

A Gamut-Mapping Framework for Color-Accurate Reproduction of HDR Images

Elena Šikudová ■ Comenius University Bratislava

Tania Pouli ■ Technicolor Research & Innovation

Alessandro Artusi ■ University of Girona

Ahmet Oğuz Akyüz ■ Middle East Technical University

Francesco Banterle ■ ISTI-CNR

Zeynep Miray Mazlumoğlu ■ Middle East Technical University

Erik Reinhard ■ Technicolor Research & Innovation

High dynamic range (HDR) imaging consists of tools and techniques to capture, store, transmit, and display images with significantly higher fidelity than can be achieved with conventional imaging techniques. An important aspect of HDR imaging involves the reproduction of images on conventional displays. Because in such cases the image's dynamic range can be much higher than the display device can accommodate, dynamic range reduction techniques need to be employed.^{1,2}

An integrated gamut- and tone-management framework for color-accurate reproduction of high dynamic range images can prevent hue and luminance shifts while taking gamut boundaries into consideration. The proposed approach is conceptually and computationally simple, parameter-free, and compatible with existing tone-mapping operators.

In many cases, tone reproduction techniques focus on range compression along the luminance dimension, either leaving chromaticities unaltered or treating color management as a separate problem³⁻⁵ In the latter case, algorithms focus on correcting or improving the appearance of the tone-mapped image. Some HDR color appearance models do integrate color

and luminance management, for the purpose of predicting the human visual response to a stimulus.^{6,7} Such algorithms can be used successfully as display algorithms, albeit still without appropriate gamut management.

To our knowledge, none of the existing algorithms take the target color gamut into consideration, and as a result, they often produce pixel values that cannot be correctly represented or displayed. We have developed a method that aims to combine the color-correction step often necessary after tone mapping with gamut management into an integrated HDR gamut-management framework that handles both lightness and chroma compression, while limiting hue shifts and luminance distortion as much as possible.

These two dimensions should be treated in conjunction for several reasons. First, human vision does not treat colorfulness separately from luminance values, as evidenced by the Hunt effect. In essence, lighter objects are seen as more colorful, and vice versa. In the context of dynamic range management, this means that the compression of luminances should be accompanied by a cor-

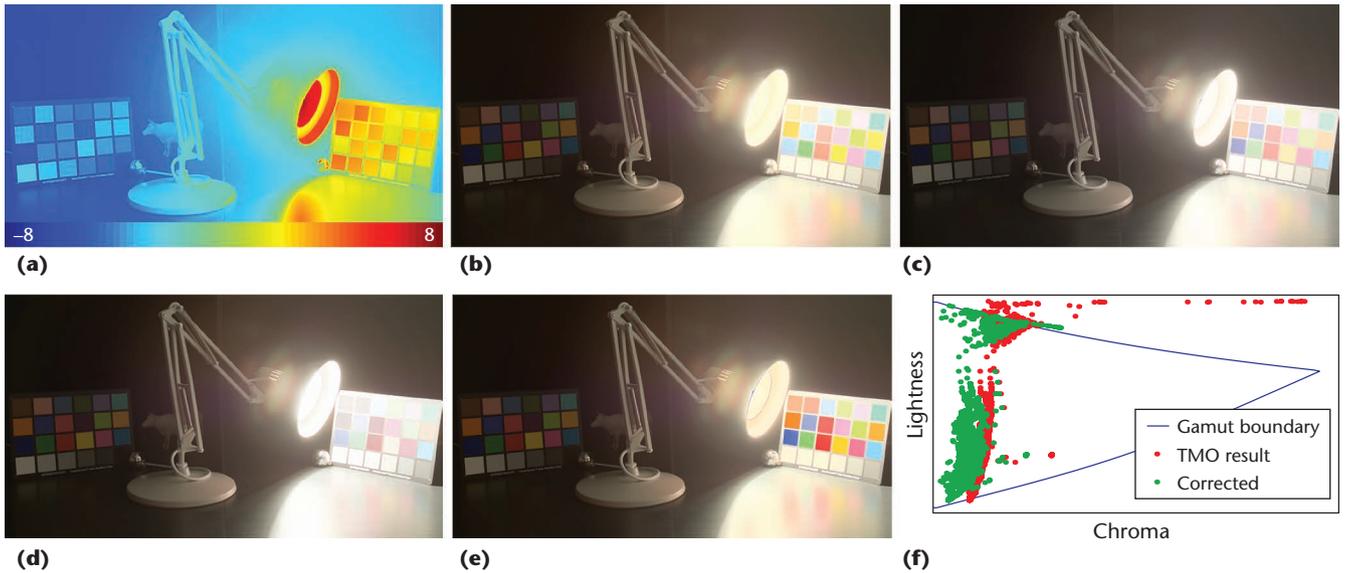


Figure 1. High dynamic range (HDR) image reproduction. This image, (a) shown in false color, was (b) tone mapped using the Photographic operator⁹ and (c) processed with our framework, (d) a color correction solution,⁴ and (e) a gamut-mapping solution.¹⁰ (f) Although the tone-mapping process has successfully compressed the luminance of the image, it has led to an oversaturated appearance and out-of-gamut pixels. Our method corrects both the oversaturated appearance and out-of-gamut pixels issues, whereas the alternative solutions only handle one of the two.

responding chroma adjustment. Additionally, current consumer display trends are pushing toward a combined increase in both the gamut and dynamic range, necessitating algorithms that can manage these aspects in conjunction.

Second, color spaces are three dimensional and bounded by their gamut.⁸ The gamut spanned by the pixels in an image may not match that of the target display. In such cases, gamut mapping involves compensating for differences in the size, shape, and location between the image and display gamuts. This constitutes a mapping from one 1D shape to another. Because the overlap between gamuts is typically large in conventional gamut-mapping scenarios, such algorithms aim to find a trade-off between moving out-of-gamut pixels inside the target gamut, while pixels that are already inside the target gamut are left alone as much as possible.

On the other hand, 1D luminance adjustment, as achieved by many tone-reproduction operators (TMOs), may create pixels that lie outside the target display's gamut (see Figure 1) along the chroma channel. These pixels typically are either not managed or treated with naive approaches³ that tend to oversaturate, as Figure 1b shows.

Our gamut-mapping framework integrates tone mapping with gamut mapping, correcting the colors after dynamic range compression, while ensuring that images fit within the target gamut. It does not require calibrated input and is parameter-free. We accomplish this by transforming the original HDR image into the CIE L*C*h* color space and then compressing both the lightness and chroma

channels, as in gamut-mapping algorithms. The lightness channel can be compressed with any existing TMO, or a scheme similar to our chroma compression can be applied to lightness as well.

Prior HDR Image Reproduction Methods

Approaches that attempt to reproduce visual content on devices of different gamuts are typically divided into two categories: gamut mapping and tone mapping. Gamut-mapping techniques deal with mapping the color gamut between devices, attempting to produce the most accurate reproduction of colors possible given the restrictions of a given device or medium.⁸ Tone mapping, on the other hand, is primarily concerned with compressing the luminance range of an HDR image or video such that the media can be visualized on a low dynamic range (LDR) display device.^{1,2}

Unlike tone or gamut mapping, color appearance models (CAMs) predict human visual perception of patches of color and images. They consider parameters relating to the scene and viewing environment and, as such, require accurate input measurements. These methods can accurately reproduce the appearance of an image for different devices and viewing conditions, but they do not take gamut boundary issues into consideration.

Gamut Mapping

Given a color space, a gamut for a device or medium can be thought of as a subspace within the color space that contains the colors that can be reproduced by that device. A new color that is outside the gamut

cannot be accurately reproduced by the device. Gamut-mapping techniques thus are used to define how colors outside this gamut should be treated or mapped to the displayable color subset. Such techniques can be categorized as global and spatial.

