Optimized Extended Gamut Color Encoding for Scene-Referred and Output-Referred Image States

Kevin E. Spaulding,▲ Geoffrey J. Woolfe,▲ and Edward J. Giorgianni

Eastman Kodak Company, Rochester, New York

Images can be broadly categorized as belonging to one of two types of image states: unrendered and rendered. Images in an unrendered image state are directly related to the colorimetry of a real or hypothetical original scene. Images in a rendered image state are representations of the colorimetry of an output image, such as a print or a CRT display. Over the years, many device-dependent and device-independent color encodings have been used to represent images in both of these image states. This has resulted in a host of interoperability problems between various systems. There would be a significant advantage to standardizing on a small number of color encodings for the purposes of storage, interchange and manipulation of digital images. While the recently agreed upon *sRGB* color encoding specification is one important step towards such standardization, there is also a need to define standard color encodings that are not limited by the color gamut of any specific device. This article will describe a family of new color encodings that have been developed to address this need. A new color encoding specification known as *Reference Output Medium Metric RGB (ROMM RGB)* is defined for representing rendered or output-referred images, and a companion color encoding specification, known as *Reference Input Medium Metric RGB (RIMM RGB)*, is defined for representing unrendered or scene-referred images.

Journal of Imaging Science and Technology 45: 418–426 (2001)

Introduction

Digital images are often encoded in terms of color spaces that are tied directly to the characteristics of actual input or output devices. Common examples of such color spaces are scanner RGB, video RGB, and printer CMY(K). However, such spaces generally are *device-dependent* in that their values can be associated with specific colorimetric values only in the context of the characteristics of the particular device on which the image is displayed or captured.

On the other hand, *device-independent* color spaces are generally meant to represent colorimetric values directly. Most often, these color spaces are based on the system of colorimetry developed by the Commission International de l'Eclairage (CIE). Examples of such color spaces include *CIE XYZ* tristimulus values and *CIELAB*. It should be noted that the specification of a color value, whether in a device-independent or device-dependent color space, does not fully specify *color appearance* unless the viewing conditions also are known. For example, two patches with identical colorimetric values can have very different color appearances, depending on the conditions under which they are viewed. Thus, the specification of image colorimetry alone does not unambiguously communicate the intended color appearance. To clearly communicate that appearance, it is also necessary to specify the viewing environment in which the image is intended to be viewed.

The fact that images exist in many different 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. This has led many people to advocate the use of a standard color encoding (or perhaps a small number of standard color encodings) for the storage, interchange and manipulation of digital images. Often, these proposals have involved specifying a particular output-device-dependent color space to be a "standard." Examples of such color spaces include *SWOP CMYK*[1] and *sRGB*.[2]

One significant problem with specifying an output-device-dependent color space as the standard is that to do so typically will limit the encodable color gamut and luminance dynamic range of images according to the capabilities of that specific output device. 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 would necessarily involve clipping many colors that could have been produced on a given hardcopy medium.

The International Color Consortium (ICC)[3] has defined a Profile Connection Space (PCS) that comprises a device-independent color encoding specification that can be used to explicitly specify the color of an image with respect to a reference viewing environment. *Device profiles* can be used in a color management system to relate the device-dependent code values of input images to the corresponding color values in the PCS, and from there, to the device-dependent output color values appropriate for a specific output device. It could be argued that the PCS could serve as the standard color encoding we are look-

Original manuscript received June 5, 2000

▲ IS&T Member

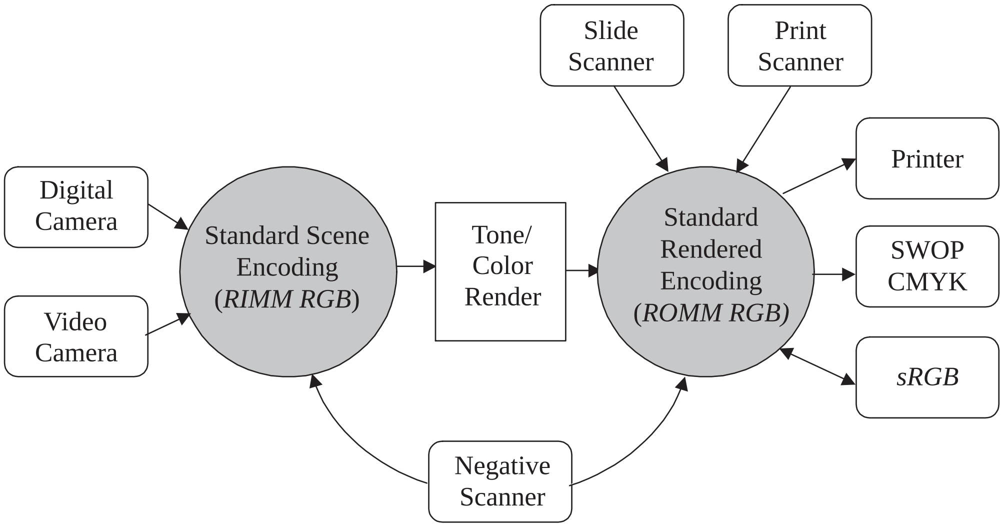

Figure 1. Image state diagram showing standard color encodings.

ing for. However, it was never intended that the PCS be used to directly store or manipulate images. Rather, it was simply intended to be a color space where profiles could be joined to form complete input–to–output color transforms. Neither the *CIELAB* nor the *XYZ* color encodings supported for the PCS is particularly well suited for many common kinds of image manipulations. Additionally, quantization errors that would be introduced by encoding images in PCS would be significantly larger than necessary because a large percentage of code value combinations would correspond to unrealizable colors. It is therefore desirable to define standard large-gamut color encodings that can be used for storing, interchanging and manipulating color images. This article will describe a new color space known as *Reference Output Medium Metric RGB* (*ROMM RGB*).[4] This color encoding is tightly coupled to the ICC PCS and is intended to be used for encoding *rendered output images* in a device-independent fashion.

