

Chapter 19

WORKING WITH PIXEL IMAGES

Pixel images are a frequent product of scientific research, whether they are generated by normal photography or as the output of a more sophisticated imaging process such as radiography. Here we explore the tools and concepts needed to edit pixel images and to integrate them with vector art in figures, as well as how to evaluate the most appropriate trade-offs when selecting file types, resolution, and compression.

Image compression

General principles

In addition to the resolution and dimension issues discussed in Chapter 17, when handling pixel images one must also consider compression strategies. Much of the information within pixel images is redundant in a technical sense: large blocks of pixels may all be the exact same color, there may be repeating elements that only need be stored once, or only a small subset of possible colors may be used. Through compression, programs and file formats reduce the amount of memory needed to store an image.

There are three image-storage strategies: no compression, **lossless compression**, and **lossy compression**. Each is associated with different file formats. At times, especially for small images, there are few or no performance costs to using uncompressed files. Lossless compression reduces the memory needed to store a file without actually losing any information, hence its name. Like an accordion, the image is compressed into a smaller space and returns to its same condition when expanded. A program achieves this by identifying and consolidating redundant information in the file, be it colors or patterns. This provides reduced file size at no cost to the appearance of the image, but it might not reduce the file size very much, particularly if the image is complex.

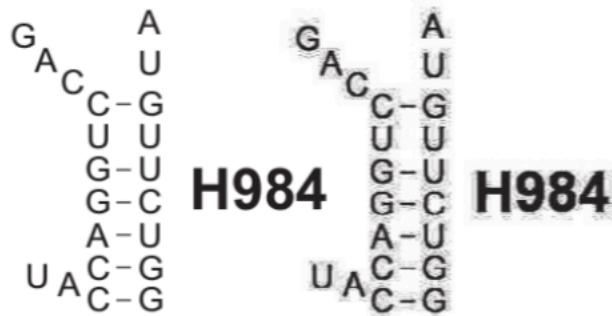


FIGURE 19.1 Comparison of PDF (left) and JPEG (right) versions of a figure. Notice the cloud of gray "dust" that surrounds the objects in the right figure, typical of JPEG compression. (The original artifacts in the compressed image above have been darkened for visibility.) This is more a function of compression methods than of insufficient image resolution.

If small file size is a more pressing concern than perfect image reproduction, you can use lossy compression. Lossy compression permanently throws away some information when the file is compressed. Similar colors may all be considered as a single color, decreasing the color depth of the image, or regions of the image may be approximated, changing the texture and lowering overall picture detail. In addition to reducing the amount of information in an image, lossy compression can introduce artifacts when the image is re-expanded (Figure 19.1). In many cases the differences introduced by lossy compression are imperceptible, but sometimes they can seriously impact the readability of an image. The JPEG format offers different compression levels, where you can choose to have smaller low-quality files or less compressed higher-fidelity images.

Implications for image workflows

Lossy image formats become problematic if an image is repeatedly opened, saved, and closed. With each iteration of this cycle, more and more image detail can be thrown away, and artifacts can accumulate. For this reason, lossy compression should not be used for images that are being repeatedly changed.

For practical reasons, most cameras and instruments that generate pixel images save them using lossy compression. This is often appropriate since far more images are generated than will ever be used, and the deficiencies of lossy formats are not particularly acute if only one round of compression is applied. Once a subset of image files is selected for further processing, they should first be resaved in a lossless image format. This will ensure that the images can be opened, edited, and resaved as many times as needed without degrading their quality.

If specimens are of particularly high value or every bit of resolution is important for the task at hand, then you should consider saving images in a lossless format immediately upon acquisition. The default file format used by most cameras and instruments can be easily changed to a lossless format.

Pixel image file formats

There is a wide and often confusing variety of pixel image file formats. They differ in several key technical respects. Some don't provide any compression; others support lossy or lossless compression. Some support transparency, while others do not. Some can store only a single image; others can support a series of images and video. Some allow more than one layer within one file but others allow only a single layer. Pixel image formats also differ in licensing. Some are open formats

TABLE 19.1 Selected image formats and their feature sets

Format	Supported features			
	Compression	Layers	CMYK	Transparency
TIFF	None, lossy, or lossless	Inconsistent	Yes	Yes
JPEG	Lossy	No	Inconsistent	No
PNG-24	Lossless	No	No	Yes
PNG-8	Lossy (via color)	No	No	Yes
GIF	Lossless (color options)	No	No	Limited
PSD	Lossless	Yes	Yes	Yes
XCF	Lossless	Yes	No	Yes
RAW	None, lossless	No	No	No

that are free for use by any program and any user, and were designed specifically to improve the interoperability of software and portability of data. Others are closed formats, owned by companies which in some cases charge royalties to developers who use these formats in their software.