Global mapping techniques can be further classified as clipping- and compression-based approaches. Clipping only changes the colors that are located outside of the destination gamut by clamping them to the boundaries of the destination gamut.⁸ Although it has the advantage of preserving within-gamut colors, this is only a viable solution if the difference between the two gamuts is small. Compression, on the other hand, adjusts all the colors of the input gamut so they match the destination gamut.⁸ Researchers have proposed different types of compression functions such as linear, piecewise linear, and sigmoidal. Compression is typically performed on both lightness and chroma components.

In contrast to the global approaches, spatial gamut mapping attempts to preserve local information. These methods will map similar out-of-gamut colors to the same color if they are spatially distant in the image, but map them to distinct colors if they share an edge.

Our goal is to extend existing work on gamut mapping for LDR, proposing a gamut-mapping management framework to work directly with HDR input data. This can be either integrated into existing TMOs or be a stand-alone solution with its own lightness compression technique for HDR luminance values.

Tone Mapping

Although tone mapping can be considered a form of gamut mapping, there are important differences. First, tone mapping is generally employed when the input image's dynamic range is vastly higher than that of the display device. Second, tone mapping is generally concerned with compressing luminances, whereas gamut mapping is concerned with compressing perceptual attributes of lightness and chroma. As such, it is possible that a tone-mapped image will contain out-of-gamut colors, which are clipped to gamut boundaries in an uncontrolled manner.

An additional concern when tone mapping the luminance channel only is that images tend to acquire an oversaturated appearance, either globally or locally, as shown in Figure 1b. Appearance aspects such as image saturation and colorfulness or image patch depend both on the chromatic information and the image luminance. Thus, to fix these color distortions, most TMOs are augmented with a postprocessing step that desaturates the im-

age by means of a manually controlled parameter.³ Psychophysical studies have linked this saturation parameter to the amount of contrast correction computed from the global tone-mapping curve.⁴ Alternatively, the amount of (de)saturation can be computed by comparing the original HDR input to the tone-mapped result.⁵ Although these methods can improve the appearance of the tone-mapped image, they are not able to consider the gamut boundaries of the target medium.

HDR Gamut-Management Framework

Although both tone- and gamut-mapping research aim to reproduce images on devices of more limited capabilities, they have remained largely disconnected areas. In this work, we bring together these two fields in an integrated gamut-management technique. Our framework incorporates tone mapping, chroma correction, and gamut management to process an HDR image for a given output gamut. Figure 2 shows a workflow of our technique.

The input to our pipeline is an HDR image I given in linear XYZ coordinates. First, the image's luminance channel, denoted $I(L)$, is compressed using any existing TMO. The resulting image's chroma channel $I(C)$ is then corrected with our chroma compression algorithm to correct for unwanted saturation resulting from the tone-mapping process. Finally, image chroma and luminance values $I(C, L)$ are processed in a gamut-management step to ensure that all pixels fit within the target gamut boundaries, denoted $G(C, L)$, while minimizing appearance changes in the image.

Luminance Compression

Because we want to ensure that our chroma compression and subsequent gamut management steps correct any issues that the tone-mapping process may have caused, the first component of our framework is luminance compression. We designed our framework to easily integrate with existing TMOs. The following steps describe how to integrate it with the commonly used Photographic operator⁹:

1. The input HDR is first converted to the color space expected by the TMO (in this case, Y_{xy}).
2. The tone-mapping curve is applied on the luminance Y , obtaining the compressed value Y_c . At this point, both local and global algorithms can be applied because our framework poses no restrictions on the type of processing.
3. The compressed luminance is inserted into the image, and the result is converted back to the XYZ color space.

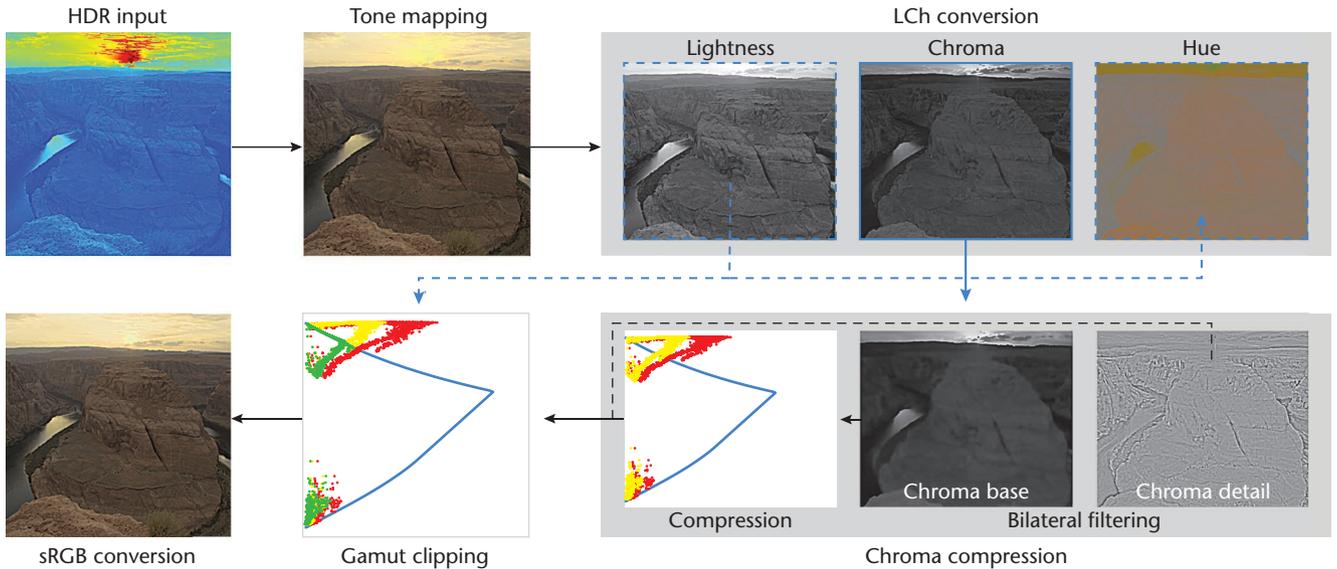


Figure 2. Overview of our framework. The input HDR image (illustrated here with a heat-map of luminance values) is first tone mapped and then converted to the LCh color space. Chroma values are then filtered using the bilateral filter and the base layer is compressed. Finally, a gamut-clipping step ensures that the compressed chroma and lightness values fit within the target gamut while minimizing appearance changes. Red indicates input, yellow chroma compressed, and green the final result.

4. Finally, the tone-mapped image I is normalized, such that the maximum Y value is 100, to allow for further processing. For the Photographic operator, output luminance values are between 0 and 1 and, as such, require scaling to the range expected by the color space used in further processing.

Although this process allows for flexibility in the choice of luminance compression, it comes at the cost of increased computational complexity resulting from additional color-space transforms because existing TMOs are not necessarily designed to operate in a color space that enables gamut manipulations. As an alternative solution, we have designed a compression solution that follows a scheme similar to our chroma compression method, which we describe in the “Lightness Compression Using Cusp Alignment” sidebar.

