Spring-Primary Mapping: Combining Primary Adjustment and Gamut Mapping for Pictorials and Business Graphics

Huanzhao Zeng, Digital Printing Technologies, Hewlett-Packard Co., 18311 SE 34th St, Vancouver, WA 98683

Abstract

A color mapping method was developed for device to device color mapping for pictorials and business graphics. The color mapping starts from gamut surface for the preference of business graphics or the tradeoff between business graphics and pictorials. As the process moves toward the interior of the gamut, the mapping is gradually adapted to the preference of pictorials. Instead of performing color mapping point-wisely, it selects a small number of points for color mapping and process other points by interpolation using relative neighbor color information. It incorporates primary adjustment and gamut mapping into a single step. Using relative neighbor color information for color mapping, it well preserves the color to color relationship. This property makes it particular good for business graphics. Applying interpolation instead of gamut mapping for the color mapping of majority of colors, it significantly improves the performance of the gamut mapping process.

***Keywords**: gamut mapping, color adjustment, hue rotation, lookup table, interpolation*

1. Introduction

Due to the gamut limitation in reproducing color, gamut mapping is essential for adapting colors from one device gamut to another. In the past fifteen years, many approaches have been developed for gamut mapping. Topics related to gamut mapping include: color appearance adjustment; color spaces for gamut mapping; how to shape source colors to fit into a destination gamut; point-wise gamut mapping versus spatial gamut mapping; and device-dependent versus image-dependent gamut mapping.

In cross color reproduction, the viewing conditions and imaging media between a source and a destination may be different. A typical example is the color transformation from a monitor display to a printer hardcopy. Color appearance adaptation or adjustment is an essential step for preferred color reproduction. This may include white point and black point adaptation, contrast adjustment, and color preference adjustment.

In early days, CIELAB color space was widely used for gamut mapping. Due to the blue hue nonlinearity in CIELAB color space, blue colors may be gamut mapped to purple [1, 2]. Different approaches were developed to fix the problem. Hue rotation in CIELAB color space is a simple approach. Braun and Fairchild developed a hue-linearized CIELAB color space for gamut mapping [3]. Marcu [4] and McCann [5] applied Munsell renotation system (MLab) for gamut mapping to solve the non-uniformity of CIELAB color space. Although a color space may work well for the gamut mapping in some gamut regions, it may have problems in other regions. For example, CIELAB works well for gamut mapping in general except for the blue region. Zeng proposed using multiple color spaces for gamut mapping to solve color space problems for gamut mapping [1]. For example, a blue-hue linearized color space is applied for gamut mapping in the blue region, and CIELAB color space is used for gamut mapping in other regions. Zeng further developed a composite color space concept to simplify the gamut mapping using multiple color spaces [6]. With the newly developed CIECAM97s and CIECAM02 color appearance models, gamut mapping has been widely applied in CAM97s JAB or CAM02 JAB [7]. However, using a composite color space is still very useful, such as for generating colorimetric mapping tables for printer ICC profiles.

The gamut mapping to reshape color gamut has been an active research topic in the past fifteen years. The simplest one is to clip source colors into the destination device gamut, ignoring the source gamut. An example is the generation of a colorimetric table (BToA1 tag) of a printer ICC profile. There are many papers address how to determine directions to clip out-of-gamut colors [8–10].

The second category is to map a source device gamut into a destination device gamut, ignoring image contents. A majority of published gamut mapping methods belongs to this category. Various geometrical mapping methods have been developed to map colors from a source device gamut into a destination device gamut. Braun and Fairchild proposed using Sigmoidal contrast enhancement for lightness rescaling to optimize contrast mapping [11]. Morovic and Luo [12], MacDonald et al [13], Lee [14] and many other researchers [15–19] developed or investigated different methods for the gamut mapping from a device gamut to another. Most of these gamut mapping methods are suitable for the mapping from a device to another for pictorials, but have difficulties in producing smooth ramps for business graphics.

The third category is the image-dependent gamut mapping. Instead of mapping from a source device gamut to a destination device gamut, it maps from the source image gamut of an image to a destination device gamut [20-22]. Image-dependent gamut mapping expects to perform better color result than image-independent gamut mapping. However, in practical, automatic image-dependent gamut mapping may not outperform the gamut mapping using a fixed source gamut due to varieties of image gamut shapes, lack of human interaction for adjusting gamut mapping parameters, and real-time performance issues.