Table 19.1 lists common pixel art file formats. The file formats you will encounter most often are PNG, JPEG, TIFF, PSD, and RAW. Most journals will request pixel art in TIFF files, which can be saved with no compression, different types of lossless compression, or JPEG compression within the TIFF file. For RGB and grayscale images without layers, PNG is an excellent lossless file format, and JPEG is a very widely supported lossy format.¹

CMYK is rarely used as the native color model for a pixel image, but it is sometimes necessary to convert to a CMYK space prior to publication. PNG does not support CMYK, and JPEG supports it inconsistently, so that CMYK files generated by one program may show up in another program as color negatives.

PSD is the native file format of Photoshop, and XCF is the native file format of GIMP. Both are largely specific to these programs, and you will usually export images to another file format when publishing them, sharing them with others, or transferring them between programs. These files can store many types of data, including text, some vector information, sophisticated layer properties, and other information that helps provide fine-scale editing control.

If you are using high-end digital cameras, you are also likely to encounter images in the RAW file format. This isn't a single file format, since each camera manufacturer has their own RAW format—for example, Nikon's RAW format is

¹PNG-8 is a lossy subset of PNG that uses 256 or fewer selected colors within the whole image. It is still widely used for web icons and non-photographic images.

NEF and Canon's is called CR2. RAW refers to pixel files that contain minimally processed raw data from imaging sensors. Some information is discarded when converting the sensor data to a JPEG or other standard image format, so saving RAW files provides greater control over downstream image adjustments such as white balance and exposure modifications. It is almost as though you can retake the photo after the fact, using different camera settings. RAW images typically allocate more information to storing each color value (the bit depth), allowing them to support a higher dynamic range. See Appendix 6 for more on colors and memory.

BMP or GIF file formats should be used only as a last resort when software has limited options for import or export. Both are relatively old and don't support many features or perform especially well across programs.

Transparency

Some programs and image file formats support the **alpha channel**, which encodes the transparency of each pixel. By default, all pixels are fully visible. It is often desirable, though, to make the portions of an image transparent when combining images, or just to simplify the composition of a figure. In vector art you can easily do this by copying the objects by themselves. In a pixel-based image, you have to edit the alpha channel mask that indicates which pixels should be visible. This is typically an additional 8-bit channel (256 levels, just like R, G, and B) stored along with the three color channels. Partial transparency is useful to convey time (for example, superimposing subsequent exposures) or spatial relationships between images (for example, a fluorescence overlay on a white-light image).

Pixel art editors

There are many pixel art editors, and, like vector art editors, each has its own set of trade-offs. Photoshop is the industry standard, especially when it comes to editing photographs. It is made by Adobe, the same company that produces Illustrator, and many of the shortcuts and navigation tools are the same in both programs. This is helpful when you frequently switch between them.

GIMP is the leading open-source pixel art editor, and is freely available for major operating systems at www.gimp.org. It is a bit more mature than its vector art analog, Inkscape, and can even do some vector art illustration. Like Inkscape, GIMP does not have quite as many features as its commercial counterparts, but is sufficient and easy to use for many common tasks.

Working with pixel images

Masks and nondestructive editing

Many programs that handle both vector and pixel art, including Illustrator, Keynote, and PowerPoint, support image masks for cropping and resizing images. **Image masks** are essentially little windows that can be modified to show only a particular region of a picture. The effect appears the same as cropping the image, except that

you can edit the mask later to show parts of the image that were previously excluded. In Photoshop, you can select a portion of an image, and subsequent adjustments can be applied only to the pixels within this masked region. Adjustment layers are also very useful in Photoshop. These apply an adjustment to all underlying layers or to a masked region, but with the advantage that the modification exists on its own layer and can be turned off or readjusted later in the process.

Some image editing and photo management programs, such as Aperture and Lightroom, implement more sophisticated forms of nondestructive image editing. In these programs, image manipulations—such as cropping, resizing, levels adjustments, color changes, and touchups—are stored as a series of commands associated with an image, rather than actual modifications to the image itself. These commands are reapplied each time the image is displayed or exported. This avoids the problems associated with resaving files in lossy formats, since the original file remains unchanged throughout the editing process. It also makes it easy to revert to any previous version of the image.

Levels adjustment

Many factors can cause images to come out with an exposure that is not quite optimal. In your image editing program, the best way to correct these is not by using the Brightness/Contrast menu item that most people gravitate to, but by using **Image ▶ Adjustments ▶ Levels...** instead. (You might also take the approach of using an adjustment layer, in this case choosing **Layer ▶ New Adjustment Layer ▶ Levels...**.) This dialog box presents a histogram of the brightness of the pixels in your image (Figure 19.2).

