

IMAGE-PROCESSING TECHNIQUES FOR THE CREATION OF PRESENTATION-QUALITY ASTRONOMICAL IMAGES

TRAVIS A. RECTOR

Department of Physics and Astronomy, University of Alaska Anchorage, AK, USA; rector@uaa.alaska.edu

ZOLTAN G. LEVAY AND LISA M. FRATTARE

Space Telescope Science Institute, Baltimore, MD, USA

JAYANNE ENGLISH

Department of Physics and Astronomy, University of Manitoba, Winnipeg, MB, Canada

AND

KIRK PU'UOHAU-PUMMILL

Gemini Observatory, Hilo, HI, USA

Received 2004 December 6; accepted 2006 September 29

ABSTRACT

The quality of modern astronomical data and the agility of current image-processing software enable the visualization of data in a way that exceeds the traditional definition of an astronomical image. Two developments in particular have led to a fundamental change in how astronomical images can be assembled. First, the availability of high-quality multiwavelength and narrowband data allow for images that do not correspond to the wavelength sensitivity of the human eye, thereby introducing ambiguity in the usage and interpretation of color. Second, many image-processing software packages now use a layering metaphor that allows for any number of astronomical data sets to be combined into a color image. With this technique, images with as many as eight data sets have been produced. Each data set is intensity-scaled and colorized independently, creating an immense parameter space that can be used to assemble the image. Since such images are intended for data visualization, scaling and color schemes must be chosen that best illustrate the science. A practical guide is presented on how to use the layering metaphor to generate publication-ready astronomical images from as many data sets as desired. A methodology is also given on how to use intensity scaling, color, and composition to create contrasts in an image that highlight the scientific detail. Examples of image creation are discussed.

Key words: techniques: image processing

Online material: color figures

1. INTRODUCTION

The prevalence of multiwavelength data in astrophysical research makes color images an appealing means of data visualization. The reduced costs for color figures in online journals has also increased their appeal for publication. How such images are made has changed with technological advancement. For many decades astronomical color images were generated using large-format photographic plates, wherein images were made by combining gray-scale images taken through red, green, and blue optical filters, e.g., Malin (1992). Because these filters approximately match the sensitivity of the color-detecting cones in the human eye, these images were considered to be roughly accurate representations of the object's appearance, discounting issues associated with the sensitivity of the human eye.

In the early 1980s, charge-coupled device (CCD) detectors began to replace photographic plates as the instrument of choice for optical astronomical research, allowing for the first time images to be created in a purely digital form. Since that time, images have mainly been assembled by assigning data sets to the red, green, and blue channels of a color image, even if the data sets did not correspond to these filters. The Hester et al. (1995) “Pillars of Creation” image of the central region of M16 with the *Hubble Space Telescope* (HST) is a famous example of such an image that uses only narrowband optical filters.

At nonoptical wavelengths, color has historically been used in a different manner. In the radio, color has been traditionally used to

encode information, such as monochromatic intensity, polarization, or velocity. In the X-ray, color images are often assembled by assigning red, green, and blue colors to photons in arbitrarily defined soft, medium, and hard X-ray bands.

Today many astronomical images are made from data sets that either are outside the optical window or do not match the characteristics of the human eye. Indeed, the role of these images has changed from being a “photograph” to being a representation of the science. They are not intended to show the “true” appearance of the object, at least not as defined by human vision, but rather to illustrate the physical properties of the object.

The development of advanced astronomical instrumentation has been contemporaneous with the advancement of computing power and, in particular, digital image-processing (IP) software for commercial applications. These IP programs, e.g., Adobe® Photoshop®,¹ Photoshop® Elements, and The GIMP,² offer unprecedented power and flexibility for digital image generation and manipulation. The combination of these technological advancements has led to a new ability in making color astronomical images, and in many ways it has led to a new methodology for creating them.

¹ Adobe and Photoshop are either registered trademarks or trademarks of Adobe Systems, Inc., in the United States and/or other countries.

² The GIMP was written by P. Mattis and S. Kimball and released under the GNU General Public License.

In particular, these programs use a “layering metaphor” to create images. This allows any number of images to be combined, or “layered,” to form the final image, and each layer can be modified independently. The advantage of the layering metaphor for astronomical images is that any number of data sets can be imported as separate layers. Each layer is then independently colorized and scaled. This allows for each layer to be scaled in a way that maximizes the contrast in just that data set, not in the whole image. Furthermore, each layer can be assigned any color, not just red, green, or blue channel colors. Each data set can therefore have a unique color, allowing the structure in each data set to be better differentiated.

For example, many multiwavelength “hybrid” images are assembled by assigning entire wave bands to the channels. As a result, spectral structure within each wave band is not resolved, and physical structure detected in only one band appears monochromatic. For example, in an image wherein radio emission is red, optical is green, and X-ray is blue, all of the stars will appear unnaturally green. If data are available in more than one optical filter, the layering metaphor can be used to assemble an image with more than one optical layer that shows the colors of the stars. The radio and X-ray data sets can also be incorporated as multiple layers to show spectral detail within these data sets as well. For data sets that contain both spectral and spatial information, e.g., spectral-line radio interferometry, most X-ray instruments, and integrated-field optical spectrographs, specific colors can be assigned to differentiate between continuum and line emission.

This flexibility opens an immense parameter space for image assembly. However, with this freedom comes ambiguity, as it is not necessarily clear what colors and scalings should be used to assemble an image, particularly for nonoptical or narrowband data sets. Since the primary goal of these images is to illustrate scientific detail, layer scalings and color assignments must be chosen carefully to maximize the contrast in the image, maximizing not just contrast in intensity but also in color. Such contrasts are naturally used by the human eye to interpret an image; thus, their proper usage will more readily convey the science.

The purpose of this paper is to demonstrate the methodology behind assembling an image with the layering metaphor. The procedural steps, from projecting each data set into a layer to the final electronic and print output, are discussed in § 2. This section serves primarily as an instructional guide. It will be of most interest to those unfamiliar with the layering metaphor. The steps are described in general terms, without reference to any particular software package. As software is constantly improving, tutorials for current software packages that use the layering metaphor can be found online.³ The methodology for using the layering metaphor to maximize color and intensity contrasts is discussed in § 3. This section is of greater value to those who have prior experience in assembling images, although in many ways the steps described in § 2 will be more clear after reading § 3. The Appendix presents an explanation on how to create photometrically calibrated images.

2. PROCEDURE

Generating a color image can be distilled into the following steps: (1) intensity scaling and projection of each data set into gray-scale images; (2) importing these images as layers into an IP software package; (3) intensity rescaling each layer to increase contrast; (4) assigning a color to each layer; (5) fine-tuning the

image, which includes overall color balance, the removal of artifacts, and orientation and framing; and (6) preparing the image for electronic distribution and print production. To ensure that your image will appear as desired on other computers and in print, it is helpful if your monitor is color calibrated and a color management workflow is established on your computer prior to image generation. These steps are briefly described in § 2.6.

The techniques described herein assume that more than one data set, preferably three or more, will be combined to produce a color image. A data set is defined as a two-dimensional image of a particular wave band, polarization, or other distinct characteristic; e.g., an optical image through a single filter, a radio image at or in a particular wave band and/or polarization, or an X-ray image over a specified energy band. These techniques are designed to take advantage of the distinct structural information in each data set for which they were obtained.

The data must first be fully reduced, e.g., optical data are bias and flat-field corrected with a standard data reduction package. The data must also be stored in a data file, e.g., a FITS format, that can be projected into an image. The image will then be imported into an IP program that can handle layers.⁴ The steps in this section are discussed in general terms, without reference to a particular software program, where possible. Note that the power of these programs is far beyond what can be explained here. The reader is therefore expected to have a basic familiarity with the data reduction and IP programs to be used.

While presented as a linear step-by-step process, it is important to emphasize the iterative nature of image creation. For example, while assigning colors to a data set it can be noted that a different intensity scaling may improve the image, thereby prompting a return to the original data set. Similarly, the flexibility of some IP software allows the exploration of many different realizations of an image simultaneously. With the variety of intensity scalings and colorings possible for each data set, the parameter space for combining all of the data sets into a single image is immense. It is recommended that the user explore many different realizations of the image before settling on a final composite image.

2.1. Scaling and Projecting Data

One of the most critical steps in the image generation is also the first step after the data are reduced. Each data set must first be projected into a gray-scale image. In this paper the terminology of “data set” is used when referring to the original, reduced data, and “image” is used when referring to a data set that has been projected into an image. A data set is projected into an image by using a scale function⁵ that converts the data values in each pixel into one of a finite number of integer pixel values. In a gray-scale image, these pixel values are shown as shades of gray. In an index-color (color-mapped) image, these pixel values are shown as predetermined color values. Only 2^8 (256) pixel values are available in an 8 bit image; whereas 16 bit images have 2^{16} (65,536) pixel values.

The dynamic range within a data set usually exceeds what can be shown directly in an image. For example, most optical CCDs have 15 bit or 16 bit converters, which can create data sets that have dynamic ranges of $>10,000$, and radio interferometric

³ See <http://www.stsci.edu/stsci/resources/imaging/papers/rector2007/>.

⁴ The ESA/ESO/NASA Photoshop FITS Liberator (Christensen et al. 2004) is a free plug-in that can scale and project FITS data files directly into Photoshop, ver. 7.0 and higher, and into Photoshop Elements, ver. 2.0 and higher. It provides a graphical interface for many of the steps described in § 2.1.

⁵ The scale function is also often referred to as the “transfer” or “stretch” function.

