A Topographic Gamut Compression Algorithm

Lindsay MacDonald,▲ Ján Morovic and Kaida Xiao

Color & Imaging Institute, University of Derby, United Kingdom

A color gamut compression algorithm has been developed based on analysis of observer judgements in a previous interactive gamut mapping experiment. The new algorithm preserves the color relationships between the original and reproduced images, by matching the local conformations of the source and destination gamut boundaries. A core gamut is constructed inside the destination gamut boundary, within which no compression occurs, i.e., color is preserved unchanged. Colors outside the destination gamut are mapped into the region between the core and destination gamut boundaries in a reversible manner. The results of an experiment are reported, indicating that the new algorithm performed well but that scope remains for further improvement.

Journal of Imaging Science and Technology 46: 228–236 (2002)

Introduction

It remains a significant problem how best to reproduce the colors of an image that lie outside the color gamut of a given device. There can be no single optimal method, because various conflicting demands must be taken into account:

- Achieve a specified reproduction objective or *rendering intent*;
- Make best use of the available color gamut of the output device;
- Not introduce any visible artifacts, such as contouring, into the image;
- Minimize computational complexity for efficient processing.

Developments in color imaging techniques over the past decade have separated the problem into three areas—device characteristics, color appearance and gamut mapping—enabling each to be studied in greater detail. Gamut compression algorithms (GCAs) in particular have steadily become more sophisticated since the crude ‘clip to range limits’ methods of early computer graphics. Studies by Morovic and Luo[1] and Braun and co-workers[2] have indicated that different algorithms are preferred by observers not only for different device color gamuts, but also for different regions of color space. Also the characteristics of individual images are crucia–generally much better results can be achieved by analysis of the distribution of colors within an image than by applying a generic algorithm.[3,4]

It is generally agreed that gamut mapping should be performed in a perceptually uniform color space such as CIELAB or CIECAM97s.[5] This provides more control over the appearance of the image, and allows the techniques of Cartesian geometry to be applied to equally scaled dimensions of lightness, chroma and hue. The CIECAM97s color model yields superior uniformity and independence of media and viewing conditions, although it is more complex to compute. Throughout this article we shall assume that a uniform color space is employed, and denote the lightness and chroma axes by L and C respectively.

Gamut Compression Algorithms

Various methods of gamut compression have been proposed, from the basic clipping of colors onto the nearest point on the gamut boundary to complex transformations of color space in which the lightness, chroma and in some cases also hue are modified. Good results for pleasing reproduction of scenes have been obtained by the CARISMA algorithm, first proposed by Johnson[6] and further developed by Morovic and Luo[1] and Green and Luo.[7] A recent evaluation by Pirrotta and co-workers[8] confirmed that it performs well for photographic images but not very well for business graphics and illustrations.

One of the problems with the majority of previous algorithms is that they attempt to map all colors in the L–C plane (at constant hue) toward a single convergence point, or ‘center of gravity’, based on the coordinates of the cusps (points of maximum chroma) of the original and reproduction gamuts. An example is the SLIN algorithm,[1] in which colors are mapped toward the point L = 50 on the lightness axis. Different rules may be employed for different cases (e.g., the relative lightness and chroma of the two cusp points), and the transformations may be non-linear, but usually all the points in the plane are governed by a single mapping formula. Such methods can result in unnecessarily large changes in the lightness of colors at the extremities of the lightness axis (notably in light yellow and dark blue–violet hues), and hence in significant changes to the overall image appearance.

Recent investigation by Kang and co-workers[9] has produced some interesting insights into gamut mapping in