Along the left side of the curve are the black pixels, in the middle the graph shows the number of grey pixels, and along the right side are the white pixels. In Figure 19.2 you can see that the image is probably a bit underexposed: most of the values are distributed between black and gray, with very few pixels brighter than gray and none approaching white. The white point slider, circled here in red, sets the level of brightness above which pixels are considered fully white. To adjust this picture for a better balance and to bring up the overall brightness while remapping the pixels across a broader intensity range, you would slide the white point slider to the left until it is near the right edge of the

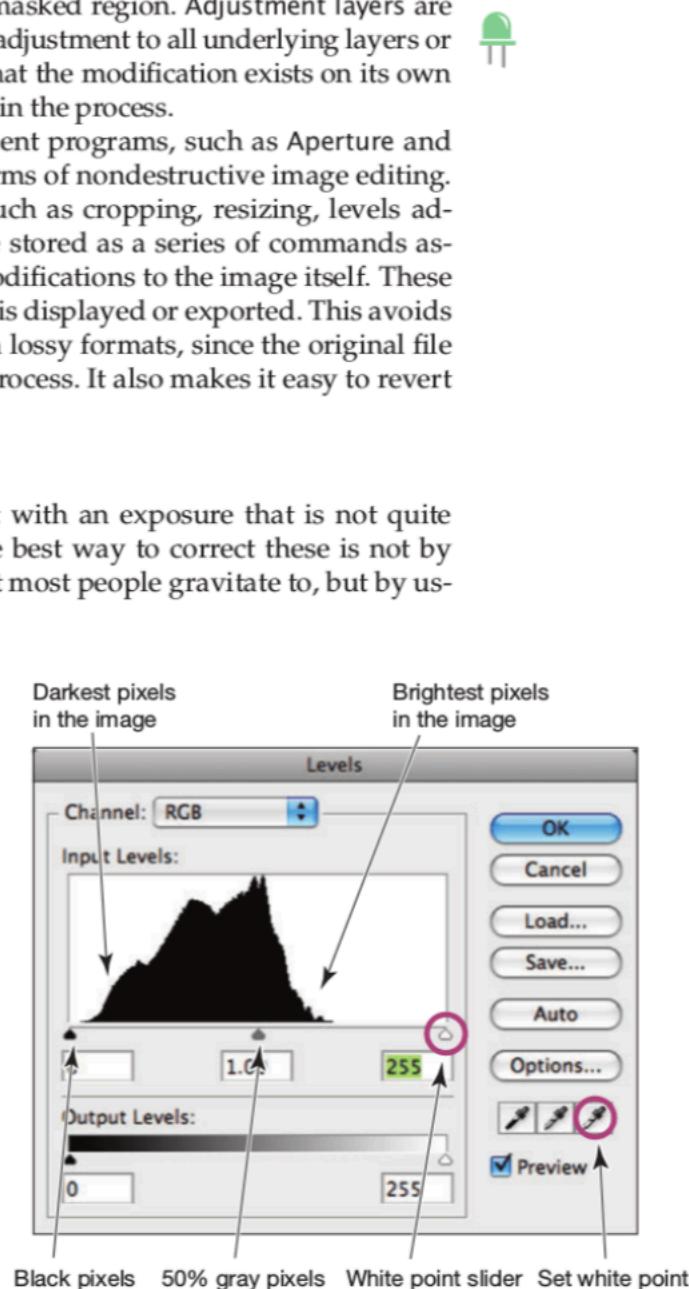


FIGURE 19.2 The Levels dialog box in Photoshop This box shows a histogram of how many pixels have each level of brightness. Other programs will have very similar views.

pixel histogram. This means that those pixels at the upper edge of the distribution would now be white instead of gray.

If there is a region of the image that you know is white, you can use the Set White Point eyedropper tool, also circled in Figure 19.2, and click inside the white part of the picture. This will set the white point to match that pixel. The white point tool also adjusts the RGB channels of the image independently, so it can change the white balance of your photo (sometimes for the better, and sometimes in inappropriate ways).

As long as these transformations are carried out as described above, and applied uniformly to the entire image, the transformation is a linear remapping of values. Therefore, this is very similar to changing the exposure on a camera and should be fully within the restrictions on image adjustment mandated by journals.

Grayscale images

In preparing scientific figures, it is sometimes clearer and almost always cheaper to produce figures in black and white—in this case meaning grayscale—rather than in color. Using grayscale images can also help your annotations to stand out against a monochromatic background. When converting to black and white in Photoshop, avoid using the **Image ▶ Mode ▶ Grayscale** menu item. Like an autoexpose function, this uses a weighted average of the red, green, and blue channels of your image, but it is not necessarily tailored to scientific images with non-standard color composition. Instead, try **Image ▶ Adjustments ▶ Black and White** and choose from the preset adjustments to see what produces the best results. This effect is very similar to the way that microscopists use (used to use?) color filters to improve the contrast when taking black and white images.

Antialiasing

A pixel image has a given resolution, and converting to a different resolution typically involves resampling the image data. Because vector art effectively has infinite resolution, converting it to pixel art is similar to downsampling a higher resolution pixel art image. Artifacts arise in this translation process when there are hard edges on a solid background, such as the edges of text characters or lines. The edge of the vector art won't exactly line up with boundaries of the pixels. Objects with perfectly horizontal and vertical edges may end up slightly smaller or larger, which is particularly noticeable for shapes that are thin to begin with. Curves and lines that aren't perfectly horizontal or vertical end up with a staircase effect along their edge, and may have variable thicknesses. If an object is thinner than a pixel it may disappear entirely. These graphical glitches are known as aliasing.

