Continuity of gamut mapping algorithms

Peter Zolliker
Klaus Simon
Empa, Swiss Federal Laboratories for Materials Testing and Research
Laboratory for Media Technology
CH-8600 Dübendorf, Switzerland
E-mail: peter.zolliker@empa.ch

Abstract. *The design of a gamut mapping algorithm (GMA) is always a compromise between preserving different competing aspects such as color, contrast, and lightness. A natural requirement of a GMA is that the algorithmic treatment of this competition has to avoid any additional artifacts such as discontinuities or loss of contrast. In this work, several common gamut mapping algorithms are studied from this aspect, resulting in the observation that problems with geometric discontinuities are widespread. For the assessment of the phenomena induced by local mapping properties, an algorithmic test is developed and applied. This new test supports both the quality check of existing as well as the development of new GMAs.* © 2006 SPIE and IS&T. [DOI: 10.1117/1.2177631]

1 Introduction

We investigate the quality of gamut mapping algorithms (GMA), with special emphasis on the mapping behavior of local features, in particular continuity and contrast losses. A special measure is proposed to make these quality-relevant features assessable, for which psycho-visual tests may not be sensitive.

Mapping a given source color space to a destination space, so-called gamut mapping, is a key feature in every color reproduction process. With the increased use of digitalization in image processing devices, the scientific interest in gamut mapping concepts has grown drastically. In principle, gamut mapping algorithms have to preserve as much as possible of the original color appearance. This is always a complex mixture of different aspects such as color saturation, contrast, and lightness. A lot of different algorithms were proposed with special emphasis on one or the other aspect. See Morovic[1] for a recent overview.

Modern algorithms try to optimize several of these aspects simultaneously, e.g., lightness and saturation in GCUSP[2] and SGCK.[3] This typically results in the use of different mapping strategies in different color regions. In general, such algorithms achieve a higher quality. However, the increasing complexity of the algorithmic design makes it difficult to prove that the algorithms do not produce visible artifacts. This is particularly the case for the assessment of the output quality by only the traditional method,

namely the application of the GMA to a small set of established test images judged by a small number of professional observers.

On the one hand, a set of common test images has the advantage of simpler comparability between different studies. Recently, to improve the comparability, the technical committee of CIE worked out and published corresponding guidelines.[3] These recommendations strive for a *universal gamut mapping algorithm*, a device independent solution with excellent results for an arbitrary input image. The discussion is still going on as to whether it is possible to identify such an algorithm or at least a universal mapping concept. The CIE committee hopes for active participation in applying and further developing these guidelines. We understand our work to be a contribution to this discussion, in particular on the question of how to measure the quality of a GMA.

On the other hand, a substantial increase of the number of test images, similar to the assessment of compression algorithms,[4] may help to increase the credibility of the test, but this does not ensure that local mapping problems are not overlooked. During the implementation of a set of GMAs, including those described in the CIE recommendations, we became aware that a series of local mapping problems were not the result of the implementation itself, but due to the algorithmic design. This problem seemed to have escaped the judgment of the psycho-visual tests. What is needed is a systematic method to study the local mapping behavior of a GMA, covering the entire gamut.

We first describe a set of gamut mapping algorithms within a common framework of basic mapping steps. In this investigation, we cover GMAs using device-to-device mapping. However, an application of the concepts to GMAs using image-to-device mapping is straightforward. In the second part, we introduce a test to assess the local mapping behavior with respect to visual artifacts. It basically consists of exploring the source gamut by mapping a sphere of radius $\Delta E = 1$ around a color. From the resulting mapped solid, we can determine its volume, its surface, and the largest and smallest mapped diameter. All those measures are good indicators of the local mapping behavior and can be collected in histograms. In the third part, we present results from the application of this test to the described GMAs for a variety of destination gamuts. Results are presented and discussed for newspaper, photographic, inkjet, and offset printing. In these applications, robust gamut
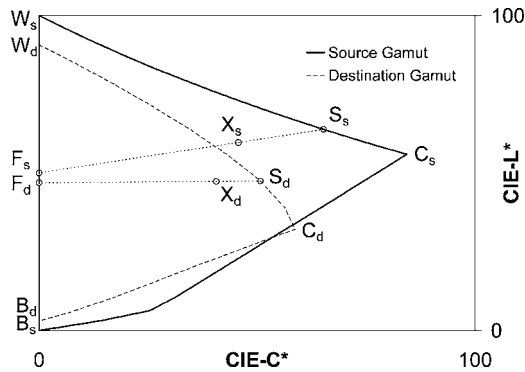
Fig. 1 General mapping scheme from a device source gamut to a destination gamut.

mapping is especially important, because the high volume of images processed does not allow for elaborate control of every individual image.

2 Gamut Mapping Algorithms

2.1 Basic Mapping Steps

A wide variety of gamut mapping algorithms can be described by a few basic mapping steps. To be able to investigate different GMAs, we implemented several of them. Our generic concept allows us to describe many of the published GMAs within a consistent notation. Note that algorithms make implicit assumptions on the shapes of the gamuts to be mapped. For three-color systems, this includes the notation of primary (one color fully saturated, the other two colors absent) and secondary colors (two colors fully saturated and the third color absent), as well as cusps (one color at maximum, a second color at minimum, the third color free). Primary and secondary colors are found on distinct corners of the gamut boundary, and the cusps form edges connecting the primary and secondary colors. It is straightforward to use these definitions also for a CMYK color system, as black (K) is used mainly to replace the common part of C, M, and Y colors.

Once a color is transformed to a device independent color space, for instance CIELAB, we can describe the mapping as follows (see Fig. 1). In our notation, the indices s and d refer to source and destination, respectively. Furthermore, we use W for the white point, B for the black point, and C for the cusp of a gamut. The term X is used for a general color and S is used for a color on the gamut's border. For every given color X_s, a focal point F_s can be assigned. Its exact position depends on the specific GMA. A ray with F_s as its origin and passing through X_s is intersected with the source gamut border, yielding S_s. Every GMA must have a defined strategy of mapping border colors S_s to corresponding destination colors S_d. Finally, the mapped color X_d can be found on the line $\overline{S_d F_d}$. Its position is defined by the 1D mapping function of the line $\overline{S_s F_s}$ to the line $\overline{S_d F_d}$. For some algorithms, an additional clipping step has to be added. This is the case if after the prior operations some of the mapped colors lie outside the destination gamut.

In the following, we list six basic mapping operations used to describe a large class GMAs. A specific algorithm may not make use of all those steps.

Working color space. The transformation from source color space into gamut mapping working color space, and from there to the destination color space, needs to be defined. We assume that these transforms include at least partial compensation for color appearance, and that the color space has a good distance measure as well as good predictors for hue, chromaticity, and lightness. Because most of the used GMAs are designed to work in CIELAB, we used this color space to allow for a acceptable comparison. However, the presented tests and basic results are independent of the working color space used, and the results are valid also for other color spaces.

Lightness compression. The source gamut is compressed in the L direction. As a result, W_s is mapped to W_d and B_s to B_d. The compression may, for example, be linear,[5] described by a sigmoidal shaped function,[6] or depend on chroma C^{*}.[2]

Focal points. Focal points are often used to define the direction of the color mapping. They may be fixed,[7] but may also depend on hue angle[2] or lightness.[8,9] In most cases, the mapping of the focal points is trivial, as they map to themselves, i.e., $F_s = F_d$.