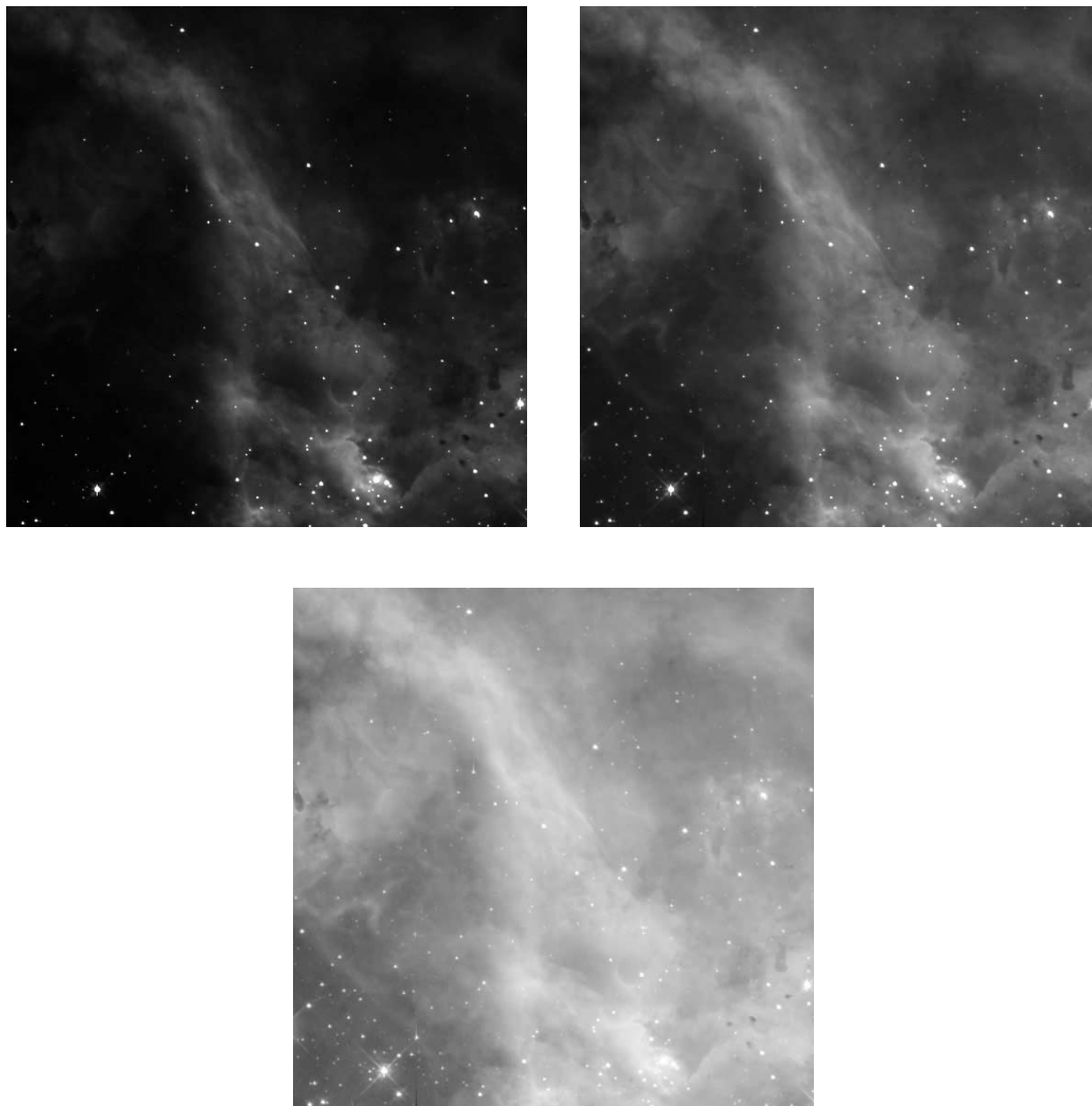


FIG. 1.—Three examples of images projected from the same data set of 30 Dor, observed with the *HST* WFPC2 and F656N filter. Each image is projected with the same maximum and minimum scaling values of 0:750 but with different scaling functions: linear (*top left*), square-root (*top right*), and logarithmic (*bottom*). The square-root scaling provides the best usage of the 256 shades of gray available. The linear image is too dark, and the logarithmic image has too low of a contrast.

observations can result in data sets with dynamic ranges of $>100,000$ (e.g., Perley 1999). An 8 bit gray-scale image is therefore incapable of displaying the structure and detail in a data set over the entire dynamic range without compromise. Sixteen-bit gray-scale images are therefore, in principle, preferable because they can retain the fine detail within a data set over a much larger dynamic range.

Unless an image is of the index-color type, color images are typically stored in a 24 bit image format that consists of 8 bits each for the red, green, and blue channels.⁶ Many IP packages are now capable of storing 16 bits channel^{-1} in a color image, generating in effect a 48 bit color image. Unfortunately, this does not nec-

essarily result in a more detailed image because computer monitors are currently capable of displaying only 8 bits channel^{-1} .⁷ Thus, while 16 bit gray-scale images better retain the detail within a data set, ultimately they cannot display it any better than an 8 bit gray-scale image. The advantage of the format is that it better retains the fine structure contained within the original data sets when it is rescaled within the IP software. Note that each rescaling of the image, e.g., with a levels or curve tool in the IP software, will result in a loss of pixel values. Within an 8 bit image, the loss of even a few of the 256 pixel values can have a significant impact on the quality of an image. Every time a 16 bit image is rescaled, there is also a loss of some of the 65,536 pixel values. But the

⁶ A 32 bit image contains an additional 8 bit “alpha” channel that describes the opacity of each pixel. This format should not be confused with 32 bit data formats.

⁷ The 16 bit data values are automatically converted into 8 bit values by the IP software for display.

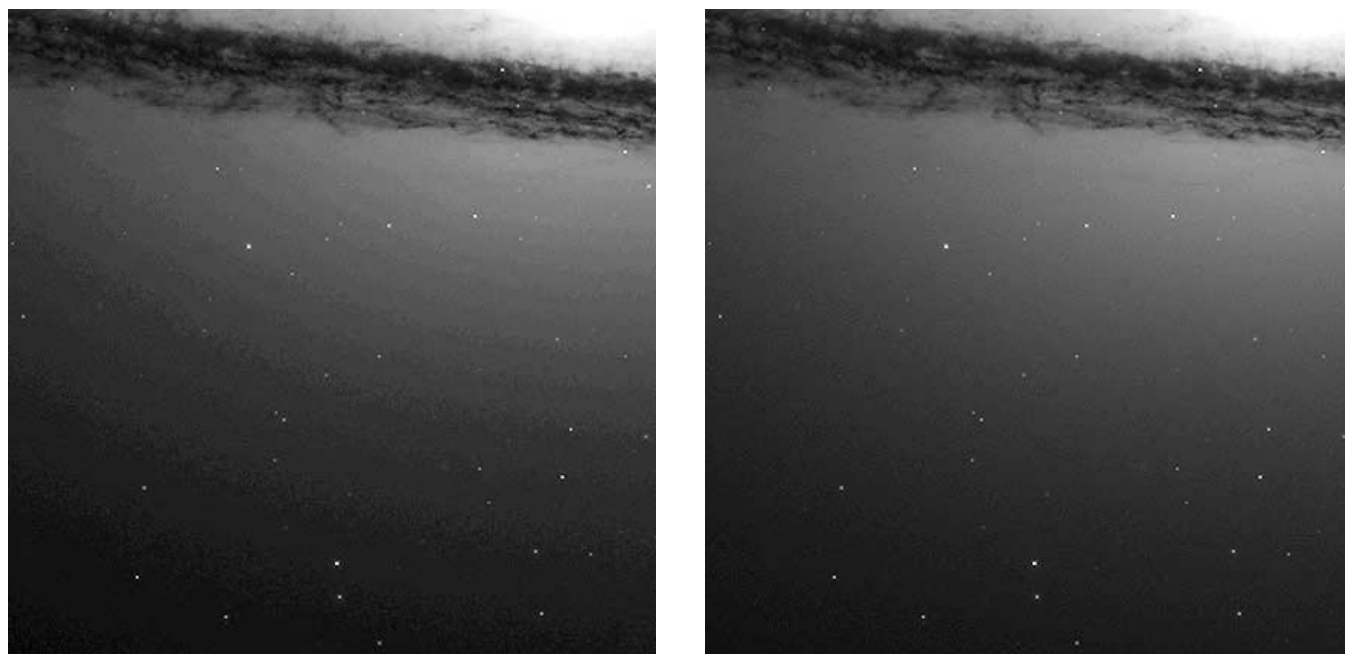


FIG. 2.—Portion of M104 observed with the *HST* ACS. Both images were projected from the same data set with a square-root function and identical minimum and maximum scale values, and then identically rescaled with a high-contrast curve adjustment in Photoshop. The image on the left was projected into an 8 bit gray-scale image and suffers from posterization due to the high-contrast rescaling. The image on the right was projected into a 16 bit gray-scale image and does not suffer from posterization. This is an extreme example that can be avoided if the data set is instead projected with a logarithmic function.

effect of this loss on the image itself is minimal because the remaining 16 bit pixel values are redistributed among the 256 values available for image display, usually resulting in no loss of image quality. Thus, high-contrast scalings are not likely to introduce posterization (see below) into the final image, allowing more flexibility in choosing the initial scaling for projecting the data into an image.

Ultimately, the 16 bit channel⁻¹ image format currently offers only a modest advantage over 8 bit, as 24 bit color image formats, the dominant standard, allocate only 8 bits each to the three color channels. While this may sound like a limitation, 24 bit color images still can generate over 16 million distinct colors, which is more than current monitors can display, and it is far more than can be printed. Thus, to some degree, scaling each data set into an 8 bit gray-scale image is an undesirable but unavoidable step. Fortunately, with the careful selection of a scaling scheme, the structure and detail of the original data set can be retained with a properly selected scaling function.

A scale function will manipulate the data values before projecting them into 16 or 8 bit gray-scale values. In the case of a linear scale function, the data are divided into bins of equal size. The bin sizes are equal to the difference between the maximum and minimum scale values, divided by the number of gray-scale values available. Data values within the same bin will be shown as the same shade of gray. For nonlinear scale functions, a mathematical function is first applied to the data before they are divided into equally sized bins. For example, a logarithmic scale function first typically applies a base 10 logarithm to the data before dividing the data into bins. It is usually advantageous, and sometimes necessary, to apply an additive and/or multiplicative adjustment to the data before applying the scale function. For example, if a data set contains negative pixel values, a constant value must be added prior to applying a logarithmic scale function to avoid undefined pixels. For most scaling functions a multiplicative adjustment to the data will affect the distribution of the gray-scale values; e.g., in the case of a logarithmic function, a multiplicative adjustment

greater than unity will shift up low data values more than high ones, increasing the contrast in faint portions of the image and decreasing it in bright regions. Additive and multiplicative adjustments have no effect on a linear scale function.