A common solution for these artifacts is **antialiasing**, the use of intermediate color values between objects with hard edges (for a simplified example of this, see the lower right panel of Figure 17.3). If the boundary of the vector lies exactly at the boundary of the pixels, then nothing need be done. If the boundary of the vector passes through a pixel, then that pixel would have an intermediate color that depends on how much of the pixel is occupied by the vector and other im-

age elements. Up close, where independent pixels can be discerned, antialiased images appear fuzzy. From a typical viewing distance, though, they appear much smoother and don't have the harsh artifacts that arise when converting vector art without antialiasing.

Most vector art applications that have the option of exporting pixel art images provide the option to perform antialiasing during the conversion. Recent versions of Illustrator give the option for a "pixel preview" and take into account the underlying pixel grid when translating text and vector art into pixels.

Layers

Layers are extremely important in pixel editing. Without layers, if you create a new object to annotate or cover the image, it will overwrite the original pixels, making it impossible to move or modify your annotation. Even effects like levels and contrast can be applied to an image through layer properties rather than changes to the image itself, making it easy to remove or adjust this effect later on.

GIMP and Photoshop both have a similar layer palette, with a blank-page icon to create a new layer and an eye icon to toggle the visibility of a layer. A drop-down menu specifies how the layers are combined; normally they are collapsed from top to bottom to create a single composite image, but you can set some layers to affect the way others are displayed or combine them in different ways. In Photoshop, when you first create or open a new document, the background layer will be locked with regard to transparency. Deleting portions of an image will revert them to the background color rather than leaving a transparent gap. To unlock, double-click on the layer named *Background* and select *unlock*.



When creating and moving selections in Photoshop, some potentially confusing behaviors may be observed. First, dragging the selection with the hollow arrow merely relocates the selection border, and not the pixels beneath it. To move the pixels themselves, you have to either hold the Command key (⌘) or switch to the black arrow tool. Second, an operation only applies to the layer highlighted in the Layers palette (and within that layer, only to selected pixels). So if you are editing without noticeable effect, make sure the correct layer is highlighted.

Feathering is a term used to describe gradually transitioning from selected pixels to non-selected pixels using a boundary of partially selected (semi-transparent) pixels. This can be an important way to modify your selection and smoothly compile images from several sources into a single image. All considerations about data integrity of course apply to use of these tools in merging images.

Colors in GIMP

GIMP has an innovative color picker, with several color space tools all in one window. When one slider is changed, the sliders for other color spaces are dynamically modified. This is nice because you can change the way you browse for colors without switching between color space windows, as is necessary in most other graphics programs. It is also very informative to see directly how changes in one color space map to changes in other color spaces.



One problem with all of this, though, is that GIMP doesn't support the CMYK color space natively. Color conversions which you can try in Photoshop will not be easy to achieve in GIMP without additional plug-ins.

Photoshop shortcuts

Like Illustrator, Photoshop has a rich set of well-thought-out keyboard shortcuts that can greatly accelerate your workflow. Many of these shortcuts are the same in Photoshop as in Illustrator, but there are additional ones as well:

- The Rectangular Marquee Tool (the dashed rectangle on the tool palette) selects rectangular regions of your image that you can then copy, cut, delete, or transform in other ways. Like the other selection tools, you can hold `shift` while you drag to select multiple regions simultaneously. If regions are overlapping they are joined together, which can allow you to select complex non-rectangular areas.
- Holding `option` while using a selection tool will *subtract* the newly selected region from the previous region, which can create holes in the middle of an existing region or remove chunks from its perimeter. You can also invert the selection, to then delete everything from a layer except the area of interest.
W Use `alt` in place of `option`.
- As with Illustrator, holding down `space` turns the cursor into a hand that moves the canvas around, and `⌘ space` will bring up the magnifying glass. When using the Rectangular Marquee Tool to drag a rectangular selection, holding `space` after you start dragging will allow you to drag so as to move the point of origin for the selection box. Letting up `space` will resume dragging out the edges of the selection box.
- Photoshop initially appears to have only one level of Undo. If you repeatedly press `⌘ Z`, it will just keep undoing and then redoing the last command. To go back further in time, choose Window ▶ History, and you will see a chronologically arranged palette displaying the commands you have issued.² Click on a command in the list to jump back directly to that point in your editing process.
- You can also use the keyboard shortcut `⌘ option Z` to step back through your history.

Command-line tools for image processing

It might seem counterintuitive that command-line tools, which entirely lack graphical user interfaces, are sometimes the most convenient way to work with image files. They are, however, particularly useful for automating repeated image manipulation tasks and for quickly extracting information about images. They also make it simple

²Multiple Undo commands first became available in Photoshop version 5. Before that time, image editing could be a stressful activity.

to script routine operations, such as performing a series of standard transformations to photographs of electrophoresis gels.

The sips program

This command-line image editor is included with OS X. It can retrieve a variety of image properties, rotate images, and perform cropping and resizing operations. In the scheme of image editors, it has limited functionality, but can work well for some common tasks.

