Chapter 17

GRAPHICAL CONCEPTS

Many scientific data are inherently visual, and the ability to clearly communicate using images is central to preparing presentations and publications. Nonetheless, scientists are usually left on their own to learn how to navigate the many trade-offs and pitfalls of the tools, settings, and file types that are available for handling images, as well as how to make basic aesthetic decisions. These decisions can have profound effects on the ability to successfully and effectively communicate, yet scientists often unnecessarily compromise their images because they are unaware of these effects. Before examining the specifics of creating and manipulating images, we will therefore look at some general aspects of images which affect their effectiveness, quality, color, and file size. These considerations will influence how you generate, store, modify, and present your images.

Introduction

Within a relatively short time, image and graphic preparation methods have completely transitioned from Rapidograph pens, rub-on letters, and darkroom chemistry to computer-based illustration and editing. At the same time, journals have shifted the burden of image production to scientists, requiring publication-ready digital files. No longer can you submit a photo that looks right; it has to be a "CMYK image, with 300 DPI at printed dimensions, saved as a TIFF with LZW compression." These variations in the way graphical information can be formatted can be confounding; even journal editors and photo managers sometimes perpetuate misconceptions through the ways they describe image requirements. In this chapter we discuss the technical aspects of images that are relevant to scientific publication, and we introduce tools and techniques to prepare figures for print or display.

General image types

Vector versus pixel

When preparing artwork, the most fundamental consideration is whether it should be a **vector**-based or **pixel**-based image. Vector art is made of independent editable lines, curves, and shapes, all of which are defined by a few key properties. A pixel-based image, also called **bitmap** or **raster art**, is made of a uniform grid of colored dots, the pixels.

In vector artwork, a line can be defined by just two endpoints. If you specify a starting point and an end point, the line will stretch between them. Move one point and all the intervening positions of the line automatically follow. In pixel art, on the other hand, a line consists of many points of a particular color, arranged on the screen next to each other (Figure 17.1). In a way, the pixel line exists only because the colors adjacent to the line are perceived as different; there are no special "line pixels" as distinct from "non-line pixels," except that we see them that way.

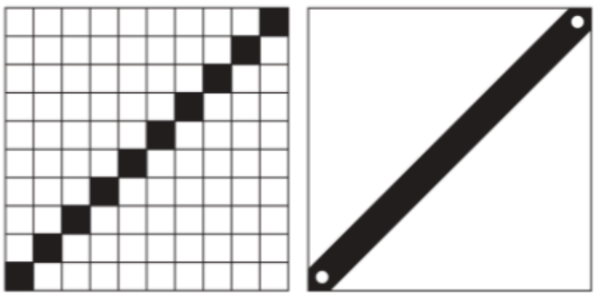

FIGURE 17.1 A 100-pixel (10 × 10) line and a two-point vector line Although it requires 1/50th the number of points, the vector line (right) is usually easier to edit and prints more clearly.

Consider the job of storing the information that describes a straight black line on a white background. For a vector line, you have to save the X and Y location of each end point, the color and width of the line, and the color and dimensions of the background. In this idealized example, there are about nine pieces of information required.[1] This amount of information stays the same regardless of what size the line appears on the screen or paper. It could be scaled up to a billboard and would still only require that handful of descriptors to recreate it. Now imagine the same situation for a pixel-based image. To estimate the amount of information it will take to record the line, you need to determine the dimensions of the image where the line will appear. Is this line on a 10 × 10 grid of pixels, or on a 100 × 100 grid? The information stored in a bitmap is the color of each point in the grid. For an image of just 10 × 10 pixels, this is 100 pieces of information; the color of each point has to be specified whether it is used to draw the line or not. For a 100 × 100 image (still relatively small), this value goes up to 10,000 pieces of information that must be stored. Fortunately, there are ways to compress this information and take advantage of the fact that many of those pixels are the same color.

There are dramatic differences between the ways that you work with vector and pixel art, and in most cases entirely different programs and file formats are used. File formats for storing vector-based images include PDF, EPS, SVG, AI, and

[1]For the moment, we are ignoring many subtleties, including that each color usually requires at least three values to describe it.

PostScript. These are discussed further in Chapter 18. Common pixel-based formats include JPEG, PNG, TIFF, BMP, and PSD, which are described in Chapter 19. Particular vector formats such as PDF, EPS, and AI can also store embedded pixel information, so they can accomodate hybrid images. When sending or transferring files, using the proper extension for your file names (`.pdf`, for example) can make it easier for the recipient to correctly identify the format.

A line in a vector-based document can be repeatedly moved, rotated, or scaled up to any size, as well as have its color and width changed, and it will always look perfectly smooth at any magnification. As you zoom in on a pixel-based image, however, the little colored squares that make up the image become more and more conspicuous, leading to jagged edges and other visual imperfections. It is also not always possible to manipulate the width, color, or position of an object once it has been created in a pixel-based world. As we saw above, not only can pixel art require thousands of times more memory, but to moderate this requirement, images are usually compressed in ways that can degrade their quality.

Deciding when to use vector art, pixel art, or both

So why does anyone use pixel art if it has so many potential shortcomings? For one thing, data are frequently collected as a grid of pixels to begin with. A photograph is the obvious example. Pixel-based images also have the advantage that they can contain a large amount of complex and interacting information, which would be hard to represent on a line-by-line, object-by-object basis. However, many scientists default to a pixel-based approach for generating images without having considered whether vector art is more suitable. This is often because they are most familiar with pixel-based art editors such as Photoshop.

A good guideline is that any graphical element you create de novo should begin as vector art and thereafter be preserved as such all the way through publication. Anything that originates as pixel art can be maintained as pixel art, unless it is more appropriate or informative as a vector tracing. Any annotations, arrows, letters, or graphs should be generated as vectors; in other words, if an image doesn't start as pixel art it should probably be created and maintained as vector art. Of course there are exceptions to these guidelines, and it is not hard to find examples of both photorealistic vector art and starkly graphical pixel art.

There are several other advantages to vector art. It is always possible to convert from vector to pixel art, but it is difficult to go the other way. Furthermore, pixel text in figures is not searchable, which means that words in pixel images will not be cataloged in search engines and will be missed by a user who is searching their document automatically. Pixel text can't be easily copied and pasted to another document, either.