A scale function needs to be chosen that will optimize the contrast and detail throughout the image. In general, a linear scale function works well when most of the structure of interest in a data set has a modest dynamic range and structural detail is very subtle. The most common nonlinear scaling transformation functions are square-root and logarithmic scales, e.g., as shown in Figure 1. A logarithmic scale often works well when there is a large dynamic range in the structure of interest but most of the structure is faint. Other nonlinear functions, such as log-log and those given in Lupton et al. (2003), should also be considered. Some programs allow user-defined scaling functions. Some scaling functions also use bins of unequal size. In particular, a histogram scale function resizes the bins so that each bin has a nearly equal number of pixels.

For any scale function, proper minimum and maximum values for the scale must be specified. All data values below the minimum scale value are “undersaturated” and will be displayed as a pure black, and all data values above the maximum scale value are considered “oversaturated” and will be displayed as a pure white. Each intermediate data value will be displayed as one of 254 shades of gray. When choosing these values consider the following compromise: if the difference between minimum and maximum scale values is large, the scaling will be coarse and subtle differences in intensity will be lost; e.g., in a linear scale from 0 to 32,768, each shade of gray will represent a range of 128 in data values (since $32,768/256 = 128$). Thus, if the values of 2 pixels both fall into the same bin of 128, they will be shown as the same shade of gray and therefore will be indistinguishable. In extreme cases, when large sections of a data set are assigned the same shade of gray, the resulting image can look posterized (Fig. 2). If the difference between the minimum and maximum scale values is small, minute differences in data values

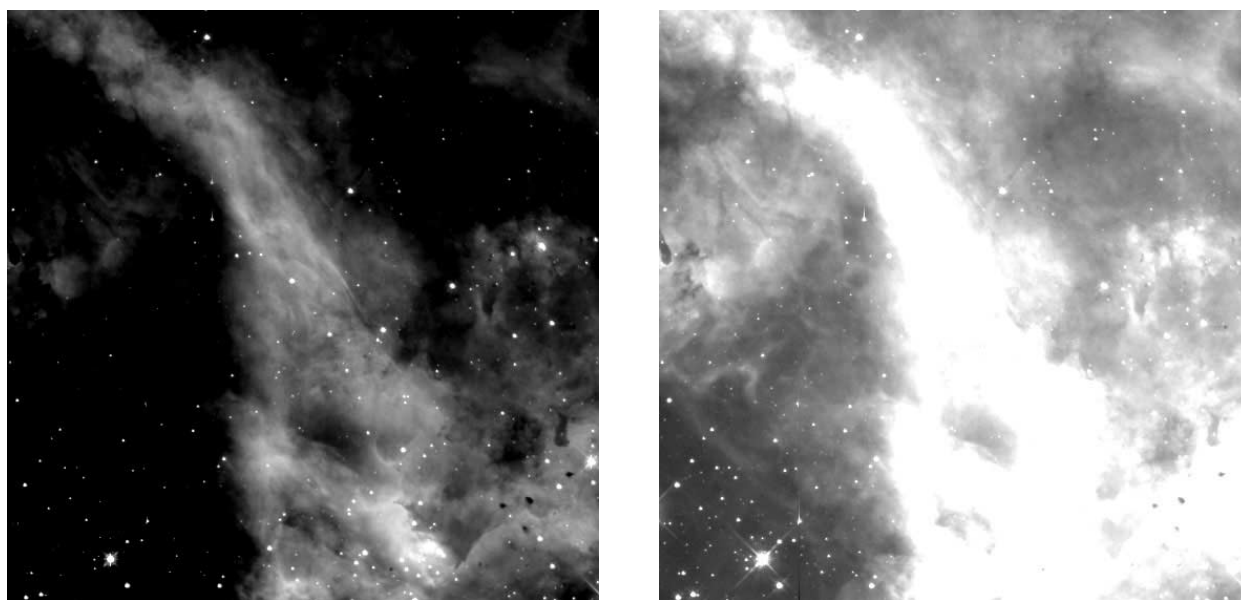


FIG. 3.—Two images projected from the data set in Fig. 1 but with different scaling functions. The image on the left contains large undersaturated (*black*) regions, and the image on the right contains large oversaturated (*white*) regions. In both cases detail in the saturated regions is lost. To recover the detail in these regions the minimum and maximum intensity scale values must be changed.

will be better resolved. But all data values outside of the scaling range will be undersaturated or oversaturated, and detail in these regions will be lost (Fig. 3). If a scaling undersaturates or oversaturates a particular region of the image, that region will be shown as pure black or pure white, respectively, and detail within the saturated regions cannot be recovered once the data set is projected.

In general, a good scaling will maximize the contrast without saturating the detail of interest. A logical starting point is to choose a scaling with a minimum data value at or slightly above the noise level of the image. Scaling the image so that some noise is visible is valuable for ascertaining the sensitivity of the data. It gives a visual cue as to what structure is significantly above the noise levels. The maximum scaling value should be set to just above the highest data point value in the source structure. In the case of optical images it is generally not a concern if the cores of stars exceed the maximum scaling value because brighter stars usually are scaled to appear white in the final images, and the unsaturated halos around these stars will indicate the color of the star. A histogram of data values may prove useful in identifying an appropriate scaling range. However, the histogram will contain stars and bad pixels that in general should be ignored when choosing a scale. Figure 4 shows examples of the same data set with different scalings to illustrate the difficulty in choosing a good scaling. It is best to search iteratively for good scaling values by displaying the image with different scaling functions.

Unless scaling each data set to a photometric scale, as described in the Appendix, it is important that a separate scaling system be determined for each data set, because the goal is to maximize the intensity contrast in each data set, and the value ranges for each data set are different. This may seem counterintuitive at first. However, using the same scaling system for each data set can lead to undesirable results. A chosen minimum and maximum scaling range may be appropriate for one data set but too wide for another, resulting in poorer resolution of the intensity scale; i.e., only a fraction of the 256 shades of gray will be available to show the detail in the data set. Worse, if the chosen scaling range is too

narrow for a particular data set, regions of the data set will be undersaturated or oversaturated. Also consider using a different type of scaling function for different *kinds* of data sets; e.g., a logarithmic scaling for broadband filters and a linear scaling for narrowband filters. In general, the same scaling function should be used for similar data sets, i.e., if the emission is primarily from the same sources. Otherwise, an artificial color gradient could be introduced into the image that is a result of the different scaling functions, not of differences in the data (R. Hurt & L. L. Christensen 2005, private communication).

In some cases the dynamic range of a data set is too large, and there is interesting structure in both the bright and faint regions of the data set that cannot be shown well with a single scaling solution. Several solutions are possible. The “unsharp mask” is an image manipulation technique that compresses dynamic range by dividing the original version of an image by a low-resolution, i.e., blurred, version. The end result is that large-scale structure is canceled and fine detail is retained, even over a large dynamic range. The technique has been used with great success in many astronomical images.⁸ Another option to compress dynamic range is to subtract a function from or divide one into the data set before projection; e.g., to generate an *HST* image of M51, the original data sets were divided by a galaxy disk model prior to projection (Scoville et al. 2001). In some cases it is useful to use more than one scaling of the data, with the intention of showing the detail in different regions of the image. The two scalings are then exported as separate gray-scale images, imported into the IP software, and combined. One approach is to blend the two images as layers with decreased opacity; e.g., in the *HST* image of M51 described above, two scalings of each data set were blended. One scaling was a logarithmic stretch, and the other scaling was linear but divided by the galaxy disk model. When

⁸ See the 1979 Anglo-Australian Observatory (AAO) image AAT 19 by D. Malin, *The Orion Nebula, M42 and M43*. The definition of an unsharp mask varies; e.g., the Photoshop unsharp mask algorithm works differently from the traditional darkroom technique.