The following commands create a new folder named `converted`, then resample each JPEG in the current folder to a width of 470 pixels, and place the new file into the `converted` directory:

```
lucy$ mkdir converted  
lucy$ sips *.jpg --resampleWidth 470 --out converted
```

Here, `*.jpg` refers to the input file list to use, and `converted` specifies the directory in which to save the images that are created. Use caution, because `sips` will overwrite images of the same name in the destination directory. To see a full list of `sips` options, type `man sips` at the command line.

ImageMagick: convert and mogrify

ImageMagick does not come with OS X, but it can be installed on OS X, Linux, and Windows. The installation procedure is described in detail in Chapter 21. You can also install it with the `port` command if MacPorts³ is installed on your computer:

```
sudo port install imagemagick
```

In Linux, you can use the command:

```
sudo apt-get install imagemagick
```



The installation is not a single program, but a collection of command-line and graphical utilities, and it has many more features than `sips`. It can adjust the levels of colors, brightness, or contrast, and perform other transformations like inverting an image. If you have a photographic data set which needs to be batch processed for analysis, this tool could save a great deal of time.

You will usually run ImageMagick by typing the `convert` command. For example, to convert a gel image from its native TIFF format to a half-size, inverted (black on white) and compressed PNG format, you could type:

```
convert gelscan.tif -resize 50% -negate gelscan.png
```

³You should install MacPorts from macports.org to make it easy to install and update command-line programs and all their dependencies. See Chapter 21 for more information.

For doing batch conversions of several files, you can generate a shell script containing several `convert` commands, as shown in Chapter 6, or you can use the `mogrify` function, which is basically `convert` applied to a group of files. The same operation above can be applied to convert all `.tif` files in a folder using this command:

```
mogrify -format png -resize 50% -negate *.tif
```



Be careful when using `mogrify` and always work on a copy of the original files, since it will overwrite without checking. A more complete description of the `convert` command and other ImageMagick functions can be found by typing `man convert` or `man mogrify`, or by referring to these sites:

www.imagemagick.org/script/convert.php
www.imagemagick.org/Usage/
www.fmwconcepts.com/imagemagick/

ExifTool

Many media files, such as photographs and audio tracks, have embedded metadata that describe how the file was created, as well as recording such other properties as time and location of creation, exposure settings, and camera type. This embedded metadata is often stored in a standard format called Exif. The Exif data can also hold tags (snippets of text that help categorize your images) and these can be accessed from database programs. TIFFs and JPEGs support Exif metadata.

These Exif data are accessible from within many image editors, but you will sometimes want to take a quick peek at many files without opening them all up. The free command-line program `exiftool` reads and writes Exif and other metadata formats. It is available for download at <http://www.sno.phy.queensu.ca/~phil/exiftool/>.

ExifTool is also a great way to recover the original creation information for photos, and could form part of an automated cataloging pipeline which sorts images without user intervention. To extract the location from several images, you could create an `exiftool` pipeline as described in Chapter 16. Some electron microscope programs will embed data about the magnification used when an image was captured. If you forget to add scale bars to your image, explore the Exif data to see if you can create a tool to print out a table of magnification values, as shown in Chapter 16 and in the corresponding example `scripts/exifparse.py`.

Image creation and analysis tools

ImageJ

ImageJ is an image analysis program developed by the United States National Institutes of Health, and is freely available for OS X, Windows, and Linux at rsbweb.nih.gov/ij. It can import a wide variety of image types, and provides

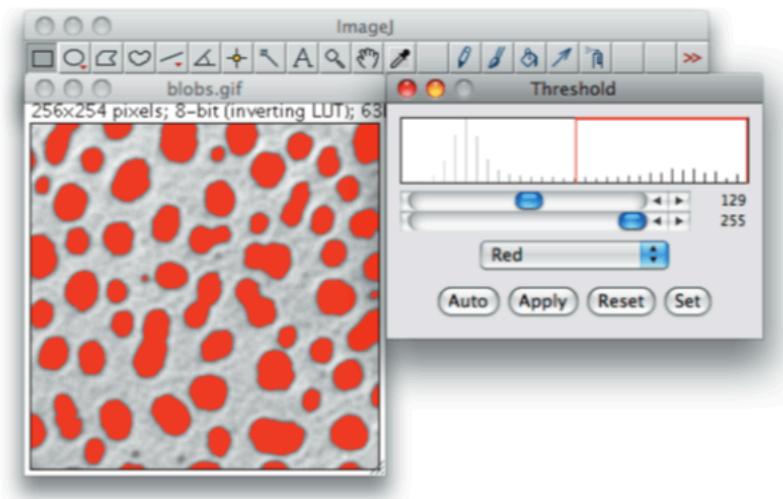


FIGURE 19.3 Setting the threshold for black and white conversion in ImageJ

a rich set of features for extracting quantitative data from them. This includes measuring the length, perimeter, and area of objects. It can also perform more specific tasks, such as gel electrophoresis analysis and 3-D reconstruction of optical slices. A wide variety of customizable plug-ins are also available, and tutorials are linked online. ImageJ is scriptable, so that macros can be recorded and then edited to replay repetitive analyses.