Original manuscript received February 14, 2001

▲ IS&T Member

©2002, IS&T—The Society for Imaging Science and Technology

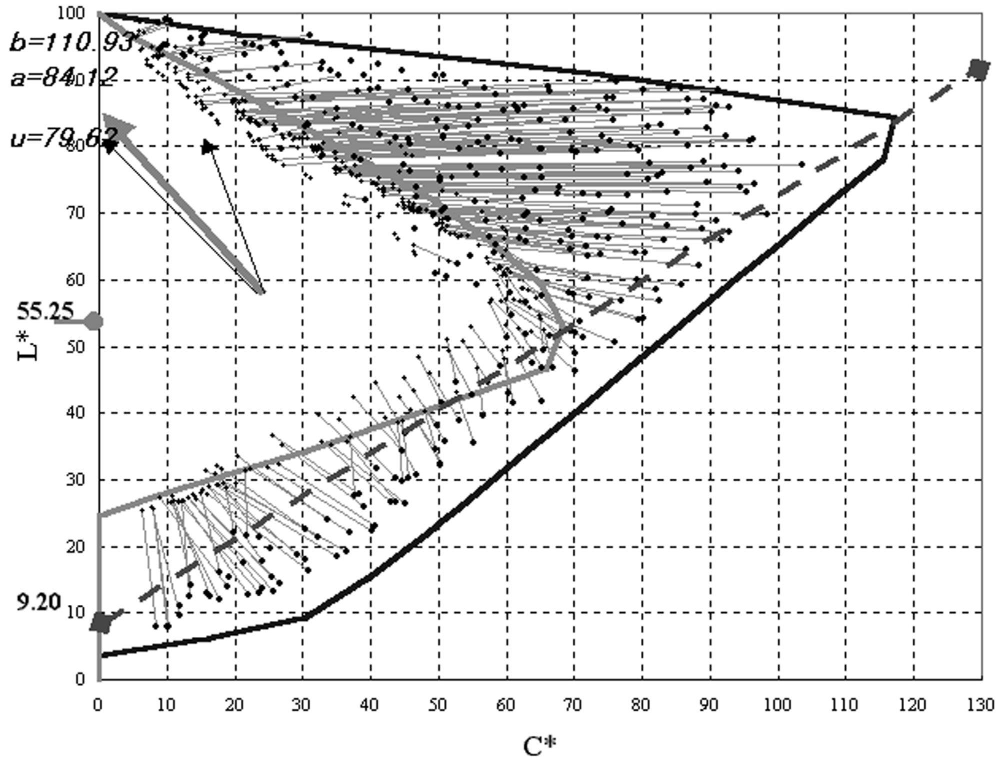

Figure 1. Gamut mapping from original (CRT, outside) to reproduction (ink jet printer, inside) for hue angles in the range 128–180° (green). The dashed line passes through cusps of the two gamuts. (Reproduced by courtesy of B.H. Kang[9])

general and gamut compression in particular. They asked observers to make interactive adjustments to an image in order to achieve the most satisfactory match to a simulated ink jet print, adjacent on the CRT display. Observers were able to adjust lightness and chroma independently in three regions (high, medium and low) and hue in six regions (red, yellow, green, cyan, blue and magenta). The results indicate clearly the trend of the translation vectors from original (CRT gamut) to reproduction (simulated print gamut), as shown in Fig. 1 for the green hue angle.

Inspection of Fig. 1 and of Kang's results for the other hues suggests the following aspects of observer preferences in gamut compression:

1. The chords between corresponding colors in the original and destination are of lengths proportional to the local distance between the two gamut boundaries. The directions of the chords change as a function of lightness, but they do not converge to a single point. Instead they are approximately normal to the destination gamut boundary for dark colors (below the destination cusp), horizontal for lightnesses between the two cusps, and inclined at an angle for light colors (above the source cusp).
2. The colors within the reproduction gamut are largely untouched, but the compression does extend a little inside the boundary, by about 10% at medium lightness values near the destination cusp. This suggests that the observers intuitively adopted a 'soft clipping' technique, in which chromatic gradations are preserved for colors near the gamut boundary while all other interior colors within a 'core gamut' are unchanged. Some variation in the thickness of the region between core and destination gamuts is evident when different hue angles are examined.

Description of the New Algorithm

A new gamut compression algorithm is proposed, which transforms values from a source color gamut into a destination color gamut, preserving the relationships between source and destination in a reversible manner. The color coordinate space is assumed to be perceptually uniform, with dimensions of hue, lightness and chroma. Hue angle is assumed to be invariant, so that the transformation maps pixel values within the lightness–chroma plane. The aim of the algorithm is to preserve the lightness and chroma differences of the source gamut, as far as possible, by means of an adaptive procedure applied consistently throughout the entire color space.

The algorithm is defined in four steps:

1. Construct the boundary of a 'core gamut', within which no colors are altered;
2. Define a distance metric along both source and core gamut boundaries;
3. Construct a set of mapping chords, connecting corresponding points;
4. Perform the gamut compression along the chords, with a 'soft-clip' function.

Step 1. Construct the Core Gamut Boundary

Let the source and destination gamuts can be expressed in a plane of constant hue, as shown in Fig. 2. Assume that both the source and destination gamuts are convex and that the source gamut lies entirely outside the destination gamut in the *L*–*C* plane. The maximum lightness of both gamuts is normalized to 100. Define L_S and L_D to be the minimum lightness of the source and destination gamuts respectively.

Let L_{Dm} be the lightness corresponding to the maximum chroma $C_{Dm} = \max_L(C_D)$, i.e., the cusp point, on

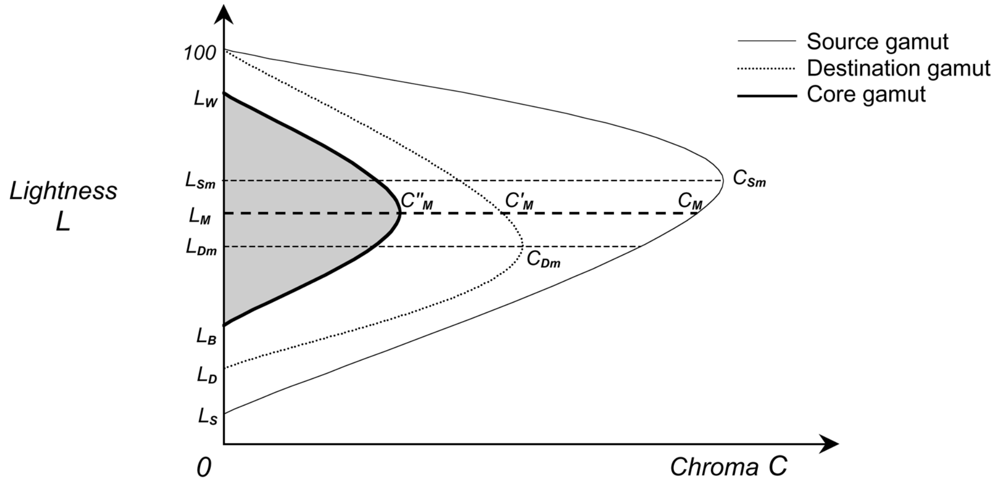

Figure 2. Gamut boundaries in the L–C plane.

the destination gamut, and let L_{Sm} be the lightness corresponding to the maximum chroma $C_{Sm} = \max_L(C_S)$ on the source gamut.