Gamut Boundary Computation

Our algorithm relies on the idea that, to avoid undesired shifts in chroma and hue as well as uncontrolled clipping for out-of-gamut colors, we should work in a perceptually decorrelated color space where these components are separated. A natural choice for this is the CIE $L^*C^*h^*$ color space, which is the cylindrical representation of CIE $L^*a^*b^*$. (In the remainder of this article, we refer to these as LCh and Lab spaces for brevity.) These two color spaces are commonly used in traditional gamut-mapping algorithms.⁸

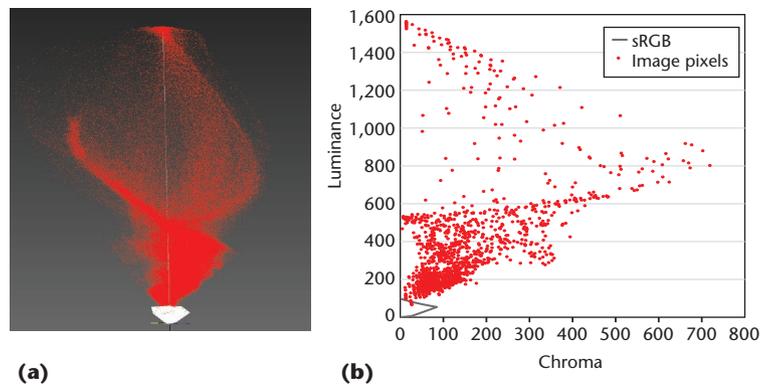


Figure 3. Differences between the source and target gamuts: (a) 3D gamuts and (b) hue slice. The input HDR image (source gamut) is in red and the sRGB color space (target gamut) is in white.

To determine the correct compression amount and assess whether a given pixel can fit within the output gamut, we need to know that gamut’s boundaries. Here, we assume that the target gamut is sRGB and thus use a D65 white point when converting to LAB. Our algorithm, however, can accommodate any alternative gamut and corresponding white point. The source or input gamut is the set of colors of the input HDR image. To compute the boundaries of both gamuts, we follow the methodology described in earlier work⁸ to obtain the sRGB gamut’s boundaries in LCh coordinates, which take the form of a triangular cusp along the chroma-lightness plane for each hue value.

Figure 3 shows the extreme differences between the source (red) and target (white) gamuts that may occur in HDR imaging. Between the source

Lightness Compression Using Cusp Alignment

Although the main goal of this article is to present our framework for managing the gamut mismatches that tone mapping causes in terms of the resulting image chroma, we have found that our approach can be directly extended to compress the lightness channel, leading to an integrated luminance and color gamut-management framework and minimizing the necessary number of color space conversions. To compress the lightness channel $I(L)$, we process each hue slice separately, similar to the process we described in the main article (see Figure 4c in the main article). Specifically, we follow four steps for each hue slice.

First, we find the global parameters that express the maximum vertical (lightness) distance from the destination gamut. The distance at the top is named SG_t and at the bottom SG_b (see Figure 4c in the main article). Both values are set to 0 when all pixels of the source gamut are already inside the destination gamut.

In the second step, the middle line L_{mid} is computed for the cusp of each hue slice as follows:

$$L_{mid} = g_b + (g_t - g_b) \frac{SG_b}{(SG_t + SG_b)}, \tag{A}$$

where g_b and g_t are the bottom and top values of the destination gamut, respectively. This equation shifts the middle line toward g_b for large SG_t , effectively compressing more of the image, and toward g_t for large SG_b , in which case more pixels are scaled linearly. That is, the equation adaptively determines a threshold that separates the source gamut into two regions that can be thought of as “light” and “dark,” with a magnitude for each determined by the ratio $SG_t:SG_b$. We treat each of these regions separately.

Third, in the light region—that is, for points above the L_{mid} —the lightness is compressed as follows:

$$I(L)_c = a_t + b_t * F(I(L) - L_{mid}). \tag{B}$$

The compression factors a_t and b_t are computed by $a_t = L_{mid}$ and

$$b_t = \frac{(1-w)(g_t - L_{mid}) + w(100 - L_{mid})}{N},$$

respectively, where the normalization factor is $N = F(g_t + SG_t - L_{mid})$ and the weight w is computed as

$$w = \frac{I(C)}{I(C) + \max(G(C))}.$$

Here $F(x)$ represents a nonlinear compression function, which is applied to pixels above L_{mid} . This could be any desired tone curve, including for instance sigmoidal compression or basic compressive functions such as roots and logarithms. Note that Equation B compresses the range $[L_{mid}, g_t + SG_t]$ to $[L_{mid}, b_t N + L_{mid}]$. Consequently, pixels with really high chroma and lightness may be mapped above the gamut boundary and therefore clipped in the following stage of our framework. The clipping process takes into account both C and L values, which ensures that such pixels will still remain bright in the final image.

In the last step, for points below L_{mid} , linear compression is used:

$$I(L)_c = a_b + b_b * (I(L) - L_{mid}). \tag{C}$$

In this case, the parameters are computed as $a_b = g_b$

and target, the differences in both lightness and chroma channels exceed a factor of 10.

Chroma Compression

For HDR imagery, a large number of pixel values may be outside the destination gamut in terms of chroma, which will generate unwanted hue shifts if they are clipped in an uncontrolled manner.

Additionally, research has shown that tone compression along the luminance dimension tends to create an oversaturated appearance in images (see Figure 1). To correct these issues, we propose two methods that compress the chroma values $I(C)$ in the image in a content-dependent manner.

Hue-specific method. This method compresses the chroma values $I(C)$ in the image using a two-step process. First, our algorithm determines a scaling factor R_h for each hue value $h = [1^\circ \dots 360^\circ]$, leading to a vector \mathbf{R} . To achieve that, we scale the

gamut boundaries $G_h(C, L)$ until they enclose all pixels within that hue slice.

Formally, we initialize the scaling factor for a hue slice $R_{h,0} = 1$. If any pixels are out of gamut for that hue slice, at each iteration step i , we increment the scale factor $R_{h,i}$ and scale the gamut boundaries as follows:

$$R_{h,i} = R_{h,i-1} + d$$

$$G_{h,i}(C, L) = \begin{bmatrix} R_{h,i} & 0 \\ 0 & R_{h,i} \end{bmatrix} G_h(C, L),$$

where the increment d is set to a small value. (In all the results in this article, $d = 0.1$.) Figure 4a illustrates this process.

Although this scaling factor could be applied to the image values directly to obtain a within-gamut result, in practice, using the full chroma range of the source gamut would likely result in extreme

and

$$b_b = \frac{(L_{mid} - g_b)}{L_{mid}}$$

Here we compress the range $[g_b - SG_b, L_{mid}]$ to $[g_b, L_{mid}]$. Our method effectively corresponds to a nonlinear compression with a linear ramp in dark areas. Such a behavior is commonly used in film. In our case, however, the image's gamut is explicitly considered to guide this compression scheme.