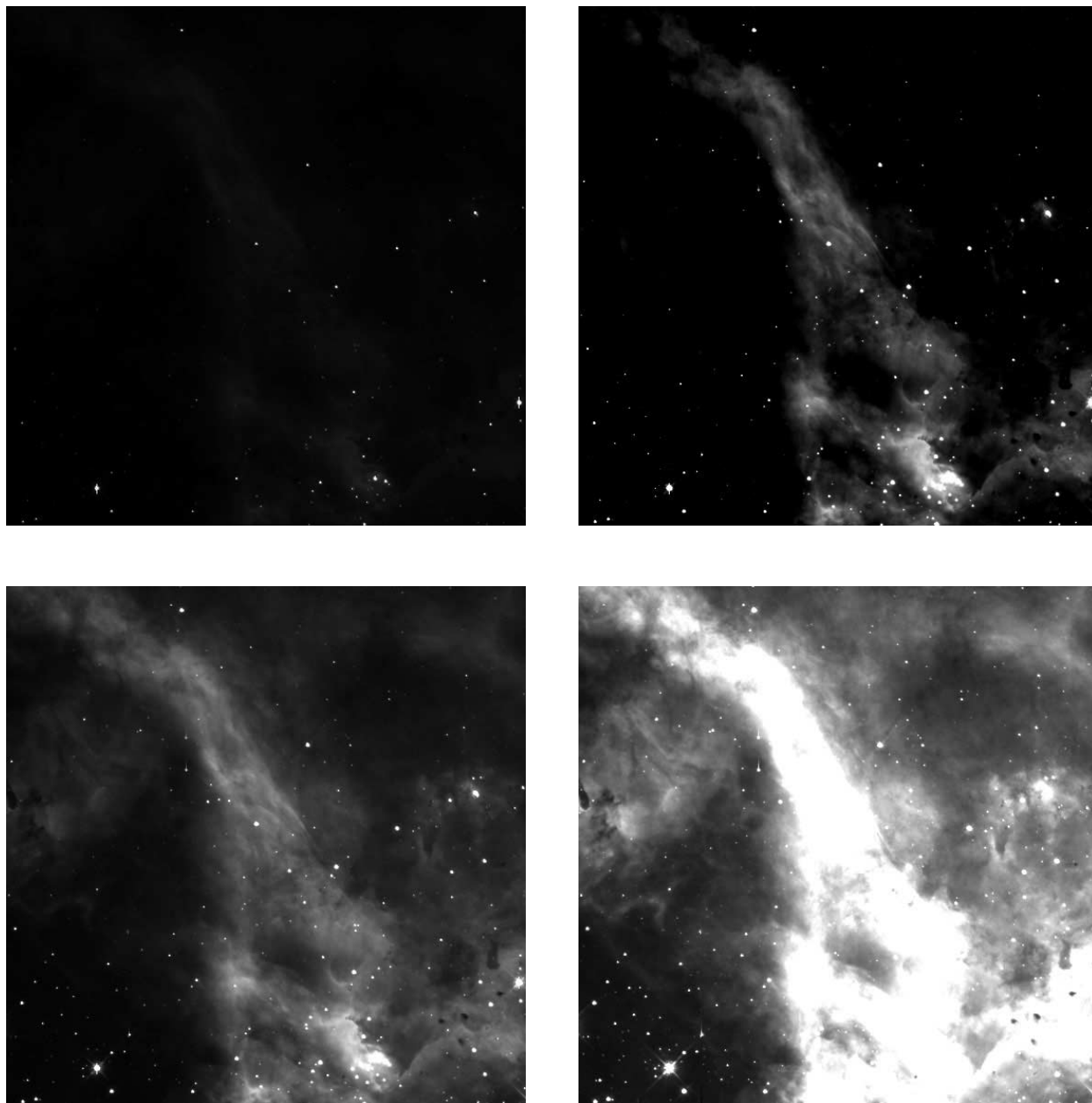


FIG. 4.— Four examples of images projected from the same data set of 30 Dor, observed with the *HST* WFPC2 and F656N filter. Each projection is done with a linear scaling but with different minimum and maximum scale values. The top left image shows a scaling of 0:4096 that encloses nearly all of the data values but as a result does not show the faint structure. The top right image shows a scaling of 100:500 that better shows the midtones but undersaturates the shadows and oversaturates a bright portion of the nebula in the lower right. The bottom left image shows a scaling of 0:500 that does not saturate shadows. The bottom right image shows a scaling of 0:150 that best shows the shadows but oversaturates the highlights. None of these scalings is sufficient to show the detail in the data set, suggesting that an alternative to the linear scaling is necessary.

blended, the overall dynamic range was compressed and the contrast in the arms was enhanced.

2.2. Importing Data into Layers in an Image-Processing Program

Once each data set has been projected into a gray-scale image, the next step is to import each image into an IP program that is capable of combining multiple “layers” of images into a single, composite image. To create a color image, each gray-scale image is imported into the IP program as a layer. The multiple images are layered onto each other. Each layer is then blended with the layers below via an algorithm. Each layer is then assigned a color. Each layer will probably also need to be rescaled within the IP program

to enhance the detail within the layer. The combination of the layers forms the resulting image.

It is important to note that layers are different from “channels,” which are the orthogonal colors that define the color space; e.g., in “RGB color” the channels are the primary colors red, green, and blue. In RGB color all other colors are reproduced by combining these three colors in varying amounts. Primary colors and the RGB color space are discussed in § 3.1.

A traditional method for creating a color image is to combine three data sets by assigning each data set to a channel, such that each data set is, in effect, assigned to one of the primary colors. The layering metaphor allows three major improvements over this method. First, an image can be created with any number of

data sets; it is not limited to the number of channels. Second, any color can be assigned to a layer, not just the channel colors, i.e., red, green, and blue in RGB color. Third, it is possible to temporarily hide a layer such that it is not visible. This is useful when modifying one layer at a time, as described in the following steps. Layers are discussed in more detail in § 2.3.

The first step is to create a master image file in the IP program that will contain all of the gray-scale images, each as a separate layer. Each gray-scale image should be opened and copied into the master image. The master file should be saved in an RGB color (24 bit) file format that retains the layers. The combination algorithm for each layer should be set to “screen” and 100% opacity. The screen algorithm acts to increase the pixel values of a particular layer, effectively brightening the image. Assuming an 8 bit channel depth, it combines the selected, “foreground” layer with the “background” layer underneath it with the following equation (Bunks 2000):

$$r = 255 - \frac{1}{255}(255 - r_f)(255 - r_b),$$

where r_f is the foreground layer red-channel pixel value, r_b is the background red-channel pixel value, and r is the resulting red-channel pixel value. This equation is applied iteratively through all the layers present. It is also used for the green- and blue-channel pixel values. From the equation it should be apparent that the resulting channel pixel value will be higher unless either the foreground or background pixel value is zero, corresponding to a pure black. The screen algorithm is used instead of simply adding the channel values for each layer to prevent the resulting value from exceeding the maximum value of 255. Note that if an image layer contains nonzero values in only one channel, combining it with other layers will only affect the values in that channel. Using the screen algorithm to combine three “R, G, and B” layers, each of which contains nonzero values in only the R, G, or B channel, respectively, is functionally identical to copying these layers into the channel values directly.

In the layering metaphor, it is possible to modify each layer without affecting the other layers. In addition, the appearance of a layer can be changed without modifying the layer itself. Some IP packages use “adjustment layers.” These are layers that do not contain image data but rather modify the appearance of the layer(s) below via an algorithm. The primary benefit of an adjustment layer is that it does not actually change the image layers below it; thus, the effects of an adjustment layer can be seen and easily undone by deleting or hiding the adjustment layer. A good way to visualize how an adjustment layer works is the example of a “colorize” adjustment layer. It is analogous to laying a color gel over an image in that both the gel and the adjustment layer will cause the image below to take on the monochromatic color of the filter. Deleting or hiding the adjustment layer is analogous to removing the filter gel. By using adjustment layers it is simple to modify or undo adjustments and try new ones, thereby making it much easier to try many different color schemes and scalings. Adjustment layers therefore allow this parameter space to be explored more efficiently. Adjustment layers are especially useful for the two important adjustments made to the layers in an image, colorizing and intensity scaling. Another advantage of adjustment layers is the ability to develop the ideal color and intensity scaling scheme with a scaled down “thumbnail” file (§ 2.4) and then simply copy the adjustment layers from the thumbnail file and apply them to the full-resolution image. This is useful when working with large files.

If adjustment layers are unavailable or not desired, an alternative approach is to duplicate a layer before modifying it. If the

result is unacceptable, delete the modified layer and restart. At each step duplicate the layer to be modified, and then after the modification blink between the original and modified layers to see the effect.

2.3. Alignment and Rescaling

Once all of the images are copied as layers into the master file, the layers need to be aligned to a common astrometric frame. It is also usually necessary to fine-tune the intensity scaling of each layer to better show detail.

2.3.1. Alignment of Layers

In general the data sets should be registered to a common astrometric frame in a data analysis package prior to exporting the image. Manipulations such as fractional pixel shifts, rotation, pixel rescaling, and higher order geometric distortions are better done in a data analysis package. However, simple registration can be done in most IP programs. If all of the data were obtained from the same telescope and instrument during the same epoch, astrometric registration is usually as simple as a linear shifting of the layers. Large-format instruments may have significant geometric distortion across the field of view. The varying thickness of filters can also introduce differential geometric distortions. If geometric distortions are not removed, a process commonly done during data reduction, they can result in problems with layer registration, as well as with mosaicking of images with significant offsets.

The layers can be aligned by selecting each layer and shifting it to a common reference layer. Layers can be more effectively aligned by temporarily inverting the gray scale of a layer and reducing its opacity, making its offset relative to other layers more apparent. Once the layer is properly aligned, invert it again to return it to normal and restore the layer’s opacity to 100%.

It is possible to combine data from different telescopes and/or instruments such that one data set partially replaces data from another, e.g., setting a high-resolution *HST* WFPC2 image into a wide-field ground-based optical image, by using a mask. After the layers are aligned, the region in the larger image that is overlapped by the smaller image can simply be masked out by copying the smaller image into the larger image’s pixel layer mask. If the superposed data are of a different type, it may be desirable to include a visible border to differentiate.

2.3.2. Rescaling Each Layer

As discussed in § 2.1 it is important to properly scale each data set before exporting it as a gray-scale image. However, additional fine-tuning of the intensity scaling for each layer is usually desirable, and IP programs usually have a wide range of image scaling tools that exceed what is possible with data reduction packages. These scaling tools remap the 256 shades of gray in each layer into the 256 shades shown on the computer. Thus, it is possible to remap a layer so that there is better contrast in both dark and bright regions of the image. When rescaling a layer, all of the other layers should be hidden so that the effect on that layer can be seen more clearly.

The levels and curve tools are particularly useful. These can be applied directly to a layer, or preferably with an adjustment layer. When creating an adjustment layer it is important to group the adjustment layer so that it only modifies the data layer below it. Otherwise, the adjustment layer will modify all of the layers below it.

