Chapter Four

Color Management for Digital Imaging Systems

Edward J. Giorgianni
Thomas E. Madden
Kevin E. Spaulding
Eastman Kodak Company

Contents

4.1 Introduction
4.2 Color Management Paradigms
4.3 Digital Color Encoding
4.4 Color Encoding Methods
4.5 Image States
4.6 Standard Image-State Color Encoding Specifications
  4.6.1 Criteria for selection of RIMM/ROMM RGB color encoding specifications
  4.6.2 ROMM RGB color encoding specification
    4.6.2.1 ROMM RGB conversion matrix
    4.6.2.2 Nonlinear encoding of ROMM RGB
  4.6.3 RIMM RGB color encoding specification
    4.6.3.1 RIMM RGB conversion matrix
    4.6.3.2 Nonlinear encoding of RIMM RGB
  4.6.4 ERIMM RGB color encoding specification
    4.6.4.1 Nonlinear encoding for ERIMM RGB
4.7 Image States in a Color Managed Architecture
4.8 Digital Color Management with JPEG 2000
4.9 Summary
References

© 2003 by CRC Press LLC
4.1 Introduction

All successful color imaging systems employ some form of color management. Color management can be defined as a means for predicting, controlling, and adjusting color information throughout the system — from the initial color capture to the formation and display of output images. In chemical and other analog-based imaging systems, color management may be implemented in various ways, including equipment calibration, chemical process control, and operator-controlled or automated color-printing adjustments. In digital imaging systems, color management is generally implemented using software designed specifically for that purpose. The principal function of that software is to process (transform) image signals derived from an input device to make them appropriate for a given output device. Digital color management can be relatively simple when applied to imaging systems that are restricted to only certain types of inputs and outputs, but, when applied to systems having a variety of different types of input and output devices and media, color management can become quite complex.

The successful implementation of digital color management depends on a number of factors, including the use of appropriate device characterization methods and suitable mathematical techniques for forming and applying image-processing transformations. More fundamental to the success of the color management, however, are the selection of an appropriate color management paradigm and the use of a correspondingly appropriate method for color encoding, i.e., a method for representing color in digital form throughout the imaging process.

4.2 Color management paradigms

Underlying every color management approach is an implicitly or explicitly defined paradigm — an underlying conceptual model that ultimately determines how an imaging system using that color management will behave. The paradigm describes the expected relationships among the input images, encoded images, and output images of the system. Although various types of color imaging systems might behave quite differently, virtually all can be described in terms of just three fundamental types of color management paradigms. These paradigms will be referred to as Types A, B, and C.¹

Color imaging systems based on a Type A color management paradigm are “input driven.” Their color encoding represents the colors of the input images, and the colors produced by their outputs match (as much as possible) the input image colors. Color copiers, for example, operate according to a Type A paradigm; the normal expectation is that an output image produced by the copier will match the image input for copying. If an intermediary image (such as a video preview) is provided, it too would be expected to match the input and output images. This generally is the paradigm that first comes to people’s minds when they think about color management. In fact, because the paradigm specifies that colors will match throughout an imaging system, the paradigm might seem to be the only one
that is needed. However, in many ways, the basic concept of the Type A paradigm is quite limited, which is why many commercial systems instead are based on Type B or Type C paradigms.

Systems based on a Type B color management paradigm are “encoding driven.” Their color encoding is based on a unifying color encoding concept that tends to reduce or eliminate the colorimetric differences inherent in the system inputs. For example, some electronic prepress systems encode color in terms of the colorimetric characteristics of a reference reflection-print medium. Colors scanned from actual reflection prints are encoded essentially in terms of their measured colorimetry. But colors scanned from photographic transparency films are re-rendered, i.e., their measured colorimetric values are altered such that they correspond more closely to those that typically would be measured from the reference reflection-print medium. The current International Color Consortium (ICC) color management system is also based on a Type B paradigm in that all input images must be re-rendered to correspond to the properties of a reference imaging medium. As in a Type A paradigm system, the colors produced by the outputs of a Type B paradigm system are expected to visually match the colors represented by the color encoding. However, unlike a Type A paradigm system, the colors produced by a Type B paradigm system’s outputs do not necessarily match the input image colors.

Systems based on a Type C color management paradigm are “output driven.” Like Type B systems, they are based on a unifying color encoding concept. However, their output colors do not necessarily match the colors represented by this encoding, because additional re-rendering is performed, subsequent to encoding, as part of the output signal processing. This deliberate additional re-rendering might be done for simulation, i.e., to make one output produce images that imitate the appearance of images normally produced by another type of output. Re-rendering also might be done to enhance output images by taking advantage of the particular capabilities of each output device or medium. For example, when an output medium having a large color gamut is used, the output signal processing might include some expansion of the gamut of the encoded colors so as to use the full capabilities of that particular medium. This paradigm is often used in digital photofinishing systems where the objective is for each output to produce the best image possible from the encoded data. As a consequence of the output-specific re-renderings and color enhancements that might be performed, images produced on different types of output devices and media generally will not (by design) match each other.

These three paradigms are sufficient for describing the basic functionality of all existing types of color-managed imaging systems. Each paradigm is widely used, and each is technically valid. Yet each produces very different color results. The most appropriate paradigm for a given system will depend on the specific application for which that system will be used. It is also possible to design systems that function according to a Universal Paradigm, in which various input and output signal processing options are provided.
Through the selection of appropriate options, such systems can be made to operate according to any of the described paradigms.

4.3 Digital color encoding

In addition to the selection of a color management paradigm appropriate for a given application, the successful implementation of digital color management requires the use of an appropriate method for digitally encoding color. The basic function of the digital color encoding is to provide a digital representation of colors for image processing, storage, and interchange among systems. Within a given color imaging system, the encoding provides a digital link between the system’s inputs and outputs.

In a simple system, having just one type of input and one type of output, color encoding can be performed prior to any signal processing. The encoding is therefore a direct representation of the color values measured by the input device. In more complex systems supporting multiple types of inputs and outputs, such an arrangement is impractical, because each combination of input and output would require a separate signal-processing transform. For example, a single-output system requires two different transforms to process color values measured by two different input devices. The number of required system transforms in this arrangement equals the product of the number of inputs and outputs. Thirty-two signal-processing transforms are required, for example, in a system having four inputs and eight outputs.

A much more efficient system results if the color signal processing is split into two parts — input signal processing and output signal processing. In this arrangement, each input and each output has its own associated transform. Each input signal processing transform converts input color-signal values to values for a defined color encoding specification, and each output transform converts values from the color encoding specification to values appropriate for the particular output. In this arrangement, the number of system transforms equals just the sum, rather than the product, of the number of inputs and outputs. For example, only 12 signal-processing transforms are required in a system having 4 inputs and 8 outputs.

The success of this approach depends on the use of an appropriate color encoding specification. The specification must allow for color information to be represented unambiguously and in a way that does not limit the desired functionality of the system. A complete color encoding specification must define two principal attributes of the color representation: a color encoding method and a color encoding data metric. The color encoding method determines the actual meaning of the encoded data, while the color encoding data metric defines the color space and the numerical units in which encoded data are expressed. Some considerations involved in the design of a color encoding data metric will be discussed later.