Border colors. For most GMAs, the mapping of the colors at the gamut surface also plays an important role in the mapping of all other colors. For a device gamut completely enclosed by the source gamut, the source gamut boundary is mapped on the destination gamut boundary. If the destination gamut has parts exceeding the source gamut, the source gamut boundary colors are usually left unchanged for those parts. An exception are GMAs allowing for gamut expansion.[10] A typically simple mapping strategy is the use of a straight line from a source border color S_s to a focal point F_s, intersecting the surface of the destination gamut, yielding S_d. Alternatively, the use of curved lines was proposed.[11]

Mapping function. Once the mapping of the border colors is known, the mapping of the bulk colors is defined along a 1-D mapping function. This mapping is usually done on straight lines, but also curved lines can be used. Different mapping types are used such as linear, clipping, or a variety of intermediate functions. Examples are shown in Fig. 2.

Clipping. Depending on the GMA, some colors may still lie outside the destination gamut even after application of the previous mapping steps. In this case, they have to be clipped to the destination gamut border. Different methods are used such as minimum distance mapping in 2- or 3-D and clipping along straight lines toward a focal point.

2.2 Description of Gamut Mapping Algorithms Used in This Work

2.2.1 Gaussian lightness compression and cusp scaling

This algorithm, called GCUSP, was first described in the thesis of Morovic[2] and was later used in several GMA
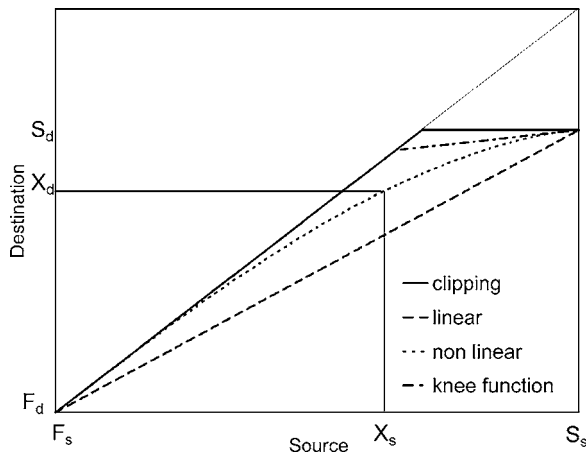
Fig. 2 Typical 1-D mapping functions: clipping, linear compression, knee function (SGCK), and nonlinear compression (SGDA).

evaluations. It consists of an initial chroma-dependent lightness compression followed by a compression toward the lightness of the cusp on the lightness axis. We found two variants of the GCUSP algorithm in the literature. At first, one[12] only compresses colors that are outside the destination device's gamut. This leads to a pronounced discontinuity at colors that are close to the destination gamut border. A second variant of the algorithm, called GCUSP99 by Pirrotta, Newman, and Lavendel[13] applies the same compression to all colors. We use the terms GCUSP1 for the first variant and GCUSP2 for the second variant, whenever a distinction is appropriate.

In terms of the mapping steps defined before, the algorithm can be described as follows.

1. CIELAB is used as working color space with full adaption of the white point.
2. Lightness compression is linear but chroma dependent, using the following function:

$$p_c = 1 - \left[\frac{C^{*3}}{(C^{*3} + 5 \times 10^5)} \right]^{1/2}. \qquad (1)$$

   Full lightness compression ($p_c = 1$) is applied to the achromatic axis and its degree is decreased as chroma increases.
3. Focal points lie on the achromatic axis. L are hue dependent and correspond to the lightness of the cusp $F_s = F_d$.
4. If a border color S_s is outside the destination gamut, it is mapped to S_d, defined by the intersection of the line $\overline{S_s F_s}$ with the destination gamut, otherwise $S_d = S_s$.
5. The mapping function uses linear compression. GCUSP1 applies linear compression only to out-of-gamut colors, GCUSP2 applies it for all colors.
6. Clipping is not needed.

2.2.2 *Sigmoidal lightness mapping and cusp knee scaling*

This algorithm, called SGCK, is one of the two GMAs recommended by CIE to be used as a reference algorithm in

psycho-visual tests. The details of this algorithm are described in the CIE guidelines.[3] This algorithm is a combination of GCUSP and the sigmoidal lightness mapping and cusp knee scaling proposed by Braun and Fairchild.[6] One difficulty for its application to the newspaper printing application was that the parameters for sigmoidal lightness mapping are not defined for L_{minr} values larger than 20. For the newsprint gamut, which has an L_{minr} of about 40, the values for the maximum available L_{minr} were used.

In terms of the mapping steps defined before, the algorithm can be described as follows.

1. CIELAB is used as working color space with full adaption of the white point.
2. Lightness compression is chroma dependent as in GCUSP and uses nonlinear compression based on a discrete cumulative normal function.
3. Focal points lie on the achromatic axis. L is hue dependent and corresponds to the lightness of the cusp $F_s = F_d$.
4. If a border color S_s is outside the destination gamut, it is mapped to S_d defined by the intersection of the line $\overline{S_s F_s}$ with the destination gamut, otherwise $S_d = S_s$.
5. The mapping function uses a knee-shaped compression.
6. Clipping is not needed.

2.2.3 *Hue preserving minimal ΔE*

The details of this algorithm, called HPMinDE, are described in the CIE guidelines.[3] It keeps colors inside the destination gamut unchanged, and maps outside colors to the destination gamut border. This is done by minimum distance clipping within planes of constant hue.

In terms of the mapping steps defined before, the algorithm can be described as follows.

1. CIELAB is used as working color space without adaption of the white point.
2. No lightness compression is applied.
3. Focal points are not needed.
4. Border colors are not needed.
5. Mapping function is not needed.
6. For a color C_s outside the destination, its mapped color is determined by the closest border color S_d with the same hue.

2.2.4 *Three-dimensional minimal ΔE*

This algorithm, called MinDE3D, minimizes 3-D color-by-color differences between the original and reproduction. A mathematical method of the minimum ΔE clipping is given by Morovic and Sun.[14] In this work, we used ΔE_{ab}^* as a distance measure. The results and systematic problems discussed next are basically the same also if other distance measures are used.

The minimum distance can be performed with any of the known color difference formulas such as ΔE_{ab}^* ΔE_{94},[15] ΔE_{CMC},[16] and ΔE_{BFD}.[17] Katoh and Ito[18] have suggested a weighted color difference formula. In terms of the mapping steps defined before, the algorithm can be described as follows.
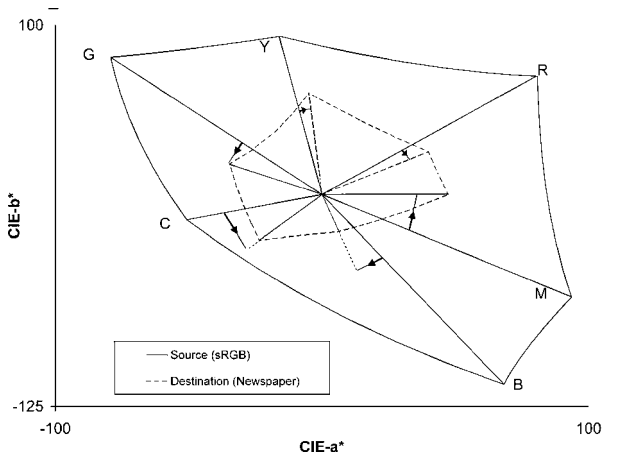
Fig. 3 Hue shift of CARISMA and SGDA algorithms: Full hue shift for primary and secondary colors.
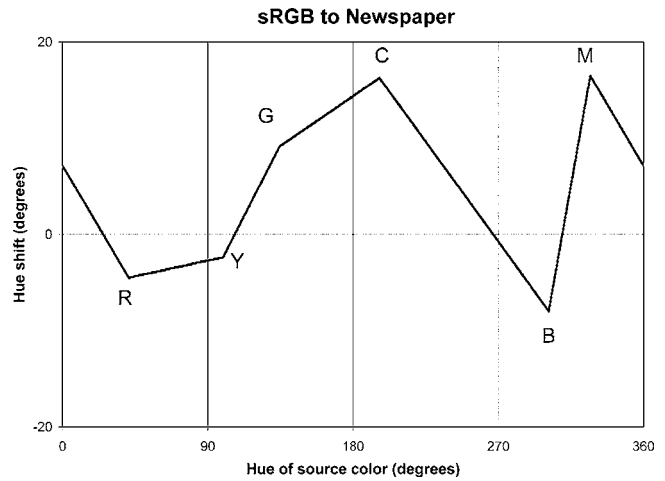

Fig. 4 Hue shift of CARISMA and SGDA algorithms: interpolated hue shift as a function of hue.

