A Spectral Gamut-Mapping Environment with Rendering Parameter Feedback

Ian E. Bell[1] and Simon K. Alexander[2]

[1]Ian Bell Consulting {iebell@sympatico.ca} [2]University of Waterloo {sk2alexa@uwaterloo.ca}

Abstract

This paper proposes a prototypical environment for gamut mapping in spectral space. Images are rendered in terms of light and material parameters by a symbolic ray tracer, and the parameter ranges are adjusted, without re-rendering, to bring the image into the output device's spectral gamut. There is a growing disparity between the high dynamic range images produced by spectral renderers and the limitations of display gamuts and low-dimensional colour management standards. While in rendering tone mapping has helped compress luminance ranges, and in colour science 3D gamut mapping has helped compress chrominance ranges, only high-dimensional spectral methods will fully bridge the gap. This paper's environment for gamuts in spectral space is a step toward spectral gamut mapping, which we demonstrate by using the ray tracer to predict feasible ranges of rendering parameters for an in-gamut image. The environment can be easily extended to support interactive or automatic image correction, and more sophisticated rendering and gamut-mapping methods of arbitrary dimensionality.

Categories and Subject Descriptors (according to ACM CCS): I.3.7 [Computer Graphics]: Color, shading, shadowing and texture

1. Introduction

Rendering algorithms have reached the heights of sophistication in the last decade. As algorithms can reproduce real-world radiances with great accuracy, image gamuts routinely exceed display capabilities. Graphics hardware has also advanced, however spectral nuances beyond RGB colour [WTP01] are not well supported by current hardware. Rendering research has therefore turned to algorithms such as *tone mapping* [THG99, DCWP02] for compressing image colours in a spatially-varying manner similar to the retinal response of the human visual system [Hun98].

Colour science has followed its own path, solving problems in device profiling and creating standards for colour management and colour appearance [Hun02]. Recent work involves spectral quantities in extending colour models [IRB02] and reproducing images between devices while preserving colour appearance [Hun02, WHD03]. Gamut-mapping algorithms have been refined [HB00, BF00, ML01], though they generally remain in 3D spaces and avoid spatial variation.

These two trends in research leave a conspicuous gap. Tone mapping, while solving luminance compression, does not fully address the remaining dimensions of spectral colour. Gamut-mapping is limited to 3D colour and generally does not vary spatially. Despite these limitations,colour device gamuts are expanding in dimensionality: Seven- and eight-ink printers are widely available, and products like Sony's DSC-F828 four-channel RGBE camera pave the way for low-end multispectral capture. What is needed now is an environment bridging this gap, supporting high-dimensional spectral gamut-mapping tools that can vary spatially while respecting colour appearance. This paper is one step toward such an environment.

Both physically-based rendering and interactive spectral modelling can benefit immediately from an environment giving feedback from out-of-gamut spectral regions. Our symbolic ray tracer allows light and material parameters to be clipped in spectral space, giving feasible range feedback for those parameters, which can be adjusted without re-rendering the image. If the image is to be printed, this is a great improvement over parameter adjustment based on monitor RGB. Also, in calculations limited to 3D colour, illuminant and sensor assumptions must be made, restricting later spectral gamut mapping options. Our spectral gamut space scales to arbitrary dimensions, providing a gamut mapping environment for future multiple-illuminant or multiple-sensor problems.

2. Conventional Device Gamuts

Conventional colour management requires *profiles* describing gamuts of both input and output devices [RFJW00]. Although input devices like multispectral cameras may also have high-dimensional gamuts [MCSP02], we will mostly be concerned with output device gamuts in this paper. Output device gamuts are intuitively simple: the colours displayed for all possible device inputs form the device gamut. As device behaviour is smoothly varying, the continuous extension of the discrete colours is more convenient to manipulate. Monitors, after gamma correction, may typically be characterized by a matrix, giving a distinctive parallelepiped gamut if plotted in a linear space. For printers, the subtractive nature of colour combination gives gamuts that, although generally smooth, can be non-convex and irregular in traditional colour spaces. The variety and irregularity of printer gamuts makes for difficult gamut mapping, and suggests that perfecting an image on a monitor may be counterproductive, if the final image destination is print. To make precise our concept of spectral gamuts, we first follow the gamut-mapping guidelines of Stone, Cowan, and Beatty [SCB88], who stress the centrality of the neutral axis.