The selection of the encoding method for a given system must be based on some color property — a particular aspect of color — that all of the inputs of that system have in common. It is that aspect of color that must be
measured and digitally encoded so as to represent color completely and unambiguously in the encoding specification. Three fundamental types of measurement and encoding methods are discussed in the following section.

4.4 Color encoding methods

Densitometric color encoding is based on input-image color measurements made according to defined sets of spectral responsivities that are not equivalent to any set of visual color-matching functions. The responsivities can be those of a particular type of densitometric instrument, such as an ISO Status A or Status M densitometer. The responsivities also can be those of an actual scanner or of some hypothetical reference scanner. Encoded colors can be expressed in terms of red, green, and blue (RGB) densities, transmittances, or reflectances; cyan, magenta, and yellow (CMY) or cyan, magenta, yellow, and black (CMYK) colorant amounts; or other values associated with the densitometric measurements. The principal advantage of this type of encoding is that it corresponds quite directly to physical measurements of input images. Therefore, transformations from scanner RGB values to densitometric values, and transformations from densitometric values to output device RGB values, generally are quite simple. That simplicity can translate into optimum signal-processing accuracy and speed. However, the use of densitometric color encoding generally is limited to situations where all system input data are derived from essentially the same input medium. This often is the case in graphic arts and motion picture applications.

Colorimetric color encoding is similar to densitometric color encoding, except that it is derived from measurements made according to the spectral responsivities of a human observer. One of the principal advantages of this method is that it is based on well-established CIE recommendations for color measurement. At first glance, colorimetric encoding would seem to offer the perfect “device-independent” method for encoding color; in practice, colorimetric encoding sometimes can be used successfully where methods based on other forms of measurements will not work. Consider, for example, a system that supports input from an assortment of reflection media with image-forming colorants — printing inks, photographic dyes, thermal-transfer dyes, etc. — having different spectral absorption characteristics. A color encoding method based on RGB densitometric measurements alone would not provide a meaningful representation of color in this system. For example, a pair of colors on two different media might look identical, but they might produce quite different RGB densitometric values. Conversely, a pair of colors on two different media might appear quite different from one another, but they might happen to produce the same RGB densitometric values. These inconsistencies occur because the spectral absorption characteristics of the colorants used in the two media are different. Visual matches therefore will be metameric, not spectral. This makes colorimetric measurement a logical choice for color encoding. By definition, metameric pairs of color stimuli will have equal colorimetric values.
It is important to remember, however, that metameric matching is viewing-illuminant dependent. So, areas of color on different media that match when viewed under one illuminant might not match when viewed under another. This means that color encoding based on standard CIE colorimetric measurements can be used to encode color from multiple reflection media, but only if two conditions are realized. First, a single reference illuminant used for metameric matching must be specified, and, second, the encoded colorimetric values must be determined according to the spectral power distribution of that reference illuminant.

A further limitation of color encoding based on standard colorimetry alone is that it will not work for images input from media designed to be viewed under different conditions. For example, reflection prints generally are designed to be viewed under typical indoor conditions, while photographic slides are designed to be projected and viewed in a darkened room. Because an observer’s perceptions will be affected by the differences in these respective viewing conditions, the colorimetric properties of reflection-print and projection-slide media must be fundamentally different. The colorimetric values measured from one type of medium will not be appropriate for use on the other. For example, if the colorimetry of a slide is measured and reproduced exactly on a reflection print, that print will appear too dark, too high in luminance contrast, and too cyan-blue in color balance. The use of colorimetric encoding therefore must be limited to media designed for one set of viewing conditions.

This is a serious problem, because three fundamentally different types of viewing environments are involved in the color imaging process, and there are limitless possible sets of viewing conditions for each of those types. On the input side of an imaging system, there are original-scene environments, i.e., the environments in which live original scenes are viewed and captured. Also, on the input side, there are input-image environments, where hardcopy and soft-copy images that are to be input to a color imaging system are viewed. Finally, there are output-image environments, where hardcopy and soft-copy images produced by a color imaging system eventually are viewed.

One means for dealing with the effects of various viewing conditions is color appearance encoding. In this technique, colorimetric values associated with one set of viewing conditions are transformed to determine a visually corresponding set of colorimetric values associated with another set of viewing conditions. The transformations, which are based on models of the human visual system, can account for differences in factors such as absolute image luminance level, image surround, and the observer’s state of chromatic adaptation. Such transformations can be used, for example, to determine the colorimetric values required for a slide projected in a darkened room to visually match a reflection print viewed in a graphic arts viewing booth. Although transformations based on a color appearance model could be used quite directly in this particular example, other types of transformations are needed when an image is to be transformed from one image state to another, as described in the next section.
4.5 Image states

In a digital color imaging system, images can exist in several fundamentally different states. The image state is a function of how an image was captured, as well as any processing that may have been applied to the image. Although the concept of an image state can be applied to all attributes of an image, such as sharpness or noise, the aspect of image state that is of importance for the current discussion relates to the interpretation of the color values of the image. For example, the color values (digital code values) of an image could correspond to the sensor RGB values from a digital camera, the CIELAB values of a reflection print, or the ISO Status M RGB density values of a photographic negative. These image examples vary in two respects. First, they each use a different color space to encode the image (sensor RGB, CIELAB, and ISO Status M densities). However, just as importantly, each of these image encodings corresponds to a distinctly different image state (an original scene, a reflection print, and a photographic negative). Even if the same color space, CIELAB for example, were used to encode all of these images, it would still not be possible (or at least not optimal) to treat the images identically. Obviously, something quite different would have to be done with the CIELAB values of a color negative relative to the CIELAB values of a print. As will be discussed in more detail later, the same also would be true for the CIELAB values of an original scene relative to those of a print.

Most digital images can be broadly categorized into two types of image states: unrendered and rendered. Images in an unrendered image state are directly related to the colorimetry of real or hypothetical original scenes. Such images are sometimes called scene-referred images. Images in this category would include raw digital camera captures and images stored in the Kodak PhotoYCC color interchange space. Images in a rendered image state are representations of the colorimetry of output images (such as a print, a slide, or a CRT display) and are sometimes called output-referred images. Many common color encodings, such as sRGB and SWOP CMYK, fall into this category. A third category of image state applies to the encoding of photographic color negatives. Unprocessed images captured by photographic color negative film scanners are in this image state, although it typically is a temporary state prior to forming a rendered image or determining a corresponding scene-referred image.

To enable the optimal use of digital images, it is important to distinguish images in an output-referred image state from those in a scene-referred image state. It is well known that the colorimetry of a pleasing rendered image generally does not match the colorimetry of the corresponding scene. Among other things, the tone/color reproduction process that “renders” the colors of a scene to the desired colors of the output image must compensate for differences between the scene and rendered image viewing conditions. For example, rendered images generally are viewed at luminance levels much lower than those of typical outdoor scenes. Consequently, an increase in the overall contrast of the rendered image usually is required to compen-
sate for perceived losses in reproduced luminance and chrominance contrast. Additional contrast increases in the shadow regions of the rendered image also are needed to compensate for the viewing flare associated with rendered-image viewing conditions.