1. CIELAB is used as working color space without adaption of the white point.
2. No lightness compression is applied.
3. Focal points are not needed.
4. Border colors are not needed.
5. Mapping function is not needed.
6. For a color C_s outside the destination, its mapped color is determined by the closest border color S_d using the chosen distance measure.

2.2.5 CARISMA

The approach used for this algorithm, developed in a project called "color appearance research for interactive system management and application" (CARISMA), was to ask color experts for professional image reproduction, and capture what they do.[5] The special characteristics of this algorithm are that it suggests changes to hue and that it uses different mapping methods for different hue sections, depending on the relative shapes of the original and reproduction gamuts. A hue shift is defined for the hue planes containing the primary and secondary colors of the source gamut half way toward the hue of the primary and secondary colors of the destination gamut (see Fig. 3). Hue shift of intermediate colors is determined by linear interpolation (see Fig. 4).

In terms of the mapping steps defined before, the algorithm can be described as follows.

1. CIELAB is used as working color space with full adaption of the white point.
2. Lightness compression is linear.
3. For the definition of focal points F_d, three cases are used, depending on the relative position of the hue shifted source cusp C'_s and the destination cusp C_d (see Figs. 5 and 6). Note that small changes in the form of the gamuts can result in very different positions of the focal point.

   - The source gamut does enclose the destination gamut, and the intersection of the line $\overline{C'_s C_d}$ with the achromatic axis is within the destination gamut. Use this intersection as F_d.

- The source gamut does enclose the destination gamut, and the intersection of the line $\overline{C'_s C_d}$ with the achromatic axis is outside the destination gamut. Use the point on the achromatic axis with the same lightness as the cusp C_d as F_d.
- The source gamut does not enclose the destination gamut: F_d has $L=0$ and C^* equals half the chroma of the destination cusp C_d.

4. Border colors S_s are first shifted in hue to S'_s. If S'_s is outside the destination gamut, S_d is the intersection of the line $\overline{S'_s F_d}$ with the destination gamut, otherwise $S_d = S'_s$.
5. The mapping function is linear. However, for some mapped lines case 3 has no valid solution [see Fig. 6(b)], and an additional mapping rule has to be defined.
6. Clipping is not needed.

Due to the fact that a description of the CARISMA algorithm is not complete,[2,19] we did not include CARISMA
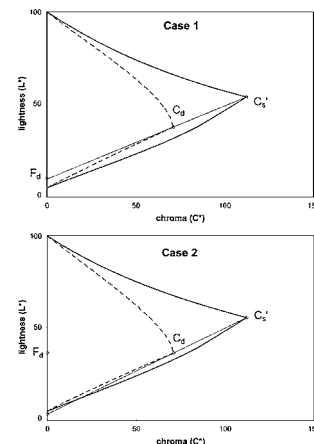

Fig. 5 Cases 1 and 2 of the CARISMA mapping. Note that small changes in the form of the gamuts can result in very different positions of the focal point.
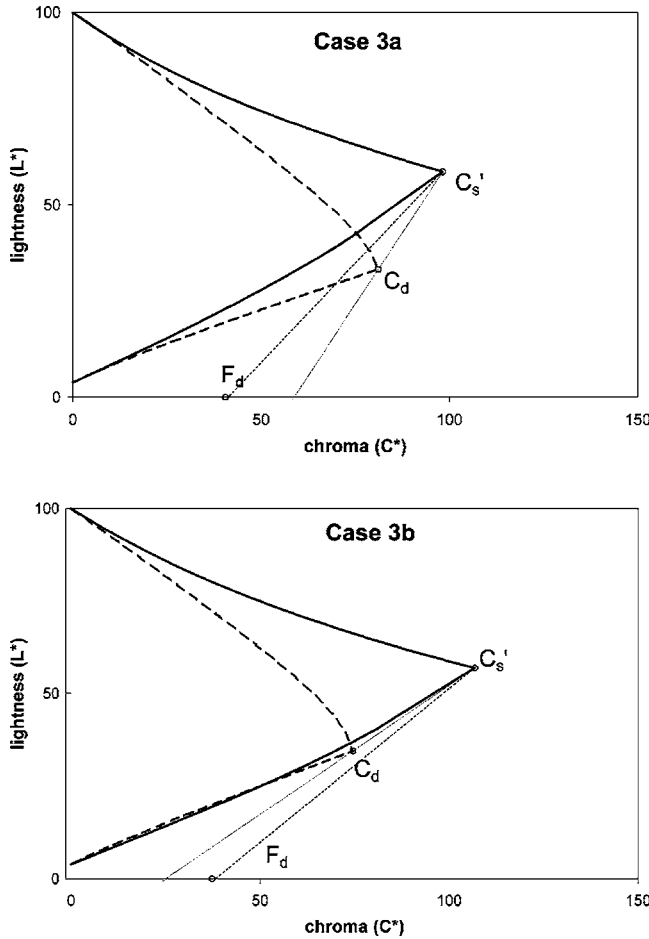
Case 3a

Case 3b

Fig. 6 Cases 3 of the CARISMA mapping. Case 3(b) shows a configuration with undefined mapping for colors close to C_s.

in our continuity assessment. Interesting aspects concerning continuity are described in the discussion section later on.

2.2.6 *Enhanced Schläpfer*

The Schläpfer algorithm[20], called SCHLP, was one of the first mapping algorithms designed for the newspaper printing gamut. It has roots similar to CARISMA to model the mapping based on the experience of printing experts. The original algorithm separates the lightness mapping completely from the chroma mapping. The chroma mapping is described only in the 2-D projection onto the chromaticity plane, leading to an incomplete description of the 3-D mapping. Some mapped colors are still outside the destination gamut after the mapping. Thus, an additional step had to be introduced.

In terms of the mapping steps defined before, the algorithm can be described as follows.

1. CIELAB is used as working color space with full adaption of the white point.
2. Lightness compression is linear.
3. The focal points F_s lie on the gray axis with the same lightness as X_s.
4. No special mapping of border colors is needed; instead, lightness is conserved and chroma is

scaled with a hue dependent ratio k defined by the ratio of the chroma of the cusps:

$$k = \min\left(\frac{C_d}{C_s}, 1\right). \tag{2}$$

5. The mapping function uses a nonlinear compression, defined by the following equation:

$$X_d = kX_s\left[\frac{X_s}{C_s}(k-1) + 2 - k\right]. \tag{3}$$

6. Clipping is done along lines toward the center of the gray axis.

2.2.7 *Smooth gamut deformation algorithm*

This new algorithm, called SGDA, incorporates two main features of the CARISMA algorithm.[5] The first uses the same explicit hue shift defined by the hue differences of the primary and secondary colors of the source and destination device. The second is the goal to map the source cusp close to the destination cusp. Different from CARISMA, the mapping is done with a nonlinear gamut compression. Furthermore, the algorithm is designed to map to the full destination gamut, thus allowing for gamut expansion as well.

