

Spatial Colour Gamut Mapping by Means of Anisotropic Diffusion

Ali Alsam¹ and Ivar Farup²

¹ Sør-Trøndelag University College, Trondheim, Norway

² Gjøvik University College, Gjøvik, Norway

Abstract. We present a computationally efficient, artifact-free, spatial colour gamut mapping algorithm. The proposed algorithm offers a compromise between the colorimetrically optimal gamut clipping and an ideal spatial gamut mapping. It exploits anisotropic diffusion to reduce the introduction of halos often appearing in spatially gamut mapped images. It is implemented as an iterative method. At iteration level zero, the result is identical to gamut clipping. The more we iterate the more we approach an optimal, spatial gamut mapping result. Our results show that a low number of iterations, 10–20, is sufficient to produce an output that is as good or better than that achieved in previous, computationally more expensive, methods. The computational complexity for one iteration is $O(N)$, N being the number of pixels. Results based on a challenging small destination gamut supports our claims that it is indeed efficient.

Keywords: spatial gamut mapping, colour reproduction, anisotropic diffusion.

1 Introduction

To accurately define a colour, three independent variables need to be fixed. In a given three dimensional colour space, the colour gamut is the volume enclosing all the colour values that can be reproduced by the reproduction device or present in the image. Colour gamut mapping is the problem of representing the colour values of an image within the gamut of a reproduction device, typically a printer or a monitor. Furthermore, in the general case, when an image gamut is larger than the destination gamut some visual image information will be lost. We therefore redefine gamut mapping as the problem of representing the colour values of an image within the gamut of a reproduction device with minimum loss of visual information, i.e., as visually close as possible.

Unlike single colours, images are represented in a higher dimensional space than three, i.e. knowledge of the exact colour values is not, on its own, sufficient to reproduce an unknown image. In order to fully define an image, the spatial context of each colour pixel needs to be fixed. Based on this, we define two categories of gamut mapping algorithms. In the first, colours are mapped independent of their spatial context [1]. In the second, the mapping is influenced by the local context of each colour value [2,3,4,5]. The latter category is referred to as spatial colour gamut mapping.

Eschbach [6] stated that although the accuracy of mapping of a single colour is well defined, the reproduction accuracy of images isn't. To elucidate this claim, with which we agree, we consider a single colour that is defined by its hue, saturation and lightness. Assuming that such a colour is outside the target gamut, we can modify its components independently. That is to say, if the colour is lighter or more saturated than what can be achieved inside the reproduction gamut, we shift its lightness and saturation to the nearest feasible values. Further, in most cases it is possible to reproduce colours without shifting their hue.

Taking the spatial context of colours into account presents us with the challenge of defining the spatial components of a colour pixel and incorporating this information into the gamut mapping algorithm. Generally speaking, we need to define rules that would result in mapping two colours with identical hue, saturation and lightness to two different magnitudes depending on their context in the image. The main challenge is, thus, defining the spatial context of an image pixel in a manner that results in an improved gamut mapping. By improved we mean that the appearance of the resultant in-gamut image is closer to the original as judged by a human observer. Further, from a practical point of view, the new definition needs to result in an algorithm that is fast and does not result in image artifacts.

It is well understood that the human visual system is more sensitive to spatial ratios than to absolute luminance values [7]. This knowledge is at the heart of all spatial gamut mapping algorithms. A rephrasing of spatial gamut mapping is then the problem of representing the colour values of an image within the gamut of a reproduction device while preserving the spatial ratios between different colour pixels. In an image, spatial ratios are the difference, given some metric, between a pixel and its surround. This can be the difference between one pixel and its adjacent neighbors or pixels far away from it. Thus, we face the problem that spatial ratios are defined in different scales and dependent on the chosen difference metric.

McCann suggested to preserve the spatial gradients at all scales while applying gamut mapping [8]. Meyer and Barth [9] suggested to compress the lightness of the image using a low-pass filter in the Fourier domain. As a second step the high-pass image information is added back to the gamut compressed image. Many spatial gamut mapping algorithms have been based upon this basic idea [2,4,10,11,12].