Psychological factors, such as color memory and color preference, also must be considered in image rendering. For example, observers generally remember colors as being of higher purity than they originally were, and they typically prefer the reproductions of skies and grasses to be more colorful than they were in the original scene. The tone/color reproduction aims of well-designed imaging systems will account for such factors.1,7

Finally, the tone/color reproduction process also must account for the fact that the dynamic range of an output device or medium usually is substantially less than that of an original scene. It is therefore typically necessary to discard and/or compress some of the highlight and shadow information of the scene to fit within the dynamic range of the rendered output image. This is shown in Figure 4.1, which illustrates a typical backlit scene. In this example, the approximate scene colorimetry was determined from a scan of a color negative. The image on the left shows a rendering of the scene appropriate for the foreground information, and the image on the right shows a rendering of the scene appropriate for the background infor-

Figure 4.1  (See color insert on page 430) Rendering of large dynamic range scene optimized for (a) foreground, and (b) background.
mation. In the first case, much of the highlight information was clipped by the rendering process. Likewise, in the second case, much of the shadow information was lost. This is illustrated further in Figure 4.2, which shows a histogram of the scene luminance data for the image shown in Figure 4.1. A conventional reflection print of this scene can reproduce only about six stops (1.8 log luminance units) of scene information within the dynamic range of the output medium. The indicated ranges show the subsets of the scene luminance information corresponding to the two images in Figure 4.1. It can be seen that only a portion of the total scene information is reproduced in either of the rendered images.

Because the colorimetry of scenes and their corresponding rendered images are intentionally and necessarily different, it would be ambiguous to represent images in both image states using the same color encoding specification. For example, if one were to send the CIELAB values for a particular image, with no information about whether the color values were original-scene color values or rendered-image color values, the recipient would not know what to do with the image values so as to make a good output image. If the CIELAB values were rendered color values appropriate for the output viewing environment, it simply would be necessary to determine the device code values needed to produce the specified colorimetry. However, if the color values corresponded to original-scene color values, it would be necessary to modify the image colorimetry by applying an appropriate tone/color reproduction transformation before producing the output image. Directly reproducing the scene colorimetry on an output image gen-

![Figure 4.2](image.png)

**Figure 4.2** Histogram of relative log scene luminance values for the scene shown in Figure 4.1. A scene luminance range of about 1.8 log units can be reproduced on a typical output reflection print. Because the dynamic range of the original scene is substantially larger than this, a subset of the image data must be selected during the rendering process. Different results are obtained depending on whether the foreground or background region of the image is selected.
erally would produce results that would be judged inferior. For example, Figure 4.3 shows a pair of images generated from the same scene. Image (a) approximately matches the colorimetry of the original scene, whereas an appropriate tone/color reproduction transformation has been used to modify the colorimetry of image (b) to produce an image that generally would be judged to have improved color reproduction.

One of the advantages of encoding images in a scene-referred image state is that such encoding provides the capability of retaining the maximum amount of image information. As was illustrated in Figure 4.1, once an image is committed to a rendered output-referred image state appropriate for printing or display, any extended dynamic range information is permanently lost.

Figure 4.3  (See color insert) Two different renderings of a scene. In image (a), the colorimetry of the rendered image closely matches that of the original scene. In image (b), the rendered image is not colorimetrically accurate, but the resulting image generally would be judged to have improved color reproduction.
Retaining the scene-referred image data preserves the maximum flexibility for the potential uses of an image. This allows for the correction of image-capture exposure errors and enables multiple renditions to be made from a given image. For example, a photographer could decide at the time an image is printed whether to optimally render the foreground information or the background information from a backlit scene. It is valuable to preserve this option because there often will not be a single “best” choice that can be made when the image is captured. For the image shown in Figure 4.1, the final decision would depend on whether the photographer was most interested in the boys in the foreground or the scenic Alps in the background. Retaining the extended dynamic-range scene information also enables other options, such as employing advanced image processing techniques to produce a print wherein both the foreground and the background are well rendered, as shown in Figure 4.4. Comparable results could not be attained starting from one of the conventionally rendered images shown in Figure 4.1.

4.6 *Standard image-state color encoding specifications*

The fact that images exist in many different image states and are expressed in terms of numerous color spaces significantly complicates the development
of software applications that use and manipulate images. For example, an image-processing algorithm that works in one color space might not have the expected behavior when used in another color space. To reduce complexity of imaging system design, it is desirable to define standard color encodings for each of the main classes of image states. This provides for the unambiguous communication of color information and allows the development of standard image-manipulation algorithms and standard color-processing paths.

Attempts to standardize color encodings typically have involved the specification of a particular output-device-dependent color space that is central to the workflow for a certain market segment. Examples of such color spaces include sRGB and SWOP CMYK. Although such standardizations can work well within the limited scope of a particular application, significant compromises are necessary to use them in other applications. For example, hardcopy media and CRT displays typically have very different color gamuts. Therefore, using sRGB (which is based on a particular CRT model) as a standard color encoding necessarily involves clipping many colors that could have been produced on a given hardcopy medium. This would be unacceptable in many hardcopy-based market segments, such as consumer photo finishing and graphic arts.

The International Color Consortium (ICC) has defined a Profile Connection Space (PCS) that comprises a color encoding specification that can be used to explicitly specify the color of an output-referred image with respect to a reference viewing environment. It could be argued that the PCS could serve as the standard color encoding specification for rendered images. However, it never was intended that the PCS be used to store or manipulate images directly. Rather, it was intended to be a color space where device profiles could be joined to form complete input-to-output color transforms. Neither the CIELAB nor the CIE XYZ color encodings supported for the PCS is particularly well suited for many common types of image manipulations. Additionally, quantization errors introduced by encoding images in PCS would be significantly larger than necessary, because a large percentage of code value combinations correspond to unrealizable colors.

Given the limitations of the existing solutions, Eastman Kodak Company has developed a family of color encoding specifications for use in the development of its digital imaging products. These specifications are being offered for use by other companies, and they also have been proposed for international standardization. The following detailed discussion of the properties of these color encoding specifications will help to clarify several topics previously discussed, including color encoding methods, color encoding data metrics, and image states.

The first of these specifications, Reference Input Medium Metric RGB (RIMM RGB), is ideal for the manipulation, storage, and interchange of images from sources such as digital cameras that naturally capture scene-referred image data. A companion specification, Reference Output Medium Metric RGB (ROMM RGB), serves a similar purpose for images from sources such as print scanners and other devices that produce images in a rendered output-referred
image state. Figure 4.5 illustrates how these standard color encoding specifications can be used as the basis for a general imaging system architecture.

Before images can be sent to an output device, such as a printer, it generally will be necessary to convert scene-state images to rendered-state images using a tone/color rendering operation. However, in the same way that a negative is much more versatile than a print, an image in a scene-referred state will be much more versatile than one in a rendered-image state. Therefore, it is desirable in many imaging systems to delay any conversion to a rendered-image state until such time that an output image is to be generated. This provides the maximum flexibility for the imaging system.