Typically, gamut mapping is performed point-wisely. Some image details may be lost due to gamut compression, or noise may be amplified due to gamut expansion. Spatial gamut mapping algorithms, in which the color information of each color and its neighbors are used, try to retain the spatial details [23-25]. Two pixels with the same color may be mapped to different colors, depending on the local spatial characteristics of an image. Since the spatial processing is image-dependent and must be performed at real-time, they can be too slow for many practical applications.

Without manual adjustments, most of gamut mapping methods has difficulties in performing gamut mapping smoothly for computer generated graphics images. Braun, Bala, and Harrington [26] developed several gamut mapping methods for business graphics images, in which they concluded their rotated device mapping performed the best of all. However, because the gamut mapping is performed in a device-dependent color space, the quality may be dependant on the printer device color space. Furthermore, the gamut mapping is not suitable for pictorials. Because printer drivers are not able to determine object types (photographic images or business graphics images, etc.) reliably, a lookup table (LUT) generated by this kind of mapping method must be blended with a LUT generated by other gamut mapping methods for driver based printing applications.

Zeng [27] presented a method to preserve primaries in ICC color management system, which is used for printing computer-generated graphics. However, a color map for the transformation from a default monitor color space to a printer device color space for business graphics must be generated. Yet, the paper does not provide information for generating a default color map.

Spaulding et al. [28] developed a gamut mapping, UltraColor gamut mapping, for a unified solution for both photographic images and presentation images. Although no detail information is provided for the trade-off of gamut morphing based on the priority of image object types (e.g. how to determine the mapping for first-primary and second-primary color ramps and other gamut surface colors), it is the only gamut mapping method known by the author for unified gamut mapping for all classes of images.

None of these gamut mapping methods provide the trade-off for the priorities of different object types, from photograph to business graphics, for priority color mapping. Nor did they use neighbor color information to speed up gamut mapping and to gamut map colors smoothly. Two similar colors must be mapped to two similar colors; and two significantly different colors must be mapped to two significantly different colors. By applying this characteristic, we need only to apply gamut mapping to a sparse sampling points of a source color space for the gamut mapping from a source color space to a destination color space, and to apply a more efficient interpolation method or geometrical reshaping method to color map other colors.

In this paper, we present an innovative approach that combines the primary adjustment and the geometric shape-adaptation to achieve the effect of gamut mapping. It is able to accurately map a source primary to a destination primary or an aimed primary which is usually somewhere between a source primary and its corresponding destination primary. The result of the color mapping is a pictorial type of gamut mapping for interior colors, and is gradually adapted toward a business graphics type of mapping as colors change from interior toward the gamut surface.

2. Basic Concept of the Algorithm

An example of the gamut difference between a source gamut and a destination gamut in Fig. 1 shows that hue rotation must be performed for primary mapping, and gamut compression or expansion must be applied to map source colors into the destination gamut. Fig. 2 shows a constant hue slice of the 3-D plot from Fig. 1. The source cusp (Cs) may be mapped to the destination cusp (Cd) so that the source white-to-Cs color ramp be mapped to the destination white-to-Cd color ramp and the source black-to-Cs ramp be mapped to the destination black-to-Cd ramp nicely. While this kind of mapping may be preferred for business graphics, it may darken colors too much for pictorials. An aimed point, Ca, may be chosen as an aimed primary for a photo rendering intent. Because the most saturated source point Cs is mapped to Ca, the printable colors from Ca to Cd may or may not be used, depending on rendering intents. The portion of the printer gamut encompassed by lines from W (the white point) to Ca to K (the black point) to W may be used as an aimed gamut boundary, and the shaded portion of the printable gamut is not used.