A completely different approach was taken by Nakauchi et al. [13]. They defined gamut mapping as an optimization problem of finding the image that is perceptually closest to the original and has all pixels inside the gamut. The perceptual difference was calculated by applying band-pass filters to Fourier-transformed CIELAB images and then weighing them according to the human contrast sensitivity function. Thus, the best gamut mapped image is the image having contrast (according to their definition) as close as possible to the original. Kimmel et al. [3] presented a variational approach to spatial gamut mapping where it was shown that the gamut mapping problem leads to a quadratic programming formulation, which is guaranteed to have a unique solution if the

gamut of the target device is convex. However, they did not apply their method to colour images.

Finding an adequate description of the surface of the gamut, commonly denoted a gamut boundary descriptors (GBDs) is an important step in any colour gamut mapping algorithm. One of the main challenges is the fact that gamut surfaces are most often concave. Many methods for finding the GBD have been proposed over the years. Recently, Bakke et al. [14] presented a survey and evaluation of the most common method, showing that the modified convex hull algorithm by Balasubramanian and Dalal [15] is generally the most reliable one.

The algorithm presented in this paper adheres to our previously stated definition of spatial gamut mapping in that we aim to preserve the spatial ratios between pixels in the image. We start by calculating the gradients of the original image in the CIELAB colour space. The image is then gamut mapped by projecting the colour values to the nearest, in gamut, point along hue-constant lines. The difference between the gradient of the gamut mapped image and that of the original is then iteratively minimized with the constraint that the resultant colour is inside the destination gamut and has the same hue as the original. The scale at which the gradient is preserved is related to the number of iterations and the extent to which we can fit the original gradients into the destination gamut.

The main contributions of this work are as follows. We first present a mathematical formulation of the gamut mapping problem in colour space. Our formulation can be extended to a higher dimensional space than three. Secondly, to avoid halos which are a main drawback in previous spatial gamut mappings techniques we introduce a modification of the algorithm presented by Alsam and Farup [16]. In [16] we observe that halos are visible in the resultant gamut mapped images at strong lightness or chromatic edges. Furthermore, those edges are generally visible in the gamut clipped image. That is to say that halos are the result of over enhancing visible edges. In this article we avoid this problem by using anisotropic diffusion [17] where the gradients of the gamut mapped image are improved based on their strength. This method is known to improve the classic strategy for a number of applications such as zooming [18] and image compression [19]. With anisotropic diffusion, the diffusion is encouraged within regions and prohibited across strong edges thus avoiding the introduction of halos. Finally, our results show that as few as ten iterations are sufficient to produce an output that is similar or better than previous methods. Being able to improve upon previous results using such low number of iterations allows us to state that the proposed algorithm is fast.

2 Spatial Gamut Mapping: A Mathematical Definition

Let's say we have an original image with pixel values $\mathbf{p}(x, y)$ (bold face to indicate vector) in CIELAB or any similarly structured colour space. A gamut clipped image can be obtained by leaving in-gamut colours untouched, and projecting the out-of-gamut colours onto the gamut surface along straight lines defined by

\mathbf{g} , the center of the gamut on the L axis and the nearest in-gamut colour. Let's denote the gamut clipped image, which is the outcome of such a conventional gamut clipping algorithm as $\mathbf{p}_c(x, y)$. Clearly, the gamut clipped image is a convex linear combination of the original image and the grey point,

$$\mathbf{p}_c(x, y) = \alpha_c(x, y)\mathbf{p}(x, y) + (1 - \alpha_c(x, y))\mathbf{g} \quad (1)$$

which can be solved for α_c :

$$\alpha_c(x, y) = \frac{\|\mathbf{p}_c(x, y) - \mathbf{g}\|}{\|\mathbf{p}(x, y) - \mathbf{g}\|}. \quad (2)$$

A spatial gamut mapped image is obtained by further compressing the pixel value towards the grey by some hitherto unknown amount $\alpha_s(x, y)$, i.e.,

$$\begin{aligned} \mathbf{p}_s(x, y) &= \alpha_s(x, y)\mathbf{p}_c(x, y) + (1 - \alpha_s(x, y))\mathbf{g} \\ &= \alpha_s(x, y)\alpha_c(x, y)\mathbf{p}(x, y) + (1 - \alpha_s(x, y)\alpha_c(x, y))\mathbf{g} \\ &= \alpha(x, y)\mathbf{p}(x, y) + (1 - \alpha(x, y))\mathbf{g}, \end{aligned} \quad (3)$$

where $\alpha(x, y) = \alpha_s(x, y)\alpha_c(x, y)$ has been introduced. A visual illustration of the parameters involved is shown in Figure 1.