The levels tool allows the minimum and maximum scale values to be set in the same way as done before in § 2.1. If the scale values were chosen appropriately when projecting the data set, little

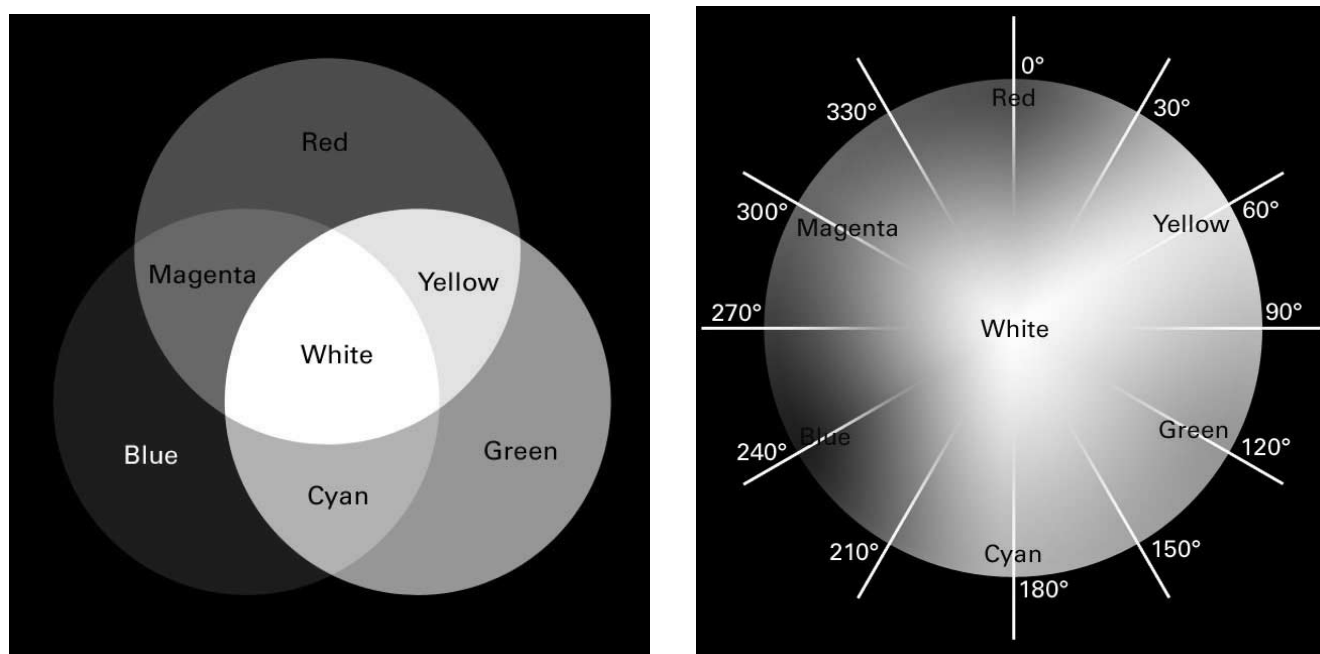


FIG. 5.—Color wheel model based on Newton's studies. *Left:* Primary colors in two systems of color mixing. The additive primaries of red, green, and blue combine to produce white at the center of the image. The secondary colors in this system, cyan, magenta, and yellow, are primaries in the subtractive system that mix to produce dark neutral gray. *Right:* Angular positions of hue given on the perimeter of the color wheel. Colors directly opposite each other on the color wheel are complementary colors. For example, yellow (60°) is the complement of blue (240°) because they are 180° apart. [See the electronic edition of the *Journal* for a color version of this figure.]

modification of these levels will be necessary within the IP software. However, if the scale values do need to be changed significantly, the original data sets should be projected again with better values. This is especially necessary if the image is being assembled in 8 bit channel⁻¹ mode.

The levels tool also allows a “gamma correction” to be applied to the layer, which is a simple, nonlinear way to remap the pixel values to the displayed intensity values on the monitor. It is a power-law function, i.e., $\text{intensity} = \text{pixel}^\gamma$. A gamma correction consists of applying the inverse of this relationship to the image before displaying: i.e., $\text{new pixel} = \text{old pixel}^{(1/\gamma)}$. A gamma correction greater than 1 will brighten the fainter regions of the image but will tend to saturate the brighter regions. Conversely, a gamma correction of less than 1 will show better detail in the bright regions but will darken the faint regions. Most monitors apply a gamma correction with a gamma value between 1.7 and 2.7 to brighten the displayed image.

The curve tool enables a more complex mapping of the pixel values in the file to the values displayed. By adjusting the slope of the line within the curve tool, the contrast can be changed for different levels of intensity. The local slope of the line determines the contrast in that region. Thus, for example, the contrast could be increased in the faint portions of the image and decreased in the bright portions.

Note that each layer can be rescaled as many times as necessary. An advantage of using adjustment layers is that multiple scalings of each layer can be tried before one is chosen. After colorizing each layer, as described in § 2.4, adjust the intensity scaling of each layer again to achieve a better color balance. As mentioned before, achieving the best intensity scaling and colorization for each layer is an interactive, iterative process.

2.4. Color and the Image

Once each layer is appropriately aligned and rescaled, the image is ready to be colorized. Color is parameterized in most

IP packages in the “HSV” color space, wherein a specific color is defined by its hue, saturation, and value. “Hue” describes the actual color, e.g., yellow or yellow-green. “Saturation,” also often referred to as intensity, is a measure of a color’s purity. A purely desaturated color is mixed with an equal amount of its “complementary” color. Drawing a line across the color wheel from a color, through the center of the wheel, to the color on the direct opposite side of the wheel defines a color’s complement. For example, blue is the complement of yellow in the RGB system. A purely saturated color contains none of its complement. A purely desaturated pixel contains no color; i.e., it is a gray scale. “Value” is a measure of the lightness of a color, i.e., the amount of white or black added. Other terms used to modify color are “tint,” “tone,” and “shade.” A “tint” is a color with added white; e.g., pink is a tint of red. A “tone” is a color with added gray, and a “shade” is a color with added black. The HSV color space has its origins in Newton’s color circle or wheel.⁹ The simple color wheel used in most IP packages is shown in Figure 5.

In the HSV system the three parameters of hue, saturation, and value define a three-dimensional color cylinder wherein all colors are contained. Hue is measured as an angular position on the wheel. In the RGB color system, 0° is red, 120° is green, and 240° is blue. The radial axis is saturation, wherein fully (100%) saturated colors lie on the outer edge of the color wheel and fully desaturated (0%) neutral gray sits at the center. The simple color wheel bisects the color cylinder, and value is the axis orthogonal to the color wheel.

Hues are created by combining primary colors that, by definition, are the fundamental colors that cannot be created by any combination of other colors. This simple color wheel model uses two different systems for creating color, each system consisting of three primary colors. The most familiar system is the additive color system in which the primaries, labeled red, green, and blue

⁹ For a history of color wheels see Birren (1969).

(RGB), add to produce white. This system is used when combining light and is the standard for computer displays and projection devices. The RGB primaries are based on the spectral sensitivities of the three classes of photoreceptors in the retina, which peak at wavelengths of about 5800, 5400, and 4400 Å, respectively (e.g., MacDonald 1999). Another system based on this color wheel is the subtractive system, in which the primary colors are cyan, magenta, and yellow (CMY). They add to produce neutral gray. Note that these primaries are the complement colors of the RGB system; e.g., cyan is the complement to red.

A colorization can be applied directly to a layer or by creating a hue/saturation adjustment layer that is grouped to the layer to be colorized. This tool will allow the adjustment of the hue, saturation, and lightness of the layer. To colorize the image, select the colorize option. Once the colorize option is selected, the hue determines the angular position on the color wheel. As each layer is adding color and intensity to an image, it is easy to saturate areas of the overall image (which will be shown as pure white) if each layer is too bright. Thus, it is necessary that each layer be darkened when choosing a color assignment, particularly if more than three layers are to be combined. To generate the color image, apply a different colorization to each gray-scale image layer. If using adjustment layers be certain to group each adjustment to its associated image layer. Strategies on choosing colors for each image layer are discussed in § 3.1.

2.5. Fine-Tuning the Image

2.5.1. Cosmetic Cleaning of the Image

Ideally, distracting instrumental effects, e.g., systematic noise, cosmic rays, CCD charge bleeds from bright stars, and seams from mosaics, will be removed as much as possible during the data reduction process. If artifacts and defects in the data are not properly removed in earlier data-processing steps, many can be removed with tools in the IP software. It is important to note that these tools should only be used to fix cosmetic defects in the image due to instrumental effects. Care must be used not to introduce artifacts or change intrinsic structure from the image. For example, sharpening filters should not be used, as they modify the source structure.

There are several filters available that are effective at reducing noise and removing cosmic rays. Gaussian blur and median filters are useful for reducing noise, but with some reduction in the resolution of the image. Filters designed to remove dust and scratches from scanned images can also be effective at removing cosmic rays and reducing noise. Images generated from radio interferometric data often suffer from dimples on the scale of the longest baseline, which can be ameliorated with a Gaussian blur. If the problems are particular to an individual data set, each layer should be fixed individually.

It is recommended that cosmetic cleaning be done while working on a duplicate of the layer that is to be edited. The duplicated layer can then be blinked on and off to compare the cosmetically corrected image to the original. Another good practice while cleaning an image is to apply a temporary “levels” adjustment layer that brightens the image. Periodic inspection of the image in a brighter mode may more easily show unwanted artifacts introduced by the cleaning process.

Note that all such manipulations must be explicitly listed to avoid ambiguity. Furthermore, they should not be applied to only selected regions of the image, unless explicitly described. Many journals have guidelines as to what image modifications are acceptable (e.g., Nature 2006).

2.5.2. Overall Color and Intensity Balance

Once the cosmetic cleaning of each layer is complete, the image must be “flattened” into a single layer. It can now be saved in a standard file format suitable for press production, e.g., a PostScript file, or for electronic distribution, e.g., a tiff or jpeg file. The flattened image should be saved as a different file so that the original, unflattened image can be returned to if necessary. After flattening the image, it is a good idea to apply overall color balance, brightness, and contrast adjustments to the image, either directly or with adjustment layers applied above all grouped layers. Color can be added or removed to balance the image; e.g., if the image appears too red, add cyan, the complementary color to red, to balance the image. Also fine-tune the brightness and contrast of the image directly or with similar adjustment layers.

As discussed in § 3.1, the eye perceives quantities such as color and brightness relatively, and uses seven contrasts to perceive information from an image. One of these is the “light-dark” contrast, or simply a relative measure of intensity. To take advantage of the full range of the light-dark contrast, each image should contain regions of pure black and pure white.¹⁰ Regions of pure white are almost always present in optical images, from the cores of bright stars. However, some scaling may be necessary to ensure a pure black is present. Once regions of pure black and white are identified, some color balance is usually necessary to ensure that the black and white regions are a neutral color.