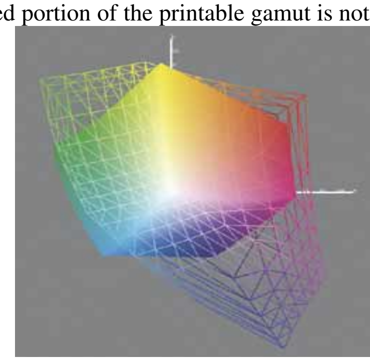

Fig. 1 *A gamut comparison between sRGB (wire-frame) and SWOP CMYK (solid) in CIE L*a*b* color space*

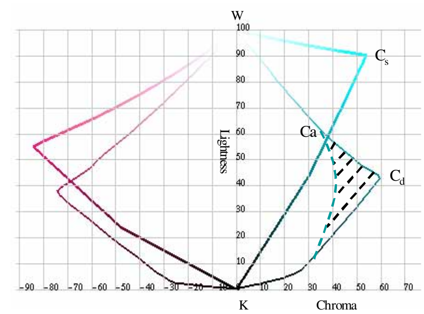

Fig. 2 *A hue slice of sRGB gamut (thick-line) and SWOP CMYK gamut (thin-line) (both black points are adjusted to zero)*

Multi-dimensional LUTs are generally used for the color transformation from one device color space into another for device to device color transformation. For example, a 3-D LUT may be generated for the transformation from sRGB color space into a printer CMYK color space. To simplify numerical descriptions, we assume the sampling of a 3-D LUT is 17x17x17. In a LUT, the colors of neighbor nodes are similar. The larger the LUT size, the smaller the color differences among neighbor nodes. In most locations, color shifts from one node to its neighbor nodes are highly predictable. These properties can be seen in an RGB LUT discussed below.

Fig. 3 illustrates the 3-D gamut of a device in a device L-S-H color space. For a 3-D 17x17x17 RGB LUT, the indexes of each of the 4913 nodes are denoted as (r, g, b) hereafter, where r, g, and b are from 0 to 16. The indexes of the eight corner nodes are: K (black): (0, 0, 0); B (blue): (0, 0, 16); G (green): (0, 16, 0); C (cyan): (0, 16, 16); R (red): (16, 0, 0); M (magenta): (16, 0, 16); Y (yellow): (16, 16, 0); and W (white): (16, 16, 16).

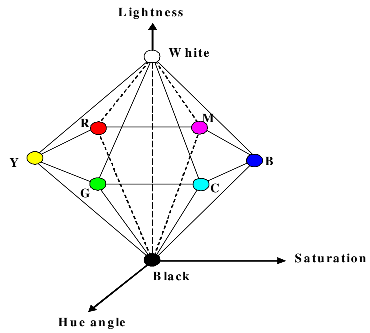

***Fig.** 3 A 3-D device gamut in an L-S-H color space.*

Let's assume that the source color space is sRGB. As the indexes of an sRGB color (r, g, b) are changed gradually from K to W , colors change gradually from black to gray to white (see Fig. 4). Knowing the white and black points of sRGB and the white and black points of a printer and knowing the tone mapping curve, we can compute the mapping of these seventeen points by interpolation instead of gamut mapping.

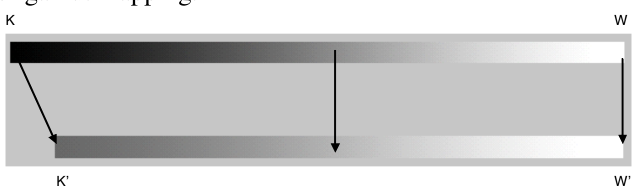

***Fig. 4** A source gray ramp from its black point (K) to its white point (W) is mapped to a destination gray ramp from its black point (K') to its white point (W'). The mapping is performed by 1-D interpolation.*

As the indexes of an sRGB color (r, g, b) are changed from W to R, colors gradually change in the red ramp from W to R. Knowing the transition relationship (or relative relationship) of these sRGB colors, we may determine the mapping for only the primary, R, and compute the mapping for other points by a more efficient approach than gamut mapping. Fig. 5 shows that a source gamut boundary color ramp from K to R to W is mapped to a color ramp in a destination gamut. There are 33 nodes on the ramp for a 17x17x17 LUT. Again, K is mapped to K' and W is mapped to W'. R (the source red primary) is mapped to R' (an aimed red primary). The details for determining an aimed primary will be discussed latter. For rest of points, the mapping can be performed by interpolation using relative neighbor color relationship instead of gamut mapping.

Fig. 6 shows another example for the mapping of a color ramp R-to-Y on the gamut boundary of a source gamut. The source red primary, R, is mapped to an aimed red primary R', and the source yellow primary, Y, is mapped to an aimed primary Y'. Again, the mapping of other colors is performed by interpolation using neighbor relative color information instead of gamut mapping. Important colors, such as a skin tones, may be mapped without hue rotation. Most of gamut mapping methods perform gamut mapping in a two-step process, hue rotation followed by mapping lightness and chroma. Contouring often happens in gamut boundary colors. In this example, lightness mapping, hue rotation, and chroma compression are performed in a single step which is very helpful for smooth color mapping.

