Efficient hue-preserving and edge-preserving spatial color gamut mapping

Øyvind Kolås and Ivar Farup; Gjøvik University College, Gjøvik, Norway

Abstract

We present a new efficient hue- and edge-preserving spatial color gamut mapping algorithm. The initial computation of the algorithm is to project all out-of-gamut colors to the destination gamut boundary towards the center of the gamut. Based on this spatially invariant hue-preserving clipping of the image, we construct a greyscale map indicating the amount of compression performed. This map can be spatially modified by applying an edge-preserving smoothing filter that never decreases the amount of compression applied to an individual pixel. Finally, the colors of the original image are compressed towards the gamut center according to the filtered map. Examples on real images show that the algorithm gives interesting results.

Introduction

Color gamut mapping algorithms (GMAs) has been an active field of research for quite some time. A good review of the state of the art of gamut mapping was given by Morovič and Luo in 2001 [1]. Recently, spatial GMAs has become an active and important field of research. Farup et al. [2] gave a thorough review of spatial GMAs along with the presentation of their novel technique. Bonnier et al. [3] suggested to group the spatial GMAs into two groups. For the algorithms in the first group, a functional to be minimized is defined. This can e.g., be a contrast measure [4] or a Retinex-related measure [5]. Optimization is then performed using standard optimization techniques [4] or a variational approach [5]. The second group of algorithms work by first performing some kind of spatially invariant algorithm (most often clipping), and then reinserting some of the high-frequency information that is lost in the process. Often, this results in an image that is again slightly out of gamut, so the process can be iterated. Multilevel gamut mapping algorithms like those of Bala et al. [6], Morovič and Wang [7], Zolliker and Simon [8], and Farup et al. [9] are examples of this approach. In this paper, we focus only on this second group of algorithms.

The main advantage of spatial GMAs is that they improve the rendering of details. However, as discussed by Farup et al. [9], this often happens at the cost of introducing some new problems.

Halos: When high frequency information is reinjected into the gamut mapped image, haloing artefacts near sharp edges can easily appear. Bala [6] argued that this can be a good thing in that it can, through an artificial Kornsweet effect, increase the strenght of edges that would else disappear. However, for most images, it is a problem that should be avoided. Farup et al. [2] suggested a method to strongly reduce haloing artefacts by the means of only changing colors close to those of the low-pass filtered image, and forcing the change to occur on lines of constant hue. Zolliker [8] more or less completely eliminated the problem of halos by using computationally expensive edge-preserving bilateral filters instead of conventional blur filters.

Hue changes: Hue preservation is considered an imporant goal in gamut mapping [1]. The high-pass information that is reinjected into the spatially filtered image is obtained by a per-channel spatial filtering. Thus, the added information does not necessarily change the image in a hue-preserving manner. It was demonstrated by Farup et al. [2] that the classical multilevel approach can severly disturb the hue close to sharp edges. They further suggested a way to force the color change to occur at constant hue.

Computation time: Conventional spatially invariant GMAs can be implemented in terms of simple 3D LUTs, and are therefore inherently fast. Spatial processing, however, can easily become time consuming. The algorithm of Bala [6] performs quite well (but introduces halos and hue changes), whereas the algorithms of Morovic and Wang [7], Zolliker and Simon [8] and Farup et al. [2] are complex and slow. In order to be feasible to apply in a printing process, GMAs should not exceed a complexity of $O(N)$, N being the number of pixels.

As seen, some of the previously proposed algorithms solve some of these problems, but thus far there exists no algorithm solving all of them. In the next section, we present a new spatial GMA that solves all of them and that gives good results. Some example results from running the algorithm on real images and gamuts are then given and discussed in the following section.

The proposed algorithm

The general idea of the proposed algorithm goes as follows. First, the image is gamut clipped. Then a map indicating the relative amount of compression needed for each pixel to achieve clipping is constructed from the original image and the gamut clipped one. This map is then smoothed by means of spatial filtering. Finally, the colors of the original image are gamut compressed according to the spatially filtered map. A graph giving the overview of the proposed algorithm is shown in Figure 1. In the following subsections, each step of the algorithm will be described in more detail.