To improve visibility in dark regions, we adopt a similar technique used for the chroma compression by making use of a percentile and processing $I(L)$ with the bilateral filter before compression. Figure A shows an example of

our tone curve for the base layer and compares it with other global TMOs.¹⁻³

References

1. E. Reinhard et al., "Photographic Tone Reproduction for Digital Images," *ACM Trans. Graphics*, vol. 21, no. 3, 2002, pp. 267–276.
2. F. Drago et al., "Adaptive Logarithmic Mapping for Displaying High Contrast Scenes," *Computer Graphics Forum*, vol. 22, no. 3, 2003, pp. 419–426.
3. M. Ashikhmin, "A Tone Mapping Algorithm for High Contrast Images," *Proc. 13th Eurographics Workshop on Rendering (EGRW)*, 2002, pp. 145–156

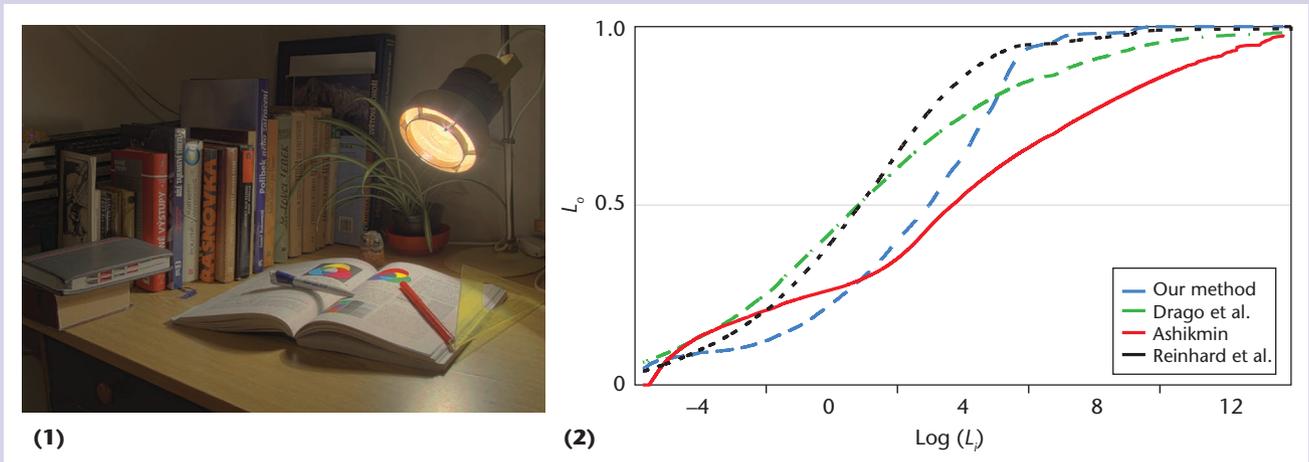


Figure A. Tone curves for the base layer. Here we compare (1) our method (HDR Gamut) with (2) three global tone-mapping operators (TMOs).¹⁻³

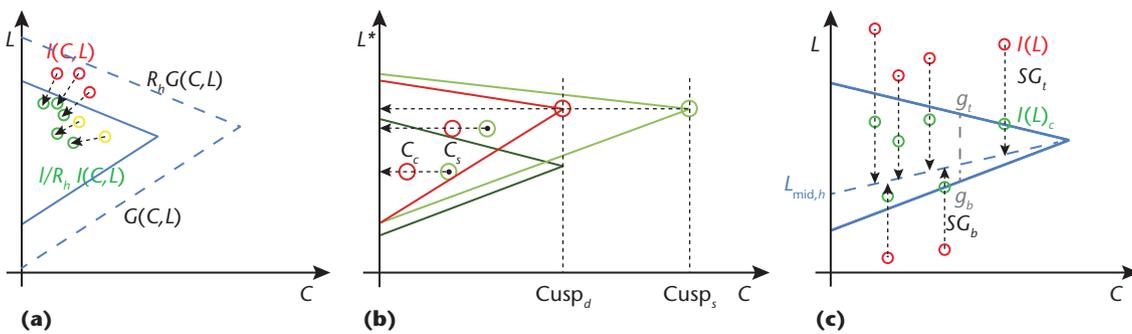


Figure 4. Compression schemes. (a) Hue-specific chroma compression, (b) global chroma compression, and (c) lightness compression.

compression due to a few outlying pixels with extremely high chroma. This is a common problem in luminance compression, where some extremely bright highlights could lead to an over-compressed result. The problem is usually countered by compressing according to a percentile of the range of values. In the case of chroma, if the compression takes into account such pixels, the resulting image may be too desaturated.

To avoid this undesirable effect, we use a percentile of the chroma range when computing \mathbf{R} . We have found that the percentile value required is content-dependent—if the pixels that require clipping are spread over the image, then a more aggressive percentile value can be selected. However, if the out-of-gamut pixels are concentrated in a small number of regions, a more gentle approach is necessary to ensure that no artifacts are created.

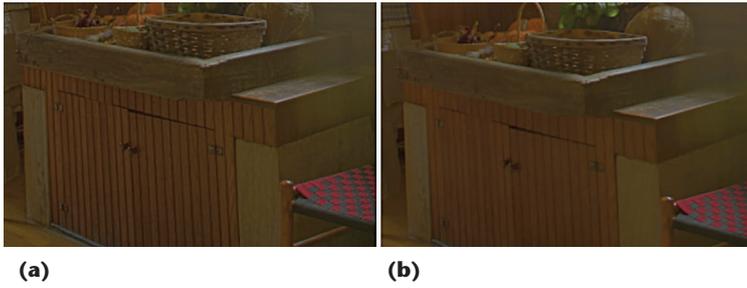


Figure 5. Applying bilateral filtering to the chroma channel. Results (a) with and (b) without bilateral filtering decomposition.

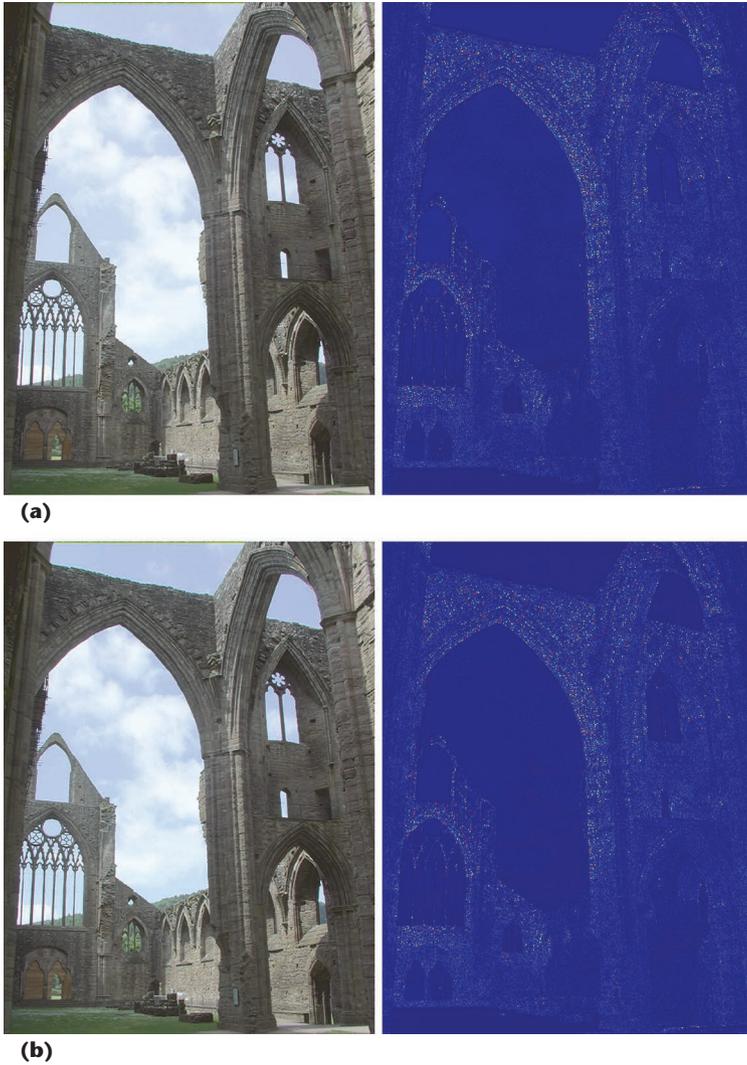


Figure 6. Results using the two chroma compression methods and their hue differences maps. (a) The hue-specific method would be suitable for a postproduction pipeline where accuracy is the primary goal; whereas (b) the global chroma linear method is better suited on the display side of an imaging pipeline, where computational resources are limited.