Rendered output images should be distinguished from images that are intended to be an encoding of the colors of an *original scene*. 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 rendered image must compensate for differences between the scene and rendered image viewing conditions.[5,6] For example, rendered images generally are viewed at luminance levels much lower than those of typical outdoor scenes. As a consequence, an increase in the overall contrast of the rendered image usually is required in order to compensate for perceived losses in reproduced luminance and chrominance. Additionally, the rendering process should compensate for viewing flare associated with rendered-image viewing conditions.

In addition, psychological factors such as color memory and color preference must be considered in image rendering. For example, observers generally remember colors as being of higher purity than they really were, and they typically prefer skies and grass 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.

Finally, the tone/color reproduction process also must account for the fact that the dynamic range of a rendered image usually is substantially less than that of an original scene. It is therefore 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 image.

Because the colorimetry of scenes and their corresponding rendered images are intentionally and necessarily different, it would be ambiguous to try to represent images in both image states using the same color encoding. For example, if someone were to send the *CIELAB* values and viewing environment description for a particular image with no information about whether the color values were scene color values or rendered image color values, the recipient would not know what to do with the image in order to make the best print. If the color values were rendered color values, it would simply be necessary to determine the device code values needed to produce the equivalent color appearance in the output-viewing environment. However, if the color values corresponded to original scene color values, it would be necessary to modify the image colorimetry by applying appropriate tone/color reproduction aims before printing the image.

Due to these and other factors, color encodings such as *sRGB* or *ROMM RGB* that are intended for encoding rendered output images are inappropriate for use in encoding unrendered scene-referred images. Rather, a color encoding that is directly related to the color of an original scene should be used. Accordingly, a companion to the *ROMM RGB* color encoding specification, known as *Reference Input Medium Metric (RIMM RGB)*, has also been defined. This encoding is intended to represent original scene color appearance. The *RIMM RGB* color encoding not only provides extra dynamic range necessary for the encoding of scene information, it is specifically defined to be an encoding of images in a scene image state.

A diagram illustrating how these standard color encodings can be used as the basis for a general imaging system architecture is shown in Fig. 1. *ROMM RGB* is ideal for the manipulation, storage, and interchange of images from sources such as print scanners and other devices that produce images in a rendered *output-re-*

ferred image state. Likewise, *RIMM RGB* serves a similar purpose for images from sources such as digital cameras that naturally capture *scene-referred* image data. It should be noted that while digital camera sensors fundamentally capture a scene-referred image, many consumer digital cameras only provide access to a fully rendered output-referred image in a color space such as *sRGB*. Therefore, only cameras that provide access to the unrendered image data can be used as sources of scene-referred images.

A color negative scanner can be used to produce either scene-state or rendered-state images. The wide dynamic range of color negative film makes it an ideal source of scene-referred image data. It is also possible to produce rendered image data from color negative film by determining the color of a print that could be made from the negative.

Before sending images 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 flexible than a print, an image in a scene-state will be much more flexible than one in a rendered-state. Therefore, it will be desirable in many imaging systems to delay any conversion to a rendered-state until the time when an output image is generated. For example, consider the case where a color negative contains a scene with a brightly lit background and a dimly lit foreground. The negative generally will contain useful image information in both of these regions, and that information can be used to make a print that is properly exposed for either one region or the other. In order to make a rendered-state image from this negative, it is necessary to make a choice about which part of the image is important. In a conventional photographic system, this is done when the negative is printed optically. However, scene-referred images can retain all of the information on the negative, which allows the delay of any decision as to how the scene is to be rendered. This retains the maximum amount of flexibility in the system.

Selection of Color Space

It is 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 output-referred *ROMM RGB* image is created from an original scene-referred image encoded in *RIMM RGB*. This is best achieved by basing the two encodings on the same color space. The criteria that were used to select this color space include the following:

- Direct relationship to the color appearance of the scene/image
- Color gamut large enough to encompass most real-world surface colors
- Efficient encoding of color information to minimize quantization artifacts
- Simple transformation to/from ICC PCS
- Simple transformation to/from video RGB (e.g., *sRGB*)
- Well-suited for application of common image manipulations such as tone scale modifications, color-balance adjustments, sharpening, etc.
- Compatible with established imaging-system workflows