Hue preserving clipping

First, we compute an intermediate relative colorimetric image where the hue of out-of-gamut colors is preserved. This is done in CIELAB by projecting an out-of-gamut color (C) towards the middle grey of the gamut along the L axis in the destination gamut. See Figure 2.

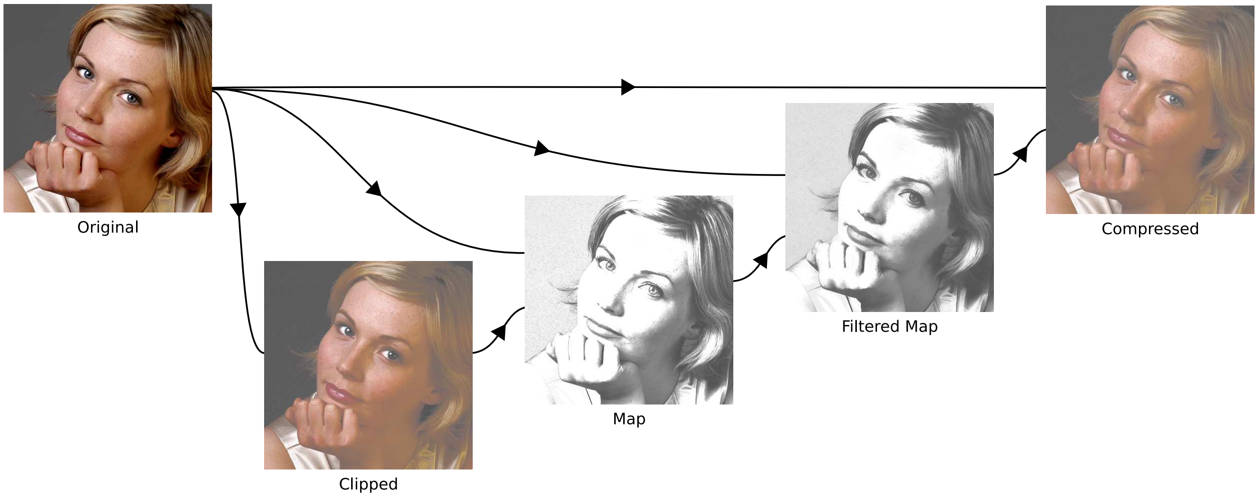

Figure 1. *Overview of the proposed spatial gamut mapping technique.*

Map computation

From the original image and the gamut clipped one, we compute a map telling us how much the individual colors of the image had to be compressed towards grey during the clipping:

$$m = \frac{||C' - g||}{||C - g||}, \tag{1}$$

where C is the color of the original pixel, C' the color of the gamut clipped image, and g is the center of the gamut on the gray axis. $||\cdot||$ denotes the L_2 norm of the colour space, i.e., the CIELAB ΔE.

The clipped image can now be computed as convex linear combination of the original image and grey using the map as the ratio,

$$C' = mC + (1-m)g. \tag{2}$$

The computed compression map is the upper bound for saturation we can reproduce for the individual colors in the image.

Spatial filtering

By modifying the generated map, we can change the amount of compression applied to each pixel. To avoid moving colors out of gamut, we must make sure to never increase values in the map. The map should be continuous in regions that are continuous in the original image.

This can be obtained by filtering the map through an edge preserving smoothing minimum filter controlled by edge information from the original. The filter we used computed the value for the center pixel of a neighbourhood as the average value of the set of pixels in the neighbourhood chosen by a color enhanced Symmetric Nearest Neighbor criteria [10] that had a compression value lower than the center pixel. We iteratively apply this filter with square neighbourhoods 33, 17, 9, 5 and 3 pixels wide. 3 × 3 pixels.

The effect of applying this filter to the map is that when the map is used we increase the compression of neighbourhoods containing out-of-gamut colors thus bringing back some of the details lost by gamut clipping.

Results and Discussion

Figure 3 shows how the algorithm performs on four real images. The images were mapped to the very small gamut of an HP Design Jet printer printing on ordinary copy paper. The central column shows the original image, and the gamut clipped image is shown on the left hand side. The result of the proposed algorithm is shown on the right hand side.