Define the core white point L_W to be the lightness of the cusp having the highest lightness over all hue angles (normally yellow):

$$L_W = \max_h(L_{Dm}) \tag{1}$$

Define the core black point L_B to be the lightness of the cusp having the lowest lightness over all hue angles (normally blue):

$$L_B = \min_h(L_{Dm}) \tag{2}$$

Define a chroma-scaling factor χ as the ratio of the chroma of the cusps of the destination and source gamuts at the given hue angle, restricted to a maximum value of 0.8. Note that this factor may vary for different hue angles:

$$\chi = \min(0.8, C_{D\max} / C_{S\max}) \tag{3}$$

Let the chroma of the destination gamut boundary be expressed as $C_D(L)$, a continuous single-valued function of L for $L_D \le L \le 100$. Then define the core gamut boundary $C_C(L)$ on the interval $L_B \le L \le L_W$ as follows:

$$C_C(L) = \chi C_D\left(L_D + (L - L_B)\left(\frac{100 - L_D}{L_W - L_B}\right)\right) \tag{4}$$

This function has a maximum value (cusp point) at lightness:

$$L_M = L_B + (L_{Dm} - L_D)\left(\frac{L_W - L_B}{100 - L_D}\right) \tag{5}$$

Step 2. Define a Distance Metric along Gamut Boundary

A key aspect of the new algorithm is its reliance upon definition of a distance metric, or 'path length', along the boundary of each gamut in the hue plane. Let the horizontal line $L = L_M$ intersect the source, destination and core gamut boundaries at points C_M, C'_M and C''_M respectively, as shown in Fig. 2. Define the parametric variable λ along the upper source gamut boundary on the interval $L_M \le L \le 100$, i.e., the light tonal region:

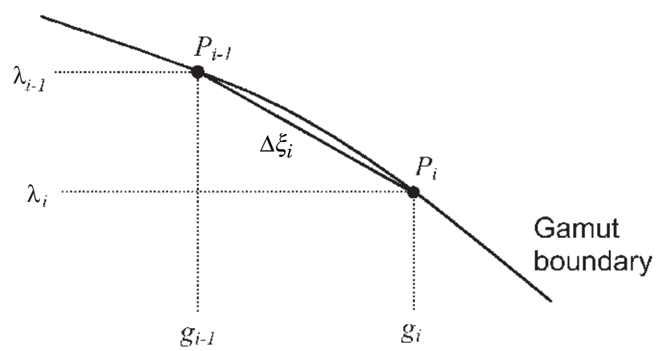

Figure 3. Approximation of gamut boundary path length by straight-line segments.

$$\lambda = \frac{L - 100}{L_M - 100} \tag{6}$$

Let the gamut boundary be represented by a chroma function $g(\lambda)$, $0 \le \lambda \le 1$, and let the path length along the source gamut boundary be represented by ξ. Then for a small change $\Delta\lambda$ in parametric lightness, the corresponding change in distance $\Delta\xi$ along the source gamut boundary is:

$$\Delta\xi = \sqrt{\Delta\lambda^2 + (g(\lambda + \Delta\lambda) - g(\lambda))^2} \tag{7}$$

When the gamut boundary is represented as a series of n straight-line segments, connecting points $P_i = (\lambda_i, g_i)$ for $i = 0 \ldots n$ along the boundary, each increment in path length can be conveniently calculated as the length of one segment, as shown in Fig. 3. The total length of the source gamut boundary from L_M to 100 is then approximated as:

$$\xi_S = \sum_1^n \Delta\xi_i = \sum_1^n \sqrt{(\lambda_i - \lambda_{i-1})^2 + (g_i - g_{i-1})^2} \tag{8}$$

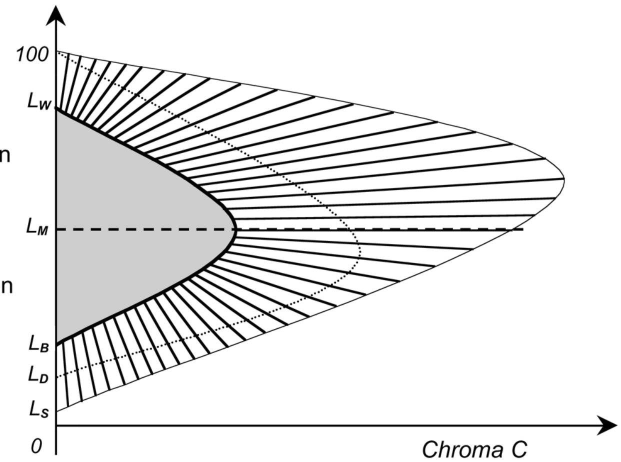

Figure 4. Chord directions in the two regions of gamut mapping.

In the limit as $\Delta\lambda_S \to 0$ the path length can be expressed by the Riemann integral:

$$\xi_S = \int_0^1 \sqrt{1+\left(\frac{dg_S}{d\lambda_S}\right)^2}\, d\lambda_S \tag{9}$$

Similar functions can be defined along the lower source gamut boundary on the interval $L_S \le L \le L_M$, i.e., the dark tonal region, by defining the parametric variable:

$$\lambda' = \frac{L - L_S}{L_M - L_S} \tag{10}$$

Similar functions again can be defined for the upper and lower regions of the core gamut boundary, i.e., above and below the line $L = L_M$.