2.1. Spectral Device Gamuts

Spectral gamuts of CRT monitors are similar to standard 3D gamuts, as no further dimensionality can be derived from a linear approach. Although our methods remain applicable, consider the more challenging question of a spectral printer gamut. Of several approaches for spectral gamuts [Pae94, Len02a, RO03], we build on the Paeth method for linear reflectance spaces [Pee93, Pae94]. Natural reflectances contain mostly low-frequency information [Har], and Principal Components Analysis (PCA) is traditional for determining the most significant basis reflectances [Len02b]. Applying PCA to many printer reflectances yields a fairly uniform first basis reflectance; we may substitute an ideal uniform reflectance with no loss in accuracy, provided all dimensions are retained. A Gram-Schmidt procedure [GL89] reveals subsequent basis reflectances. In this fashion, a standard reflectance basis and coordinate space is developed, preserving the “neutral” axis (spectra that are uniformly constant over wavelength) as prioritized above [SCB88].

Mathematically, the construction of such a spectral gamut for a printer is as follows. Since reflectances must lie between 0 and 1, we can say that a reflectance $r(\lambda)$, formed from an orthonormal PCA basis $\beta_i(\lambda)$, is constrained by

$$0 \leq r(\lambda) = \sum_i c_i \beta_i(\lambda) \leq 1 \tag{1}$$

for all wavelengths λ. Each pixel can now be described by reflectance coordinates c_i with respect to the basis. The number of coordinates depends on the basis dimension, determined by the number of wavelength samples, only three of which may be conveniently plotted (see Figure 1). Having chosen the first basis vector to be uniform, $\beta_0(\lambda) = 1$. In dealing with discrete vector quantities, however, where summation is used instead of integration, $\beta_0 = k$, a constant that ensures $\beta_0 \cdot \beta_0 = 1$. A 10 nm sampling of the visible spectrum may give 36 values across the visible spectrum, so that $k = 1/36^{1/2} = 1/6$, giving $c_0 \in [0,6]$ in the Kodak 8650 printer spectral gamut of Figure 1.

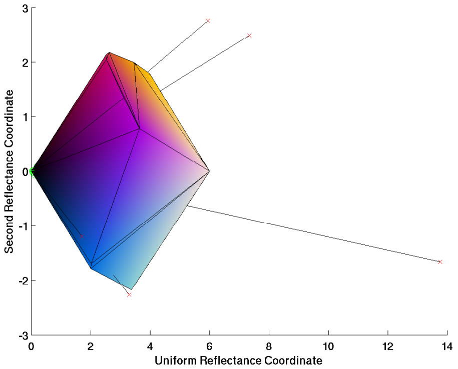

Figure 1: *Kodak 8650 printer spectral gamut (3/36 components), with blended line segments for material reflectances in the rendered scene. Clipping the segments to the gamut gives acceptable ranges for the material scale factors.*

The double inequality in (1) bounding the reflectances between 0 and 1 gives rise to pairs of parallel hyperplanes which carve out the gamut. Despite complex printer behaviour, the spectral gamut formed in this way is guaranteed to be convex, a useful feature when projecting out-of-gamut spectra into the gamut. Although measured printer reflectance spectra may have maxima less than one and minima greater than zero, the gamut can still be constructed without destroying convexity. Thus, a standard spectral gamut format can be established using linear methods [Pae94], with the basis constraining the resulting shape. This geometry can be used to provide feedback to the rendering stage.

3. Spectral Gamut Mapping

Most image spectra will not be exactly reproducible on the output device; however, selecting a uniform first basis spectrum ensures that the neutral axis is common to all gamuts. Image gamuts are therefore likely to be partly inside, and partly outside, the device gamut. Simple light and material variations scale the size of the image gamut, blending pixel spectral points into line segments. For the spectral gamut-mapping environment, we are interested in constraining light and material parameters, and thus the line segments, to lie within the device gamut. As the spectral device gamut is convex, this becomes a line clipping problem, as illustrated in Figure 1. Although we use clipping for our simple prototype, more sophisticated gamut compression is clearly possible.

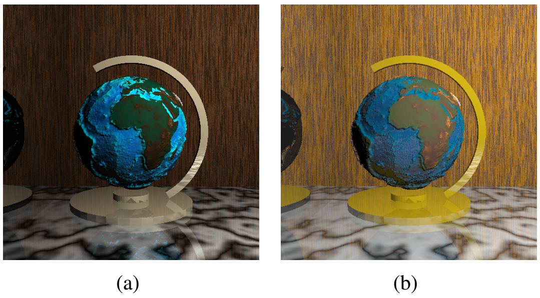
(a) (b)

Figure 2: *Globe image, rendered only once symbolically, then shaded with (-a) hand-picked RGB materials and illuminant, (-b) reflectances from the Macbeth colour checker and CIE D_{65} illuminant. (ETOPO2 global elevation data from the U.S. National Geophysical Data Center).*