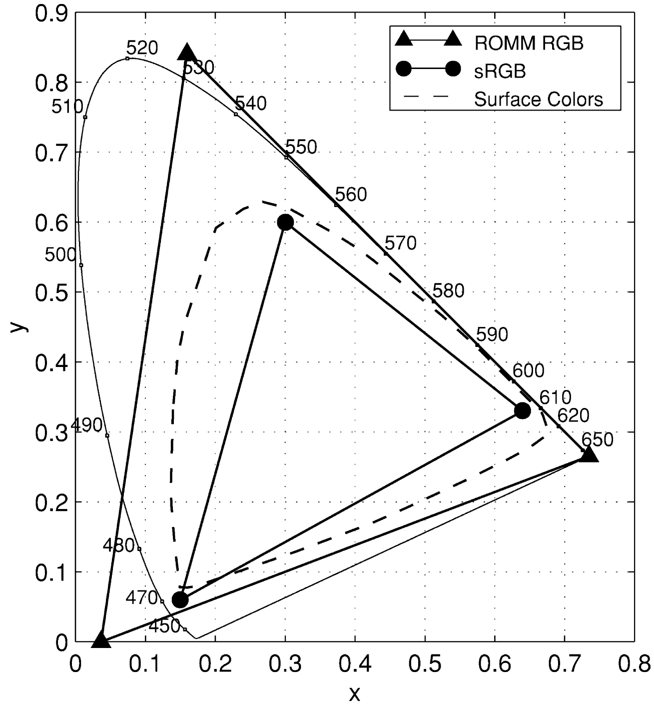

Figure 2. Comparison of primaries in *x–y* chromaticity coordinates

An additive RGB color space with an appropriately selected set of "wide-RGB" primaries is ideal for satisfying all of these criteria. When images are encoded using any such set of primaries, there is a direct and simple relationship to scene/image colorimetry because the primaries are linear transformations of the *CIE XYZ* primaries. Wide-RGB color spaces have the additional advantage that simple LUT–matrix–LUT transformations can be used to convert to/from additive color spaces such as *PCS XYZ*, video RGB (*sRGB*) and digital camera RGB.

Two of the criteria that affect the selection of particular RGB primaries are somewhat conflicting. First, their chromaticities should define a color gamut sufficiently large to encompass colors likely to be found in real scenes/images. At the same time, their use should result in efficient digital encodings that minimize quantization errors.

Increasing the gamut to encompass more colors can only be achieved by trading off against correspondingly larger quantization errors. If the chromaticities of the primaries chosen include the maximum possible color gamut (for example, choosing the *XYZ* primaries would encompass the chromaticities of the entire spectrum locus), a significant fraction of the color space would correspond to imaginary colors and to colors that would not be commonly encountered in real images. Therefore, in any encoding using such a color space, there would be large numbers of "wasted" code value combinations that would never 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 "big enough" but not "too big."

Figure 2 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* and *ROMM RGB* color encodings is that they be well suited for application of common image manipulations. Many types of common image manipulations 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 means of applying a nonlinear tone scale transformation to the individual channels of an RGB image in a scene image state. A well-designed transformation of this type will produce several desirable results:

- The luminance and color contrast of mid-tones will be increased, and the highlights and shadows will be smoothly compressed to fit within the dynamic range of the output medium.
- The chroma of in-gamut colors will be increased.
- Out-of-gamut colors will be remapped 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 relative ratios of the red, green, and blue channel data. This can lead to hue shifts, particularly for high chroma colors. Hue shifts are particularly problematic when they occur in a natural chroma gradient, having constant hue and saturation, within an image. Such gradients tend to occur when rounded surfaces are illuminated by a moderately directional light source. In such situations, chroma increases with distance from the specular highlight and then decreases again as the shadows deepen.

Such hue shifts can never be completely eliminated, so the objective when optimizing the location of the primaries was to eliminate or minimize objectionable hue shifts at the expense of less noticeable or less 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.

Hue shifts introduced by the application of nonlinear channel-independent transformations 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 flesh and dark flesh. Hue shifts in flesh tones and yellows, particularly in the direction of green, are considered to be the most objectionable. These hue shifts are most strongly affected by the location of the blue primary. Other colors that were considered to be particularly important during the optimization process were blues and reds.

There is a tradeoff 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 so as 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, starting with version 5.0, Adobe Photoshop software has incorporated the concept of a "working color space" that is different from the monitor preview color space. This is very consistent with the notion of storing/manipulating images in a wide-RGB color space. Adobe Photoshop V5.0 software places a constraint on the definition of valid working color spaces that will require the primaries to have all positive x–y–z chromaticity values. This condition is satisfied for the *ROMM RGB* primaries. For more information about using *ROMM RGB* as an Adobe Photoshop working space, see the white paper posted at www.kodak.com (search on "ROMM"). Because Photoshop software operates within a rendered-image paradigm, it is generally inappropriate to use *RIMM RGB* as a Photoshop 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 hue shifts associated with the selected *RIMM/ROMM RGB* primaries are shown in Fig. 3(a). This plot shows a series of line segments connecting a^*, b^* values before and after a nonlinear tone scale was applied to a chroma series in each of the 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. The corresponding hue shifts associated with an alternate set of wide-RGB primaries are shown in Fig. 3(b). It can be seen that the hue shifts for these primaries are significantly larger than those of the *RIMM/ROMM RGB* primaries in almost every color region.