Step 3. Construct the Mapping Chords

The L–C plane is divided into two regions, by range of lightness, as shown in Fig. 4. The mapping chords, which define the local vector directions (or 'flow lines') of gamut compression, are constructed separately in each region and generally have positive slopes in the upper region and negative slopes in the lower region.

Lower Region $L \le L_M$

Locate $n + 1$ points (L_{Si}, C_{Si}) for $i = 0..n$ at equal intervals of length ξ_S/n along the source gamut boundary. Similarly, locate $n + 1$ points (L_{Ci}, C_{Ci}) for $i = 0..n$ at equal intervals of ξ_C/n along the core gamut boundary. Typically $n = 16$.

Construct $n + 1$ chords connecting each (L_{Si}, C_{Si}) with the corresponding (L_{Ci}, C_{Ci}). The first chord ($i = 0$) will be vertical, connecting L_S with L_B along the L axis. The last chord ($i = n$) will be horizontal, connecting C_M with C''_M along the line $L = L_M$.

Upper Region $L \ge L_M$

Locate $n + 1$ points (L_{Si}, C_{Si}) for $i = 0...n$ at equal intervals of length ξ_S/n along the source gamut boundary. Similarly, locate $n + 1$ points (L_{Ci}, C_{Ci}) for $i = 0..n$ at equal intervals of ξ_C/n along the core gamut boundary. Typically $n = 16$.

Construct $n + 1$ chords connecting each (L_{Si}, C_{Si}) with the corresponding (L_{Ci}, C_{Ci}). The first chord ($i = n$) will be horizontal, connecting C_M with C'' along the line $L = L_M$ (this duplicates the last chord of the lower region). The last chord ($i = 0$) will be vertical, connecting 100 with L_W on the L axis.

Step 4. Perform the Gamut Compression

A point P in the source gamut with coordinates (L_P, C_P) is to be mapped to a point P' in the destination gamut with coordinates $(L_{P'}, C_{P'})$. The core gamut boundary represents the common sub-set of the two gamuts, within which no compression occurs. Thus $P' = P$ for all points within the core gamut.

Points outside the boundary of the core gamut are mapped as follows (see Fig. 5):

1. Find the two nearest mapping chords constructed in Step 3 on either side of P and, if not parallel, project them to intersect at point X.
2. Construct a new chord from X through P, intersecting the source, destination and core gamut boundaries at points P_S, P_D and P_C respectively.
3. In the case where the two nearest mapping chords are parallel, construct a new chord through P parallel to the other two.
4. Map P to P' along the chord using a 'soft clip' mapping function, based on the quadratic function $f(x) = x - x^2$, as shown in Fig. 6. Define the scalar variable z as follows:

$$\zeta = \frac{|P - P_C|}{2|P_S - P_C|} = \frac{\sqrt{(L_P - L_{P_C})^2 + (C_P - C_{P_C})^2}}{2\sqrt{(L_{P_S} - L_{P_C})^2 + (C_{P_S} - C_{P_C})^2}} \tag{11}$$

Then the mapping function is:

$$\begin{aligned} P' &= P && \text{for } \zeta \le 0 \\ P' &= P_C + 4(P_D - P_C)(\zeta - \zeta^2) && \text{for } 0 \le \zeta \le 1/2 \end{aligned} \tag{12}$$

Equation 12 can be implemented for mapping of the L and C components of P' as:

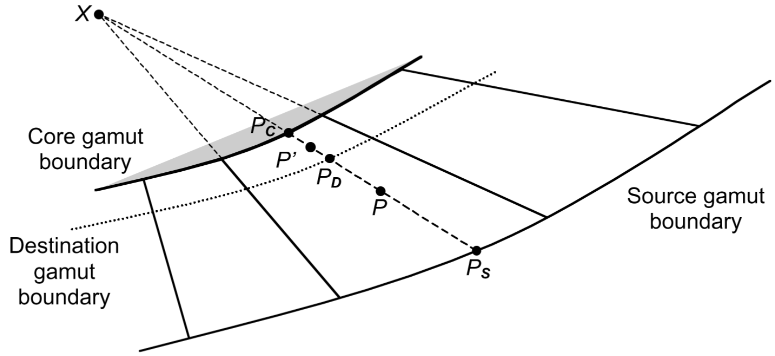

Figure 5. Construction of mapping chord.

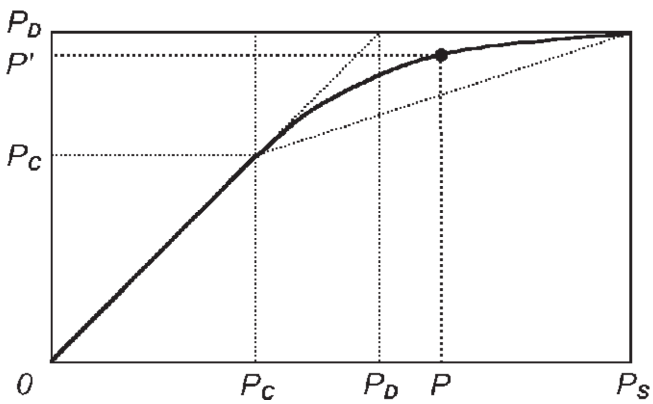

Figure 6. 'Soft-clip' gamut mapping function.

$$
\begin{aligned}
L_{P'} &= L_{P_C} + 4(L_{P_D} - L_{P_C})(\zeta - \zeta^2) \\
C_{P'} &= C_{P_C} + 4(C_{P_D} - C_{P_C})(\zeta - \zeta^2)
\end{aligned} \tag{13}
$$

In the rare case where $P_C = P_S$ (core and source boundaries intersect, resulting in a zero-length mapping chord), set $\zeta = 0$ so that $P' = P$.