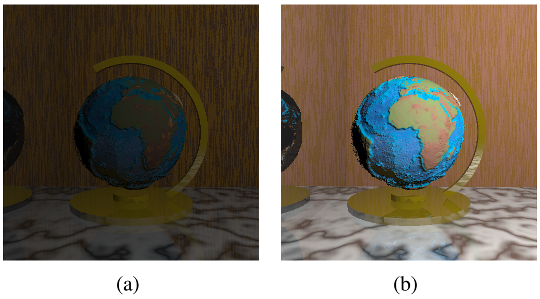
(a) (b)

Figure 3: *Globe image with (Macbeth) reflectances scaled to 0.01 and 3.00 (-a and -b resp.) of original values and clipped against the Kodak 8650 printer gamut. Spectrally, -a is the darkest and -b the lightest printable image in this range. Comparing -b with Figure 2(b), notice while Africa is lighter, the brass arm has darkened from clipping.*

4. The Symbolic Renderer

The rendering platform used is simple, but very flexible. It generalizes the parameterized ray tree approach of Séquin et al [SS89], however, our representation, while still describing all light and material interaction, is purely symbolic and does not impose any constraints on the surface models or lights. This environment provides an efficient “render once, shade often” testbed.

Common Lisp was chosen for implementation, as this facilitates mixed symbolic/floating point calculation, while allowing compilation to efficient code. For the work described here, a second pass on the tree performs partial evaluation on this representation, performing all calculations *except* the linear (or linearizable) part of light transport. For example, the Phong model may be treated as linear in the diffuse and specular components, but not the specular exponent. All geometric and nonlinear information is collapsed during this stage, leaving only arithmetic Lisp expressions in light and material variables. Once this is done, “rendering” for any set of materials and lights is a simple linear equation.

The calculation we are performing is as follows: If we let Γ be the set of pairwise products of components for all l lights and m materials, then we can calculate the λth component of the nth pixel as:

$$p_{n,\lambda} = \sum_{i=1}^{l\times m} g_{n,i}\Gamma_{i,\lambda} \quad . \tag{2}$$

By reshaping the image into an n-row matrix of pixel spectra, this calculation can be carried out as a matrix product. In general, many coefficients $g_{n,i}$ will be zero (no interaction between the ith light/material pair on the nth pixel), allowing sparse structure. These diffuse and specular $g_{n,i}$ geometry matrices remain independent of the colour space and the light and material parameters, so re-rendering is unnecessary unless the scene geometry changes. Using the PCA basis changes λ to an index over the basis spectra—data of any dimensionality may be used. In Figures 2(a) and 2(b), a globe is rendered once but shaded twice, the second time with spectral reflectances from the Macbeth colour checker.

5. Feedback from the Spectral Gamut

Having a symbolic rendering of the image allows light and material parameters to vary freely, extruding pixel spectral coordinates into line segments for clipping against the convex printer gamut. Although only a simple adjustment, clipping is sufficient to draw the image gamut into the device gamut, although we wish also to maximize contrast [SCB88], avoiding the trivial in-gamut black point solution. Using the paired planes that constructed the spectral gamut as clipping planes, we can predict acceptable scaling limits on the lighting and material parameters (Figure 1). A standard Liang-Barsky line clipping algorithm [FvDFH90] is sufficient, regardless of dimensionality. Figure 3 illustrates the globe, with materials scaled to fit in-gamut.

For more complex adjustment of parameters, consider substitution of one material for another. At a particular pixel, this will give rise to two distinct points in the spectral gamut, one for each material. Because of the linearity of the system, an affine blend of the material reflectances (akin to paint mixing) again gives a line segment in spectral space, which may be clipped to show limits on the blending parameter. Clipping can be applied to force all pixels in-gammut, or some percentage of out-of-gammut pixels may be allowed, as desired.

The complete algorithm takes this approach: 1) render the scene symbolically; 2) collapse the geometry and non-linear elements, leaving a linear system; 3) construct the light & material matrix elements in an appropriate space; 4) calculate the device spectral basis and gamut; 5) clip light & material parameters to the device gamut. Substituting the resulting parameters into the linear system yields a new, in-gammut, image.

6. Conclusions

Spectral rendering, hardware, and colour management are evolving differently, but will meet on common high-dimensional ground. Spectral gamut mapping will become a natural way to manage colour appearance. We have proposed a prototypical environment that supports these features:

- arbitrary dimensionality of pixels, including spectral data;
- symbolic ray tracing to probe the scene and record light/material interactions;
- light and material parameter range feedback for an in-gamut spectral image;
- interactive parameter adjustment without re-rendering.