4.6.1 Criteria for selection of RIMM/ROMM RGB color encoding specifications

It was desirable that the RIMM RGB and ROMM RGB color encoding specifications be defined such that they are as similar as possible to one another. Doing so simplifies the development of image-manipulation algorithms across the two color encodings. It also simplifies the rendering process in which a rendered ROMM RGB image is created from an original-scene image encoded in RIMM RGB. This desired similarity is best achieved by basing the data metrics of the two encoding specifications on the same color space. A number of criteria were used to select this color space. Specifically, the space should have the following properties:

- A direct relationship to the color appearance of the scene/image
- A color gamut large enough to encompass most real-world surface colors using non-negative tristimulus values
- An efficient encoding of the color information to minimize quantization artifacts
- A simple transformation to/from ICC PCS

![Figure 4.5 Image state diagram showing standard color encodings.](image-url)
• A simple transformation to/from video RGB (e.g., sRGB)
• Be well suited for application of common image manipulations such as tone scale modifications, color-balance adjustments, sharpening, etc.
• Be compatible with established imaging workflows

All of these criteria can be achieved by the use of an additive RGB color space with an appropriately selected set of wide-gamut primaries. When images are encoded using a set of RGB primaries, there is a direct and simple relationship to scene/image colorimetry, because such primaries are linear transformations of the CIE XYZ primaries. RGB color spaces have the additional advantage that simple transformations based on a one-dimensional lookup table (LUT), a matrix, and another LUT can be used to convert to/from additive color spaces such as PCS XYZ, video RGB (sRGB), and digital camera RGB.

However, two of the criteria that affect the selection of the particular RGB primaries are somewhat conflicting. First, the chromaticities of the primaries should define a gamut sufficiently large to encompass colors likely to be found in real scenes and images. Such colors will then be defined by non-negative tristimulus values, which generally simplifies subsequent signal processing such as tone scale modifications. At the same time, their use should result in efficient digital encodings that minimize quantization errors. Increasing the gamut to encompass more colors only can be achieved by trading off against correspondingly larger quantization errors (given a fixed bit depth). If the chromaticities of the primaries are chosen to include the maximum possible color gamut (for example, choosing the XYZ primaries would encompass the entire spectrum locus), a significant fraction of the color space would correspond to imaginary colors and to colors that would not commonly be encountered in real images. Therefore, in any encoding using such a color space, there would be large numbers of code value combinations that never would be used in practice. This would lead to larger quantization errors in the usable part of the color space than would be obtained with different primaries defining a smaller chromaticity gamut. It is, therefore, desirable to choose primaries with a gamut that is sufficiently large, but not larger than necessary.

Figure 4.6 shows the primaries selected for RIMM/ROMM RGB. These primaries encompass the gamut of real-world surface colors, without devoting a lot of space to non-realizable colors outside the spectrum locus. Also shown for comparison are the sRGB primaries. It can be seen that the area defined by the sRGB chromaticity boundaries is inadequate to cover significant portions of the real-world surface color gamut. In particular, it excludes many important high-chroma colors near the yellow-to-red boundary of the spectrum locus.

Another important requirement for the RIMM RGB and ROMM RGB color encoding specifications is that they be well suited for the application of common image manipulations. Many types of image manipulations

© 2003 by CRC Press LLC
include the step of applying nonlinear transformations to each of the channels of an RGB image (e.g., tone scale modifications, color balance adjustments, etc.). The process of forming a rendered image from a scene is one important application of this type. One way to accomplish the rendering operation is by the application of a nonlinear tone scale transformation to the individual channels of an RGB image in a scene-referred image state. A well-designed transformation of this type will have the desirable effects of increasing the luminance and color contrast in the mid-tones, compressing the contrast of the highlights and shadows, increasing the chroma of in-gamut colors, and gamut mapping out-of-gamut colors in a simple but visually pleasing way. If an input scene is represented using the RIMM RGB color encoding, the result of applying such rendering transforms will be a rendered image in the ROMM RGB color encoding.

Nonlinear channel-independent transforms will, in general, modify the ratios of the red, green, and blue channel data. This can lead to unwanted hue shifts, particularly for high-chroma colors. Hue shifts are particularly problematic in reproductions of natural chroma gradients having constant hue and saturation. Such gradients occur when rounded surfaces are illuminated by a moderately directional light source. In these situations, chroma increases with increasing distance from the specular highlight and then decreases again as the shadows deepen.

The induction of hue shifts by the application of the nonlinear channel-independent transforms can never be completely eliminated. One objective for optimizing the location of the primaries was to eliminate or minimize objectionable hue shifts, sometimes at the expense of less noticeable or less

\[ \text{Figure 4.6} \quad \text{Comparison of ROMM RGB and sRGB primaries in } x-y \text{ chromaticity coordinates.} \]
likely hue shifts. Hue shifts for a particular color can be eliminated when the color lies on one of the straight lines passing through the primaries and the white point on a chromaticity diagram.

The effects of nonlinear transforms on hue shifts were studied using a chroma series for eight color patches from the Macbeth Color Checker. These patches included red, yellow, green, cyan, blue, magenta, light skin, and dark skin. Hue shifts in skin tones and yellows, particularly in the direction of green, are considered the most objectionable. These hue shifts are most strongly affected by the location of the blue primary. Other colors that were considered particularly important during the optimization process were blues and reds.

There is a trade-off between the color gamut of the primaries, quantization artifacts, and the extent of the hue shifts that occur during rendering. If the primaries are moved out to increase the color gamut, quantization artifacts will increase, and the hue shifts introduced during the application of a nonlinear transformation generally will decrease. This results from the fact that the RGB values in real images will be distributed over a smaller range, thereby reducing the impact of nonlinear transformations. If the color gamut is decreased by moving the primaries closer together, quantization artifacts diminish, but hue shifts are generally larger, and color gamut is sacrificed.

Finally, a basic requirement for any commercially useful color encoding is that it be compatible with typical commercial imaging workflows. In many cases, Adobe Photoshop software is an important component in such imaging chains. Conveniently, Adobe Photoshop versions 5.0 and higher have incorporated the concept of a “working color space,” which is different from the monitor preview color space. This is consistent with the concept of storing/manipulating images in an extended color gamut space. Adobe has placed a constraint on the definition of valid working color spaces that requires the primaries to have all-positive \(x, y, \text{ and } z\) chromaticity values. This condition is satisfied for the ROMM RGB primaries.† (Because Adobe Photoshop software operates within a rendered-image paradigm, it is inappropriate to use RIMM RGB as a Photoshop software working color space.)