Discussion

The new algorithm is described as topographic because of the way it delineates and preserves the local features of the source and destination gamut boundaries. The objective is to map colors between corresponding points of the two gamuts, while preserving the rendering of gradations in lightness and chroma to the greatest degree possible. The technique is in some ways comparable to the mapping along curved lines proposed by Herzog and Büring[10] but is more closely attuned to the actual conformations of both source and destination gamut boundaries.

The concept of the 'core gamut' is not entirely new. Spaulding and co-workers proposed a 'gamut morphing' technique,[11] in which a football-shaped volume in the center of the source color space was constrained to be reproduced using a colorimetric model. This volume was selected to include the convex hull of typical skintone colors. Katoh and Ito proposed an 'onion-peel' method,[12] in which an invariant 'colorimetric region' lies at the core of a series of 'virtual gamut boundaries'. This region was defined by scaling the destination (printer) gamut by a constant, the 'knee point' of the chroma compression function, and was coterminous with the destination gamut at both black and white points. Their results from experiments with four test images appeared to show a slight observer preference for a core gamut of scale 50%, although the trend was very dependent on image content. Sakamoto and Urabe also defined an 'original color region' in a similar manner,[13] but their compression knee point was determined by analysis of the chroma histogram.

The definition of the core gamut is critical to the effectiveness of the algorithm. We found in our initial trials that elevating the black point of the core gamut was important in preserving gradations in the shadow regions (dark tones) of the image. The region between the lower surfaces of the core and destination gamuts is where the dark colors are mapped from a source device having significantly lower black point than the destination device (as in Fig. 1, for example). If this region is too narrow, as may happen when the core and destination gamut black points are coterminous, the shadow detail is greatly reduced or lost, and may result in contouring of the reproduced image.

The quadratic 'soft clip' function shown in Fig. 6 preserves all colors unchanged within the core gamut, while mapping colors proportionally in the region between the core gamut and source gamut boundaries. This technique was first proposed by Stone and Wallace[14] and has since been employed by various researchers.[12,15,16] Values on the source gamut boundary are mapped onto the destination gamut boundary. The monotonic 1:1 mapping preserves the relationships between values and is thus invertible, subject to the quantizing of the image coding. The quadratic mapping is constrained to lie between the clipping (upper) and linear compression (lower) bounds.

The factors L_W, L_B and χ determine the size and shape of the core gamut, which is crucial for obtaining the best results. L_B is the darkest tone of the destination gamut that remains unchanged after gamut compression, whereas L_W is the lightest tone of the destination gamut that remains unchanged after gamut compression. χ controls the radius of the core gamut, and may vary with hue.

The new algorithm assumes that the source and destination gamut boundaries are known *a priori*, and that they can be accurately represented at each hue angle. Each gamut boundary should be well-behaved, i.e., the chroma C should be expressible as a smooth and continuous single-valued function of the lightness L. Implementation should account for cases where the maximum chroma of the source gamut occurs at higher, equal, and lower values of lightness than the maximum chroma of the destination gamut. Intermediate hue angles should be interpolated between $L–C$ planes of known hues, which should be at intervals of no greater than 30° (i.e., at least 12 hue angles defined in the full hue circle).

The algorithm with an appropriate change of mapping function should also perform well in cases where the destination gamut is larger than the source gamut, resulting in gamut expansion rather than gamut compression. It should also be applicable in mixed cases where the gamut boundaries cross over, resulting in gamut expansion in some regions and gamut compression in others.

Implementation of Algorithm

The topographic gamut compression algorithm (TOPO) was implemented in the ANSI C language in the Microsoft Visual C++ programming environment. Images were converted into the CAM97s2 color space,[17] and the gamut compression performed in the perceptual dimensions of lightness (J) and chroma (C). The source device was a Barco *Calibrator V* monitor, characterized by the Berns GOG model.[18] The reproduction device was a Hewlett Packard 895c ink jet printer, characterized by fitting third-order polynomials to colorimetric densities measured from 9 × 9 × 9 printed color patches.[19] Device gamut boundaries were described using the Flexible Sequential Line Gamut Boundary (FSLGB) method developed by Morovic.[1]

The black point for the source gamut was $L_S = 3.2$ and for the destination gamut was $L_D = 12.6$. The parameters L_W, L_B used in the construction of the TOPO core gamut (Eq. 1 and 2) took the values 90.8 and 27.1 corresponding to source gamut hue angles 93° and 271° respectively. The ratio χ (Eq. 3) ranged from 0.54 to 1.03 for different hue angles, as shown in Table I. The gamut boundary descriptors were calculated for 1° intervals of hue angle. The mapping chords generated by the TOPO algorithm for six principal hue angles are shown in Fig. 7. In the few cases where the destination gamut extended beyond the source gamut (yellow and cyan hue angles) the colors were limited to the source gamut, because this version of the algorithm did not support gamut expansion.