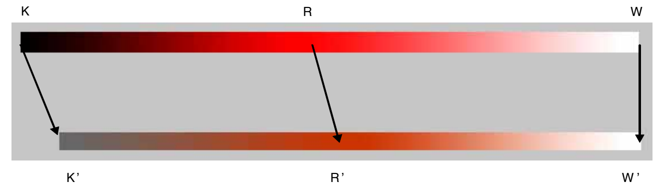

***Fig. 5** Gamut boundary colors from the K to R to W of a source gamut are mapped to the destination gamut.*

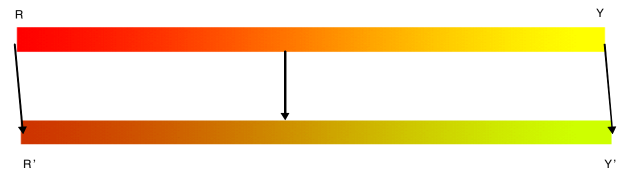

***Fig. 6** Using color similarity information to map source colors to destination colors. Simultaneously lightness mapping, hue rotation, and chroma compression can be easily performed.*

A similar approach can be applied to other group of gamut boundary colors, such as the color ramps of W-G-K, W-B-K, W-Y-K, and W-M-K.

Based on the relative color relationship of neighbor nodes of a LUT in the source color space, we can perform gamut mapping for only a small portion of points, and use a far more efficient approach to map other points.

3. The New Gamut Mapping Process

The basic process is to first map the corner colors in the gamut boundary, followed by mapping gamut boundary lines, followed by mapping gamut surface, and finally through the mapping of interior colors of a color LUT. Detail procedures are described below.

Step 1: Mapping W and K and Other Neutral Points

White point adaptation, black point adjustment, and tone adjustment are performed in this step. Other neutral nodes in a lookup table (e.g. nodes for R=G=B in an sRGB LUT) are computed using 1-D interpolation instead of gamut mapping. A nonlinear curve can be applied in this step to alter the tone curve. For a 17x17x17 3-D RGB LUT, the seventeen neutral nodes from W to K are processed.

Step 2: Mapping Primaries

Aimed primaries are determined in this step, which is very important for morphing a source gamut into a destination gamut. The lightness and hue angle of each aimed primary are determined based on the differences of the source and destination primaries, and the priority of pictorial mapping versus graphics mapping. Properly rotating hue angle and adjusting lightness are very important for mapping a source primary into a destination gamut.

The lightness and hue angle of each aimed primary may be determined by following weighting equations:

$$L_{aimed} = w_L \cdot L_{source} + (1 - w_L) \cdot L_{destination}$$
$$h_{aimed} = w_h \cdot h_{source} + (1 - w_h) \cdot h_{destination}$$

where L_{aimed}, L_{source}, and $L_{destination}$ are the aimed lightness, the source lightness, and the destination lightness of a primary, respectively; h_{aimed}, h_{source}, and $h_{destination}$ are the aimed hue angle, the source hue angle, and the destination hue angle of the same primary, respectively; w_L and w_h are the weights for the lightness and the hue angle, respectively. The weights may be determined automatically based on the difference of the gamut shapes of the source and the destination, and the preference of the color mapping (i.e. priority between pictorial and business graphics).

After lightness (L_{aimed}) and hue angle (h_{aimed}) of an aimed primary are determined, an aimed primary is determined by searching for a point that has a lightness of L_{aimed} and a hue angle of h_{aimed} on the destination gamut surface.

Step 3: Mapping Gamut Surface Points

In Fig. 7, the tetrahedron K-P1-P2-W is a portion of a 3-D device color gamut shown in Fig. 3. P1 and P2 are two source neighbor primaries. p1 and p2 are the aimed primaries corresponding to P1 and P2, respectively. The source gamut surface points are located on two triangles, W-P1-P2 and K-P1-P2. To map the source gamut surface points in two source triangles into two destination triangles, W-p1-p2 and K-p1-p2, the boundary lines are first mapped from the source to the destination, and then points on each source triangle are mapped to its corresponding destination triangle.