Performance

In the reproduced images, colors of in-gamut regions are relatively colorimetrically reproduced, regions that are out of gamut gamut and in vicinity have been compressed in a hue preserving manner. Details that are lost in the clipping process are thus clearly rendered in the final image due to the spatial filtering of the map. This is particularly evident in the detailed dark regions of the images, such as in the lower left quadrant of the camera image.

It should be noticed that the gamut mapped images in the right column do not show any signs of haloing artefacts near sharp boundaries. This is due to the use of an edge-preserving spatial filter, and is unlike the results of multilevel algorithms like those of Bala et al. [6] and Morovič and Wang [7], and to a certain degree, also that of Farup et al. [2].

Like the algorithm of Farup et al. [2], and unlike all other previously proposed spatial GMAs, the resulting images have their CIELAB hue preserved by definition. Despite this fact, there are still some noticable hue shifts in the resulting images. In particular, there is a hue shift towards purple for the blue bottle in the picnic image. This is a result of the imperfect hue constancy of the CIELAB color space [11, 12], and not a result of the proposed GMA.

The gamut clipped algorithm shown on the left part of Figure 3, shows some typical clipping artefacts. In particular, some new edges has been inserted where the colors of the image crosses the gamut boundary. Since the spatial filter used for filtering the compression map takes its edge information from the original image, and not from the computed map itself, these edges are not visible in the final images.

Figure 4 shows the comparison of five state-of-the-art GMAs. In this case, the images have been gamut mapped to the slightly bigger gamut of an Océ plotter printing on plain paper. From left to right, top to bottom, the algorithms used are the original image, hue preserving minimum ΔE clipping, SGCK [], the presently proposed algorithm, the algorithm of Farup et al. [2], and the algorithm proposed by Zolliker and Simon [8].

As suggested by Zolliker et al. [13], we also tested the algorithm on synthetical images composed of continous gradients, such as the surfaces of the sRGB cube. The resulting images did not contain the strong artificial edges typically resulting for spatially invariant GMAs for such images.

Computational Complexity

The first part of the algorithm is gamut clipping. In our implementation this is performed through the use of a lookup table computed once per destination gamut that has to be used once for each pixel in the image (alternatively, once per color in the image, but for high bit depth images, this is more or less the same). Thus the first part is $O(N)$, N being the number of pixels. The same applies to the computation of the initial compression map according to Equation (1).

The spatial filtering of the map is performed with a filter of constant size, independent of the image size. Thus, the complexity of the filtering is also $O(N)$. However, it should be noted that really high resolution images would probably benefit from using larger filters, thereby increasing the complexity of this phase. The application of the map for performing the final gamut compression of the original image by Equation (2) is also $O(N)$. Thus, as long as the size of the filter is kept constant, the whole algorithm is $O(N)$. While multilevel algorithms typically have complexities of $O(N\log(N)^2)$.

Conclusion

The proposed method for spatial GMA performs with a low computational complexity. It has shown to be promising for other applications such as tone mapping of high dynamic range images as well. The performance does not scale well with the radius of the filter used for the spatial processing. However, since the algorithm works well for quite large images with a constantly sized 33×33 kernel, this is not a major issue. Potential areas for future research include performance improvements and implementations with slightly different performance characteristics for the edge preserving minimizing/maximizing smoothing filters, and using the method for tone mapping.

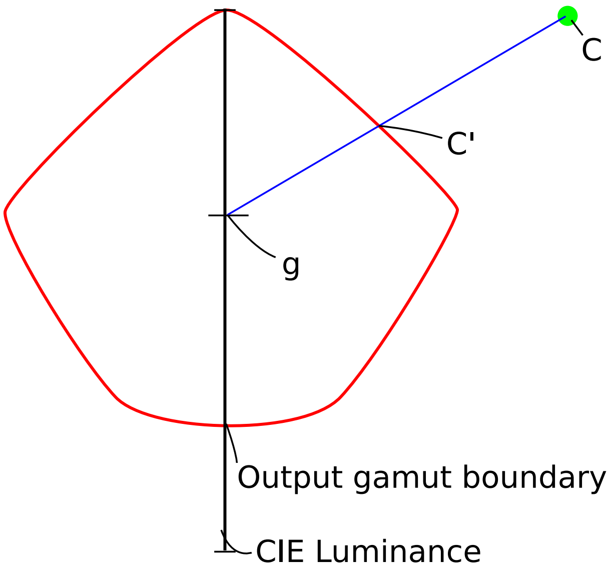

Figure 2. *Gamut clipping along straight lines towards the center of the L^* axis of the CIELAB color space within the gamut.*