For comparison of the performance of the TOPO algorithm, three other algorithms used in the previous study[1] were also applied to each test image, using the same gamut boundary data:

- MDE Minimum ΔE clipping to gamut boundary, preserving hue (in $L–C$ plane).
- LLIN Linear compression of lightness and chroma.
- GCUSP Chroma-dependent lightness compression and linear compression to cusp.

Experimental Design

In the initial testing of an earlier version of the new TOPO algorithm,[20] the performance was evaluated using the same experimental technique and test images employed previously by Morovic and Luo.[1] The observer viewed the images simultaneously side-by-side, with the source image displayed on the CRT and two reproduction prints presented in a Verivide viewing booth under a simulated D65 light source, meeting ISO 3664 viewing condition P2. Other conditions were as described in the evaluation guidelines under development by CIE TC8-03.[21] The results indicated highest observer preference for the GCUSP algorithm, followed by TOPO version 2.

The latest version (3.3) of the algorithm, as described above, was evaluated in a second experiment. A different experimental technique was employed, with simulated prints displayed on the CRT instead of real prints in the viewing booth. The simulated prints were generated by applying the gamut compression algorithm to each image in JCh space, then applying the inverse *CAM97s2* model using the monitor viewing conditions to obtain a colorimetric (XYZ) image, and finally using the GOG model to convert to monitor RGB signals. Eight test images were chosen to contain a range of tone, chroma and pictorial content, as shown in Fig. 8. Four of the images (top row) were the same as used in the previous experiment. The second group of images (bottom row) were selected from the recent Japanese SHIPP standard.[22]

Thirty-two versions of the simulated prints were produced (8 images times 4 algorithms). Twenty-one observers, all students and staff of the Colour & Imaging Institute with normal color vision and ages ranging from 21 to 50, took part in the experiment. Each observer was therefore required to make ${}^4C_2 = 6$ pair-wise comparisons per image, a total of 48 judgements per session. The observer's task was to decide which of the two simulated prints displayed on the CRT was the better overall match to the original (source) image.

Results

For each image, the 4 × 4 matrices of comparison results for each observer were averaged over the 21 observers and transformed into z-scores. Using Case V of the method proposed by Thurstone,[23] the standard deviation of the z–score values is assumed to be $\sigma = 1/(2^{0.5})$ and the 95% confidence interval CI of a z–score value A can therefore be calculated[1] as:

$$\mathrm{CI} = A \pm 1.96\frac{\sigma}{\sqrt{N}} \tag{14}$$

For the number of images and observers used in this study, CI had a value of ±0.30 for the individual images, and ±0.11 overall. The results are given in Table II and the overall z-scores for the four algorithms are plotted in Fig. 10.

Table III gives the ranking of the performance of the four algorithms for the eight test images. TOPO was ranked highest for the four SHIPP images and second for the other four images. GCUSP and LLIN generally performed poorly, except for the BUS image, where GCUSP achieved the best compromise between maintaining chroma and rendering the surfaces of the solids. The MDE algorithm performed well overall, and was ranked best for the SKI, NAT and GIRL images. This can be explained for the NAT and GIRL images because it had little effect on the skin and landscape colors, which were largely inside the printer gamut boundary. For the SKI image, which has many high-chroma colors near the gamut surface, observers evidently preferred the preservation of chroma by the MDE algorithm to the graduated rendering of the other algorithms.

Conclusions

In summary, it was found that the TOPO (version 3.3) algorithm produced very pleasing visual results for all images tested. Its performance was particularly impressive for images containing large areas of high chroma colors, such as the textile image, but it also gave very good results for images of lower average chroma because