The mapping for gamut surface points is described in following sub-steps.

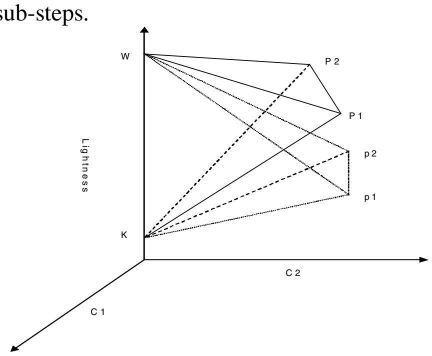

Fig. *7 A section of a source gamut (the tetrahedron K-P1-P2-W) and its corresponding aimed gamut (K-p1-p2-W).*

Step 3.1: Mapping primary ramps

There are six lines in the gamut boundary that connect the white point and a primary point and six lines that connect the black point and a primary point (see Fig. 3). These lines are: W to C, W to M, W to Y, W to R, W to G, W to B, C to K, M to K, Y to K, R to K, G to K, and B to K.

The mappings of two end nodes in each line have been obtained in previous steps. The mapping for intermediate nodes is performed by interpolation instead of by gamut mapping. Figures 4 to 6 illustrate that using interpolation works very well for mapping color ramps.

After the interpolation for each node in a line, the mapping for each point may be fine tuned so that each of them stays on the gamut surface of the destination gamut.

Step 3.2 Mapping edge points

The edge points are the points on lines connected two neighbor primaries. There are six lines to be processed in this step (see Fig. 3): R to Y, Y to G, G to C, C to B, B to M, and M to R.

Before processing these six lines, a hue rotation LUT may be generated for hue adjustment. By converting LAB values of the source and the aimed primaries into LCh, the amount of hue rotation for each primary can be computed. A hue angle mapping LUT can be generated based on the hue rotation of the six primaries.

To perform the mapping for each node on these six lines, lightness is computed through interpolation. Output hue angles are obtained by applying the hue adjustment LUT to the source hue angle. After lightness and hue angle are computed, output chroma (or chrominance A and B) is obtained by gamut clipping on gamut surface with constant lightness and constant hue angle. Direct interpolation is another alternative. Fine tuning may be required to obtain points exactly on edges.

Step 3.3: Mapping gamut surface points

Fig. 3 shows that gamut boundary can be constructed by twelve triangles. Nodes on the gamut surface are processed by interpolating in each triangle. The LAB values of a node can be computed by interpolation using a few points on the triangle. Since gamut boundary is not exactly shaped in flat plans, twelve triangles only approximately represent the gamut shape. After the interpolation, post-processing may be applied to move points to the gamut surface exactly.

Step 4: Processing Interior Points

At this step, all nodes on the gamut surface of a LUT have been mapped to the destination gamut. Except for the nodes on the neutral line, other interior nodes have not been mapped. We may draw a region for colorimetric mapping (see the green octagon in Fig. 8). For nodes on the colorimetric region, no color adjustment and gamut mapping are performed.

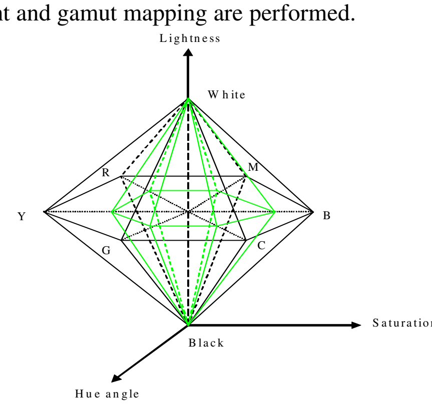

Fig. *8 Constructing a 3-D colorimetric region for interpolation*

So far, all gamut surface nodes and some interior nodes in a 3-D LUT have been mapped to the destination gamut. The rest of nodes in a LUT are interpolated using a 3-D interpolation method [29-30]. A six-weight interpolation method may be used in this step. With this method, up to six points are used for interpolation. These six points are found from six directions relative to the point to be processed: up, down, left, right, front, and back. By searching each of the six directions, a point in each direction that has been mapped to the destination gamut is used for interpolation. The weight for each point is computed based on the distance or color difference of the point to the point to be processed.