One very useful feature is the ability to automatically count particles of a certain size. This tool can be applied to ultrastructural studies, cell or organism counting, quadrat surveys, and morphological studies. To try it out, download ImageJ and choose File ▶ Open Samples... ▶ Blobs. This is one of the many demo images that are linked from within the program.

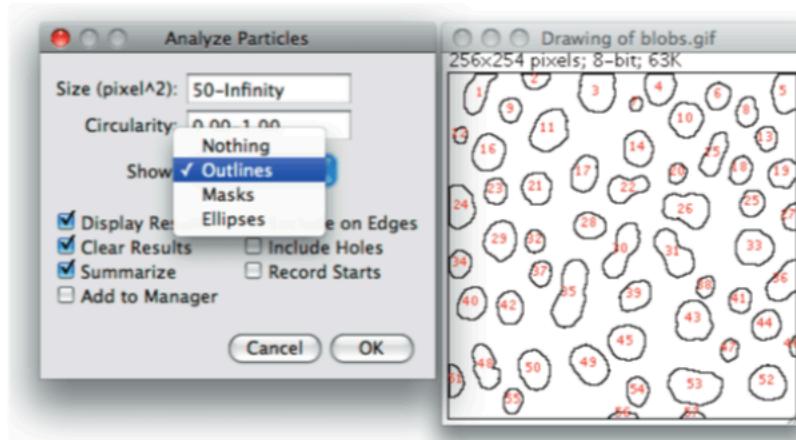
To count the blobs, you have to convert the image into a black and white image. In this case, that means just black or white pixels, with no levels of gray between. The most flexible way to do this is with the Threshold command. Choose Image ▶ Adjust ▶ Threshold... from the menu, or use the keyboard shortcut **⌘ shift T** to bring up the threshold dialog box (Figure 19.3).

Within this dialog, you can use the two sliders to set the brightness range of the image that will be converted to black (shown by the red box in Figure 19.3) or set to white. Play with the sliders to see how the red area expands and contracts, and when you have the blobs distinct from the background, click Apply and then close the Threshold box.

Your picture should have a white background with dark black blobs on it. Depending on the threshold you chose, some of the blobs might be tiny specks. Not to worry: the next step will let you filter out these non-target particles.

Now to count and measure all these particles, get out a pad of paper and a ruler. Or alternatively, choose Analyze Particles... from the Analyze menu. If you try to run this analysis without having converted your figure to black and white, you will be reminded that this only works with thresholded images.

FIGURE 19.4 The results of the Analyze Particles operation in ImageJ



To filter out the small black specks, set the minimum Size to 50 (corresponding to the total area in pixels of the blobs to count), and to visualize your results afterward, select Outlines from the Show pop-up menu (Figure 19.4). You can also choose to ignore blobs that touch the edge, or to count the total area if there are holes in your blobs. Click OK, and you should get a couple of tables summarizing the number of blobs in your image, as well as a new picture showing each blob labeled with a number. From this and from the table summarizing particle area, you can decide whether you want to count certain blobs (for example, blob 30) as representing two actual particles.

MATLAB

In a data storage context, pixel images typically consist of three 2-D matrices, with each matrix corresponding to one of the RGB values of the image. The x - y size of the matrix matches the pixel dimensions of the image. In a standard 8-bit image,⁴ each value in an array is a number from 0 to 255 representing the brightness of that color pixel. Because of their fundamentally numerical nature, images can be sliced, sectioned, analyzed, and processed relatively easily using the array tools in MATLAB. Use the `imread()` function to load an image into an x by y by 3 array. To visualize a matrix (whatever its origin) as a pixel image, use the `image()` function on the array. These can be saved with the `print(' -dpng ', '-r300 ', 'myimage.png')` command. The extra `-r300` parameter in this example indicates the resolution to use at the given screen size.

R

The analysis system R can also import JPEG files with the `ReadImage` package, and TIFF files with the `rtiff` library, installed separately. It can interface with `ImageMagick` (see above) using the `EBimage` package. It can also export image files from its command line or as part of a script.

⁴See Appendix 6 for the explanation of 8-bit images.

Animations

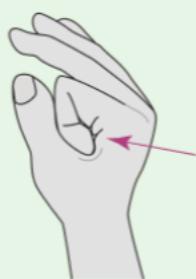
It is easy to create animations from individual images by compiling them as consecutive frames. This is ideally done with the output from a MATLAB or R program which automatically saves a uniquely named PNG each time through a loop. There are many programs which can open a batch of images in a folder and treat the files as animation. QuickTime (www.apple.com/quicktime) and the free command-line tool `ffmpeg` are two examples.

Photography

The visual documentation of organisms, experiments, and data through photography is an important aspect of biological research. Although this is not the venue for a detailed photography lesson, there are a few tips that can greatly improve the scientific pictures that you take.