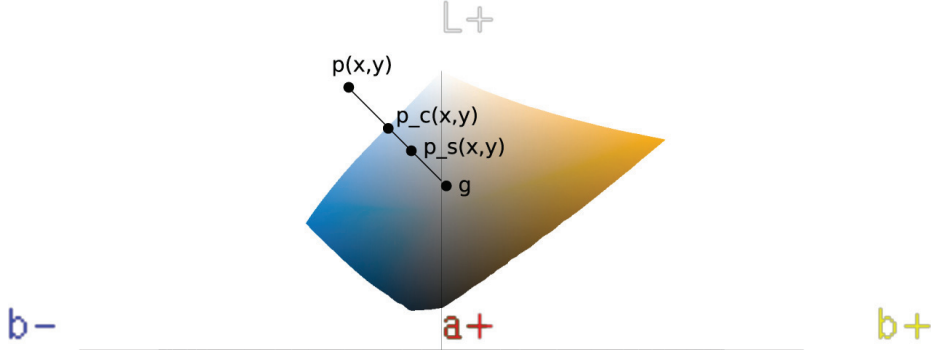


Fig. 1. A representation of the spatial gamut mapping problem. $p(x, y)$ is the original colour at image pixel (x, y) , this value is clipped to the gamut boundary resulting in a new colour $p_c(x, y)$ which is compressed based on the gradient information to a new value $p_s(x, y)$.

In a previous paper [16], we demonstrated that a reasonable spatial gamut mapping algorithm can be achieved by minimising

$$\min \int \|\nabla \mathbf{p}_s(x, y) - \nabla \mathbf{p}(x, y)\|^2 dA \quad \text{subject to} \quad \alpha_s(x, y) \in [0, 1]. \quad (4)$$