In terms of the mapping steps defined before, the algorithm can be described as follows.

1. CIELAB is used as working color space without adaption of the white point.
2. No separate lightness compression step is needed. Compression is done implicitly with step 5.
3. A hue independent focal point on the gray axis is used. Note that F_s is different from F_d, because no extra lightness compression is performed. For our implementation, we used the center of the gray axis as a focal point, i.e., $F_s = (W_s + B_s)/2$ and $F_d = (W_d + B_d)/2$.
4. The mapping of border colors consists of two steps. First, a hue shift as defined for CARISMA is applied. The amount of hue shift p_1 can be varied from $p_1 = 0$ to full shift $p_1 = 1$. We used a setting of $p = 0.5$. The determination of the position of S_d on the destination gamut cut is done by equalizing the normalized areas A_s and A_d bounded by $F_s - W_s - S_s$ and $F_d - W_d - S_d$, respectively. The normalization is done using the full areas bounded by the achromatic axis and the gamut border (see Fig. 7).
5. The mapping function is nonlinear of the following form:

$$X_d = p_2\left\{S_d\tanh\left[\frac{S_s}{S_d}\tanh^{-1}\left(\frac{X_s}{S_s}\right)\right]\right\}$$
$$+ (1 - p_2)\left(\frac{X_s S_d}{S_s}\right). \tag{4}$$

The parameter p_2 allows us to adapt the degree of nonlinearity. For our study, we used $p_2 = 0.75$. For $p_2 = 1$, this function simulates the mapping behavior of photographic paper: its saturation curve can be approximated with a tanh-type function.[21]
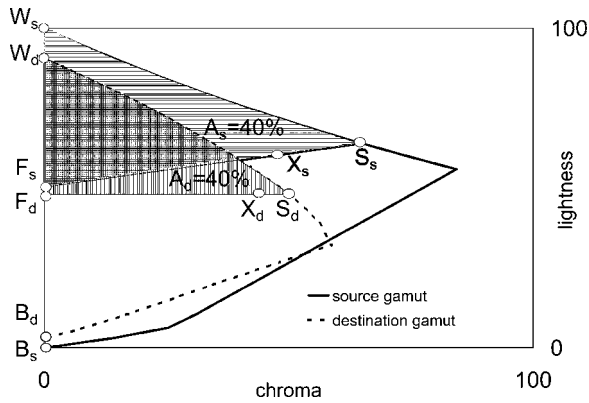
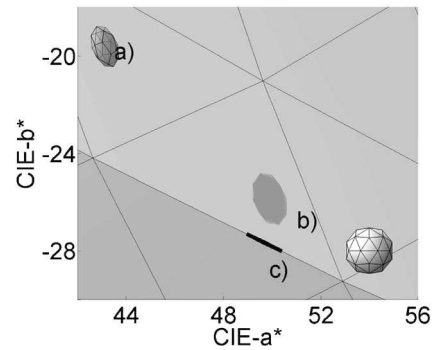**Fig. 7** SGDA: mapping scheme for surface colors using normalized area.

Fig. 8 Mapping of a unit sphere using different algorithms: (a) SGDA (mapped to an ellipsoid), (b) HPMinDE (clipped to ellipse), and (c) MinDE3D (clipped to a line on an edge of the destination gamut).

6. Clipping is not needed.

3 Experimental

3.1 Continuity Assessment

Two methods were developed to test the local mapping behavior of gamut mapping algorithms. The first determines the maximum ΔE distance of two mapped neighboring RGB colors and aims to detect discontinuities of the mapping. The second more elaborate test investigates how unit ΔE spheres are mapped. This test also extracts features assessing contrast reduction and contrast loss.

3.1.1 Mapping of neighboring RGB colors: dRGB test

The aim of this test is to determine to what extent a gamut mapping algorithm may cause artifacts in smooth color gradients in images encoded with 3×8 bits. If the difference of two colors in an original image is below the visible limit, the mapped colors should also be below this limit. Neighboring sRGB colors (i.e., $\Delta RGB = 1$ in one, two, or three colors) are good candidates for such a test, because their difference can be assumed to be close to or below the visible limit. This is justified by the fact that most RGB color spaces show maximal differences of neighboring color codes below $\Delta E = 2.5$.[22]

The test method, called the dRGB-test, can be described by the following steps.

1. Select a pair of neighboring 8-bit RGB colors.
2. Calculate the mapped RGB colors (numerically accurate without 8-bit quantization).
3. Calculate CIELAB values of the mapped colors and their distance ΔE_{ab}^*.
4. Keep the color pairs with the largest ΔE.
5. Repeat this test for a large set of color pairs.

This test can be made exhaustive: all combinations of neighboring 8-bit RGB pairs could be tested. For an overview, a subset of some million random neighboring color pairs is sufficient. The resulting color pair with the maximum ΔE can be used to construct an image containing only these two colors, allowing us to visualize to what extent artifacts may be caused by the investigated mapping algorithm.

3.1.2 Unit $\Delta E = 1$ spheres mapping: unitDE test

This test maps unit spheres in CIELAB with a radius of the just noticeable color distance ($\Delta E = 1$). For computation, the spheres are approximated by a regular polyhedron. Several features of the mapped solid are computed: the largest mapped radius ΔE_{Dmax}, the smallest mapped radius ΔE_{Dmin}, the volume V, and the total surface S of the resulting solid. From the volume V, a length ΔE_{vol} is determined corresponding to the radius of a sphere with the same volume. Analogously, ΔE_{Srf} is defined as the radius of a sphere with the same surface as S. These measures will be used to detect discontinuities and to judge the behavior concerning contrast loss. On the one hand, maximum distances ΔE_{Dmax} much larger than one are an indication for discontinuities. On the other hand, clipping to the surface results in $\Delta E_{vol} = 0$ and $\Delta E_{Dmin} = 0$. Clipping to an edge additionally yields $\Delta E_{Srf} = 0$, and for clipping to a corner we even find $\Delta E_{Dmax} = 0$.

The test method, called the unitDE test, can be described by the following steps.

1. Choose a random color in CIELAB.
2. Convert to RGB. Skip the color if it is not within the RGB cube.
3. Construct a regular polyhedron with corners of equal distance $\Delta E = 1$ to the center color.
4. Verify that all colors of the polyhedron are within the RGB cube, otherwise skip the color.
5. Calculate mapped colors in CIELAB. Examples are given in Figs. 8 and 9.
6. Determine minimum distance ΔE_{Dmin}, maximum distance ΔE_{Dmax}, volume V, and surface S of the mapped solid, and compute the derived features ΔE_{vol} and ΔE_{Srf}.
7. Collect the four features in separate histograms.