References

[1] Ján Morovič and M. Ronnier Luo. The fundamentals of gamut mapping: A survey. *Journal of Imaging Science and Technology*, 45(3):283–290, 2001.

[2] Ivar Farup, Carlo Gatta, and Alessandro Rizzi. A multiscale framework for spatial gamut mapping. *IEEE Trans. Image Proc.*, 16(10), 2007. DOI: 10.1109/TIP.2007.904946.

[3] Nicolas Bonnier, Francis Schmitt, Hans Brettel, and Stéphane Berche. Evaluation of spatial gamut mapping algorithms. In *Proceedings of IS&T and SID's 14th Color Imaging Conference: Color Science and Engineering: Systems, Technologies, Applications*, Scottsdale, Arizona, 2006. (Accepted for publication).

[4] Shigeki Nakauchi, Masahiro Imamura, and Shiro Usui. Color gamut mapping by optimizing perceptual image quality. In *Proceedings of IS&T and SID's 4th Color Imaging Conference: Color Science, Systems and Applications*, pages 63–66, Scottsdale, Arizona, 1996.

[5] Ron Kimmel, Doron Shaked, Michael Elad, and Irwin Sobel. Space-dependent color gamut mapping: A variational approach. *IEEE Trans. Image Proc.*, 14(6):796–803, June 2005.

[6] Raja Bala, Ricardo deQueiroz, Reiner Eschbach, and Wencheng Wu. Gamut mapping to preserve spatial luminance variations. *Journal of Imaging Science and Technology*, 45(5):436–443, September/October 2001.

[7] Ján Morovič and Yu Wang. A multi-resolution, full-colour spatial gamut mapping algorithm. In *Proceedings of IS&T and SID's 11th Color Imaging Conference: Color Science and Engineering: Systems, Technologies, Applications*, pages 282–287, Scottsdale, Arizona, 2003.

[8] Peter Zolliker and Klaus Simon. Retaining local image information in gamut mapping algorithms. *IEEE Trans. Image Proc.*, 16(3):664–672, March 2007.

[9] Ivar Farup, Jan H. Wold, Thorstein Seim, and Torkjel Søndrol. Generating lights with specified spectral power distributions. *Applied Optics*, 46(13):2411–2422, 2007.

[10] David Harwood, Muralidhara Subbarao, Hannu Hakalathi, and Larry S. Davis. A new class of edgepreserving smoothing filters. *Pattern Recognition Letters*, 6:155–162, 1987.

[11] Nathan Moroney. A hypothesis regarding the poor blue constancy of cielab. *Color Res. Appl.*, 28(5):371–378, October 2003.

[12] John J. McCann. Color spaces for color-gamut mapping. *Journal of Electronic Imaging*, 8(4):354–364, 1999.

[13] Peter Zolliker, Markus Dätwyler, and Klaus Simon. On the continu-

ity of gamut mapping algorithms. In *Color Imaging X: Processing, Hardcopy, and Applications*, pages 220–232, San Jose, CA, USA, 2005.

Author Biography

Øyvind Kolås is a lecturer, free-software developer and artist. He received a BSc in digital media technology from Gjøvik University College in 2005, where he started working as a teaching assistant in 2002 and is currently part of the academic staff lecturing programming, digital media-technology and image processing.

Ivar Farup received his MSc in physics from the Norwegian University of Science and Technology in Trondheim, Norway in 1994, and a PhD in applied mathematics from the University of Oslo in 2000. Since 2000, he has worked as an Associate Professor at Gjøvik University College, mainly focusing on color imaging. He is a member of IS&T.