In practice, we determine the spread of out-of-gamut pixels by computing the number of connected regions to which they belong and comparing them to the number of pixels contained within them. We have found that a ratio less than 0.01

indicates that the out-of-gamut pixels are within a few connected regions and therefore further clipping may lead to artifacts.

To maintain fine details without smoothing edges, the chroma channel of the image $I(C)$ is first processed with the bilateral filter ($\sigma_s = 0.2 \max(I_{width}, I_{height}), \sigma_r = 0.05 \max(I(C))$) obtaining a base layer $I(C)_{base}$. Either division or subtraction can be used to separate the base and detail layers. We have verified that both methods lead to similar results so we decided to use the division method to produce the detail layer $I(C)_{detail} = I(C)/I(C)_{base}$. Figure 5a shows how fine details are preserved when bilateral filtering is applied to the chroma channel (see the doors of the drawer and borders of the sink).

Even with these measures, small variations in content between adjacent hue slices may occasionally lead to discontinuities in the final image if R_h is applied directly to each slice. Smoothing the scaling vector \mathbf{R} to \mathbf{R}' will eliminate these discontinuities. To achieve this, we can use different types of smoothing functions: lbox (averaging box), loess (locally weighted regression), rloess (robust locally weighted regression), and sgolay (Savitzky Golay). We find that these four smoothing functions produce results of similar quality, so in our framework, we use lbox, which is a simpler and more computationally efficient function.

The circular nature of hues is taken into account during the smoothing step. This avoids the creation of boundaries between the hue angles of 359 and 0.

Finally, image chroma within each hue slice $I(C)_{base,h}$ is scaled as

$$I(C)'_{base,h} = \frac{1}{R'_h} I(C)_{base,h},$$

and the detail is reinjected to obtain the chroma-compressed image $I(C)' = I(C)'_{base} \times I(C)_{detail}$, where \times indicates an element-wise multiplication. Note that the reciprocal \mathbf{R}' is necessary for the final compression because \mathbf{R} was initially computed to expand the gamut boundaries until they enclosed all pixels.

Global method. The hue-specific method can successfully compress chroma and therefore maximize the use of the available gamut. However, this comes at the cost of increased computational complexity. At the same time, when it is compared with the dynamic range of the display gamut, we can see that the chroma channel's dynamic range is not extremely high. This suggests that a linear compression scheme may produce good results, as the hue difference maps in Figure 6 show.

Therefore, we propose an alternative, but simpler method based on similar premises. We envisage that the hue-specific method would be suitable for a postproduction pipeline where accuracy is the primary goal; whereas the linear method we describe here would be better suited on the display side of an imaging pipeline, where computational resources are limited.

Figure 4b illustrates the global method. The light and dark green triangles in the figure represent the source and destination gamuts, respectively, for a fixed hue angle. Our goal is to align the two chroma cusps while maintaining lightness and hue. In Figure 4b, this produces the red triangle, which represents the compressed gamut.

Chroma compression is still applied on each hue angle from 0 to 359 degrees. Instead of using different amounts of chroma compression for each hue, however, which requires additional computations, we compute the minimum of the cusps ratios across all hues. This value corresponds to the maximum necessary compression.

Similar to the hue-specific method, chroma compression is applied to the chroma channel's base layer. We compress the source chroma $I(C)_{base,h}$ to yield the compressed chroma $I(C)'_{base,h}$ as follows:

$$I(C)'_{base,h} = I(C)_{base,h} \min_{h \in [0^\circ, 359^\circ]} \left[\frac{Cusp_{d,h}}{Cusp_{s,h}} \right].$$

Note that this is only a compression when $\min[Cusp_{d,h}/Cusp_{s,h}] < 1$.

We may face issues similar to the hue-specific method when using the full chroma range of the source gamut. This will produce extremely compressed chroma results in some cases because of a few outlying pixels with extremely high chroma values. To avoid this problem, we adopted the same percentile approach used for the first method. This solution avoids over-compression but requires an additional step to manage the few pixels that may remain outside the gamut boundary.

Gamut Clipping

So far, the chroma and lightness values are processed independently, so we cannot guarantee that all pixels will be within the target gamut boundaries. As Figure 7 shows, pixels are compressed through the lightness direction during tone mapping. Pixels are guaranteed to have maximum lightness values equal to 100 nits, but they may still be outside of the target gamut boundaries (see the red pixels in Figure 7c). Applying chroma compression does not solve this problem (see the yellow pixels in Figure 7c). To ensure all pixels are within the

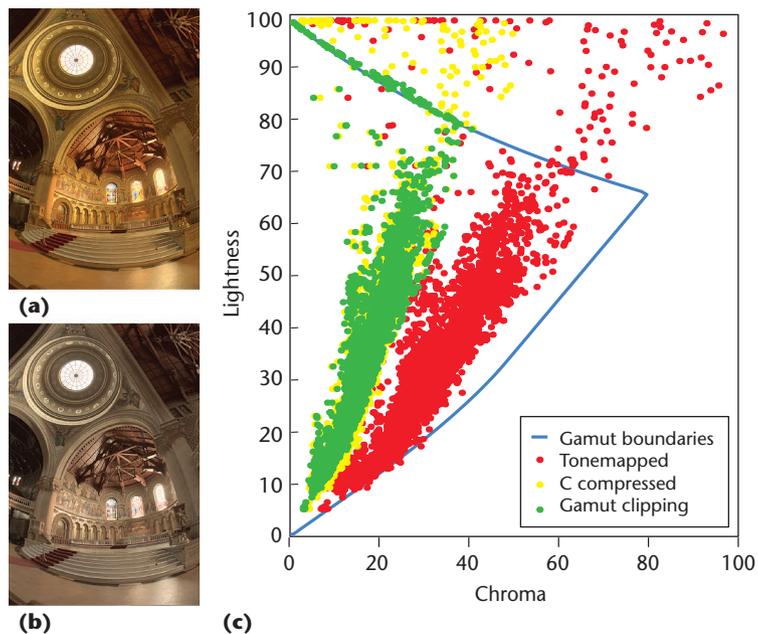


Figure 7. Gamut clipping: (a) image after tone mapping, (b) after our correction and clipping step, and (c) chroma values. The graph shows the chroma values (x axis) of pixels within a hue slice ($h = 60^\circ$) against their lightness values (y axis). After tone mapping the image's lightness channel, many pixels (shown in red) are still outside the gamut boundaries. Scaling the chroma channel moves pixels closer to the gamut boundaries (yellow). The last step in our framework ensures that pixels are mapped to values within the target gamut (green).

target gamut boundaries without further modifying pixels already in-gamut, we employ a clipping step. Because image pixels may be out of gamut both in terms of chroma and lightness, as Figure 7c shows, pixel values need to be clipped along both dimensions, which creates a trade-off between changes in lightness or in chroma for each pixel.

Although many proposals exist for defining the clipping line along which pixels should move, they are designed for scenarios where the input and target gamuts differ in terms of chromatic primaries used rather than luminance range. In our specific case, however, we have found that most out-of-gamut pixels tend to be bright, highly chromatic pixels (see the red pixels outside the gamut boundaries in Figure 7c).

Because of the narrowing of the cusp near high-L values, a delicate balance between chroma and lightness adjustments is necessary to ensure that the resulting image appearance does not change. We demonstrate this in Figure 8: the area around the sun either desaturates completely as the lightness values are near the peak of the cusp (Figure 8a) or it looks too saturated because its lightness is decreased with no corresponding chroma changes (Figure 8b).