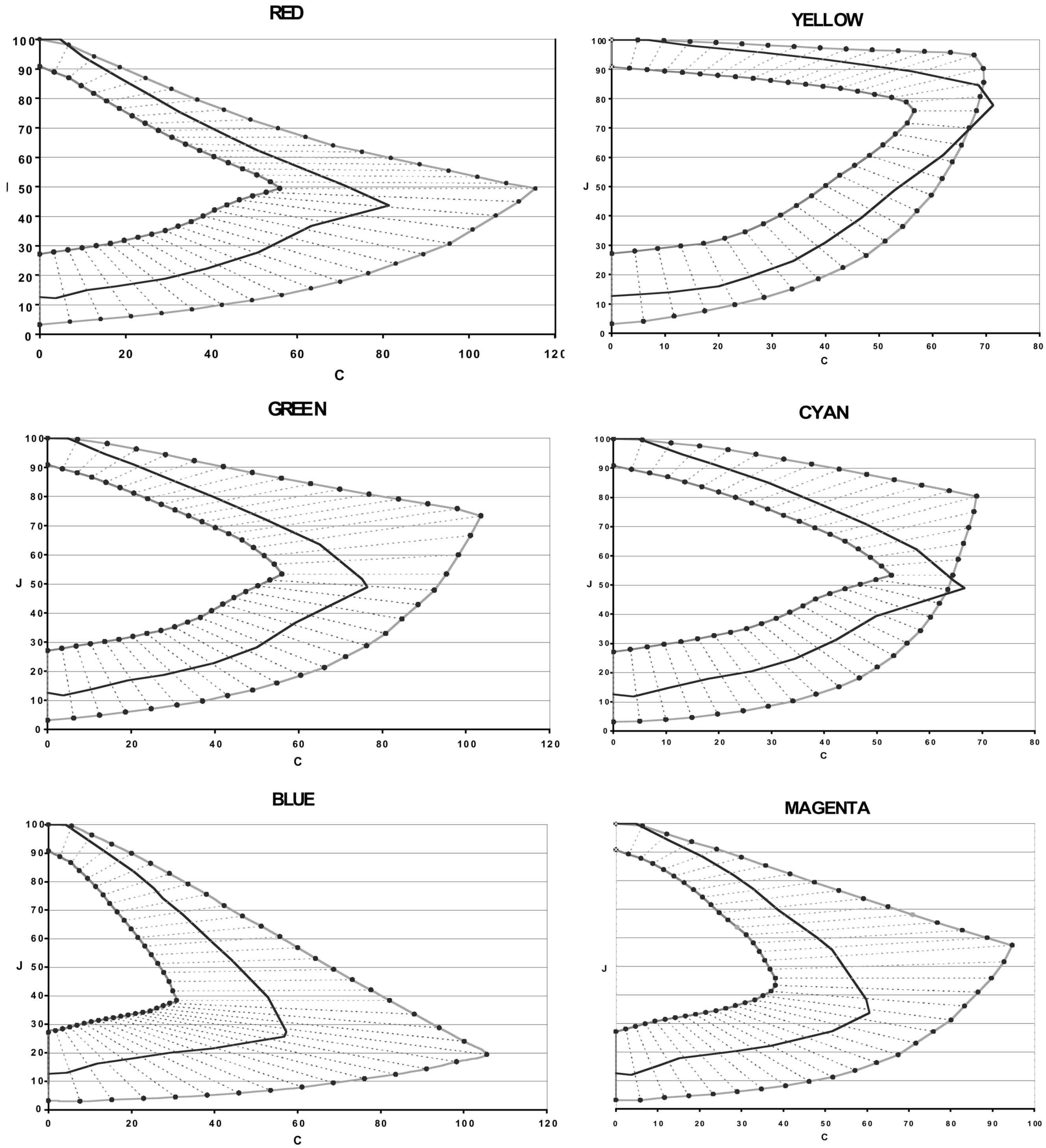

Figure 7. Construction of core gamut and mapping chords for six hue angles.

TABLE I. Ratio of Maximum Chroma (Destination/Source) for Different Hue Angles

Color	Hue angle	Destination cusp	Source cusp	Ratio C_{Dmax}/C_{Smax}
Red	26.1	81.3	115.5	0.70
Yellow	103.7	71.3	69.5	1.03
Green	143.5	76.5	103.6	0.74
Cyan	198.2	66.6	68.9	0.97
Blue	267.1	56.7	105.5	0.54
Magenta	332.8	60.6	94.7	0.64

Figure 8. The eight test images: SKI, BUS, NAT, GIRL (top row); BRIDE, HARBOR, TEXTILE, METAL (bottom row).

Figure 9. Experimental setup for comparison of the original image (center) with simulated prints (left and right).

TABLE II. Z-scores from Pair-Comparison Experiment for Each Image and Algorithm

	TOPO3	GUSP	MDM	LLIN
SKI	0.21	–0.04	0.24	–0.42
BUS	0.46	0.69	–0.47	–0.68
NAT	0.75	–0.88	0.98	–0.85
N1a	0.90	–0.83	1.14	–1.22
P1	0.98	–1.05	0.76	–0.69
P2	1.23	–1.07	1.18	–1.34
P3	0.67	–0.94	0.13	0.15
P4	0.92	–0.94	0.82	–0.80
Overall	0.59	–0.42	0.40	–0.57

TABLE III. Ranking of Algorithm Performance versus Image (1 = best, 4 = worst)

Image	TOP3	GCUSP	MDE	LLIN
Ski	2	3	1	4
Bus	2	1	3	4
Nat	2	4	1	3
Girl	2	3	1	4
Bride	1	4	2	3
Harbor	1	3	2	4
Textile	1	4	2	3
Metal	1	4	2	3
Average	1	3	2	4

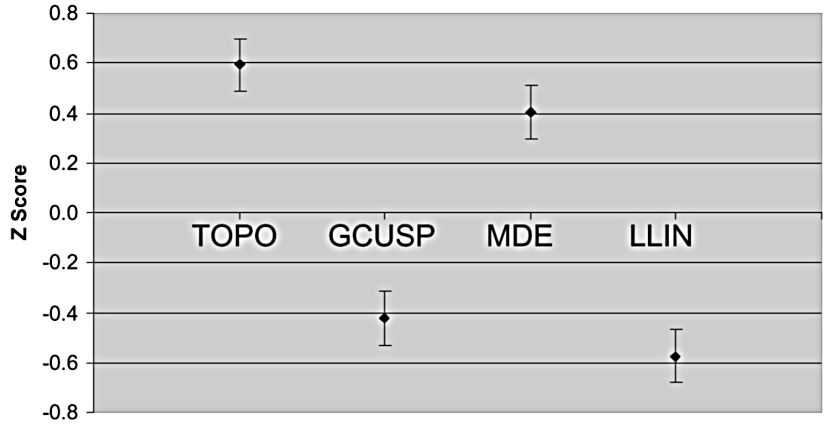

Figure 10. Overall z-scores for the four algorithms.

it maintained optimum contrast. Only for the business graphics image, which contains large regions of color near the gamut boundary, was its performance marginal because the mapping chords for medium to high lightness were close to horizontal (see Fig. 7 for example), producing a substantial reduction in the chroma of the reproduction.