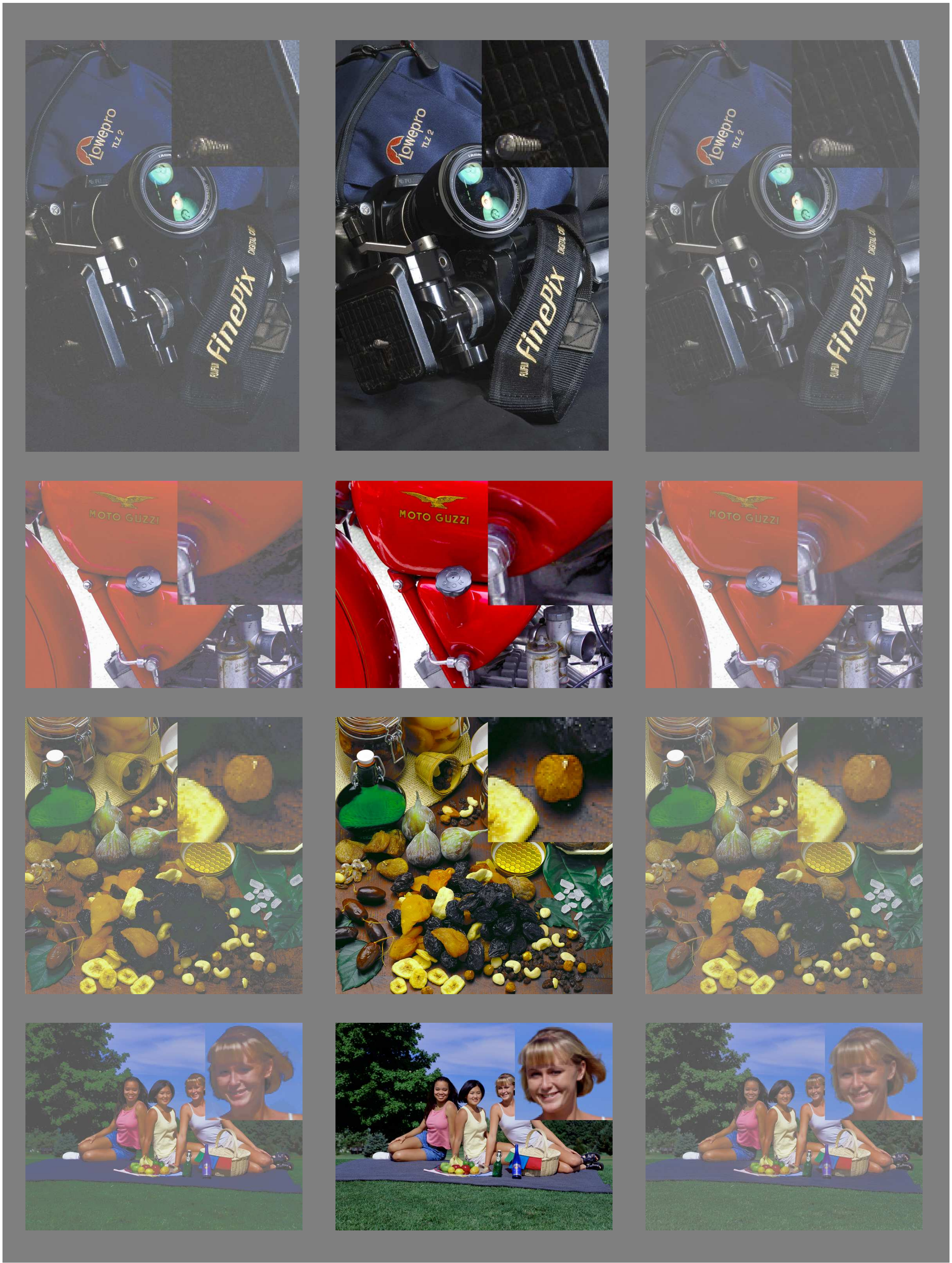

Figure 3. *Images gamut mapped using the proposed algorithm. The central column shows the original image, and the gamut clipped image is shown on the left hand side. The result of the proposed algorithm is shown on the right hand side.*