Instead, we propose a middle ground between these two extremes. In a given hue slice, for a pixel



Figure 8. Three clipping solutions used to process the same image. Because most of the out-of-gamut pixels reside in the top right corner of the cusp, (a) clipping along only the chroma direction severely desaturates these pixels. In contrast, (b) clipping only along the lightness direction leads to an unnatural, oversaturated appearance. (c) Our interpolated solution produces an image that maintains a natural appearance while all pixels fit within the gamut boundaries.



Figure 9. SCACLIP gamut mapping method example⁹ integrated with an existing tone-mapping operator (TMO).⁸ This example shows that these methods produce artifacts, which our method can remove.

$p \in I'$ we determine a point along the gamut boundaries p_{clipC} such that $p_{clipC}(C) \in G(C)$ and $p_{clipC}(L) = p(L)$. Similarly, a value p_{clipL} is determined, where $p_{clipL}(C)$ remains unchanged and $p(L)_{clipL}$ moves to the gamut boundary. Once these two points have been computed, linear interpolation is performed to map the out-of-gamut pixel to the corresponding boundary of the destination gamut.

Results

We have validated our technique using several challenging HDR images, demonstrating its ben-

efits over existing techniques, including gamut-mapping solutions, color-correction methods, and CAMs. Additionally, we show the flexibility of our approach, which can be used with existing TMOs without degrading details or introducing unwanted artifacts. We evaluate the quality of our reproduction by measuring hue changes as well as with a psychophysical study comparing our chroma compression method with alternative techniques. All the results here assume sRGB primaries for both input and output. They have been gamma corrected using the sRGB gamma correction equation and use the chroma compression method we specified earlier.

Our approach can handle challenging images and produce natural results that preserve image details while fitting within the target output gamut. Figure 1 shows the result of processing an image with our framework and other techniques. (See the online supplemental material for more images: http://sccg.sk/~sikudova/publications/cga_gamut.html). Existing tone-mapping techniques (here we use the Photographic operator⁹) can effectively compress the image’s luminance but lead to an oversaturated appearance, with many pixels still out-of-gamut (for example, colors of the Macbeth color checker in Figure 1).

Although a gamut-mapping solution such as SCACLIP¹⁰ can control this issue by moving pixels within the gamut boundaries, it could amplify the appearance of oversaturation (see Figure 1e) or introduce artifacts (see Figure 9). At the same time, although such a gamut-mapping approach modifies lightness values, it cannot sufficiently compress an image’s extreme dynamic range if used alone (see Figure 10c).

In contrast, our framework combines the advantages of tone mapping and gamut mapping (see Figures 1c and 10e). Our solution also allows for flexibility in the choice of compressive function: different functions can lead to different image appearances, as Figure 11 shows.^{9, 11–13} Despite the different tone-mapping styles, our chroma correction leads to a consistent treatment of colors in the images.

Finally, Figure 12 compares our chroma correction with other color-correction solutions that are typically applied as a postprocess to tone compression as well as the result of SCACLIP after tone mapping. Note that the three color-correction methods shown do not consider the gamut boundaries and therefore may lead to out-of-gamut pixels. In our experiments, we found that this often includes more than 10 percent of the image pixels.



Figure 10. An HDR image after being processed with different compression solutions. (a) Tone mapped with the Photographic operator.⁹ (b) Tone mapped and then processed with the SCACLIP gamut-mapping method.¹⁰ (c) Processed with SCACLIP only. (d) Processed with iCAM06.⁶ (e) Tone mapped and processed with our chroma correction step. (f) Processed with our integrated lightness and chroma compression method.

Hue Differences

Ideally, separately compressing lightness and chroma at a fixed hue should not affect hues. To assess whether our algorithm achieves this goal, we evaluated our results using color-difference metrics. Typically, color differences are computed using ΔE color difference metrics, which take into account both luminance and chromatic differences. In our case, a metric capable of separating luminance, chroma, and hue is necessary because we are only interested in preserving the hue while luminance and chroma are being compressed.

Although color-difference measurements are commonly performed in the LAB color space, LAB is not hue-linear across all hues. Instead, we use an optimized $I'P'T'$ space for color difference comparisons.¹⁴ This space is scaled and rotated with respect to IPT such that color differences in this new space are directly comparable with other color difference metrics, while preserving hue linearity.

In $I'P'T'$, a cylindrical space is then computed, where lightness $\Delta I'$ and hue Δh differences can be calculated. We use Δh instead of the perceptually scaled CIE ΔH metric because the latter scales

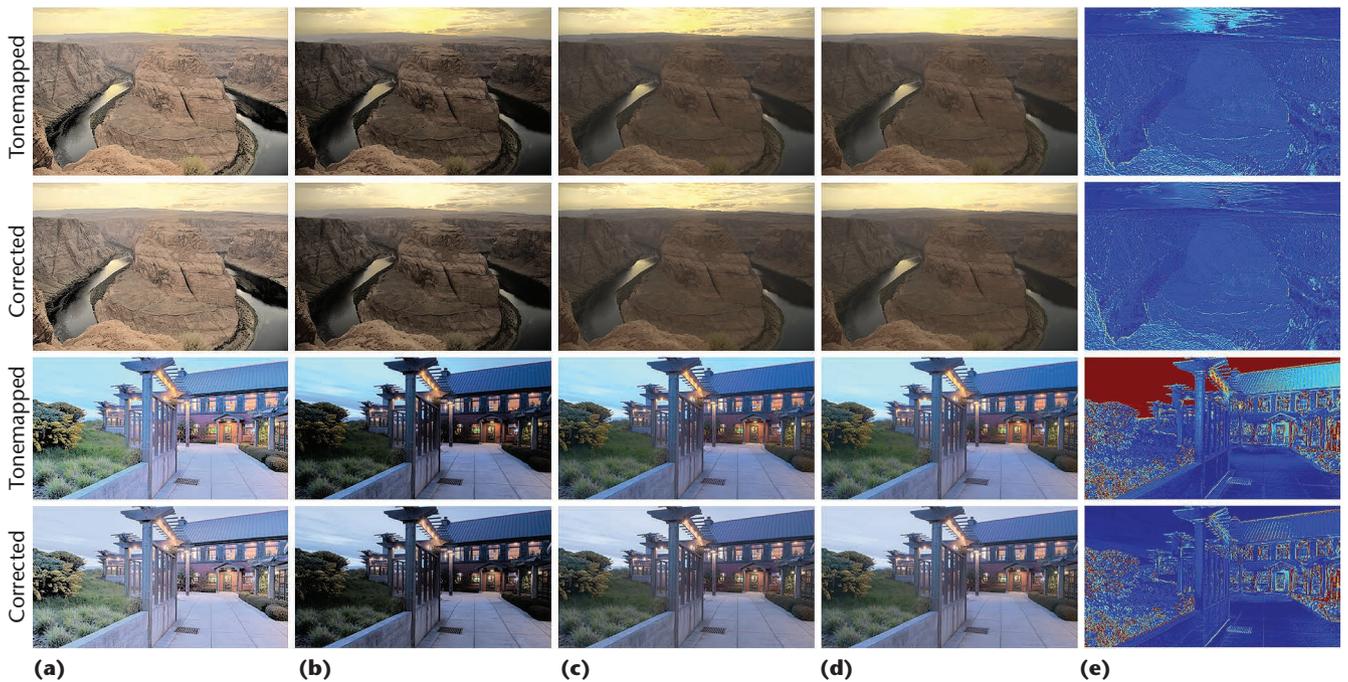


Figure 11. HDR images processed with various TMOs: (a) Li 2005,¹¹ (b) Ward 1997,¹² (c) Lischinski 2006,¹³ and (d) Reinhard 2002.⁹ The top row shows the output of the TMOs with no further processing, and the bottom row shows the output of the TMOs integrated with our chroma compression technique. (e) In the first column, we can see the hue differences between the TMO output and the HDR original (top) and between our corrected result and the HDR original (bottom) for the TMO by Reinhard 2002. The hue differences are reduced for other TMOs as well.