Following these guidelines means that, in general, vector art will be used for anything with iconic graphical elements, such as graphs, tables, schematics of experimental setups, diagrams, and flowcharts. Text should be created and kept as vector art. Pixel art is necessary for photographs, gels, most 3-D images, and pictures with subtle tonal gradations or complex visual signals like blurriness or

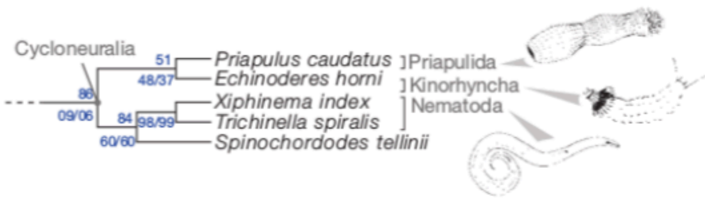

FIGURE 17.2 Detail from a figure that combines vector and pixel art This image features vector art (left) and black-and-white pixel art (images at right) in the same file.

grain. Pixel art might also be called for in plots with thousands of data points, where the vector objects begin to overlap and become cumbersome.

There are many times when it is best to use a combination of both vector and pixel information. Typically, labels for a plate of photographs should be annotated using a layer of vector art. Although the photos themselves cannot be scaled up without compromises, there are still advantages to having the annotations clear and scalable. It is possible to mix pixel and vector art in this way because most vector image file formats (such as PDF) allow pixel art objects to be embedded within them. You can, for instance, import a pixel image such as a photograph into a vector art file, and then label and annotate it with lettering and graphical elements that are themselves vector art. There is no need for all elements of an image to be pixel art just because one element is, so you can have the best of both worlds (Figure 17.2).

Ultimately, any computer-generated image that we see, whether on a display or on a printed page, has been converted to a pixel image to make it visible. Ideally this process, known as rasterization, happens at the last possible moment within the hardware generating the image—just before the ink hits the paper or the dot on the monitor is illuminated. When you are working on an image with vector content, you should defer this conversion to a bitmap as long as possible. This way you will end up with the appropriate resolution supported by the output device, while maintaining the ability to edit the image without loss of quality. Pixel art is overwhelmingly the image currency of the Web, although this may change with increasing adoption of the SVG (vector) image format, which uses XML syntax to describe vector images. In the meantime, it may sometimes be necessary to convert your vector art to an appropriately sized pixel image to post it online. Even so, it is best to delay this translation until all possible editing has been done, so that you maintain a vector version of the final image, and the pixel image is as sharp as possible.

Image resolution and dimensions

Pixel images are made up by a grid of colored pixels—for example, the 10 × 10 grid of dots described earlier. At some point you will have to deal with the question of resolution and size. There are three interconnected values associated with a bitmapped image's size:

- **Pixel dimension**: the number of pixels along the full X and Y axes of the image, for example 800 × 600 pixels
- **Physical size:** the size that the image appears on a printed page, such as 89 mm × 66 mm

- **Resolution**: the size of each pixel, expressed as the number of pixels per unit of physical dimension, usually called dots per inch (DPI) or pixels per inch (PPI)

If you know any two of these descriptors, you can determine the third via the formula: pixel dimension = physical size × resolution. However, of these, the only real dimension describing a bitmapped image is the **pixel dimension**. It is the pixel dimension that determines how much information is contained within an image, which in turn determines file size and the level of detail that can be represented. In effect, the resolution is just an arbitrary multiplier that is stored with the image file, so as to specify the size at which the image should be displayed.

The resolution, and therefore the physical dimension, can be changed at any time without fundamentally impacting the information that represents the image (Figure 17.3). For example, the actual pixels making up an image of a given pixel dimension could be a square micron or a square meter in size; either way, the information content of the image would be the same. Imagine projecting an image of our 10 × 10 line onto a screen and then moving the projector further from the screen. As you moved the projector away, the physical size of the pixels on the screen would increase (in other words, the resolution would decrease), as would the overall physical dimensions of the total image; yet even so, there would still be only 100 pixels being displayed.

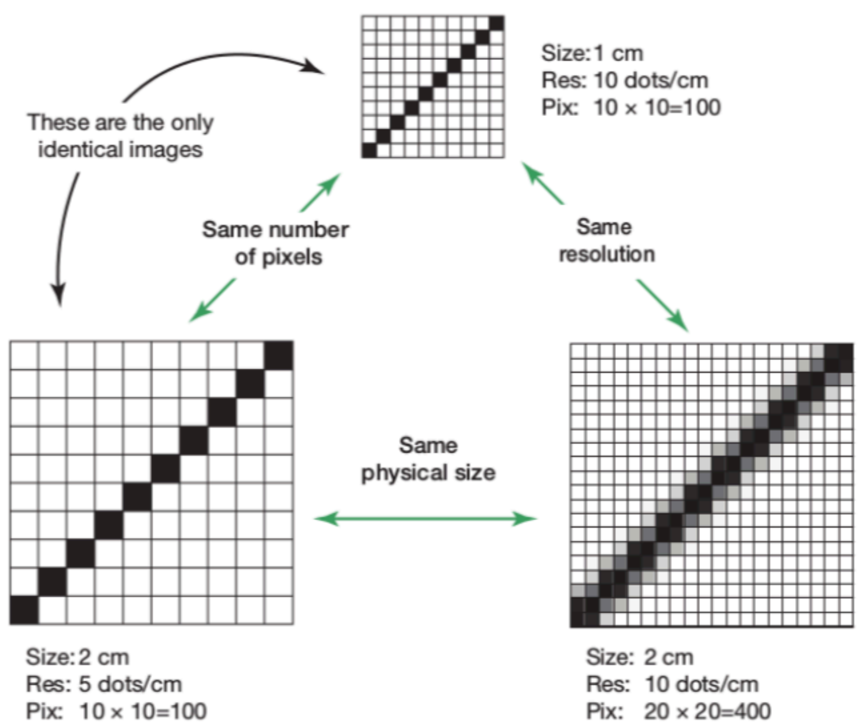

FIGURE 17.3 The relationships between physical size, resolution, and pixel dimension You can have images of different physical dimension and resolution, yet with the identical information content (pixel number).

Image resizing and the DPI misconception

Given these relationships, it should be clear why it is not informative to ask someone for an image "at 300 DPI" without specifying the associated physical dimensions. At 300 DPI, the image could be 100 pixels wide, 30,000 pixels tall, or any other dimension. There is no way to know, yet journals commonly make such requests.

Before you resize an image, you should consider the largest physical dimension at which you will conceivably use that image. Will it be considered as a cover photo for a magazine, used in a presentation, or only printed in the journal in one 89-mm wide column? Next consider the resolution required at this final dimension (often 300 DPI or 120 dots per cm, but see below). Then multiply the expected physical dimension of the image by the required resolution to get the pixel size of the image. It is best to err on the side of a larger overall pixel dimension if you don't yet know your exact requirements.

The required resolution for an image depends on how it will be shown to the user. Typical resolutions are 72 to 100 DPI for images that will only be displayed on screens (such as web content), 300 DPI for printed color images, and 600 DPI for printed black and white images. The human eye can see finer detail on a printed page than on a screen, and printers typically have much higher resolution than displays. This is why files destined for printing need higher resolution—a 90 DPI image that looks crisp on a screen will often seem blocky and fuzzy on the page. The eye can also make out greater detail when the contrast of an image is higher, which is why images need to have higher resolution when they are in black and white.