Aperture and exposure time

Aperture and f-stop The aperture is the effective size of the hole through which light enters the camera, telescope, or microscope. The aperture of most cameras can be opened up to let more light in, or made smaller to let in less light. (Some cameras, such as those typically found in mobile phones, have a fixed aperture.) It may seem strange that you would ever want to exclude light from your camera in this way, but there are other critical trade-offs that are related to aperture size. The smaller the aperture is, the better **depth of field** your image will have. Depth of field describes the distance—foreground to background—over which objects will be in focus. When using a microscope, you can get a sharper view with more depth of field if you close down the aperture in the light path. This is usually controlled with a dial or sliding knob on the microscope body between the eyepieces and the objective lens. The aperture's relationship to depth of field even applies when using an electron microscope.



GETTING A FEEL FOR HOW AN APERTURE WORKS Make a circle with your thumb and first finger and bring your hand up close to your eye, so that you are looking through the hole you have created. Collapse your index finger down until you can see just the tiniest possible pinhole of light coming through. This is what we will call your “manocle.” Now bring your other hand up, so that it enters half the field of view quite close to your face, and look partly at your hand and partly at the scene in the background. Your hand and distant objects should both be in fairly good focus. Move your “aperture”

quickly out of the way to see your hand go out of focus. You can also look through your monocle at the screen of the computer from about an inch away, seeing every pixel, then keep looking while you move your hand away to see things drop immediately out of focus.

In photography, the effective aperture size is quantified in *f-stops*, with values for a typical lens ranging from 2.8 to 32. This value is *inversely* proportional to the open area of the aperture, so the higher the number, the less light and the better the depth of field. *F*-stops are marked in unusual increments: 1.4, 2, 2.8, 4, 5.6, 8, 11, 16, 22, 32, 45. These intervals are selected so that each increment lets in one-half the light of the one below it; thus, *f*/16 lets in about half the light of *f*/11. For most scientific photography, you want the aperture to be on the small side (higher number) so that most of the subject will be in focus.

Exposure The **exposure**, the amount of time that the shutter stays open, is usually displayed by the camera as the denominator of a fraction of a second; for example, 250 means 1/250th of a second. Just like the trade-offs encountered when changing the aperture size, there can be direct effects on both the amount of light and the sharpness of the image when modifying the exposure. The reason is entirely unrelated to depth of field; rather, longer exposures let in more light but make the shot more susceptible to motion blur caused by vibration, an unsteady camera, or a moving subject.

In most situations, you should avoid using exposures slower than 1/60th of a second when the camera is handheld or the subject is moving. Longer exposures may work if the subject is absolutely still and the picture is being taken with a camera on a tripod or through a microscope. For fluorescence images, you may even need exposures of several seconds. When light is a limiting factor, as in these cases, it is usually worth the loss in depth of field to open the aperture and keep the exposure time within reasonable limits. If you have plenty of light and can operate faster (most cameras can go to 1/1000th of a second without a problem) then do it. If you are shooting macro subjects (that is, images which are very close-up) and need better depth of field, then you can use a higher *f*-stop (up to *f*/32), but optical effects (diffraction bands and edge distortion) will start to intrude.⁵ If you know what concerns will be most important for your photograph (for example, avoiding motion blur or maintaining depth of field), you will be able to adjust your settings to account for the conditions.

ISO for sensitivity If you find that the aperture is all the way open and your exposure is so long that everything is coming out blurred (the lab of a rolling ship comes to mind), you have another option at your disposal to get more out of the available light: you can adjust the sensitivity of your camera using the **ISO setting** (also called ASA or film speed). ISO formerly referred to how sensitive the film for your camera was; now it also refers to the sensitivity of a digital camera sensor. ISO typically ranges from 50 to 400, but usable speeds of 3,200 and beyond are available on good consumer cameras. The ISO scales directly to the sensitivity,⁶ so an ISO of 200 is twice as sensitive as an ISO of 100, and requires half the exposure for an equivalent image.

⁵Using your monocle, you can see the effects of distortion and diffraction as well. Look at your screen or a book from up close, and move your view around. You will see a fish-eye effect around the edges.

⁶Finally, not an inverse relationship!

As you have probably guessed, there are trade-offs for increasing the ISO, and—in an interesting parallel—they play out nearly the same for digital and film cameras. As you increase the gain and sensitivity, the sensor also becomes more susceptible to noise, visible as grainy textures and bright speckles. These may not be immediately apparent on the camera's small viewfinder, but they may be very noticeable when your photo is blown up. When light is not limiting, you should keep the ISO low (50 to 100), but as your subject matter becomes darker, faster moving, or more dimly fluorescent, you may need to bump up the sensitivity.

Illumination The best way to avoid the limitations that arise from all of these trade-offs in image quality is to provide more light, although this may not always be possible. If you are taking photographs through a microscope, check the light path to make sure there are no unneeded filters and that the illuminator apertures are open. For stereomicroscopy or macro photography, an off-camera flash on an extension cable or remotely triggered flash makes it possible to provide light directly to the specimen and greatly improve the lighting of the scene. With this additional illumination, you can achieve much faster shutter speeds, especially useful with live specimens.