During the selection of the RIMM/ROMM RGB primaries, an extensive optimization process was used to determine the best overall solution to satisfy all of these criteria. The CIELAB hue shifts associated with the selected RIMM/ROMM RGB primaries are shown in Figure 4.7. This plot shows a series of line segments connecting corresponding pairs of CIELAB \(a^*, b^*\) values, before and after a nonlinear tone scale transformation was applied to a chroma series in each of eight color directions. It can be seen that only relatively small hue shifts are introduced for the highest chroma colors in the blue and cyan directions, and the hue shifts elsewhere are virtually negligible. Overall, these hue shifts are very small compared to those

† For more information about using ROMM RGB as a Photoshop software working space, see the white paper posted at www.kodak.com (search on “ROMM”).
Chapter four: Color management for digital imaging systems

Associated with most other sets of additive primaries. Similar results were obtained when this hue-shift analysis was carried out using several other color spaces, including CIECAM97s, IPT, and the OSA_UCS color space.

4.6.2 ROMM RGB color encoding specification

Reference Output Medium Metric RGB (ROMM RGB) is designed to be an extended-gamut color encoding specification for representing the color appearance of an output-referred image. In addition to specifying the image state and color space, it is also necessary to specify an intended viewing environment to define unambiguously an encoding of color appearance. One of the requirements for ROMM RGB is that it be tightly coupled to the ICC Profile Connection Space (PCS). Color values in the PCS represent the CIE colorimetry of a defined reference medium that will produce the desired color appearance when viewed in a reference viewing environment. The reference viewing environment for ROMM RGB was based on that defined in the latest ICC draft specification and is specified to have the following characteristics:

1. The luminance level for the observer adaptive white is 160 cd/m².
2. The observer adaptive white has the chromaticity values of CIE Standard Illuminant D50 (x = 0.3457, y = 0.3585).
3. The viewing surround is average, i.e., the overall luminance level and chromaticity of the surround are assumed to be similar to that of the image.
4. There is 0.75% viewing flare, referenced to the observer adaptive white.

Figure 4.7 Hue shifts resulting from a typical nonlinear rendering transform for (a) the RIMM/ROMM RGB primaries, and (b) an alternate set of wide-gamut primaries. The hue shifts for the most important colors are visually negligible for the RIMM/ROMM RGB color encoding.
5. The image color values are assumed to be encoded using flareless (or flare-corrected) colorimetric measurements based on the CIE 1931 Standard Colorimetric Observer.

The ROMM RGB color encoding is defined in the context of a reference imaging medium associated with a hypothetical additive color device having the following characteristics:

1. Reference primaries defined by the CIE chromaticities given in Table 4.1
2. Equal amounts of the reference primaries producing a neutral with the chromaticity of D_50 (x = 0.3457, y = 0.3585)
3. The capability of producing a white with a luminance factor of F_W = 0.89 and a black with a luminance factor of F_K = 0.0030911

<table>
<thead>
<tr>
<th>Color</th>
<th>x</th>
<th>y</th>
</tr>
</thead>
<tbody>
<tr>
<td>Red</td>
<td>0.7347</td>
<td>0.2653</td>
</tr>
<tr>
<td>Green</td>
<td>0.1596</td>
<td>0.8404</td>
</tr>
<tr>
<td>Blue</td>
<td>0.0366</td>
<td>0.0001</td>
</tr>
<tr>
<td>White</td>
<td>0.3457</td>
<td>0.3585</td>
</tr>
</tbody>
</table>

Images intended to be viewed in other viewing environments, or on a medium different from the reference medium, can be encoded in ROMM RGB by first determining the corresponding tristimulus values that would produce the intended color appearance on the reference medium when viewed in the reference viewing environment. The corresponding tristimulus values can be determined by using appropriate color appearance transformations to account for the differences between the actual and reference viewing conditions. Additionally, it may be necessary to account for differences in the media characteristics.

The conversion of the PCS XYZ tristimulus values to ROMM RGB values can be performed by a matrix operation followed by a set of one-dimensional functions. This is equivalent to the operations associated with a basic monitor profile, which means that ROMM RGB can be incorporated in a system employing ICC profiles simply by using an appropriately designed display profile.

Most current implementations of the ICC PCS incorporate the concept of a reference medium wherein the black point of the reference medium is mapped to Y_{PCS} = 0, and the white point of the reference medium is mapped...
to $Y_{PCS} = 1.0$. Therefore, to relate actual CIE image colorimetry to PCS XYZ values, an appropriate normalizing transformation is required as follows:

$$X_{PCS} = \frac{(X - X_K) X_W}{(X_W - X_K) Y_W}$$

$$Y_{PCS} = \frac{(Y - Y_K)}{(Y_W - Y_K)}$$

$$Z_{PCS} = \frac{(Z - Z_K) Z_W}{(Z_W - Z_K) Y_W}$$

(4.1)

where $X, Y, Z =$ CIE image tristimulus values

$X_{PCS}, Y_{PCS}, Z_{PCS} =$ PCS tristimulus values

$X_W, Y_W, Z_W =$ tristimulus values of the reference medium white point ($X_W = F_W X_0 = 85.81$, $Y_W = F_W Y_0 = 89.00$, and $Z_W = F_W Z_0 = 73.42$, where $X_0 = 96.42$, $Y_0 = 100.00$, and $Z_0 = 82.49$)

$X_K, Y_K, Z_K =$ tristimulus values of the reference medium black point ($X_K = F_K X_0 = 0.2980$, $Y_K = F_K Y_0 = 0.3091$, and $Z_K = F_K Z_0 = 0.2550$)

4.6.2.1 ROMM RGB conversion matrix

Given the defined primaries shown in Table 4.1, the following matrix can be derived to compute the linear ROMM RGB values from the PCS image tristimulus values:

$$\begin{bmatrix}
R_{ROMM} \\
G_{ROMM} \\
B_{ROMM}
\end{bmatrix} = \begin{bmatrix}
1.3460 & -0.2556 & -0.0511 \\
-0.5446 & 1.5082 & 0.0205 \\
0.0000 & 0.0000 & 1.2123
\end{bmatrix}\begin{bmatrix}
X_{PCS} \\
Y_{PCS} \\
Z_{PCS}
\end{bmatrix}$$

(4.2)

As required by the definition of ROMM RGB, this matrix will map image tristimulus values with the chromaticity of $D_65$ to equal ROMM RGB values. A neutral with a $Y_{PCS}$ value of 1.0, corresponding to the reference medium white point, will map to linear ROMM RGB values of 1.0. Likewise, the reference medium black point will map to linear ROMM RGB values of 0.0.

4.6.2.2 Nonlinear encoding of ROMM RGB

A nonlinear quantization function is used to store the ROMM RGB values in integer form. A simple gamma function nonlinearity incorporating a slope limit at the dark end of the intensity scale is defined for this purpose.

© 2003 by CRC Press LLC
where \( C \) is \( R \), \( G \), or \( B \); \( I_{\text{max}} \) is the maximum integer value used for the nonlinear encoding; and

\[
E_t = 16^{1.8(1 - 1.8)} = 0.001953
\] (4.4)

For the baseline 8-bit configuration, \( I_{\text{max}} \) is equal to 255. The linear segment of the nonlinearity is used to impose a slope limit to eliminate reversibility problems that otherwise would result from the infinite slope of the gamma function at the zero point. Twelve-bit and 16-bit versions of ROMM RGB are also defined. The only difference is that the value of \( I_{\text{max}} \) is set to 4095 or 65535, respectively. In cases where it is necessary to identify a specific precision level, the notations ROMM8 RGB, ROMM12 RGB, and ROMM16 RGB are used. Table 4.2 shows some sample encodings for a series of neutral patches of specified \( Y_{\text{PCS}} \).