You can safely change the resolution or physical dimensions of a pixel image without impacting its quality or further limiting future manipulation—this is the equivalent of moving the projector closer and further from the screen. It is easy to undo these changes just by entering a new resolution or physical dimension. If, however, you make any changes that affect the pixel dimensions of the image—in other words, if you decide to resize it—the image will be **resampled** by the editing program (Figure 17.4). There are two possible ways to resample. First, you can downsample your image, by reducing the image's overall pixel dimensions; if you do this, then you will be throwing away information, as existing pixels are combined into a smaller number of new pixels. Second, you can upsample the image, or increase the pixel dimensions; here you will be attempting to create more information through interpolation. There are very few circumstances where it makes sense to increase the pixel dimensions of an image—usually this just unnecessarily increases file size.[2]

If you will be performing any image adjustments to your file, such as adjusting the levels of brightness, do these before you resize the image. This is to avoid introducing artifacts, or imperfections that tend to be left over after processing. If you increase the pixel dimensions of an image it can amplify the artifacts that arise

[2] One circumstance where upsampling is sometimes necessary is when a photo editor insists on a "higher resolution" of a photo that is already of suitable pixel dimension.

from any earlier adjustments, and if you reduce the pixel dimensions, it will actually reduce the artifacts from adjustments that have been applied prior to resizing.

You should minimize the number of times an image is resampled, and defer resampling as long as possible. Resampling results in a loss of quality, due both to the information that is thrown away and to artifacts of the resampling process itself, and this damage can't be undone simply by returning the image pixel dimensions to their original values. The primary motivation for reducing the pixel dimensions of an image, apart from resizing it for final distribution, has historically been to minimize file size to save disk space and make manipulations less computationally intensive. These concerns are less pressing than they were even a few years ago, given the large performance improvements of personal computers. If you are going to send an image by email or embed it into a presentation, then resampling a copy is good practice. A full screen image for most projectors will be no more than 1024 × 768 pixels, so you will probably get better results during your presentation if you use a duplicate that has been reduced in size to these dimensions. You will generally want to save a copy of the unmodified image before performing any size modifications or otherwise irreversible changes.

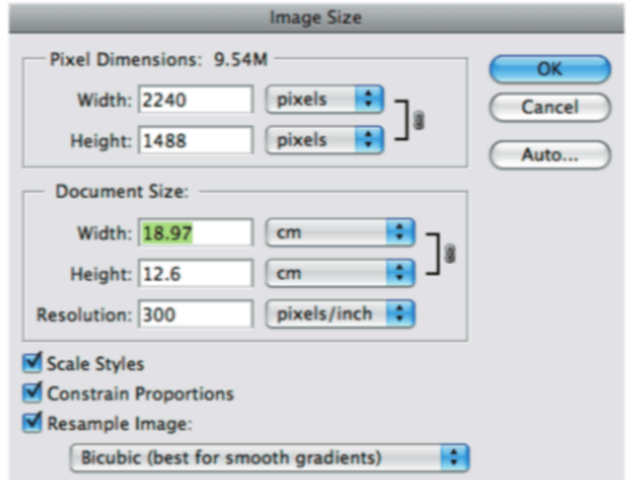

FIGURE 17.4 The Photoshop Image Size dialog box There are options for setting the pixel dimensions, document size, and resolution. Additional options are available to control resampling. If Resample Image is not checked, then your pixel dimension, and thus the true image size, will not change.

Typically the proportion of length to width is fixed during image resizing to avoid distortion.[3] If you resize the width to 50%, then the proportional value for the length should be determined automatically. In cases of resizing to fit a journal's column, don't worry about the height; just fix the horizontal dimension to the appropriate width. You may get better results resizing by integer multiples (for example, half-size or quarter-size) rather than irregular percentages (for example, 43%); this is because with the former, the program can in effect get rid of every other pixel.

Because vector art is largely resolution-independent, these resizing issues mainly become relevant when a vector image must be rasterized for use on a web page or in a presentation. Some vector resizing operations may not adjust the size of lines or text proportionally, so you can affect the overall look of your image.

A few journals are still reluctant to accept vector formats for submitted graphics, despite their superior printability. In these cases we have had some success

[3]There is an impressive method called "seam carving" which can resize images non-proportionally without distorting their content. This is available as Content Aware Scaling in the most recent versions of Photoshop and as liquid-rescale in the command-line tool Imagemagick.

contacting the person responsible for page layout, and seeing if they can handle a vector format. In all likelihood, they will welcome the improvement. If not, you will need to rasterize the image based on the guidelines above for dimension × resolution. Choosing an appropriate image compression format will also be important for your rasterized vector images, as described in Chapter 19 on working with pixel art. When saving graphics files in EPS format, you have the option of embedding the font descriptions into the image, to assure a match with the publisher's system. If you are using non-English typefaces, you can also convert text to outlines; although this means that the text will no longer be editable, the characters will look right on any computer.

Image colors

Color models and color space

In addition to deciding up-front whether an image should be pixel or vector art (or perhaps both), you also need to consider the document's **color model**. The color model establishes the basis of the color system used to describe your image. It essentially determines what the primary colors are and what happens when they are mixed. The two most common color models are **CMYK** (**c**yan, **m**agenta, **y**ellow, blac**k**) and **RGB** (**r**ed, **g**reen, **b**lue). CMYK is the system used in operations that deal with ink and pigment and is named after the colors of ink that are used in printing. RGB is the standard descriptor for computer displays and digital photographs and is named for the three colors that each element of a display emits.

Many graphics programs allow you to set the color mode of each image file. In Photoshop, for example, this capability is located in the Image ▸ Mode menu, while in Illustrator it can be found in the File ▸ Document Color Mode menu. Other programs, such as Inkscape, work entirely within RGB.

The CMYK color model is based on the absorption of light by pigments. It is therefore a subtractive system, in which higher amounts of each pigment lead to darker colors. A white piece of paper with no pigment has CMYK values of `[0,0,0,0]`, where each value is the percent of cyan, magenta, yellow, and black ink deposited. As you begin to add pigment, color emerges, and the overall brightness is decreased. A CMYK of `[100,0,0,0]` is cyan, while `[100,0,100,0]` is cyan plus yellow, resulting in a dark green. Even when you add in 100% of all the colored inks, they can't absorb all the light, so the resulting image is not truly black. This and other darker tints require the addition of black ink to the mix. You can see how there might be several ways to achieve a similar dark color, either by the addition of more of each colored pigment or by the addition of more black. This turns out to be a common problem when converting between color spaces, as we will describe shortly.