Definition of *ROMM RGB*

In addition to defining a color space, it is also necessary to specify an intended viewing environment in order to unambiguously define 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,[7] and is defined to have the following characteristics:

- The luminance level for the observer adaptive white is 160 cd/m^2. (This luminance level is approximately equivalent to that of a perfect white Lambertian reflector illuminated with 500 lux as specified in ISO print viewing standard 3664,[8] and also in the ICC.1:2001 color profile specification.[3])
- The observer adaptive white point has the chromaticity values of CIE Standard Illuminant D_{50} (x = 0.3457, y = 0.3585).

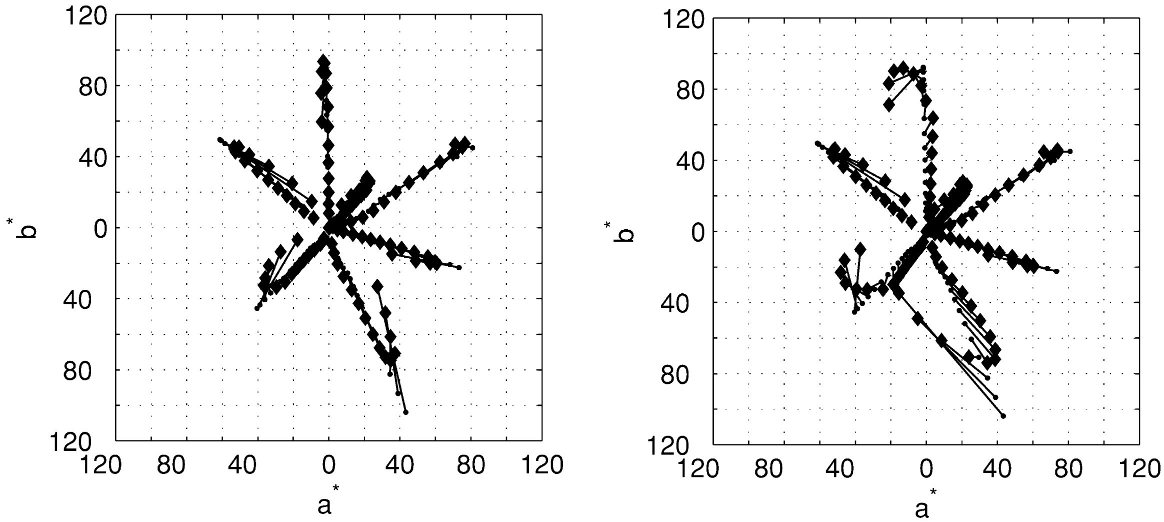

Figure 3. Hue shifts resulting from a typical nonlinear rendering transform for: a) the *RIMM/ROMM RGB* primaries, and b) an alternate set of wide-RGB primaries.

TABLE I. Primaries/White Point for Reference Imaging Medium

Color	*x*	*y*
Red	0.7347	0.2653
Green	0.1596	0.8404
Blue	0.0366	0.0001
White	0.3457	0.3585

- The viewing surround is average. (In other words, the overall luminance level and chrominance of the surround is assumed to be similar to that of the image.)
- There is 0.75% viewing flare, referenced to the observer adaptive white.
- 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:

- Reference primaries defined by the CIE chromaticity values given in Table I.
- Equal amounts of the reference primaries produce a neutral with the chromaticity of D_{50}. ($x = 0.3457$, $y = 0.3585$)
- The capability of producing a white with a luminance factor of of $F_W = 0.89$, and a black with a luminance factor of $F_K = 0.0030911$

The conversion of the PCS *XYZ* tristimulus values to *ROMM RGB* values can be performed by a matrix operation, followed by a set of 1–D functions. This is equivalent to the operations associated with a basic CRT profile. This means that *ROMM RGB* can be used conveniently in a system employing ICC profiles using an appropriately designed monitor profile.

Most current implementations of the ICC PCS incorporate the concept of a reference medium where the black point of the reference medium is mapped to $Y_{PCS} = 0$, and the white point of the reference medium is mapped to $Y_{PCS} = 1.0$.[9] Therefore, to relate actual CIE image colorimetry to PCS *XYZ* values, an appropriate normalizing transformation is required:

$$\begin{aligned} X_{PCS} &= \frac{(X - X_K)}{(X_W - X_K)} \frac{X_W}{Y_W} \\ Y_{PCS} &= \frac{(Y - Y_K)}{(Y_W - Y_K)} \\ Z_{PCS} &= \frac{(Z - Z_K)}{(Z_W - Z_K)} \frac{Z_W}{Y_W} \end{aligned}, \qquad (1)$$

where X, Y and Z are the CIE image tristimulus values, X_{PCS}, Y_{PCS} and Z_{PCS} are the PCS tristimulus values, X_W, Y_W and Z_W are the tristimulus value 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 $Z_0 = 82.49$), and X_K, Y_K and Z_K are the tristimulus value of the reference medium black point ($X_K = F_K X_0 = 0.2980$, $Y_K = F_K Y_0 = 0.3091$ and $Z_K = F_K Z_0 = 0.2550$)

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

$$\begin{bmatrix} R_{ROMM} \\ G_{ROMM} \\ B_{ROMM} \end{bmatrix} = \begin{bmatrix} 1.3460 & -0.2556 & -0.0511 \\ -0.5446 & 1.5082 & 0.0205 \\ 0.0000 & 0.0000 & 1.2123 \end{bmatrix} \begin{bmatrix} X_{PCS} \\ Y_{PCS} \\ Z_{PCS} \end{bmatrix}. \qquad (2)$$

As required by the definition of *ROMM RGB*, this matrix will map image tristimulus values with the chromaticity of D_{50} 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.