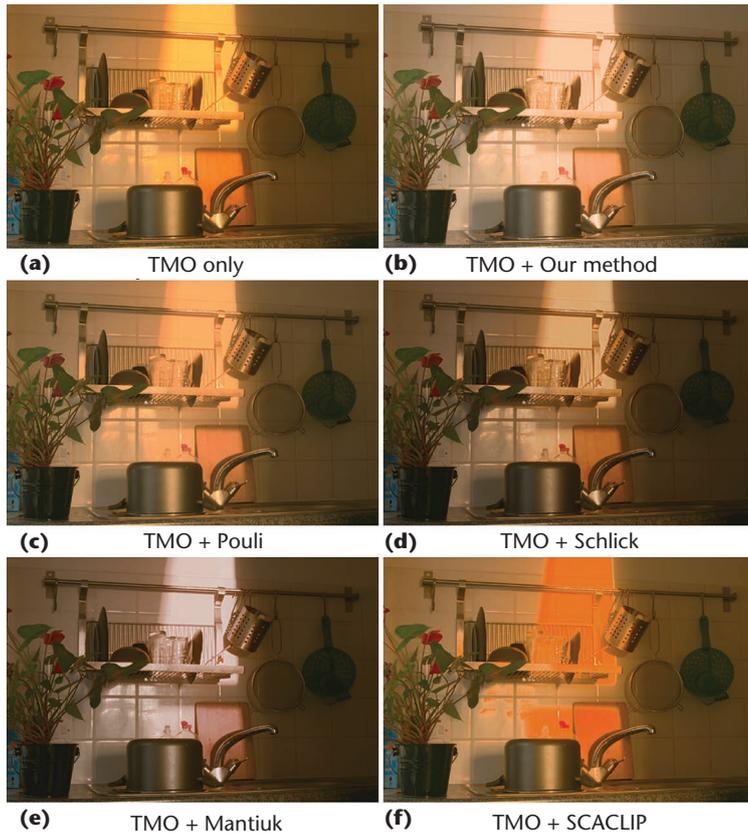


Figure 12. Tone-mapped results. (a) The tone-mapped image was corrected using (b) our method as well as (c, d, and e) existing color-correction solutions.³⁻⁵ The existing methods aim to correct the oversaturated appearance resulting from many tone-mapping solutions and operate as a postprocess on the image, without any gamut considerations. (f) The final image is the result of combining the SCACLIP gamut-mapping method with a TMO.

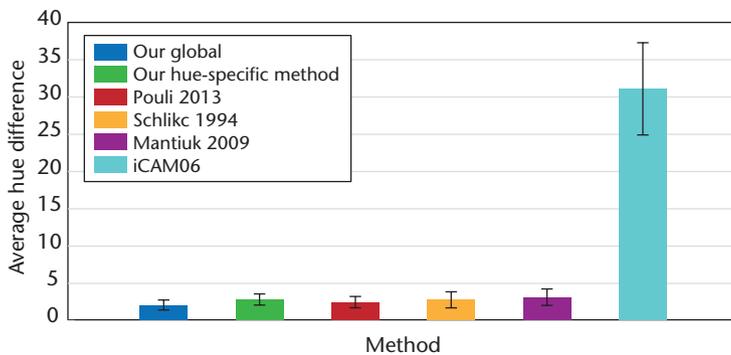


Figure 13. Hue differences and standard errors. We show the Δh graph computed over 20 images for the two proposed methods as well as alternative techniques.

hue differences by colorfulness to account for perceptual effects and is therefore not suitable for our particular application.

Because hue is defined on a circle, we compute Δh for a given pair of hues h_t and h_c as follows:

$$\Delta h = \min(|h_t - h_c|, |\min(h_t, h_c) + 2\pi - \max(h_t, h_c)|).$$

Figure 13 shows the Δh graph computed over 20 images for several methods.³⁻⁶ We use these methods to adjust saturation after the luminance dynamic range has been adjusted to the display capability using the Photographic operator.⁹ When using the iCAM06 method,⁶ the luminance dynamic range is compressed using its compression technique. Although iCAM06 was developed to reproduce the correct colors under different illumination conditions and in the context of HDR imaging, it introduces a large hue shift when compared with our proposed technique.

Psychophysical Evaluation

Typically, when compressing an image’s gamut for a particular display, the goal is to preserve the image’s color appearance and general quality as much as possible while conforming to a more limited gamut. To assess our method’s ability to preserve image quality despite gamut restrictions, we performed a psychophysical study. We used a two-alternative forced choice design, with the linearly scaled HDR reference shown at the same time. This allowed us to assess the fidelity of the color reproduction of the processed images compared with the HDR input.

This experiment had 13 participants (eight males and five females), who were between 22 and 25 years old and all had normal or corrected-to-normal vision as well as normal color vision. Based on a pilot study comparing our chroma compression method with the iCAM06 model,⁶ the SCACLIP gamut compression method,¹⁰ and the Mantiuk⁴ and Pouli⁵ correction methods, we opted to compare our method with the Mantiuk and Pouli methods in a complete experiment because we found participants preferred the other methods significantly less. Our evaluation also included the uncorrected tone-mapped image as a baseline result.

In our evaluation, the 13 participants viewed the differently processed results for six scenes in pairwise comparisons between the alternative methods. We asked them to select the image in each case that reproduced the color closest to the HDR image, which was shown as linearly scaled on the same screen. This experiment used a colorimetric calibrated NEC MultiSync P241W sRGB monitor. It is not possible to accurately reproduce the HDR ground truth on this monitor, so we allowed users to control the exposure for the HDR image manually so they could more accurately compare the processed results with the ground truth image.

Figure 14 and Table 1 show the detailed results. We performed significance analysis on the experi-

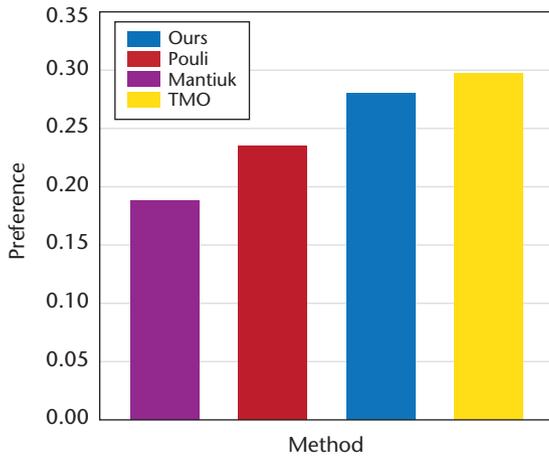


Figure 14. Psychophysical evaluation results. Aggregated preference results from our experiment for the different tested methods. The graph shows the average total preference per method.

ment results, computing the χ^2 value and agreement coefficient. Overall, the χ^2 value was 21.23 with an agreement coefficient of 0.033. At a 0.05 percent significance level, the critical χ^2 value is 12.59, indicating that our results are significant. We note however that the agreement between participants was not very high. By further analyzing our results for individual images, we observed that participants agreed in their choices for some images, while agreement was lower for other images. We also observed that images with higher agreement were generally more saturated and colorful.

Based on our findings, we repeated our analysis but split the images into two groups depending on overall saturation. For the saturated group, $\chi^2 = 53.38$ (agreement coefficient 0.315), and $\chi^2 = 6.69$ (agreement coefficient 0.002) for the less saturated group. These results suggest that the benefit of our method is more visible in more colorful images with higher saturation, which is to be expected because these images are more likely to have out-of-gamut pixels.