Most IP software contains a tool that measures the RGB value of any pixel. Analyze the chosen region of pure black and make sure that the pixel values of the RGB channels, not layers, are less than 10 each and that the value of each channel is on average roughly equal to the others. If one channel is systematically higher or lower than the other two, adjust the color balance to add or remove intensity in the shadows from that particular channel. For example, if the selected region of pure black consistently shows the *R* channel to be ~5 counts higher than the *B* and *V* channels, then balance the colors by subtracting red (which is equivalent to adding cyan) until the three channels are more in balance. The same process should be repeated for the regions of pure white, adjusting the color balance in the highlights until the channels are well balanced, each with a value greater than 245 or so.

2.6. Color Display and Reproduction of the Image

Image generation by the above steps will hopefully generate a color image that looks correct on your computer’s monitor. Unfortunately, a major difficulty with the electronic distribution and print production of digital images is that the appearance of an image can vary significantly on different output devices. For example, an image that looks properly balanced on one computer monitor may appear too dark on another, or the image may look too red when printed on a particular printer.

It is anticipated that the image will be used in one or more of the following three ways: The image will be distributed electronically to be viewed on another display device, e.g., via the Web or an electronic journal. Or the image will be printed directly to an in-house printer, e.g., on a personal or wide-bed printer. Or the image will be sent as a file to be printed on an out-of-house printer, e.g., in a paper journal or on a professional press. Each of these scenarios requires that the image be prepared so that it will look as expected. Fortunately, the effective use of “color management,” before and after image generation, can give reliable image reproduction on other output devices. Precise color management

¹⁰ In reality pure blacks and whites are difficult to print. These regions should therefore be close to, but not pure, black or white.

and calibration can be difficult; it has led to the development of an entire subindustry. For more information, Fraser et al. (2003) offer a thorough review of the subject.

3. STRATEGY OF IMAGE CREATION

3.1. *Definition of Color Schemes*

Astronomical images often invoke questions of veracity. Historically, terms such as “natural color” and “true color” have been used to describe optical images obtained with filters that roughly match the sensitivity of the cones within the human eye. Terms such as “false color” and “pseudocolor” are often used to describe images assembled with other methods, implying that such images are fabricated.¹¹ However, the goal of these images is data visualization, not a portrayal of reality as defined by human vision. Color and intensity scaling therefore serve a different role.

In this paper the term “representative color” is used when describing images assembled from multiple data sets of different wave bands in a manner different from that for natural color. This term is used because each data set encodes properties of a physical phenomenon, and colors are assigned to properly represent it. In practice, nearly all astronomical images generated from professional data fall into this category, because the data sets are seldom obtained through only broadband optical red, green, and blue filters. And these data sets are usually scaled and projected to maximize detail in the manner described in § 2.1. A representative-color scheme is said to be in “chromatic order” if the filters are assigned color based on the wavelength of their passbands. If the data sets are not assigned color in order of wavelength, the order is known as “composite order.” Examples of both are given in § 3.3.1.

Here the term “pseudocolor” is used to refer to a distinct technique wherein a monochromatic image is converted into a color image by mapping gray levels into colors according to a previously defined color lookup table (LUT). In a pseudocolor image, the color changes as a function of the value of a single physical property represented in the image, e.g., polarization, velocity, or monochromatic flux density, thereby creating a multi-colored image. Note that a pseudocolor image itself is not monochromatic unless all of the colors in the LUT are of the same hue; e.g., the popular “heat” pseudocolor scheme is not monochromatic because the LUT contains red, yellow, and colors of intermediate hue. Images generated from natural and representative color schemes are fundamentally different from pseudocolor images.

3.2. *Choosing a Color Scheme*

With the ability to assemble images from any number of data sets, each independently projected, scaled, and colorized, the parameter space available for image assembly is immense and largely unbound. However, one is not free to perform any image manipulation one chooses, as the intention is to retain scientific accuracy. Clearly, it is unacceptable to move, modify, or delete structure within the data. What other manipulations are then allowable, and even desirable? Color and intensity scaling schemes should be used that maximize the color and intensity contrast within the object to convey the most information about the source. In particular, “visual grammar,” defined as the collection of ele-

ments that affect the interpretation of an image, can be used as a guide for finding a composition and color scheme that highlights aspects of the astronomical object while retaining scientific accuracy.

Choosing an appropriate color scheme can be unclear. For example, when working with optical data, a simple approach to color assignment has been to assign to each data set the “visible” color of its filter. The visible color is essentially the passband of the filter as perceived by the human eye; i.e., what one would see if one looked through the filter against a bright, white light. For this approach to work well, several filters that cover most of the visible range of light must be used (Wainscoat & Kormendy 1997). However, the overall color of the image can become skewed if the colors of the filters are not balanced; e.g., an image that is created from V , R , I , and $H\alpha$ data sets would contain too much red because three of the four data sets would be assigned a reddish color, resulting in a loss of color contrast. Furthermore, images often include data from wavelengths outside the visible regime. Thus, there is a need to define a more generalized methodology for color selection.

Appropriate color assignments will depend on the data sets to be combined and the science to be illustrated. When choosing colors it is also important to consider the fact that our eyes, and indeed our senses in general, function by detecting relative differences; e.g., a line appears to be long only when a shorter line is present for comparison. Color in an image is similarly intensified or weakened by the contrast of the color’s properties with other colors in the image (Itten 1970). Thus, contrasts between colors can be used to highlight or deemphasize elements of the image. Black and white images have only one contrast scheme, that of light to dark, i.e., the intensity contrast. However, there are seven contrasts in color images (Itten 1970, 1990). These contrasts are given in Table 1. An example of an astronomical image that strongly shows a particular contrast is also given for each contrast, although an image usually consists of more than one contrast. Other examples are also given in § 3.2.

In addition to these contrasts, color can be used to imply physical characteristics, such as depth, temperature, and motion. For example, cool colors are often perceived to be behind warm colors (Fig. 6).¹² This is a result of our everyday experience, wherein distant objects, such as mountains, appear to be bluer because of scattered light from the foreground air mass; see Lynch & Livingston (2001) for examples. Thus, the warm-cool color contrast can be used to create an image that appears to have depth by selecting colors for data layers that, when combined, produce warm colors for the objects that should appear closer and cool colors for those that should appear farther away. In addition, as the names imply, cool colors are perceived to be physically colder than warm colors, as a result of our experience with reddish flames and bluish ice.¹³

In general a good starting strategy when assigning color is to space the color assignments evenly around the color wheel; e.g., if creating an image with three data sets, assign undiluted colors that are 120° apart on the color wheel. This is known as a triadic color scheme and is the simplest example of hue contrast. The traditional RGB color scheme falls into this category because the primary colors red, green, and blue are separated by 120° . This selection represents the extreme instance of the hue contrast in the same way that black-white represents the extreme of the light-dark contrast. Similarly, the secondary colors

¹¹ In reality no astronomical image accurately represents the appearance of an object, as the human eye’s sensitivity to color is very complex and nonlinear. Ultimately such arguments are rhetorical, as the purpose of a telescope is to show what the eye cannot see.

¹² Hues that contain blue are collectively known as cool colors, whereas hues that contain red are known as warm colors.

¹³ This is of course in contrast to the Planck spectrum.

TABLE 1
COLOR CONTRASTS

Contrast	Description	Example
Hue.....	The juxtaposition of saturated colors, including black and white. These hues can be randomly selected.	Antares and ρ Oph ^a
Light-dark.....	The juxtaposition of brightnesses: for example, a dark green beside a light blue.	IC 2944 (<i>Hubble</i> Heritage Team 2002)
Cold-warm	The juxtaposition of warm colors (those that contain yellow or red) with cool colors (those that contain blue). See § 3.2 for more on these colors.	Trifid Nebula (Malin 1992)
Complementary	The juxtaposition of colors directly opposite on the color wheel. See § 3.2 for more about split complements and compensating tones.	Carina Nebula (<i>Hubble</i> Heritage Team 2000)
Simultaneous	The juxtaposition of colors that are not complements in order to generate, e.g., physiological color mixing or the appearance of motion along common borders between colors.	Europa ^b
Saturation	A pure color surrounded by dull, less pure colors. The decrease in saturation to produce a dull color can be produced by adding white, black, gray, or complementary colors.	SN 1987A (Burrows et al. 1994)
Extension.....	Differently sized areas of color juxtaposed to achieve balance between strong and weak colors. Itten (1970) lists the appropriate balances for each color.	Tarantula Nebula (Brandl et al. 2004)

NOTES.—These seven color contrasts can be used in conjunction with the color wheel based on Newton’s experiments to choose color schemes. For each contrast an example image is given as a reference. Note that many images have more than one contrast.

- ^a See the 1979 AAO image UKS 4 by D. Malin, *Antares and the Rho Ophiuchi Dark Cloud*.
^b See the 1998 NASA image PIA01641 from the JPL/NASA *Galileo* Project, *Double Ridges, Dark Spots, and Smooth Icy Plains on Europa*.

cyan, yellow, and magenta are also separated by 120° and can be used. However, one is free to use any colors that are separated by 120°. Similarly, if there are four data sets, they should be spaced at 90° intervals around the color wheel. This is known as a tetradic color scheme. If there are only two data sets one can assign colors that are separated by 180°, known as a complementary color scheme. Note that it is impossible to produce an image that fully samples the color space with only two data sets. However, this limitation can be overcome by generating a third data set that is the average of the existing two. This technique has been successfully used in several images, including an *HST* NOAO image of the Helix Nebula (Meixner et al. 2003).