Note that the randomization is made in CIELAB and not in RGB to get a uniform distribution over the working color space. The calculation is made on a set of typically one million samples. In our tests, we used the CIELAB as
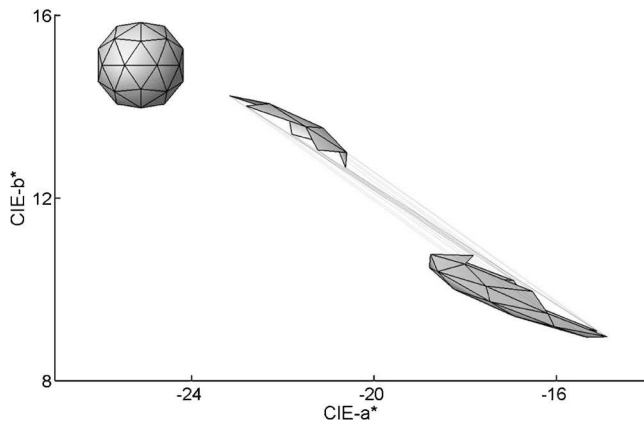
Fig. 9 Mapping of a unit sphere showing discontinuity: the colors are mapped to two separated parts.

working space and ΔE_{ab}^* as a distance measure. The test can be used for other color spaces and distance measures.

3.2 *Gamut Boundary Description*

For the description of the involved source and destination device, we need to have access to a model describing the transformations to and from a device independent color space such as XYZ. Depending on the device, this may be defined by a mathematical transformation, a color reproduction model, or an international color consortium (ICC) profile. For this work, we used sRGB as a source gamut. As output device, we selected four widely used applications: newspaper, photographic, inkjet, and standard offset printing. Color measurement data from standard ICC profiles were used to build the gamut boundary description for newspaper printing,[23] photographic printing,[24] inkjet printing,[25] and for standard offset printing on coated paper.[26]

The gamut boundary was approximated by a set of gamut surface points at the vertices of a grid of triangles. The description used is basically the "segment maxima gamut boundary descriptor."[27] We used the description where the position of the grid colors are not defined by fixed spherical angles. In this way, corners and edges of the gamut boundary, which are usually well defined for media gamuts, can be modeled to a higher precision, because grid points can be put exactly on corners and edges.

It is important to verify that the gamut boundary description does not induce discontinuities or other visual artifacts. Two points need special attention. First is the effect of numerical approximations, especially if ICC profiles are used, and second is the influence of the size of the grid. To check the validity of the data processed by the ICC profiles, color reproduction models were used in parallel to verify the results of the continuity tests. For all used profiles, we alternatively constructed gamut boundaries based on measured surface colors, on which the profile calculation was based. For photographic paper, a model-based description[28] of the color reproduction was used. In general, the local mapping behavior is not sensitive to the number of faces used for the description of the gamut boundary. An important exception is GMAs, which rely on minimum distance determinations. A discussion of this topic is given later.

Except where noted, we used a 1440 face grid. In Fig. 10, we show typical grids of the sRGB gamut compared to newspaper, photopaper, offset, and inkjet gamuts.

4 Results and Discussion

4.1 *Artifacts Caused by Gamut Mapping*

Artifacts in images are a quite common problem in the printing industry. In many cases, their source cannot be determined because of the complexity of the color management work flow. If we want to visualize artifacts caused by GMAs, we have to use reliable input and output device descriptions and idealized images. Images showing color gradients are useful, because the eye is especially sensitive to irregularities in such images. Gradual color changes are common in graphic arts but also in natural images, such as sky, snow, and illuminated homogeneous walls.

As an illustration, Fig. 11 shows the kind of perceptual effects that are produced if continuity aspects are not considered in the mapping design. One example is taken from the GCUSP1 algorithm, which makes a conceptual distinction between out-of-gamut colors and in-gamut colors. This distinction can readily be seen in the mapped images. The second example using hue preserving minimal ΔE (HPMinDE) shows the effect of minimum distance mapping to a polynomial approximation of a concave surface using a coarse grid of 72 faces. Such artifacts occur only in small color regions. They could escape detection, even if a large number of arbitrary images were used. With the proposed tests, we are able to find critical color regions for a given GMA and printing application. It is then straightforward to construct images visualizing the extent of such an artifact. In the following discussion, we show how to determine and visualize such artifacts. We concentrate on two topics: discontinuities and loss of contrast.

4.2 *Discontinuities*

Testing the mapping of neighboring RGB colors is suitable to investigate and visualize discontinuities. The test was applied for every combination of a set of GMAs and the two destination gamuts for newspaper and photographic printing. The RGB color pairs with the maximum ΔE were used to build an image of a small square on a square background using the two colors. Examples are shown in Fig. 12 for the newspaper gamut using different GMAs. The left side of a patch shows the original color pair and the right side the mapped color pair. On the left side we see no difference between the center part and the background. The right side shows the maximal visual effect caused by the mapping. If the right side shows no or only very little visual difference, the GMA can be considered as uncritical with respect to discontinuities, as is the case for the smooth gamut deformation algorithm (SGDA) and enhanced Schläpfer (SCHLP) algorithms.

For numerical comparison, the ΔE_{Dmax} feature of the unitDE test is a more suitable measure, because we compare the mapping of color pairs with equal ΔE distance. The results are shown in Table 1 for our set of GMAs and two different destination devices. *A priori*, it is not evident what limit has to be set on ΔE_{Dmax} to define a visual relevant discontinuity. If we consider the visual results from the dRGB test, a limit in the range $5 < \Delta E < 8$ seems to be reasonable. Using $\Delta E = 6$ as a limit, SCHLP and SGDA
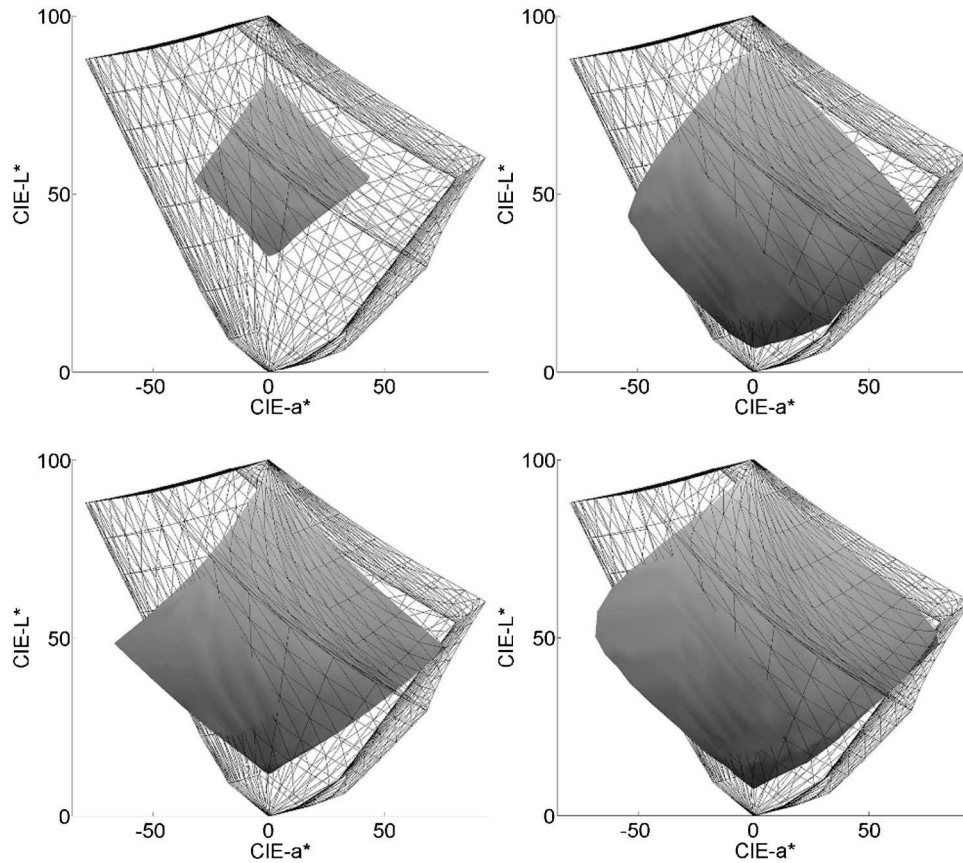
Fig. 10 Printing gamuts (solid) compared to sRGB gamut (wire grid) for newspaper printing (top left), photo paper (top right), coated offset printing (bottom left), and inkjet printing (bottom right). The projection on the L^*-a^*-plane of CIELAB is shown.

would be considered as uncritical, HPMinDE as being close to the limit, and SGCK, 3-D minimal ΔE (MinDE3D), and both variants of GCUSP as unacceptable. This is in good agreement with the results obtained from the dRGB test. A closer analysis of the reasons for high ΔE_{Dmax} values revealed that all but SGDA and SCHLP show discontinuities. Three different reasons were identified: the largest discontinuities, found in GCUSP1, are at the boundaries of regions with different mapping strategies.