### Table 4.2 Sample Neutral Patch Encodings

<table>
<thead>
<tr>
<th>( Y_{\text{PCS}} )</th>
<th>ROMM8 RGB</th>
<th>ROMM12 RGB</th>
<th>ROMM16 RGB</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.00</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0.001</td>
<td>4</td>
<td>66</td>
<td>1049</td>
</tr>
<tr>
<td>0.01</td>
<td>20</td>
<td>317</td>
<td>5074</td>
</tr>
<tr>
<td>0.10</td>
<td>71</td>
<td>1139</td>
<td>18236</td>
</tr>
<tr>
<td>0.18</td>
<td>98</td>
<td>1579</td>
<td>25278</td>
</tr>
<tr>
<td>0.35</td>
<td>142</td>
<td>2285</td>
<td>36574</td>
</tr>
<tr>
<td>0.50</td>
<td>174</td>
<td>2786</td>
<td>44590</td>
</tr>
<tr>
<td>0.75</td>
<td>217</td>
<td>2490</td>
<td>55855</td>
</tr>
<tr>
<td>1.00</td>
<td>255</td>
<td>4095</td>
<td>65535</td>
</tr>
</tbody>
</table>

### 4.6.3 RIMM RGB color encoding specification

Reference Input Medium Metric RGB (RIMM RGB) is a companion color encoding specification to ROMM RGB that can be used to encode the colorimetry of an unrendered scene. Both encodings utilize the same wide-gamut color space defined by the primaries and white point given in Table 4.1. The reference viewing conditions used to encode scene color...
values for RIMM RGB are typical of outdoor environments and are defined as follows:

- The luminance level for the observer adaptive white is 15,000 cd/m².
- The observer adaptive white has the chromaticity values of CIE Standard Illuminant D50 (x = 0.3457, y = 0.3585).
- Viewing surround is average, i.e., the overall luminance level and chrominance of the surround is assumed similar to that of the scene.
- There is no viewing flare for the scene other than that already included in the scene colorimetric values.
- The scene color values are assumed to be encoded using flareless (or flare corrected) colorimetric measurements based on the CIE 1931 Standard Colorimetric Observer.

Scenes captured under conditions different from those of the reference viewing environment can be encoded in RIMM RGB by first determining the corresponding tristimulus values that would produce the intended color appearance in the reference viewing environment. For some applications, the intended color appearance may be an estimate of the appearance of the original scene if it had been captured in the reference viewing environment. For other applications, it may be desirable to encode the color appearance of the scene in its particular capture viewing environment. In this case, corresponding tristimulus values can be determined by using appropriate color appearance transformations to account for the differences between the actual and reference viewing conditions.

### 4.6.3.1 RIMM RGB conversion matrix

Because ROMM RGB and RIMM RGB use a common color space, the conversion from scene tristimulus values to corresponding linear RIMM RGB values can be accomplished using the same conversion matrix that was given in Equation 4.2, except that the input tristimulus values are scene XYZ values rather than PCS XYZ values.

\[
\begin{bmatrix}
R_{RIMM} \\
G_{RIMM} \\
B_{RIMM}
\end{bmatrix} = \begin{bmatrix}
1.3460 & -0.2556 & -0.0511 \\
-0.5446 & 1.5082 & 0.0205 \\
0.0000 & 0.0000 & 1.2123
\end{bmatrix} \begin{bmatrix}
X_{D50} \\
Y_{D50} \\
Z_{D50}
\end{bmatrix}
\]

(4.5)

*Note:* The scene XYZ values are normalized such that the luminance of a correctly exposed perfect white diffuser in the scene will have a value of \( Y_{D50} = 1.0 \).

### 4.6.3.2 Nonlinear encoding of RIMM RGB

Because the dynamic range of unrendered scenes is generally larger than that of the medium specified for ROMM RGB, a different nonlinear encoding must be used. The RIMM RGB nonlinearity is based on that specified by...
Recommendation ITU-R BT.709\textsuperscript{16} (formerly known as CCIR 709). This is the same nonlinearity used in the Kodak PhotoYCC color interchange space encoding implemented in the Kodak Photo CD system\textsuperscript{4} and is given by

\[
C'_{RIMM} = \begin{cases} 
0; & C_{RIMM} < 0.0 \\
\frac{I_{\text{max}}}{V_{\text{clip}}}4.5C_{RIMM}; & 0.0 \leq C_{RIMM} < 0.018 \\
\frac{I_{\text{max}}}{V_{\text{clip}}}1.099C_{RIMM}^{0.45} - 0.099; & 0.018 \leq C_{RIMM} < E_{\text{clip}} \\
I_{\text{max}}; & C_{RIMM} \geq E_{\text{clip}}
\end{cases} \tag{4.6}
\]

where \(C\) is either \(R\), \(G\), or \(B\); \(I_{\text{max}}\) is the maximum integer value used for the nonlinear encoding; \(E_{\text{clip}} = 2.0\) is the normalized scene luminance level that is mapped to \(I_{\text{max}}\); and

\[
V_{\text{clip}} = 1.099E_{\text{clip}}^{0.45} - 0.099 = 1.402 \tag{4.7}
\]

For the baseline 8-bit/channel RIMM RGB configuration, \(I_{\text{max}}\) is 255. In some applications, it may be desirable to use a higher-bit-precision version of RIMM RGB to minimize any quantization errors. Twelve-bit and 16-bit per channel versions of RIMM RGB are also defined. The only difference is that the value of \(I_{\text{max}}\) is set to 4095 or 65535, respectively. In cases in which it is necessary to identify a specific precision level, the notations RIMM8 RGB, RIMM12 RGB, and RIMM16 RGB are used.

4.6.4 ERIMM RGB color encoding specification

The RIMM RGB color space is defined to have an extended luminance dynamic range that can encode information up to 200\% of the luminance value associated with a normally exposed perfect (100\%) diffuse white reflector in the scene. This should be adequate for many input sources, such as digital cameras, which themselves have a somewhat limited dynamic range. However, for some inputs, most notably scanned photographic negatives, a greater luminance dynamic range is required to encode the full range of captured scene information. For example, consider the histogram of scene luminance data previously shown in Figure 4.2. The RIMM RGB encoding would only retain scene information up to a log relative scene luminance value of 0.3. A significant portion of the scene information would be lost with a RIMM RGB encoding in this case. To provide an encoding that can retain the full range of captured scene information, a variation of the RIMM RGB color space, Extended Reference Input Medium Metric RGB (ERIMM RGB), is defined.
As with RIMM RGB, ERIMM RGB is related directly to the colorimetry of an original scene. The nonlinear encoding function is the only encoding operation that is different. For ERIMM RGB, it is desirable to increase both the maximum scene luminance value that can be represented as well as to reduce the quantization interval size. The size of the quantization interval is directly related to the minimum scene luminance value that can be accurately represented. To satisfy both the extended luminance dynamic range and reduced quantization interval requirements simultaneously, it is necessary to use a greater bit precision for ERIMM RGB. A minimum of 12 bits per color channel is recommended.