Nonlinear Encoding of *ROMM RGB*. A nonlinear quantization function is used to store the *ROMM RGB* values in an integer form. A simple gamma function

TABLE II. Sample Neutral Patch Encodings

Y_{PCS}	ROMM8 RGB	ROMM12 RGB	ROMM16 RGB
0.00	0	0	0
0.001	4	66	1049
0.01	20	317	5074
0.10	71	1139	18236
0.18	98	1579	25278
0.35	142	2285	36574
0.50	174	2786	44590
0.75	217	3490	55855
1.00	255	4095	65535

nonlinearity incorporating a slope limit at the dark end of the intensity scale is defined for this purpose:

$$C'_{ROMM} = \begin{cases} 0; & C_{ROMM} < 0.0 \\ 16\ C_{ROMM}\ I_{\max}; & 0.0 \le C_{ROMM} < E_t \\ (C_{ROMM})^{1/1.8} I_{\max}; & E_t \le C_{ROMM} < 1.0 \\ I_{\max}; & C_{ROMM} \ge 1.0 \end{cases}, \quad (3)$$

where C is either R, G, or B, $I_{\max}$ is the maximum integer value used for the nonlinear encoding, and

$$E_t = 16^{1.8/(1-1.8)} = 0.001953 \cdot \quad (4)$$

For the baseline 8–bit configuration, $I_{\max}$ is equal to 255. The linear segment of the nonlinearity is used to impose a slope limit to minimize reversibility problems because of the infinite slope of the gamma function at the zero point. 12– and 16–bit versions of *ROMM RGB* are also defined. The only difference is that the value of $I_{\max}$ is set to 4095 or 65,535, respectively. In cases where it is necessary to identify a specific precision level, the notation *ROMM8 RGB*, *ROMM12 RGB* and *ROMM16 RGB* is used. Table II shows some sample encodings for a series of neutral patches of specified Y_{PCS}.

Inverse of *ROMM RGB* Encoding. It is also necessary to define an inverse transformation to convert *ROMM RGB* values back to rendered image PCS values. This can be done by simply inverting the nonlinear function given in Eq. 3 and then applying the inverse of the matrix given in Eq. 2.

The first step is to undo the nonlinear encoding of the *ROMM RGB* values. This will convert the signals back to linear *ROMM RGB* values:

$$C_{ROMM} = \begin{cases} \dfrac{C'_{ROMM}}{16\ I_{\max}}; & 0.0 \le C'_{ROMM} < 16\ E_t\ I_{\max} \\ \left(\dfrac{C'_{ROMM}}{I_{\max}}\right)^{1.8}; & 16\ E_t\ I_{\max} \le C'_{ROMM} \le I_{\max} \end{cases}, \quad (5)$$

where, as before, C is either R, G or B.

To convert the *ROMM RGB* values to the corresponding D_{50} PCS tristimulus values, it is simply necessary to multiply by the inverse of the matrix given in Eq. 2.

$$\begin{bmatrix} X_{PCS} \\ Y_{PCS} \\ Z_{PCS} \end{bmatrix} = \begin{bmatrix} 0.7977 & 0.1352 & 0.0313 \\ 0.2880 & 0.7119 & 0.0001 \\ 0.0000 & 0.0000 & 0.8249 \end{bmatrix} \begin{bmatrix} R_{ROMM} \\ G_{ROMM} \\ B_{ROMM} \end{bmatrix} \quad (6)$$

As expected, when this matrix is applied to linear *ROMM RGB* values that are equal, tristimulus values with the chromaticity of D_{50} are obtained.

Conversion Between *ROMM RGB* and *sRGB*. In many cases, it will be necessary to convert *ROMM RGB* values to a video RGB representation for display on a CRT. This can be accomplished by combining the *ROMM RGB* to PCS transformation described in the previous section with an appropriate PCS to video RGB transformation for the CRT. Consider the special case of a CRT that responds according to the *sRGB* specification.[2] Because *sRGB* is defined using a D_{65} white point, and the PCS is defined using a D_{50} white point, the first step in the conversion of PCS values to *sRGB* values must be a D_{50}–to–D_{65} chromatic adaptation. This can be accomplished using a simple von Kries transformation as follows.

$$\begin{bmatrix} X_{D65} \\ Y_{D65} \\ Z_{D65} \end{bmatrix} = \begin{bmatrix} 0.9845 & -0.0547 & 0.0678 \\ -0.0060 & 1.0048 & 0.0012 \\ 0.0000 & 0.0000 & 1.3200 \end{bmatrix} \begin{bmatrix} X_{D50} \\ Y_{D50} \\ Z_{D50} \end{bmatrix} \quad (7)$$

The Hunt–Pointer–Estevez cone primaries[10] were used to derive this chromatic adaptation transform. Alternatively, other primaries or chromatic adaptation transforms could also be used.