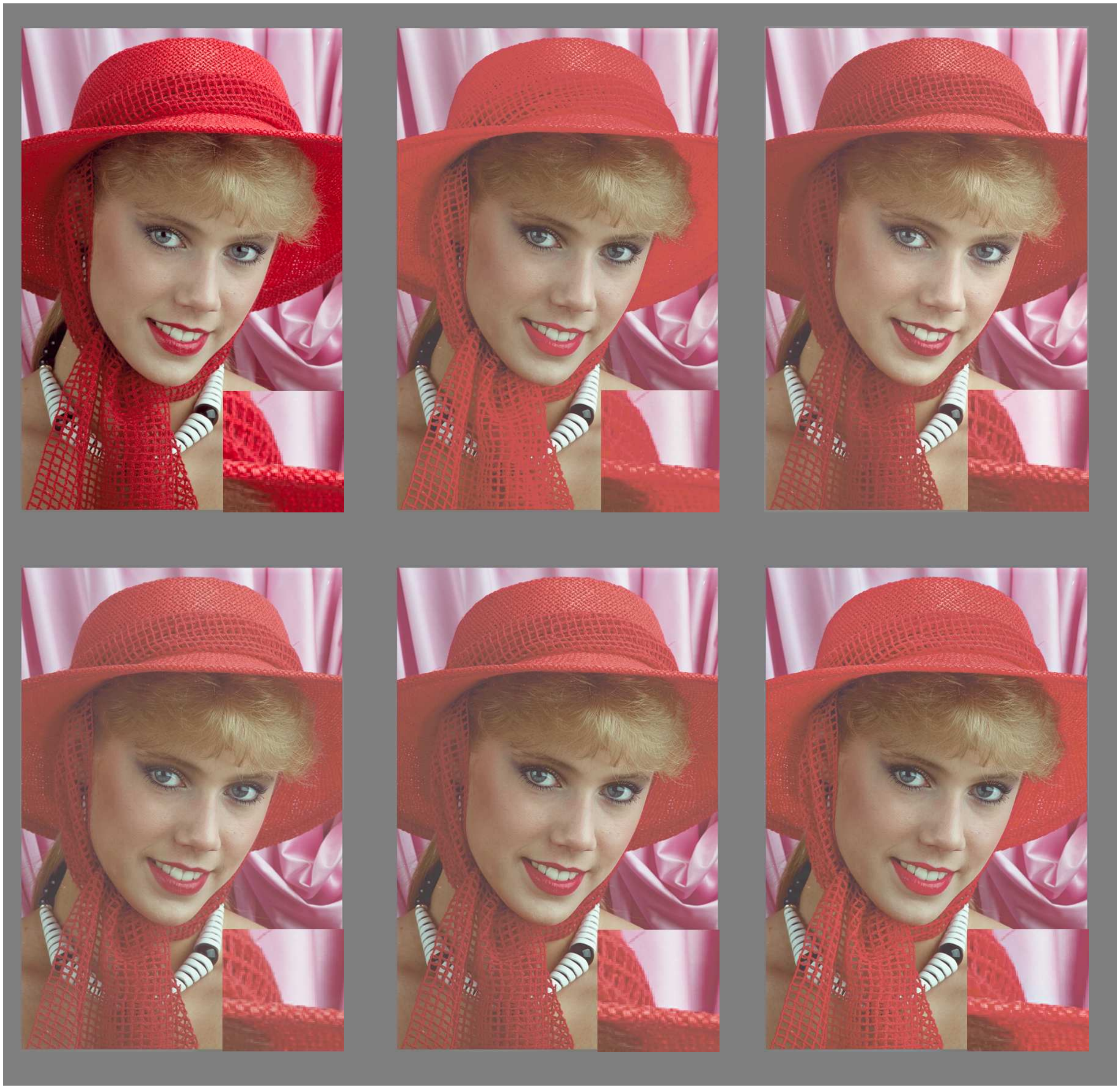

Figure 4. *Image gamut mapped using various spatially variant and invariant techniques. From left to right, top to bottom, the original image and results produced by, hue preserving minimum* ΔE *clipping, SGCK [], the presently proposed algorithm, the algorithm of Farup et al. [2], and the algorithm proposed by Zolliker and Simon [8].*