4.6.4.1 Nonlinear encoding for ERIMM RGB

A modified logarithmic encoding is used for ERIMM RGB. A linear segment is included for the very lowest luminance values to eliminate the non-invertibility of a strictly logarithmic encoding at the dark end of the scale. The encoding was defined such that the linear and logarithmic segments match in both value and derivative at the boundary. In equation form, this encoding is represented by

\[
C_{ERIMM} = \begin{cases} 
0; & C_{RIMM} \leq 0 \\
\left( \frac{0.0789626}{E_t} \right) C_{RIMM} \left( I_{\text{max}} \right); & 0 < C_{RIMM} \leq E_t \\
\left( \frac{\log C_{RIMM} + 3.0}{5.5} \right) I_{\text{max}}; & E_t < C_{RIMM} \leq E_{\text{clip}} \\
I_{\text{max}}; & C_{RIMM} > E_{\text{clip}}
\end{cases}
\] (4.8)

where \( C \) is \( R, G, \) or \( B; I_{\text{max}} \) is the maximum integer value used for the nonlinear encoding; \( E_{\text{clip}} = 10^{2.5} = 316.23 \) = the upper scene luminance limit that gets mapped to \( I_{\text{max}} \), and

\[
E_t = e / 1000 = 0.00271828
\] (4.9)

is the break point between the linear and logarithmic segments, \( e \) being the base of the natural logarithm. For a 12-bit encoding, \( I_{\text{max}} \) is 4095, and for a 16-bit encoding, \( I_{\text{max}} \) is 65535. In cases in which it is necessary to identify a specific precision level, the respective notations \( ERIMM12 RGB \) and \( ERIMM16 RGB \) are used.

To compute ERIMM RGB values, Equation 4.8 should be used in place of Equation 4.6 in the previously described procedure for determining RIMM RGB values. Examples of RIMM RGB and ERIMM RGB encodings for neutral patches at different scene relative luminance levels are shown in Table 4.3. It can be seen that the range of relative luminances that can be represented in ERIMM RGB is greatly extended from that of RIMM RGB.
The use of color management systems, such as that developed by the ICC, is becoming increasingly common in a variety of digital imaging applications. Color management systems typically are based on an architecture in which the color response of an input device is characterized using an input profile, which describes the relationship between the device code values and color values in some profile connection space (PCS). Similarly, the color response of an output device is characterized using an output profile, which describes the relationship between the PCS color values and the corresponding device code values needed to produce colors having those values. The PCS used in the ICC color management architecture, and the color encodings used in virtually every other color management system, are defined to be in a rendered output-referred image state. This type of PCS, generally based on reflection-print media viewed in indoor viewing environments, greatly complicates the use of a color management architecture based on the image-states paradigm. For example, a traditional input profile cannot be used for an input device that captures scene-referred data if it is desired to convert the image data to the standard scene-referred color encoding (i.e., (E)RIMM RGB). This is because the output of such a profile would be PCS color values in a rendered-image state. As discussed previously, the process of rendering an image from a scene-image state to a rendered-image state typically will involve an irreversible loss of information. Thus, transforming an image into RIMM RGB by combining a device-to-PCS profile with a PCS-to-RIMM RGB profile would seriously compromise the quality of the resulting image.

However, this does not mean that traditional color management architectures must be discarded altogether to build an imaging system around the image-state paradigm previously shown in Figure 4.5. Rather, it simply means that conventional input/output profiles cannot be used in the imaging

---

Table 4.3 Sample Scene Luminance Encodings

<table>
<thead>
<tr>
<th>Relative Luminance</th>
<th>Relative Log Luminance</th>
<th>RIMM8 RGB</th>
<th>RIMM12 RGB</th>
<th>ERIMM12 RGB</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.001</td>
<td>-3.00</td>
<td>1</td>
<td>13</td>
<td>119</td>
</tr>
<tr>
<td>0.01</td>
<td>-2.00</td>
<td>8</td>
<td>131</td>
<td>745</td>
</tr>
<tr>
<td>0.10</td>
<td>-1.00</td>
<td>53</td>
<td>849</td>
<td>1489</td>
</tr>
<tr>
<td>0.18</td>
<td>-0.75</td>
<td>74</td>
<td>1194</td>
<td>1679</td>
</tr>
<tr>
<td>1.00</td>
<td>0.00</td>
<td>182</td>
<td>2920</td>
<td>2234</td>
</tr>
<tr>
<td>2.00</td>
<td>0.30</td>
<td>255</td>
<td>4095</td>
<td>2458</td>
</tr>
<tr>
<td>8.00</td>
<td>0.90</td>
<td>NA</td>
<td>NA</td>
<td>2906</td>
</tr>
<tr>
<td>32.00</td>
<td>1.50</td>
<td>NA</td>
<td>NA</td>
<td>3354</td>
</tr>
<tr>
<td>316.23</td>
<td>2.50</td>
<td>NA</td>
<td>NA</td>
<td>4095</td>
</tr>
</tbody>
</table>
chain until the point at which the image is ready to be committed to a final output rendering. Fortunately, most color management systems provide for the concept of a *device link profile* that can be used to bypass the PCS and go directly from an input color space to an output color space. (Typically, such device link profiles would be created by cascading an input profile with an output profile, but this is not a requirement.)

Figure 4.8 illustrates this approach in more detail. Device link profiles are used to transform scene-referred input images into *RIMM RGB*. Input-image sources might include digital cameras (when unrendered sensor RGB values are available), as well as color negative film scanners (when special transformations are used to extract scene colorimetry from scanned densitometric values). In this case, not only is *RIMM RGB* used as a stopping point where images can be stored or edited, but it also becomes the output color space for the device profiles, effectively serving the role of a “scene-referred profile connection space.” At the point when it is finally desired to produce an image on an output device such as a printer or CRT, a conventional input profile can be used to render the *RIMM RGB* image to the PCS. This profile would include the desired system tone/color reproduction characteristics. A conventional output profile then can be used to transform the PCS image to the appropriate output device code values.

Conventional input profiles can be used for input devices, such as print/slide scanners and CRTs, where the input images already are in a rendered image state. These input profiles can be combined directly with output profiles to produce an image for a particular output device. Alternatively, the input profile can be combined with a *ROMM RGB* profile to convert the image to *ROMM RGB* for the purposes of storage, interchange, or editing. Because *ROMM RGB* is a simple LUT/matrix away from ICC
PCS XYZ values, it falls within the class of color encodings that can be represented with a simple display profile.