The *sRGB* color space is defined using the phosphor primaries associated with Rec. 709. The conversion from D_{65} tristimulus values to the linear RGB values associated with these primaries is given by the following inverse phosphor matrix:

$$\begin{bmatrix} R_S \\ G_S \\ B_S \end{bmatrix} = \begin{bmatrix} 3.2406 & -1.5372 & -0.4986 \\ -0.9689 & 1.8758 & 0.0415 \\ 0.0557 & -0.2040 & 1.0570 \end{bmatrix} \begin{bmatrix} X_{D65} \\ Y_{D65} \\ Z_{D65} \end{bmatrix}. \quad (8)$$

Finally, the desired *sRGB* code values can be computed by applying the appropriate nonlinearity and integerizing:

$$C'_S = \begin{cases} 255(12.92 C_S); & C_S \le 0.0031308 \\ 255(1.055 C_S^{1/2.4} - 0.055); & C_S > 0.0031308 \end{cases}, \quad (9)$$

where C is either R, G, or B. It should be noted that *sRGB* is not defined with respect to the same reference medium/viewing environment as the ICC PCS. Strictly, this would imply that the image should be re-rendered to account for these differences. However, in practice, most ICC profiles used for *sRGB* simply map the PCS white/black points to the corresponding *sRGB* white/black points as is described in the transformation described here.

Conversion from *ROMM RGB* values to the *sRGB* code values can therefore be accomplished by applying the inverse *ROMM RGB* nonlinearity given in Eq. 5, followed by the matrices given in Eqs. 6, 7 and 8, followed by the *sRGB* nonlinearity given in Eq. 9. The three sequential matrix operations can be combined by cascading the matrices together to form the following single matrix:

$$\begin{bmatrix} R_S \\ G_S \\ B_S \end{bmatrix} = \begin{bmatrix} 2.0564 & -0.7932 & -0.2632 \\ -0.2118 & 1.2490 & -0.0372 \\ -0.0152 & -0.1405 & 1.1556 \end{bmatrix} \begin{bmatrix} R_{ROMM} \\ G_{ROMM} \\ B_{ROMM} \end{bmatrix}. \quad (10)$$

Thus, the transformation from *ROMM RGB* to *sRGB* can be implemented with a simple LUT–MAT–LUT chain.

It should be noted that not all colors that can be encoded in *ROMM RGB* will be within the *sRGB* color gamut. As a result, it will be necessary to perform some sort of gamut mapping to limit all of the colors to the appropriate gamut. The simplest form of gamut mapping is just to clip all of the linear *sRGB* values to the range 0.0 to 1.0 before applying the nonlinearity of Eq. 9. However, this approach can result in noticeable hue shifts in certain cases. Superior results can be obtained using more sophisticated gamut-mapping strategies.

The conversion from *sRGB* back to *ROMM RGB* is simply an inverse of the steps that were just discussed. First, the inverse of the *sRGB* nonlinearity given in Eq. 9 is applied to determine the linear RGB_s values:

$$C_S = \begin{cases} \dfrac{\left(C'_S/255\right)}{12.92}; & C'_S \le 0.04045 \times 255 \\ \left(\dfrac{\left(C'_S/255\right)+0.055}{1.055}\right)^{2.4}; & C'_S > 0.04045 \times 255 \end{cases} . \quad (11)$$

Next, the inverse of the matrix in Eq. 10 is used to compute the linear RGB_{ROMM} values,

$$\begin{bmatrix} R_{ROMM} \\ G_{ROMM} \\ B_{ROMM} \end{bmatrix} = \begin{bmatrix} 0.5230 & 0.3468 & 0.1303 \\ 0.0892 & 0.8627 & 0.0481 \\ 0.0177 & 0.1095 & 0.8729 \end{bmatrix} \begin{bmatrix} R_S \\ G_S \\ B_S \end{bmatrix} . \quad (12)$$

Finally, the *ROMM RGB* nonlinearity given in Eq. 3 is applied to determine the *ROMM RGB* values.

As noted above, many colors that can be represented in *ROMM RGB* color encoding are outside the gamut of *sRGB*. As a result, the process of mapping an image from *ROMM RGB* to *sRGB* and back again generally is not lossless. Therefore, it should be emphasized that, whenever possible, a video RGB color space should not be used as an intermediate color space during the process of manipulating a *ROMM RGB* image. Rather, the image manipulations should be applied to the *ROMM RGB* image directly, and the *ROMM RGB* to *sRGB* transformation should be used to provide an image for video preview purposes only. It should be remembered that when previewing extended gamut images on a video display, colors outside the video gamut will not be accurately portrayed. Therefore appropriate caution should be exercised when applying modifications to extended gamut images.

On the other hand, if an original image is in a video RGB color space, it should be possible to convert the image to *ROMM RGB* for manipulation purposes, and then convert it back to the video RGB color space again with only minimal losses due to quantization effects. These effects can be reduced to negligible levels by using the 12-bit/channel or 16-bit/channel versions of *ROMM RGB*. However, it should be noted that if the manipulation process creates any color values that are outside the video RGB gamut, these values will be clipped when the processed image is converted back to the original color space.

Definition of *RIMM 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-RGB color space defined by the primaries and white point given in Table I. 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/m^2.
- The observer adaptive white point has the chromaticity values of CIE Standard Illuminant D_{50}: $x = 0.3457$, $y = 0.3585$. (In cases where the chromaticity of the observer adaptive white for an actual scene differs from that of the reference conditions, an appropriate chromatic adaptation transformation must be applied to the image data.)
- Viewing surround is average. (In other words, the overall luminance level and chrominance of the surround is assumed to be 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.