The system may be expanded to adaptively adjust parameters until certain gamut-mapping criteria are met, or to probe a complex scene for light/material interaction statistics before a sophisticated renderer is applied. Nonlinear shading methods may be linearly approximated. There are many interactive possibilities, most immediately the ability to use spectral feedback in exploring in-gamut combinations of lights and materials. Furthermore, the flexibility implied by delaying commitment to particular shaders, illuminants, or sensors allows experimentation with rendering to multiple devices in different viewing environments. As spectral data becomes a greater part of future rendering and colour management applications, such flexible methods will become increasingly important.

References

[BF00] BRAUN G. J., FAIRCHILD M. D.: General purpose gamut-mapping algorithms: Evaluation of contrast-preserving rescaling functions for colour gamut mapping. *JIST 44*, 4 (July/August 2000), 343–350. 1

[DCWP02] DEVLIN K., CHALMERS A., WILKIE A., PURGATHOFER W.: Tone reproduction and physically based spectral rendering. In *EG 2002, State of the Art Reports* (2002), pp. 101–123. 1

[FvDFH90] FOLEY J., VAN DAM A., FEINER S., HUGHES J.: *Computer Graphics: Principles and Practice*, second ed. Addison-Wesley, 1990. 3

[GL89] GOLUB G., LOAN C. V.: *Matrix Computations*, second ed. John Hopkins U.P., Baltimore, 1989. 2

[Har] HARDEBERG J.: On the spectral dimensionality of object colours. In *CGIV'2002*, pp. 480–485. 2

[HB00] HERZOG P. G., BÜRING H.: Optimizing gamut mapping: Lightness and hue adjustments. *JIST 44*, 4 (July/August 2000), 334–342. 1

[Hun98] HUNT R. W. G.: *Measuring Colour*. Fountain Press, England, 1998. 1

[Hun02] HUNT R. W. G.: Application of a model of color appearance to practical problems in imaging. *Proceedings of the IEEE 90*, 1 (January 2002), 57–63. 1

[IRB02] IMAI F., ROSEN M., BERNS R.: Comparative study of metrics for spectral match quality. In *CGIV'2002* (2002), IS&T, pp. 492–496. 1

[Len02a] LENZ R.: Spaces of spectral distributions and their natural geometry. In *CGIV'2002* (2002), IS&T, pp. 249–254. 2

[Len02b] LENZ R.: Two stage principal component analysis of color. *IEEE Transactions on Image Processing 11*, 6 (June 2002), 630–635. 2

[MCSP02] MARTINEZ K., CUPITT J., SAUNDERS D., PILLAY R.: Ten years of art imaging research. *Proceedings of the IEEE 90*, 1 (January 2002), 28–41. 2

[ML01] MOROVIČ J., LUO M. R.: The fundamentals of gamut mapping: A survey. *JIST 45*, 3 (May/June 2001), 283–290. 1

[Pae94] PAETH A. W.: *Linear Models of Reflective Colour*. PhD thesis, University of Waterloo, 1994. 2

[Pee93] PEERCY M. S.: Linear color representations for full speed spectral rendering. In *20th CGIT* (1993), ACM Press, pp. 191–198. 2

[RFJW00] ROSEN M. R., FAIRCHILD M. D., JOHNSON G. M., WYBLE D. R.: Color management within a spectral image visualization tool. In *8th IS&T/SID CIC* (2000), pp. 75–80. 2

[RO03] ROSEN M., OHTA N.: Spectral color processing using an interim connection space. In *11th IS&T/SID CIC* (2003), pp. 187–192. 2

[SCB88] STONE M. C., COWAN W. B., BEATTY J. C.: Color gamut mapping and the printing of digital color images. *ACM TOG 7*, 4 (Oct. 1988), 249–292. 2, 3

[SS89] SÉQUIN C. H., SMYRL E. K.: Parameterized ray-tracing. In *SIGGRAPH'89* (1989), ACM Press, pp. 307–314. 3

[THG99] TUMBLIN J., HODGINS J. K., GUENTER B. K.: Two methods for display of high contrast images. *ACM Trans. Graph. 18*, 1 (1999), 56–94. 1

[WHD03] WENGER A., HAWKINS T., DEBEVEC P.: Optimizing color matching in a lighting reproduction system for complex subject and illuminant spectra. In *EG Symp. Ren.* (2003), pp. 249–259. 1

[WTP01] WILKIE A., TOBLER R. F., PURGATHOFER W.: Combined rendering of polarization and fluorescence effects. In *12th EG Work. on Rend.* (2001), pp. 197–204. 1