It should be noted that the input profiles used for rendered images intended to be viewed in environments significantly different from the reference viewing environment defined for the PCS must also include appropriate viewing environment transformations. For example, photographic slides are typically intended to be viewed in a darkened room. The colorimetric characteristics of slide films are designed with a higher luminance contrast so as to produce pleasing images in that viewing environment. Therefore, an input profile for a slide scanner must not only account for the colorimetric characteristics of the scanner; it must also include an appropriate transformation that will determine the visually equivalent colorimetry for the PCS reference viewing environment. It may also be necessary for the profile to perform some amount of re-rendering of the image to map the extended dynamic range of the slide film into the reflection-print-like dynamic range of the PCS reference medium.

During the process of working with images that are stored in the RIMM/ROMM RGB color encodings, it frequently will be desirable to preview the image on a video display. In a color-managed system, this can be accomplished by combining the appropriate RIMM RGB or ROMM RGB profile with a display profile for the particular video display. Because RIMM/ROMM RGB are based on a simple additive color space, a simple display-type profile using only a LUT followed by a matrix generally can be used to get to PCS XYZ. Likewise, the output profile for the video display would comprise a matrix followed by a gamma-function nonlinearity. For cases where processing speed is a critical concern, these operations can be combined, yielding a simple LUT–matrix–LUT processing chain that can be implemented directly and optimized for speed.

An example of an imaging chain for a representative system utilizing the standard image state architecture is shown in Figure 4.9. The input device for this example is a color negative film scanner. A device link profile is used to convert the raw film scanner image to a corresponding ERIMM RGB image. This profile accounts for the characteristics of the scanner as well as the characteristics of the film used to capture the image. Once the image is in ERIMM RGB, many different types of algorithms can be used to operate on the image. For example, a scene balance algorithm can be used to automatically color balance the image to correct for any variations in capture illumination and/or film processing, or an advanced tone scale algorithm could be used to properly darken the background of a backlit scene. ERIMM RGB is an appropriate color encoding for applying many types of image-processing algorithms, but it is especially important that algorithms utilizing the extended dynamic range scene information of the encoding be applied in ERIMM RGB before the image is rendered to an output-referred state.

After all scene-state image manipulations have been applied, the image can be rendered to produce a corresponding rendered-state image. In this example, the image is converted to a ROMM RGB representation where
further operations will be applied. This conversion can be applied by combining an ERIMM RGB input profile with a ROMM RGB profile. The ERIMM RGB input profile is used to impart the system tone/color reproduction aims relating the scene color values to the corresponding rendered image color values. These aims may be application dependent. For example, consumer photographers generally prefer higher contrast and higher saturation images than those preferred by professional portrait photographers. In many cases, acceptable tone/color reproduction characteristics can be achieved by applying a simple tone reproduction curve to the ERIMM RGB scene-exposure values. In this case, the ERIMM RGB to ROMM RGB transformation will involve only a simple one-dimensional LUT.

Once the image is in ROMM RGB, additional rendered-state image operations can be applied. For example, text annotations and a creative border could be added to the image, or the image could be composited with an image from a print scanner, etc. The final ROMM RGB image can then be printed by applying a ROMM RGB profile and an output profile for the particular output device.

4.8 Digital color management with JPEG 2000

Historically, many desktop imaging applications have been designed based on the assumption that the digital image stored in a file is ready to display directly on a CRT. This assumption has caused significant interoperability problems for applications that have attempted to store images with other color encodings. For example, if an application were to open a ROMM RGB image and send the color values directly to a video display, the image would appear very desaturated, because the image was encoded using a set of high-

© 2003 by CRC Press LLC
chroma primaries rather than video primaries. Special software must be used to open and/or color manage images stored in various color spaces, and, as a result, images stored in spaces other than video RGB cannot be used by a large number of applications. The overall situation has effectively made it impractical to use color spaces other than video RGB for most consumer applications.

JPEG 2000 is a new file storage format that has been recently standardized. One of the requirements that have been built into the format specification is that all JPEG 2000 compliant file readers must be able to properly decode an image stored in any color encoding specification that can be defined using a restricted class of ICC profiles. In particular, the supported ICC profile formats include any display-type profile that utilizes a LUT-matrix transformation to get to PCS XYZ. Both the (E)RIMM RGB and the ROMM RGB color encoding specifications can be represented using profiles that fall within this definition. As a result, images can be stored using these color encoding specifications without sacrificing interoperability. Applications designed to manipulate images in the color spaces of these encodings will be able to do so. Other applications can simply use the attached ICC profile to convert the image to a video RGB color space (e.g., sRGB) or to some other color space for which the application was designed.

4.9 Summary

In digital imaging systems, the principal role of color management is to transform image signals derived from one or more input devices to signals that are appropriate for a given output device. Digital color management can be relatively straightforward when applied to simpler systems, but it becomes quite complex when applied to systems having a variety of different input and output types.

The successful implementation of digital color management depends on a number of factors, including the use of appropriate device characterization methods and suitable mathematical techniques for forming and applying image-processing transformations. In addition, an appropriate color management paradigm must be determined for the particular system being developed. Virtually all current color-managed imaging systems are based on one of three basic paradigms. A “universal” paradigm, in which various input and output signal processing options are supported, has also been defined. Through the selection of appropriate options, systems based on this all-inclusive paradigm can be made to operate according to any of the three basic paradigms.

Successful color management also requires the use of an appropriate method for encoding color. Three basic color encoding methods were described in this chapter. Densitometric color encoding is based on input-image color measurements made according to any of various defined sets of spectral responsivities. Colorimetric color encoding is derived from measurements made according to the spectral responsivities of a standard human
observer. Color appearance encoding is an extension of basic colorimetric encoding. In this method, colorimetric values associated with one set of viewing conditions are transformed to determine a visually corresponding set of colorimetric values associated with another set of viewing conditions. The transformations account for differences in a number of factors — including absolute image luminance level, image surround, and the observer’s state of chromatic adaptation — that influence an observer’s perception of color.

In addition to an appropriate color encoding method, a properly designed data metric must also be used. A data metric defines the color space and numerical units in which encoded data are expressed. The combination of a color encoding method and data metric forms a complete color encoding specification. The selected color encoding specification must be consistent with the state of the image to be encoded. The image state is a function of how an image was captured and subsequently signal processed. Most digital images can be categorized into two types of image states: unrendered and rendered. Images in an unrendered state are directly related to the colorimetry of real or hypothetical original scenes. Images in a rendered state are encoded representations of the colorimetry of output images. Transformations beyond those based on color appearance alone are needed when an image is to be transformed from one image state to another. Image-state transformations are greatly facilitated by the use of appropriate color encoding specifications.

Eastman Kodak Company has developed a family of such specifications for use in the development of its digital imaging products. These specifications have been proposed for international standardization. Reference Input Medium Metric RGB (RIMM RGB) is designed for the manipulation, storage, and interchange of images from sources that naturally capture scene-referred (unrendered) image data. Reference Output Medium Metric RGB (ROMM RGB) serves a similar purpose for images from sources that produce images in an output-referred (rendered) image state. Images encoded in terms of RIMM RGB or ROMM RGB are fully compliant with the JPEG 2000 file storage format.

References

15. Interpretation of the PCS, appendix to Kodak ICC profile for CMYK (SWOP) input, ANSI CGATS/SC6 N 254, June 3, 1998.