***RIMM RGB* Conversion Matrix.** Because *ROMM RGB* and *RIMM RGB* use a common color space, the conversion from the scene tristimulus values to the corresponding linear *RIMM RGB* values can be accomplished using the same conversion matrix that was given in Eq. 2, except that the input tristimulus values are the scene *XYZ* values rather than the PCS *XYZ* values.

$$\begin{bmatrix} R_{RIMM} \\ G_{RIMM} \\ B_{RIMM} \end{bmatrix} = \begin{bmatrix} 1.3460 & -0.2556 & -0.0511 \\ -0.5446 & 1.5082 & 0.0205 \\ 0.0000 & 0.0000 & 1.2123 \end{bmatrix} \begin{bmatrix} X_{D50} \\ Y_{D50} \\ Z_{D50} \end{bmatrix} . \quad (13)$$

Note that 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 $Y_{D50} = 1.0$.

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. In particular, the encoding must allow for scene luminance values larger than $Y_{D50} = 1.0$ because typical scenes often contain important information at levels well above the luminance of a correctly exposed perfect white diffuser. These extended luminance levels can correspond to specular highlights, areas of the scene that are illuminated more brightly than the main subject area, fluorescent colors, and self-luminous objects within the scene. The *RIMM RGB* nonlinearity is based on that specified by Recommendation ITU–R BT.709 (Rec. 709).[11] (This recommendation was formerly known as CCIR 709.) This is the same nonlinearity used in the *PhotoYCC* Color Space encoding implemented in the *Kodak Photo CD* System,[6] and is given by Eq. 14.

$$C'_{RIMM} = \begin{cases} 0; & C_{RIMM} < 0.0 \\ \left(\dfrac{I_{\max}}{V_{clip}}\right) 4.5\ C_{RIMM}; & 0.0 \le C_{RIMM} < 0.018 \\ \left(\dfrac{I_{\max}}{V_{clip}}\right) \left(1.099 C_{RIMM}{}^{0.45} - 0.099\right); & 0.018 \le C_{RIMM} < E_{clip} \\ I_{\max} & C_{RIMM} \ge E_{clip} \end{cases}, \tag{14}$$

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

$$V_{clip} = 1.099 E_{clip}{}^{0.45} - 0.099 = 1.402\ . \tag{15}$$

For the baseline 8–bit/channel *RIMM RGB* configuration, $I_{\max}$ is 255. In some applications, it may be desirable to use a higher bit precision version of *RIMM RGB* to minimize any quantization errors. 12– and 16–bit/channel versions of *RIMM RGB* are also defined. The only difference is that the value of $I_{\max}$ is set to 4095 or 65,535, respectively. In cases in which it is necessary to identify a specific precision level, the notation *RIMM8 RGB*, *RIMM12 RGB* and *RIMM16 RGB* is used.

Inverse Encoding for *RIMM RGB*. To convert from *RIMM RGB* back to the corresponding scene colorimetry, it is only necessary to invert the nonlinear encoding as shown in Eq. 16 (below), and then apply the same inverse matrix that was given for *ROMM RGB* in Eq. 6.

***ERIMM RGB* Color Encoding.** The *RIMM RGB* color space is defined to have a luminance dynamic range that can encode information up to 200% of the exposure value associated with a normally exposed perfect (100%) diffuse white reflector in the scene. This should be adequate for many applications such as digital cameras. However, for some applications, most notably scanned photographic negatives, this luminance dynamic range is insufficient to encode the full range of captured scene information. For these cases, a variation of the *RIMM RGB* color space is defined, referred to as *Extended Reference Input Medium Metric RGB (ERIMM RGB)*.

As with *RIMM RGB*, *ERIMM RGB* is directly related to the colorimetry of an original scene. The nonlinear encoding function is the only encoding step that is altered. For *ERIMM RGB*, it is desirable to increase both the maximum scene exposure 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 exposure value that can be accurately represented. In order to satisfy both the extended luminance dynamic range and the reduced quantization interval requirements simultaneously, it is necessary to use a higher minimum bit precision for *ERIMM RGB*. A minimum of 12–bits/color channel is recommended in this case.

Nonlinear Encoding for *ERIMM RGB*. A modified logarithmic encoding is used for *ERIMM RGB*. A linear segment is included for the very lowest exposure values to overcome the non-invertibility of the logarithmic encoding at the dark end of the tone 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 Eq. 17 (shown below), where C is either R, G, or B; $I_{\max}$ is the maximum integer value used for the nonlinear encoding; $E_{clip} = 10^{2.5} = 316.23$ is the upper exposure limit that gets mapped to $I_{\max}$; and $E_t = e/1000 = 0.00271828$ is the breakpoint between the linear and logarithmic segments, e being the base of the natural logarithm. For a 12-bit encoding, $I_{\max}$ is 4095, and for a 16-bit encoding $I_{\max}$ is 65535. In cases in which it is necessary to identify a