4. Results and Discussion

Four sRGB images (see Fig. 9) were chosen to test the color quality of the new gamut mapping method (SPGMA) against two other gamut mapping methods (CUSP-FIT A and CUSP-FIT B) that have been used for successful product developments. These two methods have the following properties: gamut mapping in constant hue slices; white point adaptation and black point adjustment prior to gamut mapping; hue rotation as a separated step prior to gamut mapping; the differences of the lightness and chroma of the source cusp and the destination cusp, and the lightness of the color to be process are used to determine the mapping direction (or anchor point); a core section is determined for colorimetric mapping; colors in the source gamut are mapped to the destination gamut with nonlinear chroma compression; gamut mapping in CIECMA02 color space; CUSP-FIT B completely rotate the source cusp to fit into the destination cusp, while CUSP-FIT A does not map the source cusp completely to the destination cusp if lightness of two cusps are very different. White point adaptation, black point adjustment, tone adjustment, preference color adjustment, and hue rotation are performed exactly the same for all three methods.

***Fig**. 9 Images for psychophysical experiment: Barn (upper-left), Hikey (upper-right), Kids (lower-left), and Graphics (lower-right).*

Three color LUTs, generated using these three gamut mapping methods to map sRGB into printer device color space, were applied to print these four images on high-gloss photo media using an HP photo-smart inkjet printer. Eleven observers conducted the experiment. A rank order method is applied to compute psychophysical scales. The results for the three pictorial images and the fourth computer generated graphics image are analyzed separately.

Fig. 10 shows the psychophysical experimental result from the three pictorial images. SPGMA performs the best for Barn, CUSP-FIT B performs the best for Kids, and all three gamut mapping methods perform the same for Hikey. The main reason that SPGMA performs the best for Barn is that trees have brighter green. The main reason that SPGMA is less preferred to CUSP-FIT A for Kids is that red is weaker.

The average scores from all three pictorial images are also included. In overall, SPGMA performs slightly worse than CUSP-FIT B but slightly better than CUSP-FIT A. From observers' comments, the differences among the three gamut mapping methods applying to these three images are very subtle. If the lightness mapping of the red primary is set to be closer to the printer primary, SPGMA should produce stronger red and the score should be improved. The nice feature of the SPGMA is that the lightness mapping of each primary can be set outside the gamut mapping, while other two gamut mapping do not have the freedom to adjust lightness mapping for different primaries separately. Properly setting the lightness mapping parameters for the six primaries, the author believe SPGMA's score will be better than or equal to any of the other two methods.

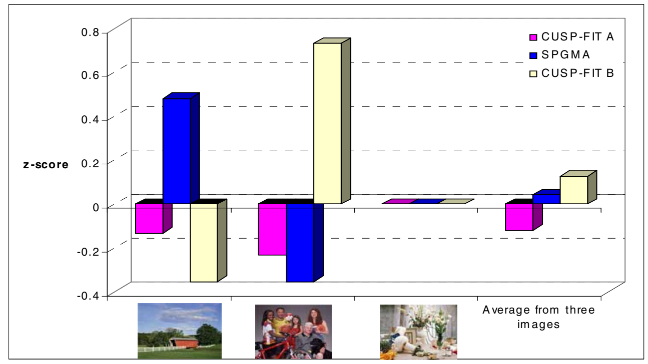

***Fig. 10** z-scores for the performance of the three pictorial images*

The experimental result from the Graphics image is shown in Fig. 11. SPGAM performs clearly better than other two methods, because its mapping is much smoother than the other two for every section of the graphics image. CUSP-FIT B maps source cusp closely to the printer cusp, thus the overall saturation is stronger than that from either of the other two, but the contouring pulls its score down. CUSP-FIT A has the worst contouring. SPGMA makes almost contour free mappings for the graphics image. Since the color map generated using SPGMA in this experiment is used for the trade-off between pictorials and computer graphics, six sRGB primaries are not mapped to the most saturated primaries of the printer gamut. Because of that, a few observers did not pick SPGMA as the best. Further optimizing the aimed lightness of the six primaries should improve the result. By optimizing a separate aimed primary set to generate a separate color map for graphics, the mapping for graphics using SPGAM can be further improved.