We believe that the TOPO algorithm has potential for further refinement in several ways. First, improved definition of the core gamut will permit better adaptation to the relative sizes and shapes of both source and destination gamut boundaries. Second, knowledge of the statistics of individual images will permit the algorithm's behavior to be optimized by mapping only those colors in an image outside the destination gamut boundary. Third, an augmented version of the algorithm for gamut expansion will permit chroma to be enhanced in regions of color space where the destination gamut is larger than the source gamut.

References

1. J. Morovic and M. R. Luo, Developing Algorithms for Universal Color Gamut Mapping, *Color Imaging: Vision and Technology*, L.W. MacDonald and M.R. Luo, Eds. John Wiley and Sons, 1999, pp. 253–282.
2. K. M. Braun, R. Balasubramanian and R. Eschbach, Development and Evaluation of Six Gamut-Mapping Algorithms for Pictorial Images, *Proc. IS&T/SID's Seventh Color Imaging Conf.*, IS&T, Springfield, VA, 1999, pp. 144–148.
3. L. W. MacDonald and J. Morovic, Assessing the Effect of Gamut Compression in the Reproduction of Fine Art Paintings, *Proc. IS&T/SID's Third Color Imaging Conf.*, IS&T, Springfield, VA, 1995, pp. 194–200.
4. J. Morovic and P. L. Sun, Methods for Investigating the Influence of Image Characteristics on Gamut Mapping, *Proc. IS&T/SID's Seventh Color Imaging Conf.*, IS&T, Springfield, VA, 1999, pp. 138–143.
5. L. W. MacDonald, Gamut Mapping in Perceptual Color Space, *Proc. IS&T/SID's First Color Imaging Conf.*, IS&T, Springfield, VA, 1993, pp. 193–196.
6. A. J. Johnson and co-workers, Color Appearance Research for Interactive System Management and Applications (CARISMA), *Report WP2-19: Color Gamut Compression* (1992).
7. P. J. Green and M. R. Luo, Developing the CARISMA Gamut Mapping Model, *Proc. Color Image Science 2000*, University of Derby, Derby, UK, 2000, pp. 244–256.
8. E. Pirrotta, T. Newman and L. Lavendel, A "Universal" Gamut Mapping Algorithm?, *Proc. Color Image Science 2000*, University of Derby, Derby, UK, 2000, pp. 313–320.
9. B. H. Kang, M. S. Cho, J. Morovic, and M. R. Luo, Gamut Compression Algorithm Development Using Observer Experimental Data, *Proc. Color Image Science 2000*, University of Derby, Derby, UK, 2000, 183–196.
10. P. G. Herzog and H. Büring, Optimizing Gamut Mapping: Lightness and Hue Adjustments, *J. Imaging Sci. Technol.* **44**(4), 334–342 (2000).
11. K. E. Spaulding, R. N. Ellson and J. R. Sullivan, UltraColor: A new gamut mapping strategy, *Proc. SPIE* **2414**, 61–68 (1995).
12. N. Katoh and M. Ito, Applying Non-linear Compression to the Three-Dimensional Gamut Mapping, *J. Imaging Sci. Technol.* **44**(4), 328–333 (2000).
13. K. Sakamoto and H. Urabe, Development of XYZ/sRGB-SCID and Color Gamut Compression, *Proc. IS&T/SID's Seventh Color Imaging Conf.*, IS&T, Springfield, VA, 1999, pp. 212–216.
14. M. C. Stone and W. E. Wallace, Gamut Mapping Computer Generated Imagery, *Proc. Graphics Interface '91*, ACM, New York, 1991, pp. 32–39.
15. T. Hoshino and R. S. Berns, Color Gamut Mapping Techniques for Color Hard Copy Images, *Proc. SPIE* **1909**, 152–164 (1993).
16. E. Granger, Gamut Mapping for Hard Copy using the ATD Color Space, *Proc. SPIE* **2414**, 27–35 (1995).
17. C. Li, M. R. Luo and R.W.G. Hunt, The CAM97s2 model, *Proc. IS&T/SID's Seventh Color Imaging Conf.*, IS&T, Springfield, VA, 1999, pp. 262–263.
18. R. S. Berns, Methods for characterizing CRT displays, *Displays* **16**(4), 173–182 (1996).
19. T. Johnson, Methods for characterizing color printers, *Displays* **16**(4), 193–202 (1996).
20. L. W. MacDonald, J. Morovic and K. Xiao, A Topographic Gamut Mapping Algorithm based on Experimental Observer Data, *Proc. IS&T/SID's Eighth Color Imaging Conf.*, IS&T, Springfield, VA, 2000, pp. 311–317.
21. http://www.color.org/tc8-03/
22. K. Sakamoto and H. Urabe, Standard High Precision Pictures: SHIPP, *Proc. IS&T/SID's Fifth Color Imaging Conf.*, IS&T, Springfield, VA, 1997, pp. 250–244.
23. L. L. Thurstone, A Law of Comparative Judgment, *Psych. Rev.* **34**, 273–286 (1927).