Table 1 Maximum mapped ΔE for various GMAs for newspaper and photographic printing.

	Newspaper	Photo	Offset	Inkjet
SCHLP	3.6	2.4	3.2	2.3
SGDA	2.8	3.7	3.9	3.5
HPMinDE	5.3	3.7	4.4	3.9
MinDE3D	11.6	6.5	3.7	9.0
SGCK	8.6	8.6	7.4	8.1
GCUSP2	15.4	10.8	9.5	9.6
GCUSP1	26.9	32.2	32.5	32.1

For SGCK and GCUSP2, multiple solutions in cutting a mapping line with the gamut boundary were identified as the cause for discontinuities. Finally, for hue preserving minimal ΔE (HPMinDE) and MinDE3D, minimum distance mapping to concave surfaces turned out to be problematic. For HPMinDE, the discontinuities are small, and may not cause visual artifacts in practical applications. Interestingly, MinDE3D shows much larger values for ΔE_{Dmax} compared to HPMinDE for three of the four studied destination gamuts. This is caused by a more pronounced convexity of the gamut border in the hue direction rather than perpendicular to it. An analysis of the direction of the maximal mapped color difference for MinDE3D showed dominant hue components for those three destination gamuts.

In the following, we discuss these three causes for discontinuities in detail.

4.2.1 *Intersecting lines with gamut boundary*

Many algorithms use intersections of lines with gamut borders for the definition of chroma or lightness compression. The intersecting angle has a strong influence on the derivative behavior. Small angles between the mapping line and gamut surface may cause an amplification of noise in images. Usually, no explicit precautions are made to avoid small intersecting angles. Even more, if algorithms are used in applications with gamut boundaries different from the

Fig. 11 Artifacts in color gradients: GCUSP1 algorithm showing visible boundaries of regions with different mapping strategies (top), and minimum distance mapping to polygonal approximations of concave gamut border (bottom).

Fig. 12 Artifacts caused by discontinuities using gamut mapping to newspaper gamut: maximum ΔE for GCUSP1, SGCK, MinDE3D (top, left to right) and HPMinDE, SGDA, and SCHLP (bottom, left to right). The left half of each square shows the original color and the right half shows the mapped colors.

one for which the method was optimized, the intersecting behavior is not under control and may even cause ambiguous results. For example, applying the GCUSP2 algorithm to the small lightness range of newsprint media turns out to be problematic. The large deformation of the source gamut due to the chroma-dependent lightness compression caused some mapping lines to have triple intersection with the gamut border (see Fig. 13). In this case, the mapping is ambiguous and will cause discontinuities. The SGCK algorithm is based on the same C-dependent L compression, thus it shows the same problem.

4.2.2 *Boundaries of regions with different mapping strategies*

This source for discontinuities may occur at the boundary of color regions treated with different algorithms or parameters. An example is the GCUSP2[12] algorithm, which treats out-of-gamut colors and in-gamut colors differently. If source and destination gamuts are not too different, the resulting discontinuity may not be visible for most images, but for small destination gamuts, it produces disturbing artifacts. Very recently, Cho, Kim, and Ha[29] proposed a color space division with parabolic functions to define the mapping direction. The mapping direction is claimed to be con-
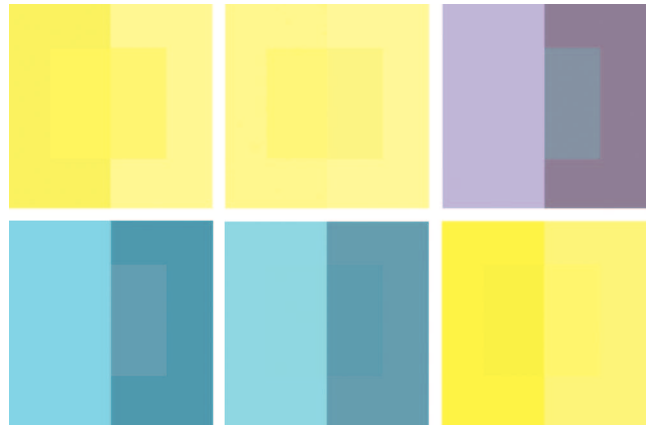
stant within a region. However, without special precautions, the mapping of colors at the region border will not be continuous.

4.2.3 *Minimum distance mapping*

As long as a gamut surface is convex, the minimum distance mapping always has a unique solution. However, if colors have to be mapped to concave surfaces, unforeseen problems arise for colors that are at a distance that exceeds the radius of the curvature of the surface. This problem exists for 2- and 3-D mappings. Ambiguous mapping and therefore discontinuities exist as soon as the center of the sphere describing the curvature lies within the source gamut. Minimum distance mapping should be avoided for all such destination gamuts. Using a polyhedral description of concave gamut surfaces will always cause discontinuities for minimum distance mapping. An illustration is shown in Fig. 14. Thus, to avoid visual discontinuities, a finer polyhedral grid may reduce discontinuity problems for minimum distance GMAs.

4.3 *Contrast Loss*

The most pronounced contrast loss can be found for algorithms using clipping. An illustration is given in Fig. 15. These images are constructed with the two farthest apart

Fig. 15 Visualization of artifacts caused by contour loss: MinDE3D (left) and HPMinDE (right). The left half of a square shows the original colors and the right half shows mapped colors.
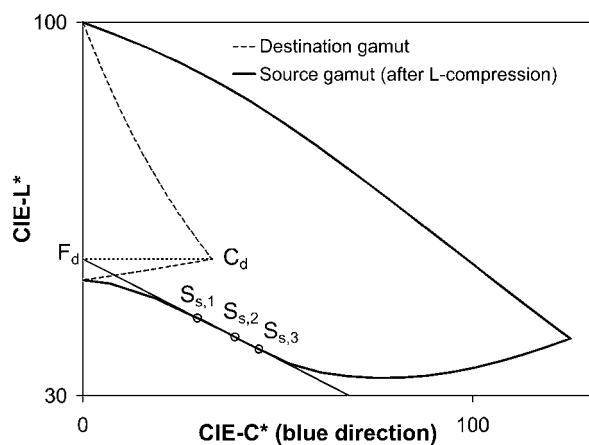
Fig. 13 Multiple intersections of a mapping line with source gamut ($S_{s,1}, S_{s,2}, S_{s,3}$). Example from GCUSP2 for mapping sRGB to newspaper gamut.