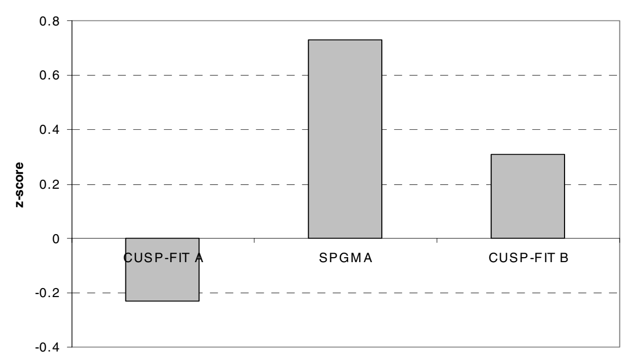

***Fig. 11** z-scores for the Graphics image*

One important characteristics of SPGMA is that it performs gamut mapping for gamut corner points only, and uses relative neighbor color information to derive the mapping for other points. By using neighbor color information for gamut mapping, it guarantees smooth mapping. By using interpolation methods for color mapping, the gamut mapping efficiency is dramatically improved.

Using a 17^3 LUT for SPGMA is generally large enough for mapping from a device gamut to another. A 9^3 LUT is mostly not large enough. A 33^3 LUT is necessary only if the source color space is a non-physical huge color space or the input or output color space is highly non-linear.

5. Conclusions

Different from other gamut mapping methods that perform gamut mapping point-wisely for each node of a lookup table, the new gamut mapping method uses color similarity information of neighbor colors to aid color mapping. It performs gamut mapping only on a small amount of points, and the color mappings of other points are performed using more efficient interpolation methods. Primary adjustment is seamlessly incorporated into the gamut mapping step, therefore it further improves the performance. Because the color similarity information of neighbor colors are used for gamut mapping, the color to color relative relationship is well retained. Based on the settings for primaries mapping, the color mapping for gamut surface colors can be configured for the preference of business graphics or for the tradeoff between business graphics and pictorials. As the processing goes from gamut surface colors to interior colors, the color mapping is gradually adapted to the preference of pictorials.

Setting aimed primaries properly, a color map produced by this method can be used for the trade-off of both photographic images and business graphics, or for a priority mapping between different image types. Because gamut mapping is performed only on a small percentage of nodes, the color mapping is highly efficient.