The RGB color model is better suited to devices like monitors and projectors, which emit light rather than absorb it. If you look at your monitor with a magnifying glass, or even put a tiny drop of water on it, you will see that what appears to be white is really three brightly lit colored pixels. Higher values indicate more photons of whichever of the three colors are being combined, until the maximum is reached for each of these values; added together, the maxima produce white. For this reason, RGB is called an additive color system. A larger range of colors can be produced by an RGB device, both in the sense of brightness and how saturated the colors look, than can be captured on a printed page (Figure 17.5). RGB values are usually given on a scale of 0–255. In RGB space, `[0,0,0]` represents black (all colors turned off), while `[255,255,255]` represents white (all colors turned on at their maximum value). Note that higher values in RGB signify more light, while higher values in CMYK signify more absorption (less light).

ENCODING IMAGE DATA The range of 0–255 for RGB values may seem arbitrary until you think about the binary values used to encode it. One byte of computer memory is 8 binary digits (or bits), and therefore can store values from 0–255 (2^8=256, with zero being the 256th value). So by dedicating one byte of memory to each of the red, blue, and green pixel values, it makes efficient use of memory in defining the color space. You will see that the number 256 recurs in many places in computer operations, as described in Appendix 6.

With the standard 8 bits per color, a total of 16,777,216 colors (2^{24} or 256^3) can be defined. You may hear about 16-bit or even 32-bit color, as supported by some cameras and image-editing programs. Here, two bytes (65,536 values) are used to define each RGB value, leading to a very large tonal range. With a high tonal range, the subtlety of gradations that can be captured for each channel is greatly increased; as a result, features like details in shadows can be seen. However, image compression methods such as JPEG can alter this range and noticeably affect the quality of an image.

Color space is the actual range of real colors, independent of how they are described. Neither RGB nor CMYK can describe every color in the full color space, which is why the color model you select can have a big influence on the appearance of your final image. RGB describes a larger portion of color space than does CMYK, and the colors that can be described with CMYK are a subset of those that can be described with RGB.

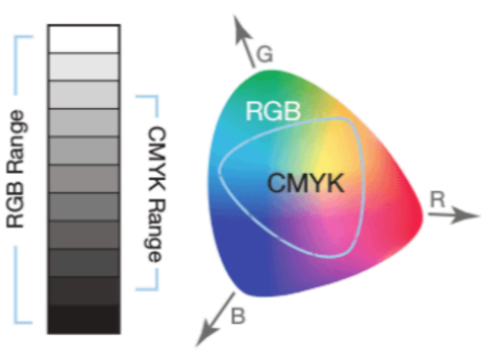

FIGURE 17.5 Schematic representation of color range In RGB, increasing values correspond to brighter pixels and thus lighter colors, while in CMYK they represent more ink, thus darker colors. An RGB model can represent colors that are both brighter and darker than possible in CMYK. Another way to look at it is that the 2-D color space represented in CMYK is a subset of that found in RGB.

Converting between color models

When converting a file from RGB to CMYK, colors in RGB that are out of the range of CMYK need to be compressed, or translated, to colors that are within CMYK. There is no one right way to do this, and the way this compression is done can have very different results. Publishers request CMYK files because they want to make sure that color conversion decisions meet author's intentions and needs. Color conversion can render a figure unintelligible if elements that had different colors in RGB are translated to the same color in CMYK (Figure 17.6). Photographs may no longer bear a reasonable likeness to the original subject (especially if they have saturated colors), or images may just appear ugly with a slightly different color scheme. Making the conversion before the files leave the author's hands avoids these problems and reduces the number of changes that would need to be made when proofing pages.

Bright colors and blacks in particular are difficult to translate from RGB to CMYK. Figure 17.6 demonstrates a particular case where the translation changes the meaning of the figure. We can't accurately show you an example of the RGB image on the printed page; however, Figure 17.6 is also available as an RGB image in `examples/MapColors.pdf`, so that you can see the actual colors on your computer display. Compare the three pairs of color values in the white boxes. Each is distinct in the RGB version (the red is somewhat subtle), but they are nearly indistinguishable in the CMYK transformations. When someone downloads a PDF of your paper and prints it out, this is how the image will look to them. Knowing this in advance, you can control the transformation of colors, and design your images (and your color schemes) to look good and convey information both on a display and in print.

When converting from RGB to CMYK, a variety of options are available that provide control over the translation process. The default settings in a photo editing program will often have adjustable limits on the amount of black ink the program assigns to a color, and this provides one of the most important controls for adapting the translation process to an image. Some programs like Inkscape don't support CMYK colors, though, and MATLAB can only export CMYK in certain formats (TIFF and EPS).

When working with photos it is best to stay in RGB-space for as long as possible. Many of the adjustments you might want to make are exclusively or more effectively implemented in the larger RGB color space. If your image is destined for display on the Web or in a projected presentation, you can leave it as RGB. For an image to be printed, however, it must ultimately be converted to amounts of ink on white paper, meaning the RGB values will probably need conversion to a CMYK color space.[4] If you are handling printing yourself, contact the prepress or art de-

---

[4]Some newer printers will accept RGB files directly, but you will still be relinquishing control over the conversion process. If you are printing a design with only a few colors, you can use process colors, such as the Pantone matching system. These special ink mixtures are reliably referenced to a book of printed swatches so that colors match exactly with your expectations.

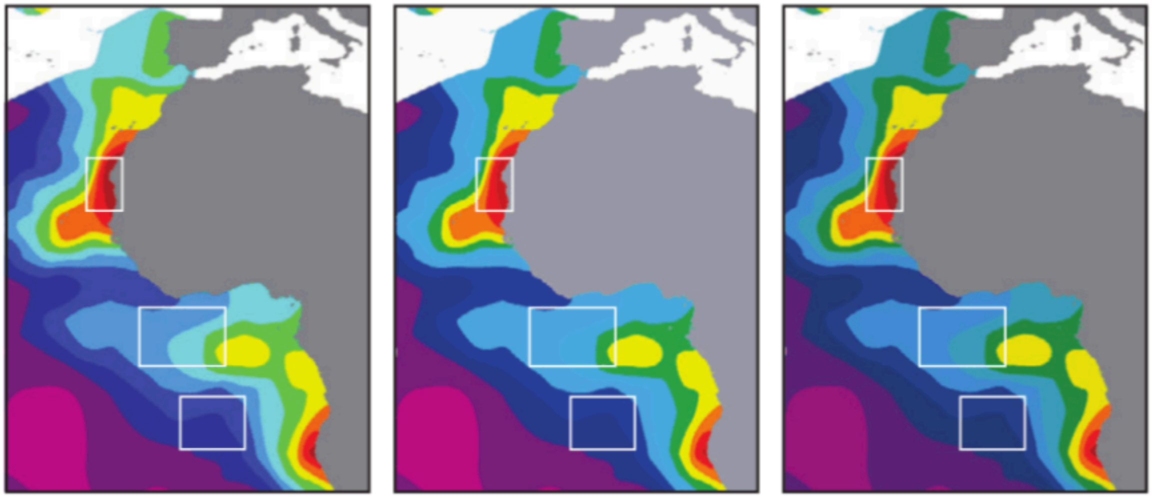

FIGURE 17.6 The same image represented in different color spaces To see the true RGB color, open the example file `MapColors.pdf`. RGB is how the image appears on the screen, CMYK with coated paper is how it would print in a journal, and CMYK with uncoated paper is how it would appear on a laser printer. Given this selection of colors, information about the oceanographic features in the white boxes is lost due to color mapping issues.

partment to determine their requirements for output. They can inform you of the proper formats, color profiles, and color conversion settings for their facilities.