Choosing colors that are evenly spaced on a color wheel improves the sampling of the color space; i.e., there will be a wider range of colors within the image. Also, for this reason it is important to assign fully saturated colors to each layer, because only fully saturated colors can combine to produce all of the colors available in the color wheel. Note that white and black can be included as colors in the contrast of hue scheme, so any

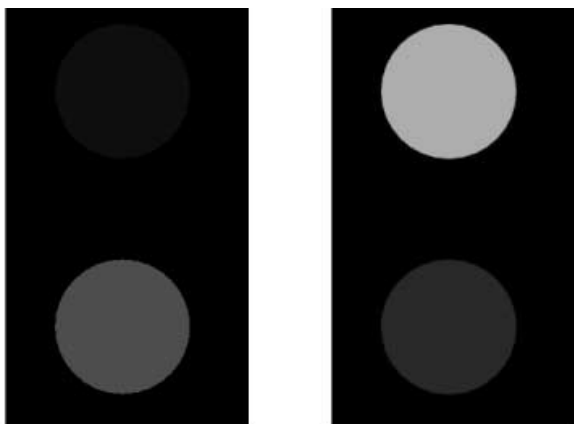


FIG. 6.—Warm and cool colors. When viewing the panel on the left, cover the panel on the right with your hand. Although the two circles in the panel on the left are of identical size, the red circle (*light gray*) appears to be closer than the blue one (*dark gray*). This demonstrates the differences between warm and cool colors. Using blue to represent motion toward the viewer can be achieved by warming up the blue by adding yellow and using a contrast of saturation, i.e., using a dull red. An example is given in the panel on the right. [See the electronic edition of the *Journal* for a color version of this figure.]

pure whites and blacks that appear in the final image will strengthen this contrast.

Therefore, based on the number of available data sets, one should attempt to distribute the hues assigned to the images in a manner that maximizes the usage of the color space. If the overall color of the image is unacceptable after the initial assignments, the color assignments can be swapped between layers. The hue values for each layer can also be changed to see the effects on the overall image. Once approaching a scheme that is close to ideal, slightly adjust the hue for each layer. Even changes of only a few degrees in hue can have a significant impact on the appearance of the image.

Note that a given contrast can be supplemented using other contrasts. For example, if an image contains the colors of orange-red and cyan, it has the complementary contrast, as well as a warm-cool contrast wherein the orange will appear in the foreground. Since dark colors in a light background also jump forward, one can use the light-dark contrast to further create depth.

It is important to emphasize that this approach does not advocate hard and fast rules of color assignment. Such fixed rules are impossible considering the diversity of data sets that may be used to assemble an image. Instead, several different color schemes should be tried as a starting point. Then, note which contrasts can result after the initial color assignment, and then adjust the color scheme, either by choosing different colors or by swapping the color assignments among layers, to achieve these contrasts. Section 3.3 discusses some of the different color schemes commonly used and indicates in more detail how colorizing a layer can produce a color contrast.

3.3. Color Schemes and Examples

3.3.1. Examples of Representative-Color Images

An example of an image produced using the chromatic ordering scheme is the Hester et al. (1995) *HST* image of M16. The data were obtained with the *HST* WFPC2 with the F502N ([O III] λ 5012), F656N ($H\alpha$ λ 6564), and F673N ([S II] λ 6732) narrow-band filters. To generate the image, the three data sets were colorized blue, green, and red, respectively, in a triadic color scheme. Filter F502N was assigned blue because it has the shortest wavelength passband, filter F673N was assigned red because it has the longest wavelength passband, and filter F656N was assigned green because its passband is of intermediate wavelength compared

to the other two. The chromatic ordering for the *HST* M16 image is not a natural color scheme because the filters were not assigned their visible colors. That is, when viewed against a bright, white light, the F502N filter appears to be green, not blue, to the human eye. Similarly, the F656N filter looks a deep red, not green. Only the F673N filter is assigned its perceived color, red. It is also not a natural color scheme because it uses narrowband filters that pass only a very small fraction of the visible spectrum. Furthermore, rather than photometrically calibrating the projected images they are balanced so that the $H\alpha$ data do not dominate and turn the image completely green. In comparison, a natural color image of M16¹⁴ shows the nebula as deep red, again because of the strong $H\alpha$ emission from the nebula. This image contains only the light-dark contrast because it is monochromatic. Detail in the image is therefore lost.

The color assignment chosen in the above example is an extreme version of the hue contrast. The image is also an example of a split complementary color scheme. The blue (from the [O III] filter) and the green (from the $H\alpha$ filter) combine to generate a background of greenish-cyan, whose complement is a slightly red orange. Areas of the pillars are yellow-orange and orange-red, which are on either side of this orange on the color wheel; hence, the complement is split. Also, the RGB assignment to each layer, along with the intensity stretches, ensures that the combined data sets produce cyan, magenta, and yellow, resulting in a strong contrast of hue in the undiluted primaries of the CMY subtractive system (see § 2.4). Where the intensity of [O III] (blue) and $H\alpha$ (green) are balanced, cyan appears in the background. The $H\alpha$ (green) and [S II] (red) images combine to produce yellow along the edge of the pillars. Also, while the centers of the stars are white due to the equal combination of RGB, their halos are magenta because the F673N filter broadens the point-spread function. The two supporting contrasts, (split) complementary and light-dark, further enhance the detail in this image.

Another factor to consider when choosing a color scheme is the structure within each data set. If each data set is scaled as described in § 2.1, the intensity of the structure of interest in each layer should be about the same. However, sometimes this is not possible, as some structures will appear in some data sets but not all. For example, an image of IC 405¹⁵ was generated from data obtained in three filters: [O III], $H\alpha$, and [S II]. No nebular signal was detected in the [O III] filter, effectively reducing the image to two data sets, as far as structure in the nebula was concerned. Thus, a nearly complementary color scheme (60° yellow for $H\alpha$ and 270° violet for [S II]) was chosen to show the structure in the nebula. The [O III] data set was assigned a blue (240°) color to enhance the bluish color of the hot O-class star, AE Aurigae, at the center of the image. Note that this image contains two contrasts, the complementary color contrast of yellow and violet and the extension contrast, wherein violet occupies roughly 3 times the area of the image as yellow (see Table 1).

3.3.2. Examples of Images from More Than Three Data Sets

While not employing a natural color scheme, the *HST* image of M16 is nonetheless generated with three data sets that are assigned to the triadic RGB channel colors. However, as discussed in § 2.2, with the layering metaphor it is possible to use any number of data sets, and to assign to each one any color desired.

For example, an image of NGC 6822 for the Local Group Survey¹⁶ was generated from data that were obtained in eight filters: *UBVRI*, [O III], $H\alpha$, and [S II]. Since each filter provides additional information about the object, the resulting image contains more richness of detail, e.g., in this case a larger diversity of stellar colors.

Note that it can also be the case that an image can be enhanced by the lack of complete wavelength coverage. Indeed, the reason for narrowband filters is to increase the contrast between the emission-line and continuum regions in an object. Thus, one needs to consider whether or not additional data sets will improve the detail visible in the image.

It is important to use a different color for each data set. Otherwise, distinct information from each of the data sets is lost. Thus, the triadic RGB color scheme cannot be used in an image created with more than three data sets. In the above example, colors were chosen for each data set to approximately match the visible color of the filter.¹⁷ Furthermore, each layer was scaled to best show the detail in the data set, not to a photometric scale; e.g., the $H\alpha$ layer was scaled to emphasize the H II regions within the galaxy. Thus, this image does not have a natural color scheme.

An image of ESO 172-2, a bipolar reflection nebula, is an example that uses the layering metaphor with four data sets to encode polarization information as color in a manner different from that for pseudocolor. The image (*Hubble* Heritage Team 2005) was generated from data sets from the *HST* ACS WFC with the F606W filter. Three of the data sets were also obtained with polarization filters of different orientation; the fourth data set was obtained without a polarization filter.

3.3.3. Examples of Nonoptical and Multiwavelength "Hybrid" Color Images

The layering metaphor can also be used to assemble images outside of the optical. An example at radio wavelengths using the layering metaphor is an image of the supernova remnant Cassiopeia A.¹⁸ Note that this is not a pseudocolor image because the color scales as a function of intensity at three frequency bands, 1.4, 5.0, and 8.4 GHz. Color assignments were chosen to generate a warm-cool color contrast between the edges and center of the remnant. The redder center of the remnant therefore appears to be closer, implying the spherical nature of the remnant. Note that an important issue to consider when creating a nonoptical image, particularly at radio and X-ray wavelengths, is that these images often lack stars, which can make registration of the images nontrivial.

Hybrid images combine data from widely separated wave bands, e.g., radio and X-ray, that have been obtained with different telescopes and instruments. An example is that of a color image of the Cygnus region produced from radio and far-infrared data (English & Taylor 2001). In the image, 74 and 21 cm data from the International/Canadian Galactic Plane Survey¹⁹ were assigned "rose" (a tint of red) and green colors, respectively; they were then combined with 60 and 25 μ m data from *IRAS*, which were assigned turquoise and blue, respectively. As well as containing

¹⁴ See the B. Shoening 1973 image of the Eagle Nebula in the NOAO Image Gallery.

¹⁵ See the T. A. Rector & B. A. Wolpa 2002 image of AE Aurigae in the NOAO Image Gallery.

¹⁶ See the P. Massey et al. 2001 image of Barnard's galaxy (NGC 6822) in the NOAO Image Gallery.

¹⁷ Technically, this was not completely possible, as the passbands for the *U* and *I* filters are outside of the visible range of the human eye.

¹⁸ See the L. Rudnick et al. 2002 image of Cassiopeia A in the NRAO Image Gallery.

¹⁹ The CGPS is supported by the National Research Council of Canada and the Natural Sciences and Engineering Research Council of Canada.

a light-dark color contrast, this color scheme resulted in a warm-cool contrast causing the yellowish and reddish supernova remnants to appear in the foreground.

