of the
max-intensity projection.

Question: How many nuclei are detected in the projection? Which is
more accurate in terms of nuclei detected?
Note that measuring the fluorescent intensity in the 3D image will be more
accurate because of the loss of information in the z-dimension of the
max-intensity projection.
threshold in the pre-processor and the BOA merger in the post-processor are critical parameters. Raising the intensity threshold will eliminate background signal or dim nuclei. Increasing the BOA merger radius will merge larger regions into the same nucleus.

**Question:** What change did you make in the parameters and is the nuclear detection more accurate, or are there more mistakes?

### Scientific Conclusion (Workshop Slides 55-56)

**Question:** Looking at the BOA-intensity results, how does mCitrine-ATML1 fluorescence compare between nuclei? Does ATML1 concentration differ between cells?

The results table can be imported into Excel or R for further quantitative analysis of the data. To examine the variability in intensity a histogram can be plotted and the coefficient of variation (CV = SD/mean) of the nuclear intensities can be calculated.

We find that mCitrine-ATML1 concentrations varies between cells. We further find that mCitrine-ATML1 fluctuates up and down in sepal epidermal cells. We find that if ATML1 reaches a high concentration during the G2 phase of the cell cycle, the cell is likely to become a giant cell. However, if the ATML1 concentration is lower during the G2 phase of the cell cycle, the cell is likely to divide and make small cells.

### Step 5. Testing Your Own Image (Workshop Slide 57)

Use your own image from your research or a scientific image you find on the web that looks interesting. Simple images that have only the structure you are trying to detect (i.e., nuclei or plasma membranes) in one color tend to work well.

**Hints:**

1. Remember to split color channels and put the color channel you want to analyze in grayscale before running COSTANZA on it (as described above). Running COSTANZA on a composite image is a common problem, which either generates errors or generates BOAs in structures present in the other color channel that you are not trying to detect.

2. As a reminder, COSTANZA detects high intensity regions (white) on dark (black) backgrounds (see Pre-processing in COSTANZA). If the object you want to detect is dark with a white background, use invert image as the first step of the pre-processor. Remove preprocessor steps then add invert image and then add the other pre-processor steps again.

3. Third, COSTANZA runs on the topmost image window. When you start building up results, make sure you bring the original image back to the top before pressing run (otherwise COSTANZA will try to run on one of the result images it just generated).

4. In the COSTANZA general menu, you can check Display internal working stack, which will provide the image after the pre-processor is applied. Median and mean filters can blur the image excessively causing problems. These filters can be removed altogether, or the parameters can be adjusted until the image is no longer excessively blurred.

5. If COSTANZA is generating errors, double check that the image is a single channel and grayscale first. Remove pre-processor and post-processor steps, until it runs without errors. Then add them back one at a time, adjusting the parameters. Also try saving the original image as a .jpeg.

6. If COSTANZA is not detecting anything, remove pre-processor and post-processor steps and/or lower intensity thresholds until you detect too many structures. Then slowly add back processor steps and/or increase thresholds until you achieve a good match with the image.

7. Making some manual measurements of features (such as background intensity, nuclear intensity, distance between the nuclei, etc.) can help in choosing good parameters for the COSTANZA segmentation.

**Question:** What feature do you want to detect in the image?

1. Test the image in COSTANZA starting with the parameters you used for ATML1 nuclei. (Note that if your image has a light background you will have to invert it first, which you can do in the pre-processor by add Invert Image. COSTANZA follows the preprocessor queue in order from top to bottom. So, to invert the image first, you will have to remove the existing preprocessor steps, add invert image, and add back the preprocessor steps.

2. Change the parameters of the pre-processing and post-processing to try to improve the segmentation.

**Question:** What parameters did you change? What range of values did you try? Which values worked best?

**Question:** Was COSTANZA a good method for segmenting your image? If not, what else might you try?

### Step 6. Two Other Images for You to Explore

The first is “Maximum intensity projection of pAR180 8 day 1st leaf ab 10X 1A.lsm”

This is an image of nuclei (channel 1, initially green) in the epidermis of the first rosette leaf developing between the petioles of the cotyledons in Arabidopsis. Chlorophyll autofluorescence is shown in channel 2 (initially red).

To detect the nuclei, before processing the image, use Image>Color>Split Channels to separate the two channels. Then use Image>Color>Channels Tool>More to change the image to greyscale.

Set Scale in Fiji (may be imported automatically from .lsm file):

- Distance in pixels: 1.3249
- Known distance 1.00
- Pixel aspect ratio 1.0
- Unit of length: µm

The second is “pAR169 cauline leaf abaxial 20X 1C slice.tif”

This is a single optical section taken with a confocal microscope of the abaxial (back) epidermis of an Arabidopsis thaliana cauline leaf. There is one channel, currently displayed in gray showing a fluorescent plasma membrane marker (pAR169 ML1::mCitrine-RC(2)A).

Before running COSTANZA, Set Scale in Fiji:

- Distance in pixels: 0.9635
- Known distance 1.00
- Pixel aspect ratio 1.0
- Unit of length: µm

This image allows you to test the ability of COSTANZA to detect cells instead of nuclei. Hint: to do so the first step in the
COSTANZA pre-processor queue should be invert image. You will have to remove all other steps to add invert image first and then add back other steps.

WORKSHOP CONCLUSIONS
Fiji has many more features than we have explored. In addition, plugins are widely available for many specialized tasks: http://rsbweb.nih.gov/ij/plugins/index.html. I hope you will find Fiji and related programs useful.

Adrienne H. K. Roeder
School of Integrative Plant Science, Section of Plant Biology
Weill Institute for Cell and Molecular Biology
ahr75@cornell.edu
ORCID ID: 0000-0001-6685-2984

ACKNOWLEDGMENTS
This work was supported by the National Science Foundation (NSF) IOS CAREER 1553030 to AHKR. Thanks to Mary Williams for insightful comments and editorial support.

RECOMMENDED READING AND REFERENCES
(This is a list of a few sources to help the reader gain entry into a large body of literature. We apologize to those whose work is not included.)

Computational Image Analysis

Microscopy

Image Manipulation

Biological Background on ATML1
Additional Examples of Plant Biology Papers Employing Image Analysis

(Additional references for eager students illustrating the application of image processing to exciting questions in plant biology. Note this small sampling is far from exhaustive.)