Fig. 16 Histograms of ΔE_{vol}, ΔE_{Srf}, and ΔE_{Dmax} of mapped dE=1 spheres for SGDA (top) and MinDE3D (bottom) algorithms.

colors, resulting in visually identical mapped colors. The results are shown for HPMinDE and MinDE3D on the newspaper gamut. From the unitDE test, three different types of clipping can be distinguished: clipping to a sur-
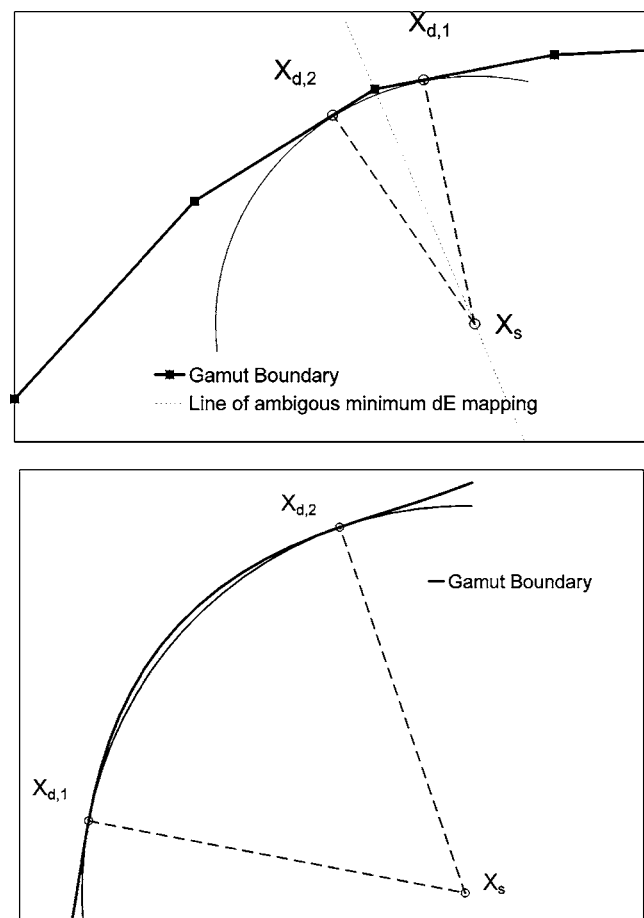

Fig. 14 Illustration of bivalent mapping to a polygonal approximation of a concave gamut boundary (top) and to a smooth, concave gamut boundary (bottom).
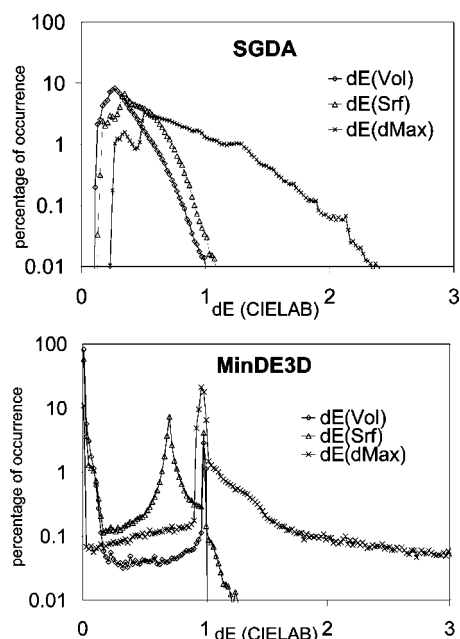
face, clipping to an edge, and clipping to a corner. From the histograms we can extract the percentage P of clipping of these three types:

$$P_{Srf} = H_{vol}(0) - H_{Srf}(0), \tag{5}$$

$$P_{edge} = H_{Srf}(0) - H_{Dmax}(0), \tag{6}$$

$$P_{corner} = H_{Dmax}(0). \tag{7}$$

The histograms H_{Srf}, H_{vol}, and H_{Dmax} are shown in Fig. 16 for newsprint gamut and the two GMAs, MinDE3D and SGDA. Note that the percentage scale is logarithmic. This allows us to also visualize small occurrences of a specific ΔE entry. MinDE3D, as a typical clipping algorithm, shows several distinct maxima in the histograms. H_{Dmax} shows two maxima, one at $\Delta E = 0$ for the clipped colors, and the other at $\Delta E = 1$ for the unaltered colors. H_{Srf} shows an additional broad peak at $\Delta E = 0.7$. Its presence can be explained by the fact that a sphere clipped to a plane has half its original surface. The corresponding radius ΔE_{Srf} is thus $\sqrt{1/2}$. SGDA shows no distinct peaks, but more importantly the histogram shows no entries close to $\Delta E = 0$ and above $\Delta E = 2.5$.

Table 2 shows the percentage of the three types of clipping in MinDE3D, HPMinDE, and SCHLP, for the four investigated printing applications. The other investigated algorithms SGDA, GCUSP, and SGCK, are not shown in the table because they do not use clipping at all. If MinDE3D or HPMinDE is used, the amount of clipping to edges is especially high for the newspaper printing application. For MinDE3D, more than 10% of the colors are clipped to a corner, losing all its contrast information. This indicates that contrast loss gets to be a severe problem if the destination gamut is small compared to the source

Table 2 Percentage of the three types of contrast loss in MinDE3D, HPMinDE, and SCHLP.

Clipping to	Surface P_{Srf}	Edge P_{edge}	Corner P_{corner}	Total
MinDE3D				
Newspaper	30	51	13	93
Photo	40	12	1	53
Offset	43	15	1	59
Inkjet	30	9	1	40
HPMinDE				
Newpaper	42	51	1	93
Photo	36	7	0	53
Offset	46	13	0	59
Inkjet	33	7	0	40
SCHLP				
Newspaper	55	0	0	55
Photo	26	0	0	26
Offset	36	0	0	36
Inkjet	18	0	0	18

gamut. Note that for MinDE3D and HPMinDE, the total percentage of clipped colors corresponds to the percentage of out-of-gamut colors.

Several gamut mapping studies suggest that clipping is preferred over compression.[6,11,30,31] Conservation of color saturation seems to be more important than contrast conservation of saturated colors. The previous results show that if clipping is used in the design of a GMA, at least the clipping of color regions to lines or to points should be avoided to restrict the loss to, at most, one of the three dimensions of the color contrast.

For small destination gamuts, however, contrast loss is a major problem if a GMA is designed based on maximization of color saturation. Particularly important is the preservation of luminance contrast, because human depth perception is based mainly on lightness gradients. Thus, it is not surprising that GMAs developed for newspaper printing[5,19,20] all use compression as a major mapping strategy.

5 Conclusion

From a geometrical point of view, gamut mapping can be described as a transformation from one 3-D solid into another. Based on today's knowledge of computational geometry, it is not surprising that the previously described continuity problems of GMAs have a large influence on the overall quality of its performance. In this context, it is worth also to note that in computational geometry, accuracy and continuity aspects were underestimated for a long

time.[32] Recently, continuity problems in color science applications were also discussed for the implementation of the CIEDE2000 difference formula.[33]

The discussed local mapping problems deserve special attention when designing a universal gamut mapping approach. Mapping algorithms designed for industrial applications must avoid visual failures, even if they occur only rarely. The strong and disturbing effect of this kind of problem may be increased by the fact that the human depth perception view is based mainly on lightness gradients. The tests that are developed check for the occurrence of contrast losses and continuity problems. Note that these tests can also be adapted to test color management systems for local mapping properties of ICC profiles.

The use of a controlled gamut mapping design can avoid such problems. This is particularly important for GMAs that use image-based gamut descriptions such as alpha shapes[34] and flow shapes.[35] The shape of these gamuts do not allow the use of minimum distance methods and call for special attention if intersections of lines with gamut borders are used to define the mapping functions. Thus, new concepts for gamut mapping are needed for this application.