Another example is a radio and optical hybrid image of the Triangulum galaxy, M33, that was produced with five data sets.²⁰ Data through four optical filters (*B*, *V*, *I*, and $H\alpha$) were combined with 21 cm $H\text{ I}$ spectral line radio observations obtained with the NRAO²¹ Green Bank Telescope, NRAO Very Large Array, and ASTRON Westerbork Radio Synthesis Telescope.²² The optical data sets were assigned approximately visible colors, except for the *I*-band image. That was assigned an orange color to differentiate it from the $H\alpha$ layer that was assigned a pure red. A violet color was assigned to the 21 cm data set because it is a mixture of red and blue. The red in the violet implies a relationship to the $H\text{ II}$ regions that are prominent in the $H\alpha$ layer. In appearance, it causes the $H\text{ I}$ and $H\text{ II}$ regions to blend rather than appear as overlays. The blue in the violet implies that the gas in the $H\text{ I}$ regions is cooler than in the $H\text{ II}$ regions. It also creates a cold-warm color contrast. A similar example is a hybrid image of B33 generated from optical data and a ^{12}CO map (Pound et al. 2003). Again, a blue-violet color was chosen to imply the colder nature of the ^{12}CO emission. These images are therefore composite color schemes because the radio data are assigned a bluish color even though these data sets are of the longest wavelength.

4. SUMMARY

The quality of modern astronomical data, and the technologies now available to manipulate them, allow for the creation of images in a way that exceeds the traditional definition of an astronomical image. These images are important tools for scientific illustration and visualization.

A practical guide has been presented to generate astronomical images from research data by using image-processing programs that use a layering metaphor. This metaphor allows any number of astronomical data sets to be combined, and each data set can be assigned any color. With this technique, images with as many as eight data sets have been produced. This is an improvement over traditional methods, wherein only three data sets can be combined, and only with the primary colors of the color space. In the layering metaphor, each data set can be individually scaled and colorized, creating an immense parameter space to be explored. These IP programs offer excellent flexibility in the creation of images, allowing easy exploration of this parameter space.

A strategy and examples have also been presented on how to use color and composition to create images that highlight the scientific detail within an image by using the color and intensity contrasts that the human eye uses to differentiate detail within an image. This strategy is particularly relevant for generating images with data sets outside of the optical window and for data sets with limited-wavelength coverage, e.g., emission-line optical filters. With visual grammar one can imply qualities that a two-dimensional image intrinsically cannot show, such as depth, motion, and energy. The effective use of these techniques can result in an image that effectively conveys the science within one's data.

²⁰ See the T. A. Rector et al. 2002 image of the Triangulum galaxy (M33) in the NRAO Image Gallery.

²¹ The National Radio Astronomy Observatory is a facility of the National Science Foundation operated by Associated Universities, Inc., under a cooperative agreement.

²² The Westerbork Synthesis Radio Telescope is operated by the ASTRON (Netherlands Foundation for Research in Astronomy) with support from the Netherlands Foundation for Scientific Research.

APPENDIX

PHOTOMETRIC CALIBRATION

There are cases in which it is desirable to retain the relative flux densities between the filters that are being used to create a color image, e.g., to show in a planetary nebula the relative amounts of $H\alpha$ emission in red and $[\text{O III}]$ emission in green. Alternatively, one may be creating a mosaic from a number of chips. The photometric calibration of each chip will ensure a more accurate consistency in intensity across the image field. This is especially important at the low-intensity levels if the intention is to show faint structures.

When photometric calibration data are available, one can convert the counts per pixel in each filter into $\text{ergs s}^{-1} \text{cm}^{-2}$ values for each pixel. This is done at the data-processing stage using the arithmetic tasks in the data reduction software. This conversion will assign a flux density to each pixel for the bandwidth of that filter. This is consistent with each filter subsequently being assigned to a monochromatic color, i.e., to one wavelength or frequency.

It is beyond the scope of this paper to describe the photometric calibration process because it is unique for every telescope; e.g., the standard technique for optical ground-based photometry is well described in Landolt (1992). However, the calibration of the *HST* WFPC2 will be used as an example of how to convert from instrumental units, e.g., $\text{counts s}^{-1} \text{pixel}^{-1}$, to physical units of monochromatic flux density, e.g., $\text{ergs s}^{-1} \text{cm}^{-2} \text{\AA}^{-1} \text{pixel}^{-1}$. The conversion is done with the following equation, which is based on one given in the WFPC2 handbook (Koekemoer et al. 2003):

$$\text{flux density} = \frac{\text{DN} \times \text{PHOTFLAM} \times 10^{18}}{\text{EXPTIME}}.$$

In this equation, DN is the instrument's number of counts in each pixel, EXPTIME is the exposure time in seconds, and PHOTFLAM is the calibration constant, in units of $\text{ergs DN}^{-1} \text{cm}^{-2} \text{\AA}^{-1}$. Each of the four WFPC2 chips and filter combination has a different PHOTFLAM value. Normally the exposure time and PHOTFLAM parameters in the header of the FITS file for each chip are used. Since the PHOTFLAM parameter is of order 10^{-18} , for ease of data manipulation and viewing, the equation above has been modified such that the units are $\sim 10^{-18} \text{ ergs s}^{-1} \text{cm}^{-2} \text{\AA}^{-1}$. Since only relative photometry is of concern here, the absolute values here need not be retained. The calibration step is applied before a single data set is projected from the WFPC2 mosaic so that the calibration is constant across the data set. Naturally, each filter and each exposure time must also be calibrated separately. After calibration, each data set will consist of pixel data values that are independent of exposure time, wavelength, and the performance characteristics of the telescope and instrument.

Once the data sets are properly flux-density-calibrated, they are ready to be projected into images as described in § 2.1. To retain the relative value of the flux density between filters, first ensure that the same minimum and maximum data values are set in each data set. Then, when projecting each data set into an image, use the identical intensity scaling function for each data set, i.e., use the same minimum and maximum scale values, as well as identical gamma and scaling functions. Once imported into an IP program, any rescalings of the image must also be applied equally to all of the layers if relative photometric accuracy is to be retained.

Images generated in the above manner are photometrically calibrated in the sense that each layer is relatively calibrated to the others. However, that still does not address the issue of the complex response that the human eye has to light and color. No image-generation system is capable of conveying quantitatively the relative brightness of objects. Any image will convey that one star is brighter than another, regardless of the intensity scaling, but

to convey that a star is, e.g., 2 mag brighter is not possible because the eye is very poor at quantifying the intensity of light. Similarly, it is poor at quantifying color beyond simple relative measures, e.g., “bluer” and “much bluer.” Thus, in general photometric calibration of an image may not be worth the effort in most cases for the simple result of an aesthetically pleasing image, and there are significant disadvantages incurred, as discussed in § 2.1.

REFERENCES

- Birren, F. 1969, *Principles of Color: A Review of Past Traditions and Modern Theories of Color Harmony* (New York: Van Nostrand Reinhold)
- Brandl, B. R., Carey, S., Giovanni, G. F., Houck, J., Megeath, S. T., Stauffer, J., & Weedman, D. 2004, *The Tarantula Nebula* (SSC press release 2004-01a; Pasadena: SSC)
- Bunks, C. 2000, *Grokking the GIMP* (Indianapolis: Pearson Education)
- Burrows, C., et al. 1994, *Hubble Finds Mysterious Ring Structure around Supernova 1987a* (STScI-1994-22; Baltimore: STScI), <http://hubblesite.org/news/1994/22/>
- Christensen, L. L., Nielsen, L. H., Nielsen, K. K., Johansen, T., Hurt, R. L., Levay, Z., Fosbury, B., & Hook, R. 2004, *FITS Format Liberation DIY Astronomical Images with the Expert Touch* (ESA press release heic0412; Noordwijk: ESA)
- English, J., & Taylor, A. R. 2001, *S&T*, 102, 52
- Fraser, B., Bunting, F., & Murphy, C. 2003, *Real-World Color Management* (Berkeley: Peachpit)
- Hester, J., et al. 1995, *Embryonic Stars Emerge from Interstellar “Eggs”* (STScI-1995-44; Baltimore: STScI), <http://hubblesite.org/news/1995/44/>
- Hubble* Heritage Team. 2000, *Light and Shadow in the Carina Nebula (NGC 3372)* (STScI-2000-06; Baltimore: STScI), <http://hubblesite.org/news/2000/06/>
- . 2002, *Thackeray’s Globules in IC 2944* (STScI-2002-01; Baltimore: STScI), <http://hubblesite.org/news/2002/01/>
- . 2005, *Hubble Catches Scattered Light from the Boomerang Nebula* (STScI-2005-25; Baltimore: STScI), <http://hubblesite.org/news/2005/25/>
- Itten, J. 1970, *The Elements of Color: A Treatise on the Color System of Johannes Itten, Based on His Book the Art of Color* (New York: Wiley)
- . 1990, *The Art of Color: The Subjective Experience and the Objective Rationale of Color* (New York: Wiley)
- Koekemoer, A., et al. 2003, *WFPC2 Instrument Handbook*, Ver. 8.0 (Baltimore: STScI)
- Landolt, A. 1992, *AJ*, 104, 340
- Lupton, R., Blanton, M. R., Fekete, G., Hogg, D. W., O’Mullane, W., Szalay, A., & Wherry, N. 2004, *PASP*, 116, 133
- Lynch, D. K., & Livingston, W. 2001, *Color and Light in Nature* (New York: Cambridge Univ. Press)
- MacDonald, L. W. 1999, *IEEE Comput. Graphics Appl.*, 19(4), 20
- Malin, D. 1992, *QJRAS*, 33, 321
- Meixner, M., et al. 2003, *Iridescent Glory of Nearby Planetary Nebula Showcased on Astronomy Day* (STScI-2003-11; Baltimore: STScI), <http://hubblesite.org/news/2003/11/>
- Nature*. 2006, 439, 891
- Perley, R. A. 1999, in *ASP Conf. Ser. 180, Synthesis Imaging in Radio Astronomy II*, ed. G. B. Taylor, C. L. Carilli & R. A. Perley (San Francisco: ASP), 296
- Pound, M. W., Reipurth, B., & Bally, J. 2003, *AJ*, 125, 2108
- Scoville, N., et al. 2001, *Hubble Reveals the Heart of the Whirlpool Galaxy (M51)* (STScI-2001-10; Baltimore: STScI), <http://hubblesite.org/news/2001/10/>
- Wainscoat, R. J., & Kormendy, J. 1997, *PASP*, 109, 279