Color gamut and color profiles

The **color gamut** of a display is the number and range of different colors it can produce. A display with a high gamut can produce many colors, with fine gradations between them, while a display with a low gamut produces fewer colors. It can be difficult or impossible to distinguish small variations in color and intensity on a low gamut display, especially in the case of red or dark blue on a black background, and cyan or yellow on white backgrounds. One may spend hours working on a presentation on a standard computer display, and then stand up in a room full of people and realize that critical colors are indistinguishable on the projector. This is because many projectors have a shallow color gamut, while even inexpensive computer displays can show millions of colors.

Dealing with variable color gamuts raises many of the same issues as dealing with different color models—in both cases, the colors of an image are translated and compressed to a smaller part of color space. In fact, different color models are used precisely because they anticipate the reduced color gamut of particular output devices, such as printers in the case of CMYK.

Even though an RGB value of [255,0,0] represents "pure red," it will not look exactly the same on different displays. Similarly, 100% cyan ink will appear different depending on the paper and printing process used (see Figure 17.6). It would be quite cumbersome, however, if a different color model had to be used for every single model of computer display, projector, printer, and mobile phone screen.

Information about the color space of an output device can be stored in a standard format known as a **color profile**. Using color profiles, it is possible to specify how the same image will look on different types of output devices. If you know the color profile for a particular printer, for instance, it is possible to anticipate how an image will look when printed to it. In Photoshop, you can get a preview of what the image will look like in various target color spaces using the Proof Setup and Proof Colors items of the View menu.

In addition to color models and color profiles, it is also possible to specify the color space of a file. For present purposes, to insure that you are working in the largest possible color space, you should adjust your color settings in your graphics programs so that you are using the selection in Color Settings... called **Adobe RGB 1998** by default. Web applications cannot represent this larger color space, so web images should be viewed in the commonly used sRGB color space. As for converting a file's color profile, we won't address the mechanics of that here, but you can find out more by searching the Web and reading about "rendering intent." In Photoshop, conversion settings are found in the Edit ▸ Convert to Profile... menu option.

Color choices

At least 7% of males have some degree of color blindness, which usually affects the ability to tell red and green apart. For this reason, you should avoid using red and green to distinguish features of your figures. If you would like to use colors similar to red and green, the simplest fix is to use magenta, which can be differentiated from green, in place of red. You can check how your image will appear to the color blind at `vischeck.com` if you are uncertain about your color selection. The goal is not to create a figure that will look the same to everyone, but to make sure that the information that is conveyed through color can be perceived as distinct. Versions of Photoshop including CS4 and higher also have a filter built in for a color-blind simulator.

At the same time, you may want to consider how your image might appear in grayscale. Some journals offer the cost-saving option of printing the figures in grayscale in the journal itself, but retaining color for no additional cost in digital reprints. Another reason to consider the implications of grayscale is that people may also have to photocopy your paper from a journal. You should check the overall tonality of your color choices to see how they translate into lighter and darker grayscale values. Although there are many options for converting color to black and white, you can get a quick preview of this conversion using the same Convert to Profile... option used in Photoshop to convert from RGB to CMYK.

In addition to the RGB versus CMYK issues illustrated in Figure 17.6, you should also consider color gradients when choosing a color scheme. Color gradients are often used to convey information in a figure, and to be clear what values in your figure are higher than others, you should choose a color scheme with an intuitive progression from cool to warm colors (or dark to light). Some programs have default color palettes in which similar colors are used to represent the lowest and highest values, or in which the similar colors reappear through the palette. These cause the data to be difficult or impossible to interpret at a glance, or lead to confusing banding patterns in contour plots. See the file `examples/ODVpalette.pdf` for one particularly problematic color scheme. There are many online and desktop color palette generators that can help you create color palettes, including `kuler.adobe.com`, `mypantone.com`, `colorbrewer2.org`, and `degraeve.com/color-palette`.

HEX VALUES FOR COLORS You can refer to RGB colors by their numeric values (for example, `[9,141,255]`) but in editing programs and on the Web, you will often see color encoded using the corresponding **hexadecimal** values (in this case, `#098DFF`). This seeming gibberish is actually a straightforward way of representing the same information, but using three base-16 or hexadecimal digits (`09, 8D, FF`), rather than the normal base 10. (See Appendix 6 for an explanation of hexadecimal notation.) The maximum value of a two-digit hex number is FF, which represents 255. It is not a coincidence that this is also the maximum value for each RGB color. It is relatively easy and efficient to represent three numbers ranging 0–255 using only six characters, in a way that corresponds to how the computer stores the values in memory. You can see the hex values for colors in the "web palettes" of many color pickers, including the system color picker available in TextWrangler.

Summarizing the decision-making process

While there are many options and potential file formats for any particular graphics job, most of the decision-making process can be distilled into just a few points. They are recapped here, and portrayed in Figure 17.7 as well.

Use RGB vector art wherever possible, making conversions to CMYK and pixel art as necessary for final presentation. It is often useful to generate diagrams and graphs for publications using a CMYK color palette from the start, so that you have fewer surprises when your fluorescent green and hot pink color scheme becomes disappointingly non-distinct upon printing.