Overall, we found that although the Mantiuk method⁴ was chosen significantly fewer times, all other alternatives (our method, the Pouli method,⁵ and the uncorrected tone-mapped only images) were not found to be significantly different, suggesting that the participants found our method to be as visually pleasing as the uncorrected tone-mapped results. Because mapping out-of-gamut pixels inside the available gamut always presents a trade-off in visual quality, our method could be expected to offer a somewhat lower visual quality when compared with the Pouli method.⁵ However, our results show this not to be the case. The advantage of our proposed method therefore is the inclusion of unobtrusive gamut management.

Table 1. Preference matrix showing the number of times the method in each row was chosen over the method in each column.

Method	Mantiuk	Pouli	Ours	TMO
Mantiuk	0	31	28	29
Pouli	47	0	34	29
Ours	50	44	0	37
TMO	49	49	41	0

Given that both the image and display color space used in our psychophysical evaluation were sRGB, gamut management was only necessary due to potential out-of-gamut issues introduced by the tone-mapping process. The necessity for accurate gamut management, however, is likely to increase in the near future given the current consumer display trends toward higher dynamic range and wider gamut. The recent standard behind Ultra HD¹⁵ specifies a considerably larger gamut than the previous widely adopted color gamut standard,¹⁶ while concurrent proposals are pushing toward defining content at 4,000 or even 10,000 nits of peak luminance. At the same time, no displays exist that can achieve a full gamut or these luminance levels. Consequently, both tone mapping and gamut management will be necessary to ensure that content is displayed as intended. ❏

Acknowledgments

This work was partially supported by the Ministry of Science and Innovation Subprogramme Ramon y Cajal RYC2011-09372, TIN2013-47276-C6-1-R from the Spanish government, 2014 SGR 1232 from the Catalan government, and EC PARTHENOS (GA number 654119) from the Italian government.

References

1. E. Reinhard et al., *High Dynamic Range Imaging: Acquisition, Display and Image-Based Lighting*, 2nd ed., Morgan Kaufmann, 2010.
2. F. Banterle et al., *Advanced High Dynamic Range Imaging: Theory and Practice*, CRC Press, 2011.
3. C. Schlick, "Quantization Techniques for Visualization of High Dynamic Range Pictures," *Proc. 5th Eurographics Workshop on Rendering*, 1994, pp. 7–18.
4. R. Mantiuk et al., "Color Correction for Tone Mapping," *Computer Graphics Forum*, vol. 28, no. 2, 2009, pp. 193–202.
5. T. Pouli et al., "Color Correction for Tone Reproduction," *Proc. 21st Color and Imaging Conf. (CIC)*, 2013.

6. J. Kuang, G.M. Johnson, and M.D. Fairchild, "iCAM06: A Refined Image Appearance Model for HDR Image Rendering," *J. Visual Communication and Image Representation*, vol. 18, no. 5, 2007, pp. 406–414.
7. E. Reinhard et al., "Calibrated Image Appearance Reproduction," *ACM Trans. Graphics*, vol. 31, no. 6, 2012, article no. 201.
8. J. Morovič, *Color Gamut Mapping*, 1st ed., Wiley, 2008.
9. E. Reinhard et al., "Photographic Tone Reproduction for Digital Images," *ACM Trans. Graphics*, vol. 21, no. 3, 2002, pp. 267–276.
10. N. Bonnier et al., "Spatial and Color Adaptive Gamut Mapping: A Mathematical Framework and Two New Algorithms," *Proc. 15th Color Imaging Conf.*, 2007, pp. 267–272.
11. Y. Li, L. Sharan, and E.H. Adelson, "Compressing and Companding High Dynamic Range Images with Subband Architectures," *ACM Trans. Graphics*, vol. 24, no. 3, 2005, pp. 836–844.
12. G. Ward, H. Rushmeier, and C. Piatko, "A Visibility Matching Tone Reproduction Operator for High Dynamic Range Scenes," *IEEE Trans. Visualization and Computer Graphics*, vol. 3, no. 4, 1997, pp. 291–306.
13. D. Lischinski et al., "Interactive Local Adjustment of Tonal Values," *ACM Trans. Graphics*, vol. 25, no. 3, 2006, pp. 646–653.
14. S. Shen, "Color Difference Formula and Uniform Color Space Modeling and Evaluation," PhD dissertation, Rochester Inst. of Technology, 2009.
15. Recommendation ITU-R BT.2020: Parameter Values for Ultra-High Definition Television Systems for Production and International Programme Exchange, Int'l Telecommunication Union, 2012.
16. Recommendation ITU-R BT.709-3: Parameter Values for the HDTV Standards for Production and International Programme Exchange, Int'l Telecommunication Union, 1998.

Elena Šikudová is an assistant professor in the Department of Applied Informatics, Faculty of Mathematics, Physics and Informatics, at Comenius University. Her research interests include computer vision, HDR imaging, visual perception, and image processing. She coauthored the book *Computer Vision: Object Detection and Recognition* (Wikina, 2014, in Slovak). Šikudová has a PhD in geometry and topology from Comenius University. Contact her at sikudova@sccg.sk.

Tania Pouli is a researcher at Technicolor Research & Innovation. Her research interests include high HDR and color imaging, particularly in the context of movie postproduction. Pouli has a PhD in image statistics and applications

from the University of Bristol. She is the lead author of the book *Image Statistics in Visual Computing* (CRC Press, 2013). Contact her at taniapouli@gmail.com.

Alessandro Artusi is a Ramon y Cajal fellow at the University of Girona. His research interests include color science, image processing, image compression standardization, HDR imaging, image-quality assessment, and energy-aware algorithms. Artusi has a PhD in computer science from the Vienna University of Technology. He is a member of the ISO/IEC WG1 and WG11, known as JPEG and MPEG respectively, where he has coedited the standard ISO/IEC-18477 JPEG XT. He is a coauthor of the book *Advanced High Dynamic Range* (CRC Press, 2011) and author of the upcoming book *Image Content Retargeting Maintaining Color, Tone, and Spatial Consistency* (CRC Press, 2016). Contact him at artusialessandro4@gmail.com.

Ahmet Oğuz Akyüz is an associate professor in the Department of Computer Engineering at Middle East Technical University. His research interests include computer graphics and image processing with an emphasis on HDR and color imaging. Akyüz has a PhD in computer science from the University of Central Florida. Contact him at akyuz@ceng.metu.edu.tr.

Francesco Banterle is a postdoctoral researcher in the Visual Computing Laboratory at ISTI-CNR. His research interests include HDR imaging and rendering. Banterle has a PhD in engineering from Warwick University. He is the first coauthor of the book *Advanced High Dynamic Range* (CRC Press, 2011) and the editor in chief of the *Journal of Graphics Tools*. Contact him at frabante@gmail.com.

Zeynep Miray Mazlumoğlu has a BS in computer engineering from Middle East Technical University. Her research interests include image processing and computer graphics. Contact her at zmmazlumoglu@gmail.com.

Erik Reinhard is a distinguished scientist at Technicolor France. His research interests include color transfer, color imaging, HDR imaging, human visual perception, and image statistics. Reinhard has a PhD in computer science from the University of Bristol. He has coauthored books on high dynamic range imaging, color science, computer graphics, and natural image statistics, was the founding editor in chief of the *ACM Transactions on Applied Perception*, and is an associate editor of the *ACM Transactions on Graphics* and *ACM Transactions on Applied Perception*. Contact him at reinhard.erik@gmail.com.



Selected CS articles and columns are also available for free at <http://ComputingNow.computer.org>.