References

1. J. Morovic, "Gamut mapping," in *Digital Color Imaging*, G. Sharma, Ed., The Electrical Engineering and Applied Signal Processing Series, Handbook, Chap. 10, pp. 639–685, CRC Press, Boca Raton, FL (2003).
2. J. Morovic, "To develop a universal gamut mapping algorithm," PhD Thesis, University of Derby, UK (1998).
3. *Guidelines for the Evaluation of Gamut Mapping Algorithms*, Central Bureau of the CIE, Vienna, CIE Publication 156 (2004).
4. U. Steingrimsson and K. Simon, "Perceptive quality estimations: JPEG2000 versus JPEG," *J. Imaging Sci. Technol.* **47**(6), 572–585 (2003).
5. A. J. Johnson, "Colour appearance research for interactive system management and applications—CARISMA," Report WP2-19 Colour Gamut Compression, see http://www.derby.ac.uk (1992).
6. G. J. Braun and M. D. Fairchild, "Image lightness rescaling using sigmoidal contrast enhancement functions," *J. Electron. Imaging* **8**(4), 380–393 (1999).
7. P. Laihanen, "Colour reproduction theory based on the principles of colour science," *IARAIGAI Conf. Proc. Adv. Printing Sci. Technol.* **19**, 1–36 (1987).
8. B. H. Kang, M. S. Cho, J. Morovic, and R. M. Luo, "Gamut compression algorithm development on the basis of observer experimental data," in *Eighth Color Imaging Conf. Color Sci. Syst. Appl.* **8**, 268–272 (2000).
9. C. S. Lee, C. H. Lee, and Y. H. Ha, "Parametric gamut mapping algorithms using variable anchor points," *J. Imaging Sci. Technol.* **44**(1), 68–73 (2000).
10. L. MacDonald, J. Morovic, and K. Xiao in *Colour Image Science: Exploiting Digital Media*, L. W. MacDonald and M. R. Luo, Eds., Chap. 14, pp. 291–317, John Wiley and Sons, New York (2002).
11. H. Buering and P. Herzog in *Colour Image Science: Exploiting Digital Media*, L. W. MacDonald and M. R. Luo, Eds., Chap. 16, pp. 343–353, John Wiley and Sons, New York (2002).
12. J. Morovic and M. R. Luo, "Evaluating gamut mapping algorithms for universal applicability," *Color Res. Appl.* **26**(1), 85–102 (2001).
13. E. Pirrotta, T. Newman, and L. Lavendel, "A universal gamut mapping algorithm?," *Proc. Colour Image Sci. Conf.*, pp. 313–320 (2000).
14. J. Morovic and P. L. Sun, "Non-iterative minimum de gamut clipping," *9th Color Imaging Conf.*, **9**, 251–256 (2001).
15. "Industrial color difference evaluation," *CIE Publication 116*, Central Bureau of the CIE, Vienna (1995).
16. F. J. J. Clarke, R. MacDonald, and B. Rigg, "Modification of the jpc97 colour-difference formula," *J. Soc. Dyers Colour.* **100**, 128–132 (1984).

17. M. R. Luo and B. Rigg, "Bfd(l:c) colour difference formula, parts 1 and 2," *J. Soc. Dyers Colour.* **103**, 126–132 (1987).
18. N. Katoh and M. Ito, "Gamut mapping for computer generated images (ii)," *Proc. IS&T/SID 4th Color Imaging Conf.*, pp. 126–129 (1996).
19. P. J. Green and M. R. Luo, "Extending the CARISMA gamut mapping model," *J. Imaging Sci. Technol.* **46**(1), 33–43 (2002).
20. UGRA, "UGRA Gamcom Version 1.1 – program for the color gamut compression and for the comparison of calculated and measured values," UGRA, St. Gallen (1995).
21. P. Zolliker, "Fast gray calibration for digital photographic printers," in *First Euro. Conf. Colour Graphics, Imaging, Vision (CGIV)* **1**, 438–441 (2002).
22. U. Steingrimsson and P. Zolliker, "Assessment of rgb encoding for color imaging," in *Workshop Farbbild-verarbeitung*, D. Droege and D. Paulus, Eds., pp. 103–110, Universität Koblenz-Landau, Der Andere Verlag (2004).
23. "Zeitung_QUIZ_28_02.03V2.icm from Ifra," see http://www.ifra.com/.
24. "AveragePhotoPrinterRGB.icc from Arbeitsgruppe Photogamut," see http://www.photogamut.org/.
25. "AverageInkjetMatt.icc from Arbeitsgruppe Photogamut," see http://www.photogamut.org/.
26. "ISOcoated.icc from the European Color Initiative ECI," see http://www.eci.org/.
27. J. Morovic and M. R. Luo, "Calculating medium and image gamut boundaries for gamut mapping," *Color Res. Appl.* **25**(6), 394–401 (2000).
28. R. Meier and P. Zolliker, "A color model for digital photo printing on silver halide paper," in *First Euro. Conf. Colour Graphics, Imaging, Vision (CGIV)*, **1**, 442–447 (2002).
29. Y. H. Cho, Y. T. Kim, and Y. H. Ha, "Gamut mapping based on color space division for enhancement of lightness contrast and chrominance," *J. Imaging Sci. Technol.* **48**(1), 66–74 (2004).
30. R. S. Gentile, E. Walowitt, and J. P. Allebach, "A comparison of techniques for color gamut mismatch compensation," *J. Imaging Technol.* **16**, 176–181 (1990).
31. E. D. Montag and M. D. Fairchild, "Psychophysical evaluation of gamut mapping techniques using simple rendered images and artificial gamut boundaries," *IEEE Trans. Image Process.* **6**(7), 977–989 (1997).
32. F. P. Preparata and M. I. Shamos, *Computational Geometry: An Introduction*, Springer-Verlag, New York (1985).
33. G. Sharma, W. Wu, and E. N. Dalal, "The CIEDE2000 color-difference formula: Implementation notes, supplementary test data, and mathematical observations," *Color Res. Appl.* **30**, 21–30 (2005).
34. T. J. Cholewo and S. Love, "Gamut boundary determination using alpha-shapes," in *7th Color Imaging Conf.*, **7**, 200–204 (1999).
35. J. Giesen, E. Schuberth, K. Simon, and P. Zolliker, "Toward image-dependent gamut mapping: fast and accurate gamut boundary determination," *Proc. SPIE* **5667**, 201–210 (2005).

Peter Zolliker has a degree in physics from the Swiss Federal Institute of Technology, and received a PhD in crystallography from the University of Geneva in 1987. From 1987 to 1988 he was a postdoctoral fellow at the Brookhaven National Laboratory, New York. From 1989 to 2002 he was a member of the research and development team at Gretag Imaging, working on image analysis, image quality, setup, and color management procedures for analog and digital printers. In 2003 he joined Empa, Material Science and Technology, Switzerland, where his research is focused on digital imaging, gamut mapping, and color sciences. He is a member of the IS&T.

Klaus Simon received his master in computer science in 1983, and his PhD degree in efficient algorithms and data structures in 1987 from the University of Saarland, Germany. Between 1987 and 1988 he worked as a research and development engineer in scientific imaging at the Bodenseewerk Gerätetechnik mbH (Perkin Elmer). He joined the Swiss Federal Institute of Technology (ETH) Zürich as a senior research associate in 1988 and was appointed as an assistant professor for computer science in 1992. Since 1999 he has been leading the media technology group at Empa, Material Science and Technology, Switzerland. His research interests include efficient algorithms, probability theory, scientific imaging, and teaching.