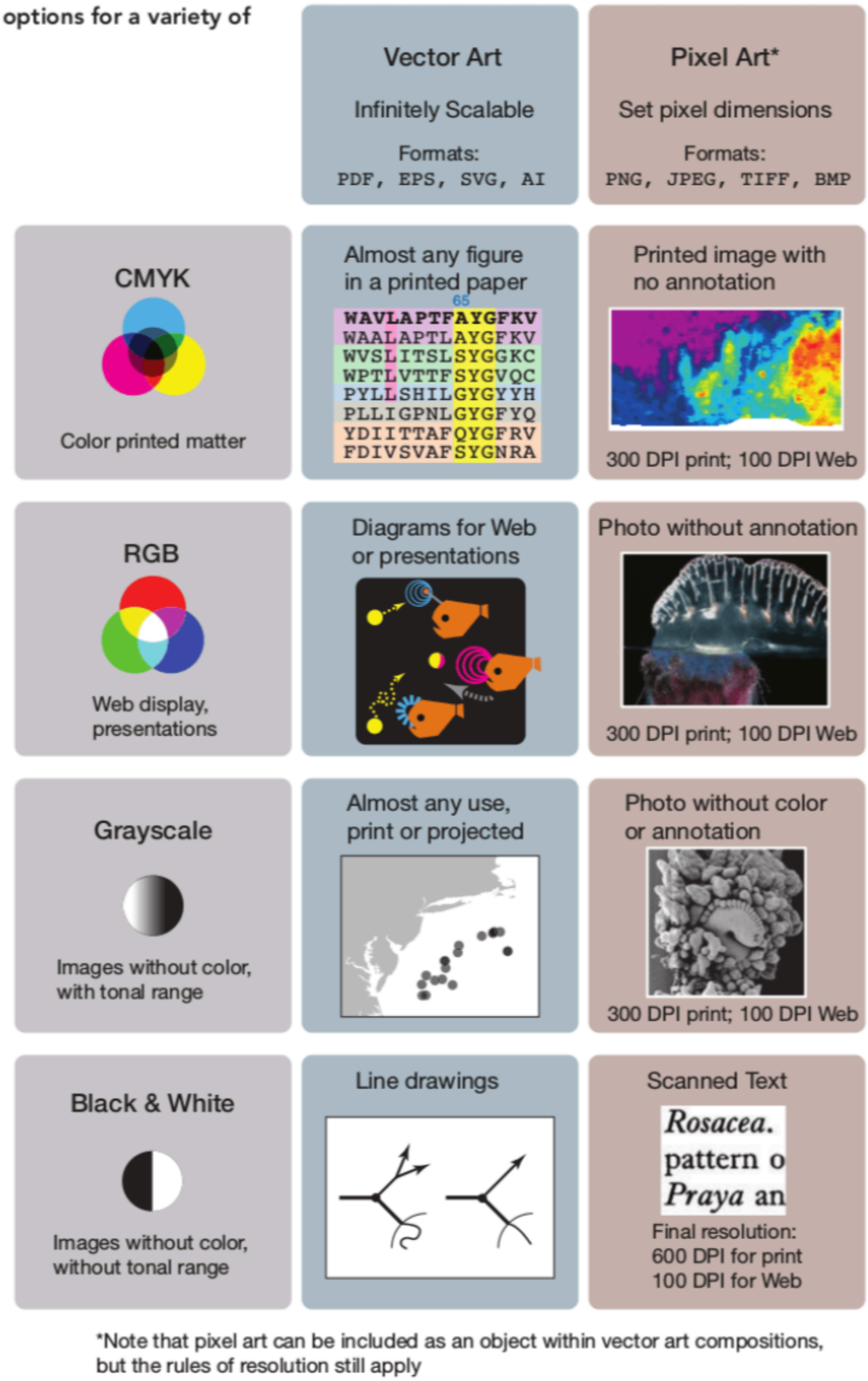

FIGURE 17.7 Graphical options for a variety of scientific uses

Layers

Layers in image editing programs can be thought of in a very physical sense, as if they were transparent pages in a sketchbook. Each graphical element is on one of these pages. The layers can be reshuffled to bring some groups of elements in front of others.

Layers are most useful, though, as containers for organizing the objects that make up an image. Often, text annotations will be on their own layer, graph data points on another, and graph axes or scale bars on another. You can place scratch work or other items you don't want to include in the final image in their own layer, and then make this layer invisible. This allows you to preserve reference images that help you remember how the image was generated, or to save rejected bits of a figure that you may want to use later, without managing different files. Layers can also be locked, so that you can edit some elements of an image without modifying other elements you wish to remain unchanged. Locking some layers also makes it easy to quickly select a subset of graphical elements from a complex image.

Even though layers greatly improve the efficiency of drawing, they are often underused by people new to image editing. When layers are used to their best advantage, a typical scientific figure may end up with a dozen or more. In fact, it is difficult to have too many layers. Layers are supported by all but the most basic graphics editors; however, they are not preserved when an image is saved to JPEG or PNG file formats. This means that maintaining layers while working on pixel images constrains your selection of file formats to either PSD (the default format in Photoshop format), TIFF (also available in Photoshop), or XCF (a format found in GIMP, an editing program discussed later Chapter 19).

General considerations for presenting data

There is, of course, much more to preparing an image than decisions about color space, vector versus pixel art, and file format. None of these technical issues will make a difference one way or the other if the image isn't designed to communicate effectively in the first place. There are a variety of aesthetic issues that arise repeatedly in scientific illustrations. When poorly addressed, these problems can make it difficult and time consuming for your audience to grasp what you are trying to say, or your audience may miss your point entirely. These issues are critical to keep in mind when you are acquiring image data, drawing new artwork from scratch, and laying out figures and slides.

Eliminate visual clutter

One of the most common problems with scientific illustrations is clutter—distracting graphical elements that don't add information. Extraneous graphics are sometimes added deliberately with the intent of making the image more appealing, while at other times they may be the passive result of not optimizing graphs or other artwork generated by an analysis program. Bar charts in 3-D might be common in annual reports, but are a liability for scientific papers (Figure 17.8).

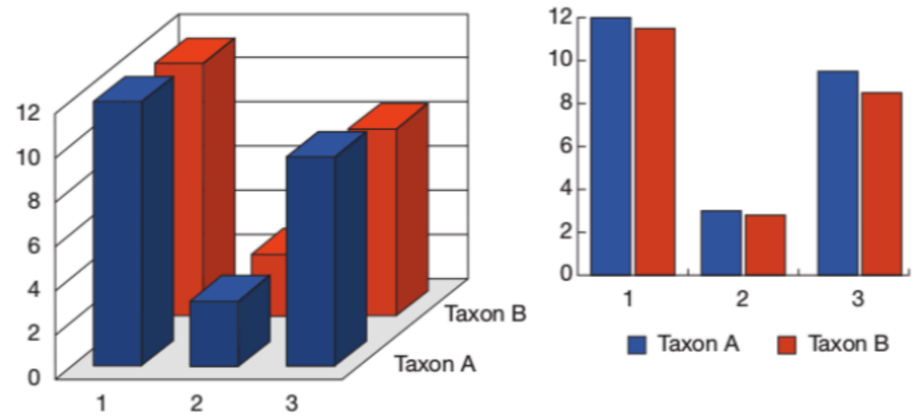

FIGURE 17.8 Bar graphs of the same data in 3-D and 2-D Although most scientists stick to the clearer and more informative 2-D versions, it is not uncommon to encounter 3-D plots produced by spreadsheets in the literature and especially in slide presentations. The plot at the left obscures the fact that the data from Taxon A, in blue, are slightly higher than the corresponding values from Taxon B.