References

1. H. Zeng, “Gamut Mapping in Multiple Color Spaces”, Proc. SPIE Color Imaging: Device-Independent Color, Color Hardcopy, and Graphic Arts V, Vol. 3963, 301-306 (2001).
2. N. Moroney, “Assessing hue constancy using gradients”, Proc. SPIE Color Imaging: Device-Independent Color, Color Hardcopy, and Graphic Arts V, Vol. 3963, 294-300 (2001).
3. G. Braun and M. Fairchild, “Color Gamut Mapping in a Hue-Linearized CIELAB Color Space”, Proc. IS&T/SID 6th Color Imaging Conference, 163-168 (1998).
4. G. Marcu, “Gamut Mapping in Munsell Constant Hue Sections”, Proc. IS&T/SID 6th Color Imaging Conference, 159-162 (1998).
5. J. McCann, “Color Gamut Measurements and Mapping: the Role of Color Spaces”, Proc. SPIE Color Imaging: Device-Independent Color, Color Hardcopy, and Graphic Arts IV, Vol. 3648, 68-82 (1999).
6. H. Zeng, “Gamut Mapping in a Composite Color Space”, Proc. IS&T NIP17: International Conference on Digital Printing Technologies, Vol. 17, 797-800 (2001).
7. J. Morovic, “Verification of Gamut Mapping Algorithms in CIECAM97s Using various Printed Media”, IS&T/SID 6th Color Imaging Conference, 55-56 (1998).
8. J. Morovic and P. Sun, “Non–Iterative Minimum DeltaE Gamut Clipping”, Proc. IS&T/SID 9th Color Imaging Conference, 251-256 (2001).
9. M. Ito and N. Katoh, “Three-dimensional Gamut Mapping Using Various Color Difference Formulae and Color Spaces”, Proc. SPIE Color Imaging: Device-Independent Color, Color Hardcopy, and Graphic Arts IV, Vol. 3648, 83-95 (1999).
10. M. Gestel and H. Draaisma, “Color Gamut Mapping along Electrostatic Field Lines”, Proc. IS&T NIP17: International Conference on Digital Printing Technologies, 801-803 (2001).
11. G. Braun and M. Fairchild, “Image Lightness Rescaling Using Sigmoidal Contrast Enhancement Functions”, Proc. SPIE, 3648, 96-107 (1999).
12. J. Morovic and M. Luo, “Evaluating Gamut Mapping Algorithms for Universal Applicability”, Color Research and Application 26:85–102 (2001).
13. L. MacDonald, J. Morovic, and K. Xiao, “A Topographic Gamut Compression Algorithm”, Journal of Imaging Science and Technology, 46: 228-236 (2002).
14. C. Lee, et al, “Gamut Mapping Algorithm Using Lightness Mapping and Multiple Anchor Points for Linear Tone and Maximum Chroma Reproduction”, J. Imaging Sci. Technol., 45(3): 209-223 (2001).
15. H. Motomura, “Gamut Mapping Using Color-Categorical Weighting Method”, Proc. IS&T/SID 8th Color Imaging Conference, 318-323 (2000).
16. J. Morovic and M. Luo, “Gamut Mapping Algorithms Based on Psychophysical Experiment”, Proc. IS&T/SID 5th Color Imaging Conference, 44-49 (1997).
17. H. Motomura, “Categorical Color Mapping Using Color Categorical Normalized Distance Transformation”, J. Imaging Sci. Technol., 48(6): 548-561 (2004).
18. P. Green and M. Luo, “Extending the CARISMA Gamut Mapping Model”, J. of Imaging Sci. Technol., 46(1): 33-43 (2002).
19. Y. Cho, Y. Kim, and Y. Ha, “Gamut Mapping Based on Color Space Division for Enhancement of Lightness Contrast and Chrominance”, J. Imaging Sci. Technol., 48(1), 75-83 (2004).
20. J. Morovic and P. Sun, “The Influence of Image Gamuts on Cross-Media Colour Image Reproduction”, Proc. IS&T/SID 8th Color Imaging Conference, 324-328 (2000).
21. R. Saito and H. Kotera, “Image-dependent three-dimensional gamut mapping using gamut boundary descriptor”, J. Imaging Sci. Technol., 13(3), 630-638 (2004).
22. I. Farup, J. Hardeberg, and M. Amsrud, “Enhancing the SGCK Colour Gamut Mapping Algorithm”, Proc. CGIV: 2nd European Conference on Colour Graphics, Imaging and Vision, 520-524 (2004).
23. J. Morovic and Y. Wang, “A Multi-Resolution, Full-Colour Spatial Gamut Mapping Algorithm”, Proc. IS&T/SID 11th Color Imaging Conference, 282-287 (2003).
24. R. Bala, R. deQueiroz, R. Eschbach, and W. Wu, “Gamut Mapping to Preserve Spatial Luminance Variations”, J. Imaging Sci. Technol., 45(5), 436-443 (2001).
25. R. Eschbach and R. Bala, “Simple spatial processing for color mappings”, J. Electronic Imaging, 13(1), 120-125 (2004).
26. K. Braun, R. Bala, and S. Harrington, “Gamut-Mapping Techniques for Business Graphics”, IS&T/SID 7th Color Imaging Conference, 149-153 (1999).
27. H. Zeng, “Primary Preservation in ICC Color Management System”, Proc. SPIE Color Imaging: Device-Independent Color, Color Hardcopy, and Graphic Arts VII, Vol. 4663, 207-216 (2002).
28. K. Spaulding, R. Ellson, and J. Sullivan, “UltraColor: A new gamut mapping strategy”, Proc. SPIE Device-Independent Color Imaging II, Vol. 2414, 61-68 (1995).
29. H. Zeng, “3-D Color Separation Maximizing the Printer Gamut”, Proc. SPIE: Color Imaging: Device-Independent Color, Color Hardcopy, and Graphic Arts VIII, Vol. 5008, 260-267 (2002).
30. H. Zeng, “Special Interpolation to Minimize Grain in Printer Color Separation”, Proc. SPIE: Color Imaging: Device-Independent Color, Color Hardcopy, and Graphic Arts X, Vol. 5667, 424-433 (2005).

Author Biography

Huanzhao Zeng is an expert-level color scientist at Hewlett-Packard Company. He received his M.S. degree in Imaging Science from Rochester Institute of Technology in 1996. His research interests include device color calibration, gamut mapping, ICC color management, image enhancement, digital halftoning, and 3-D visualization.