$$C_{RIMM} = \begin{cases} \dfrac{V_{clip} C'_{RIMM}}{4.5\ I_{\max}}; & 0 \le C'_{RIMM} < \dfrac{0.081\, I_{\max}}{V_{clip}} \\ \left(\dfrac{\dfrac{V_{clip} C'_{RIMM}}{I_{\max}} + 0.099}{1.099}\right)^{1/0.45}; & \dfrac{0.081\, I_{\max}}{V_{clip}} \le C'_{RIMM} < I_{\max} \end{cases}, \tag{16}$$

$$C'_{ERIMM} = \begin{cases} 0; & C_{RIMM} \le 0 \\ \left(\dfrac{0.0789626}{E_t}\right) C_{RIMM}\ I_{\max}; & 0 < C_{RIMM} \le E_t \\ \left(\dfrac{\log C_{RIMM} + 3.0}{5.5}\right) I_{\max}; & E_t < C_{RIMM} \le E_{clip} \\ I_{\max}; & C_{RIMM} > E_{clip} \end{cases}, \tag{17}$$

$$C_{ERIMM} = \begin{cases} \left(\dfrac{C'_{ERIMM}\, E_t}{0.0789626\, I_{\max}} \right); & 0 < C'_{ERIMM} \le 0.0789626\, I_{\max} \\ \text{anti}\log\left[\left(\dfrac{5.5\, C'_{ERIMM}}{I_{\max}} \right) - 3.0 \right]; & 0.0789626\, I_{\max} < C'_{ERIMM} \le I_{\max} \end{cases} \tag{18}$$

specific precision level, the notation *ERIMM12 RGB* and *ERIMM16 RGB* is used.

To compute *ERIMM RGB* values, Eq. 17 should be used in place of Eq. 14 in the procedure described above for determining *RIMM RGB* values. Examples of *RIMM RGB* and *ERIMM RGB* encodings for neutral patches at different scene exposure levels are shown in Table III. It can be seen that the range of exposures that can be represented in *ERIMM RGB* is extended relative to *RIMM RGB*.

Inverse Encoding for *ERIMM RGB*. The nonlinear function given in Eq. 17 can be inverted to determine an inverse *ERIMM RGB* encoding function as shown above in Eq. 18.

TABLE III. Sample Scene Exposure Encodings

Relative Exposure	Rel. Log Exposure	*RIMM8 RGB*	*RIMM12 RGB*	*ERIMM12 RGB*
0.001	–3.00	1	13	119
0.01	–2.00	8	131	745
0.10	–1.00	53	849	1489
0.18	–0.75	74	1194	1679
1.00	0.00	182	2920	2234
2.00	0.30	255	4095	2458
8.00	0.90	NA	NA	2906
32.00	1.50	NA	NA	3354
316.23	2.50	NA	NA	4095

Conclusion

Most images can be classified into one of two different image states: a rendered *output-referred* image state, or a *scene-referred* image state. A family of large-gamut color encoding specifications, based on a wide-RGB color space having optimized color primaries, has been defined for the storage, interchange and manipulation of images in these different image states. *Reference Output Medium Metric RGB (ROMM RGB)* is a large-gamut device-independent color encoding designed to be used for rendered images. It is tightly coupled to the ICC PCS, and it is compatible for use as an Adobe Photoshop working color space. *ROMM RGB* is associated with a specified encoding reference viewing environment and imaging medium, thereby enabling unambiguous communication of image color appearance and enhancing interoperability. *Reference Input Medium Metric RGB (RIMM RGB)* is based on the same color space as *ROMM RGB* and is designed for encoding the color appearance of unrendered scenes. It is associated with a set of encoding reference viewing conditions typical of outdoor scenes. An extended dynamic range version of *RIMM RGB,* known as *ERIMM RGB*, also has been defined. This color encoding is particularly well suited for encoding images from high-dynamic-range image sources such as color photographic negative films. The fact that these color encoding specifications are based on the same wide-RGB color space facilitates the development of common image-processing algorithms and simplifies the transformations between the different color encodings.

References

1. *Graphic Technology—Color Characterization Data for Type 1 Printing*, ANSI/CGATS TR 001 (1995).
2. *Multimedia Systems and Equipment—Colour Measurement and Management – Part 2–1: Colour Management—Default RGB Colour Space – sRGB*, IEC 61966–2–1 (1999).
3. International Color Consortium, *File Format for Color Profiles*, ICC.1:2001.
4. *Photography—Electronic still picture imaging—Reference Output Medium Metric RGB Color encoding: ROMM–RGB*, PIMA 7666:2001.
5. R. W. G. Hunt, *The Reproduction of Colour*, 5th ed., Fountain Press, London, 1995.
6. E. J. Giorgianni and T. E. Madden, *Digital Color Management: Encoding Solutions*, Addison-Wesley, Reading, MA, 1998.
7. Lars Borg, *The Profile Connection Space, ICC Votable Proposal Submission 19.10*, June 8, 2000.
8. *Viewing conditions – For graphic technology and photography*, ISO 3664:2000.
9. “Interpretation of the PCS,” appendix to *Kodak ICC Profile for CMYK (SWOP) Input*, ANSI CGATS/SC6 N 254, June 3, 1998
10. M. D. Fairchild, *Color Appearance Models*, Addison Wesley, Reading, MA, 1998, p. 250.
11. Basic parameter values for the HDTV standard for the studio and for international programme exchange, *Recommendation ITU–R BT.709* (formerly *CCIR Recommendation 709*).