Clutter is often added in an attempt to compensate for existing clutter. If you find yourself thinking that you need to add an element to an image to improve readability, think instead of ways to achieve the same goal by removing existing elements or reorganizing what is already present.

The default settings of spreadsheet programs can generate graphs with jarring combinations of visual elements: darkly pigmented background colors, extraneous borders, thin hairlines which don't print well, and a default panoply of colors for the lines and symbols of the graph which don't convey additional information.

Use transparency for overlapping data

It is often a struggle to represent large amounts of data on a single plot. The number of lines or points on the graph can obscure other values that lie behind them, leaving an amorphous blob of data that can be difficult to interpret (Figure 17.9A,C,E). In some cases, when the density of points is an important aspect of the data, it will be worthwhile to make a contour plot for the data—essentially a top-down view of a 2-D histogram—in which the color of the plot represents the density of points in the plot. An easier way to present many overlapping values is to make the values partially transparent (Figure 17.9B,D,F). This will allow the lines and points in the background of the figure to show through, so that where there are more data points, the colors will be darker. This imparts a histogram-like perspective onto a traditional 2-D plot, and it can turn a messy figure into a compelling one.

Make effective use of space

Screen presentations Whether you are laying out a slide that many viewers will see from twenty meters away, or designing a figure that is part of a paper with a firm

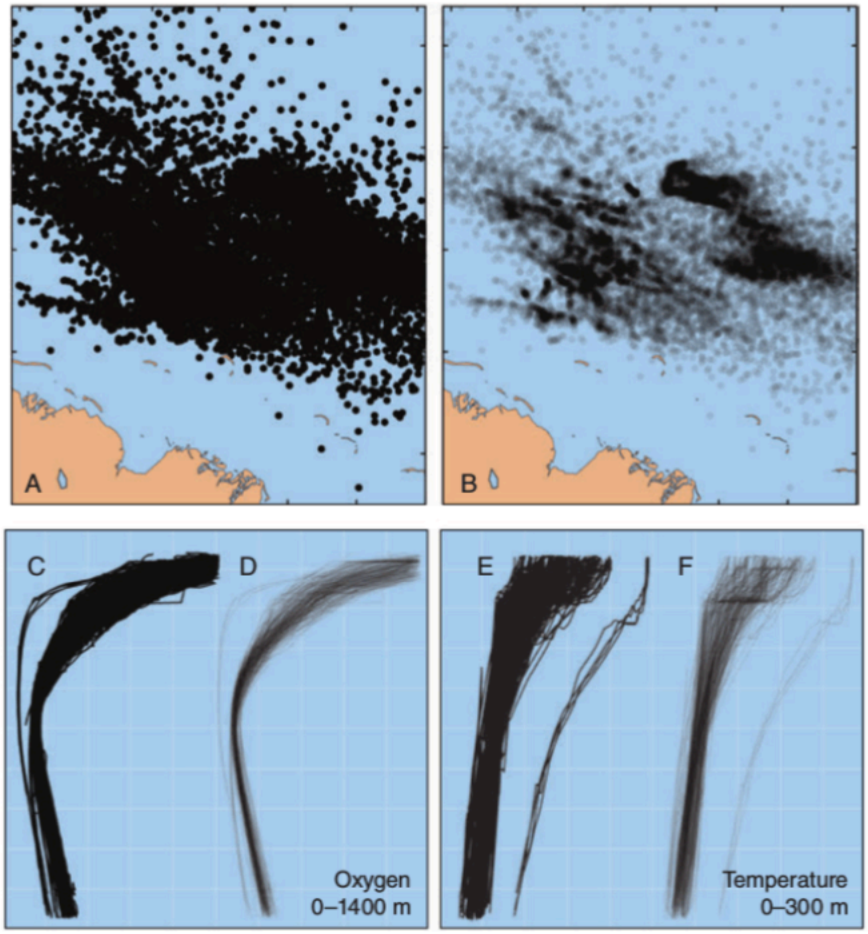

FIGURE 17.9 The added information available when using transparent points and lines in a plot The same data series are shown in each pair of plots (A and B; C and D; E and F), but structure which had been obscured in the opaque versions can be seen in the transparent versions. For example, in B, showing many thousands of geolocated whale calls, the area in the upper right has a higher density of call locations—a distinction that is not visible in A. In F, showing temperature profiles obtained during submarine dives, the depth where the vehicle sits and the thermometer equilibrates is visible as a horizontal band of dense coverage near the top of the profiles.

page limit, making effective use of space is critical. A common mistake is to bunch up elements in a figure, leaving areas of empty space that, if used, would have made the image much clearer without increasing its overall footprint. In presentations, scientists sometimes make graphical elements too small, then apologize by saying, "I know you can't read this, but…" When you are putting together your presentation, a good rule of thumb is to back up from the computer—at least three meters away—and see if you can quickly read and recognize text and labels on your slides. The projection screen for many audience members will be about the size of your fist held at arm's length, so view your screen from the same perspective.

Borders cut into the available image area, and should therefore be avoided. Three-dimensional shapes are sometimes used to present two-dimensional data, but this takes up added space without adding information. Edges and drop shadows can not only add visual noise, but may distort the conclusions of the data.

As discussed earlier, pictures used in slides of a presentation do not need to be much larger than 1024 × 768 pixels (thus, less than 1 megapixel in file size), although even camera phones can take images larger than 5 megapixels. The presentation will take more memory on disk and may stutter during projection if you don't use smaller copies of your photos. Some presentation programs can even support vector images, allowing you to project text and graphics crisply at any resolution when you embed PDFs directly into your slides.

Print figures Before starting work on what will be a printed figure, it is good to know the physical dimensions of the figure as it will appear in print. Nearly all journals provide figure size guidelines in their online instructions. When you first create a figure, pick the color space, set up a bounding box of the desired size, and create your figure to fit within that box. You can put the bounding box on its own locked layer (see Chapter 18) so that it doesn't interfere with your workflow. It's no use to work on details of your masterpiece filling a full sheet of paper, only to have it shrunken so those features become nearly invisible in the journal.

When operating within this constrained space, for most purposes the line size should be no smaller than 0.5 points. Hairlines generated by some programs are 0.1 pt, which is beyond the limit of the 0.35 pt thickness that most printers can render accurately. A line with a thickness of 1 point is 1/72nd of an inch wide, so 0.25 pt is 1/288th of an inch—roughly equivalent to two tiny pixels on a 600 DPI printer—and near the limits of visibility. Thin lines like these might look okay on your screen, but that is because the program is using a full screen pixel to represent something that is actually much smaller. View at about 300% to 400% magnification to get a more accurate idea of how things will look when printed. At this magnification, one pixel on the screen will correspond to one pixel printed at 300 DPI.