Overexposure and underexposure In addition to checking your images for focus, you should make sure that the exposure is not clipping, or failing to register fully, the light or dark values of your picture. You want to capture the full range of values in the scene, and not have areas that are overexposed ("blown out") or underexposed. Most cameras can display a histogram of the relative number of pixels at each level, from black to white, present in your image; this is similar to the Photoshop levels dialog box in Figure 19.2. Figure 19.5 shows one such plot for an overexposed image. Unless this is a picture of an object sitting on a pure white background, you can tell from this graph that the highlights are extending off the right side of the plot. You can see that there is a large number of pixels with a value of pure and near white, and that these are not distributed uniformly. In other words, the brightest portion of the image is well off-scale, and other regions of the image that would normally be some shade of gray are also being shown as pure white.

There is often a temptation or a tendency to adjust exposure times so that microscope images, especially of fluorescence, look bright and saturated. If you are fully saturating pixels on your detector, though, you are losing a great deal of information that will not be recoverable. It is often better for an image to be a bit *underexposed*, rather than overexposed, because information can be recovered up to the point that the detector becomes saturated and the values get

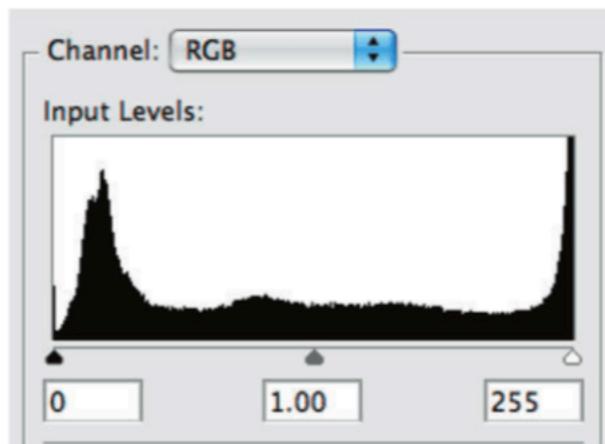


FIGURE 19.5 Levels display from an image with overexposed highlights

clipped at the upper threshold. As long as your intensities aren't going beyond the edge of the levels diagram, you will be able to adjust the exposure after the fact, through linear remapping as described earlier in this chapter.

Color balance

When taking photographs, especially through a microscope, the most common error is to have the camera's **white balance** set incorrectly. Sources of illumination have different intrinsic temperatures—the orange glow of a candle is cooler than the blue flame of a gas torch or stove. In the context of photography, incandescent lights have a yellow tint compared with the bluish light of a strobe or sunlight, so cameras have built-in settings to correct for these differences.

White balance is usually found under the WB or AWB menu on a camera. Set it manually according to the demands of your lighting setup. Use the little light bulb icon if you are using the dimmable light source on a microscope or other incandescent lights, and use the lightning bolt if you are using a strobe. Outdoor or natural light should of course use the setting for sun or clouds, since daylight has a strong blue component to the camera sensor. For other illumination sources, such as LEDs, you may have to experiment a bit. The fluorescent bulb setting was designed for use with fluorescent room lighting, but it actually works well for capturing fluorescent microscope images, since it falls somewhere between the yellow of incandescent bulbs and the blue of a strobe.

Once you are aware of these white balance issues, you will notice right away from the blue or yellow tint of your photos and the photos of others if the white balance was set incorrectly. This sometimes happens when moving back and forth between taking bright-field and fluorescent scenes on a microscope. If your images have already been captured with an incorrect color balance, you can still make adjustments to the hue with careful processing in an image editing program. If your images were taken in RAW format, then you can set the white balance at a later time without losing any image quality. RAW, despite its larger size, also has other advantages, such as a larger potential color space.

Automatic versus manual operation

By default, most cameras have the capability to automatically navigate the settings and trade-offs discussed above, including white balance, aperture, exposure, and ISO. In fact, at times it may be difficult to find out how to change settings manually. Scientific images, however, often present unique challenges that aren't addressed by one-size-fits-all consumer options. Automatic settings, especially white balance and exposure, will rarely give accurate results. You will need to switch to a manual mode and adjust your settings. It is good practice to bracket your options by taking exposures both brighter and darker than you think you might need. You can then select the best one when you are viewing the images on a large monitor and not rushing to finish an experiment.

SUMMARY

You have learned:

- The general strategies for compressing pixel image data, and their implications for working with images
- The trade-offs between the common pixel art file formats, including JPEG with lossy compression and PNG with lossless compression
- The basics of two popular pixel image editors, Photoshop and GIMP
- Tools to edit images at the command line, including ImageMagick and sips
- How to extract quantitative data using image analysis tools like ImageJ
- Some principles for taking photographs

Moving forward

- Identify recurrent uses of photography in your research (gel documentation, microscopy, etc.), and see if there are ways to improve or automate repetitive portions of the workflow.
- Open one of your images in Photoshop or GIMP and try adjusting the levels controls to get comfortable with how they affect your picture.
- Visit the gallery at processing.org to see examples of interactive plots and try out Processing, a programming environment for graphics. Processing has a low barrier to entry, and can interact with hardware like Arduino as well (see Chapter 22). Other related images are shown in the gallery section of nodebox.net.