Given current practices for accessing scientific publications, the resolution at which the journal will print your article is almost irrelevant. Most users will either see the image on their screen, or print a PDF copy to their not-as-fancy printer, so aim for maximum visibility under those conditions. This is also a reason to try to get the figures embedded into your article as vector art rather than a bitmapped conversion: then they will look good at any magnification.

Consistency

You will often have little control over the original layout of artwork, which makes it difficult to have consistent colors and aesthetics across the figures in a paper or presentation. Artwork may be from a collaborator or generated by analysis tools that have limited display options. It may be from a program that has lots of options, but all of them are bad. By default, the lines may be too thick or too thin, the text poorly proportioned or in a different typeface, the colors inconsistent, and some graphical elements superfluous. To give your figures a consistent appearance and fix sub-par starting material, you will need to edit almost every image.

If you can get the artwork in vector format, you are in good shape. You can run it through a vector art editor (discussed in the next chapter) and adjust the appearance of each element as much as you like, so as to improve clarity and tweak the style (provided that this doesn't alter the scientific interpretation of the image, of course). If the program you are using doesn't provide an option for exporting vector art directly to a file, you can almost always generate a vector art file by printing the artwork to a PDF. If you can't obtain the original image in a vector format, it is often worthwhile to take the time to trace it, or add vector elements to the basic data plot. (Tracing will be explained in the next chapter.)

Consistency also refers to the relationship between graphical elements and the data they represent. It is confusing to the reader when in one figure, red is treatment and blue is control, but in a later figure, green is treatment and red is control. Because image editors give you complete control over the appearance of your plots, you are not constrained by the default settings used by any given plotting program. You can add information that may not be easy to display in the original program—for example, white circles to represent daylight data, black for night, and grey for crepuscular data. You can take different types of plots generated by several programs, all of which may have very different appearances, and edit them to create an integrated and consistent feel. This will help your reader focus on differences in data and results, rather than on puzzling stylistic elements.

Maintaining data integrity

Any tools that make it possible to prepare and edit scientific images also provide the opportunity to fundamentally alter the interpretation of the data. Editing graphs to eliminate extraneous information, such as irrelevant data series, or to alter the aesthetics of the graph, such as changing the color of data points, isn't usually a problem, but deleting or moving points within a series certainly would be. Likewise, cropping pictures usually isn't a problem, but erasing or adding photographic elements within them clearly would be.

Intentionally misleading your audience is, of course, outright fraud, but even without malicious intent, certain image modifications can unintentionally alter the interpretation of the data. Many journals now provide lists of best practices or else strict rules for how images (and pixel art in particular) may be modified. Be sure to pay close attention to these guidelines and rules. When it comes to pixel art, transformations that uniformly impact the entire image, such as changes in brightness or contrast, are usually not problematic. However, modifications of parts of an image, such as the use of the clone tool in Photoshop to clean up some elements of a picture, are often expressly prohibited—even for seemingly innocent tasks like removing specks and dust particles.

As opposed to nonuniform changes, fewer issues arise with uniform transformations; however, even these can sometimes be problematic. The most common issues arise when the saturation of an image such as a fluorescent micrograph is increased to a misleading point: either the noise is brought up to look like part of the signal, or low-level fluorescence is suppressed, making the signal appear artificially prominent. Keep track of the modifications you make, and if you think there is any chance that a modification might alter a reader's interpretation of an image, be sure to explicitly state the modifications in the legend.

Why you should avoid PowerPoint

You may have noticed that PowerPoint has not been mentioned anywhere, either as an editor or an image format, even though it is frequently used by scientists to edit and share images. This is not an accidental omission on our part. The primary reason you should avoid using PowerPoint for tasks other than presentations is that color modes, image compression and resolution, and other key factors are not immediately obvious when images are embedded in PowerPoint presentations, and file types and quality can be automatically converted in nonintuitive ways. If you send a PPT or Word file to a colleague, you won't know how it will appear on their screen or printer, and they will be largely unable to open the images it contains or integrate them into any other useful program. It is best to use editors and file formats that don't obscure these critical image attributes or add layers of unnecessary complexity.

If you need to create a multi-page document containing several figures, do not embed them into a Word or PowerPoint document because the figures will become uneditable. Instead, you can open one of the figures in Preview on OS X, or Adobe Acrobat on other systems. In Preview, if you show the sidebar, which contains thumbnails of the document's pages, you can drag other images into that container, rearrange their order, and save them together as a single PDF file.

SUMMARY

You have learned:

- The difference between vector and pixel art
- Why to use pixel art for photographic images, and prefer vector art for most everything else
- The relationships in pixel art between pixel dimension, physical dimension, and resolution
- Why to use RGB color for Web content, photos, and presentations
- What layers are, and what they can be used for
- General considerations for graphically presenting scientific data

Moving forward

- When you see distinctive graphical elements in journal articles, Web sites, or news stories, try to figure out what it is that visually works and doesn't work with them. Use these inspirations for your own visual communication tasks.

- Open some of your images in an image editor and see how changing the color space affects them.
- In your computer's display settings, for whatever operating system you are using, take note of the color profile you are using; then try changing it to see what effect the change has on how images look.
- View the gallery of data visualization approaches at `http://vis.stanford.edu/protovis/ex/`.
- Be inspired by the stunning graphs and uses of transparency at `tinyurl.com/pcfb-flights` and at `tinyurl.com/pcfb-eigen`.

Recommended reading

References are given in order from the most highly recommended, general texts at the top to the more specialized books at the bottom.

Tufte, Edward Rolf. *The Visual Display of Quantitative Information*. Cheshire, CT: Graphics Press, 2004.
This classic is required—not recommended—reading for any scientist.

Wainer, Howard. *Graphic Discovery: A Trout in the Milk and Other Visual Adventures*. Princeton, NJ: Princeton University Press, 2005.
A historical perspective on the progression of approaches to data graphics.

Nelson, Roger B. *Proofs without Words: Exercises in Visual Thinking*. Washington, DC: The Mathematical Association of America, 1997.
Elegantly mute graphical proofs of mathematical principles. Inspires novel ways to present relationships between data.

Cleveland, William S. *Visualizing Data*. Summit, NJ: Hobart Press, 1993.

Cleveland, William S. *The Elements of Graphing Data*. Summit, NJ: Hobart Press, 1994.
These two books present data in several different ways to highlight how the clarity of the message is dependent on its presentation. Available at `hobart.com`.

Steele, J., and N. Iliinsky, eds. *Beautiful Visualization*. Sebastopol, CA: O'Reilly Media, 2010.
A collection of chapters written by scientists explaining the methods and motivations for some of their complex graphics generated to display data. In some cases, the examples include the programs used.