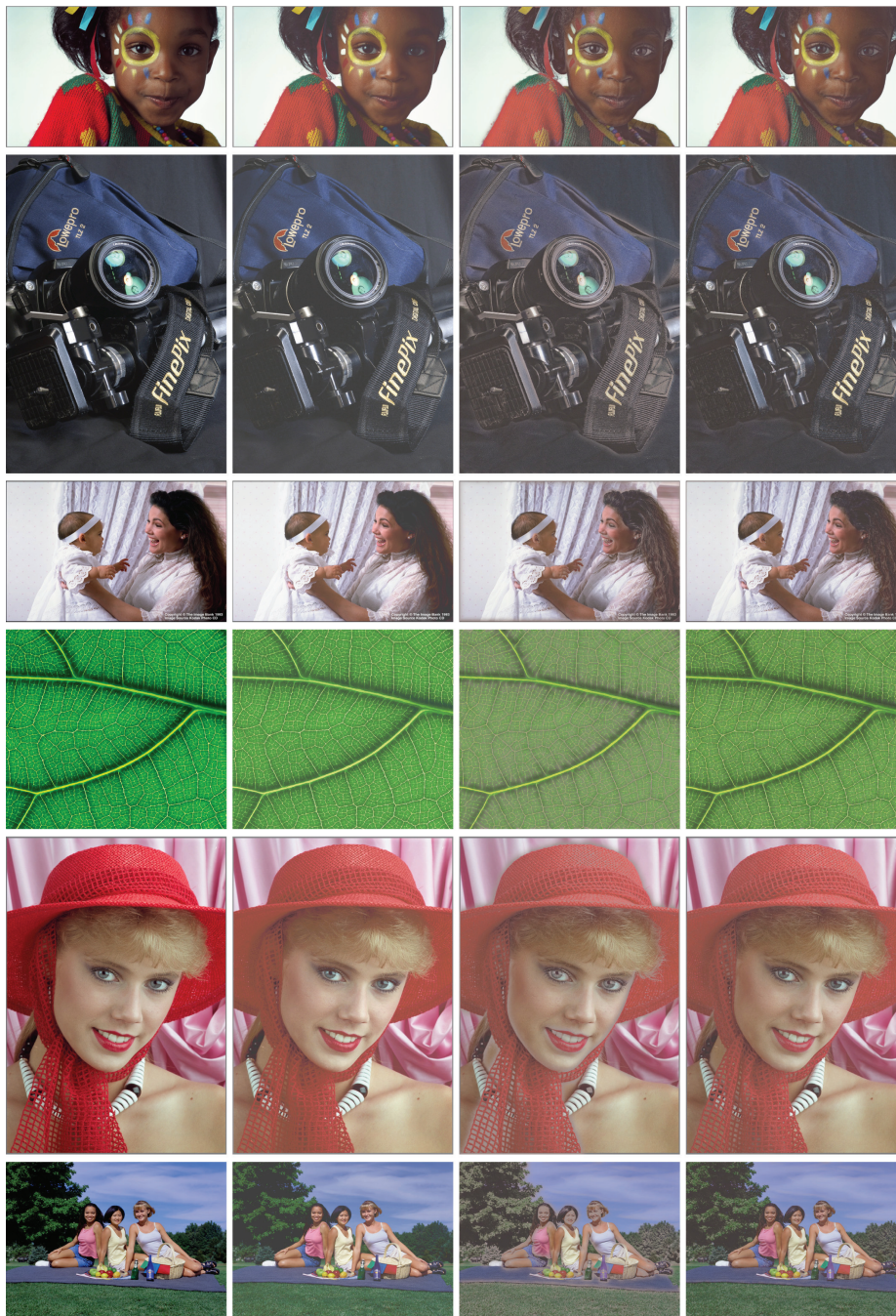


Fig. 2. A set of gamut mapped images. From left to right: original, SGCK [21], variational GMA [16], and proposed solution with $\kappa = 2$, $\alpha_{min} = 0.8$ and $N_{it} = 100$.

A numerical solution to this problem was found by solving the corresponding Euler–Lagrange equation,

$$\nabla^2(\mathbf{p}_s(x, y) - \mathbf{p}(x, y)) = 0 \quad (5)$$

subject to the same constraint using a finite difference method with Jacobi iteration with homogeneous Neumann boundary conditions to ensure zero derivative at the image boundary. The problem was reduced from three to one dimension using the constraint (3) and least squares optimisation.

One of the problems with this approach – a common problem with diffusion based image processing – was the creation of halos near strong edges. In this paper we overcome this problem by employing a non-linear diffusion algorithm. Specifically, we propose to exchange the simple diffusion equation with an anisotropic diffusion equation inspired by Perona and Malik [17]:

$$\nabla \cdot (D(x, y)\nabla(\mathbf{p}_s(x, y) - \mathbf{p}(x, y))) = 0, \quad (6)$$

where the diffusion coefficient is a function of the image properties, $D(x, y) = D(\mathbf{p}(x, y), \nabla\mathbf{p}(x, y))$. In reference [16], we transformed the vectorial diffusion equation to a scalar equation for α_s using the constraint (3) and least squares optimisation. With the more advanced diffusion coefficient of Perona and Malik, this is not a straight forward procedure. Thus, in order to simplify the problem further, we solve this equation for the grayscale versions of the original, p , and gamut mapped images, p_s , where the grayscale image is obtained by taking the second norm of the colour vectors. The final colour gamut mapped image can be obtained based on the assumption that it is a convex linear combination of the original image and the neutral gray color at any pixel position, Equation (3).

The diffusion coefficient is chosen in accordance with Perona and Malik:

$$D(x, y) = \frac{1}{1 + (|\nabla p(x, y)|/\kappa)^2}, \quad (7)$$

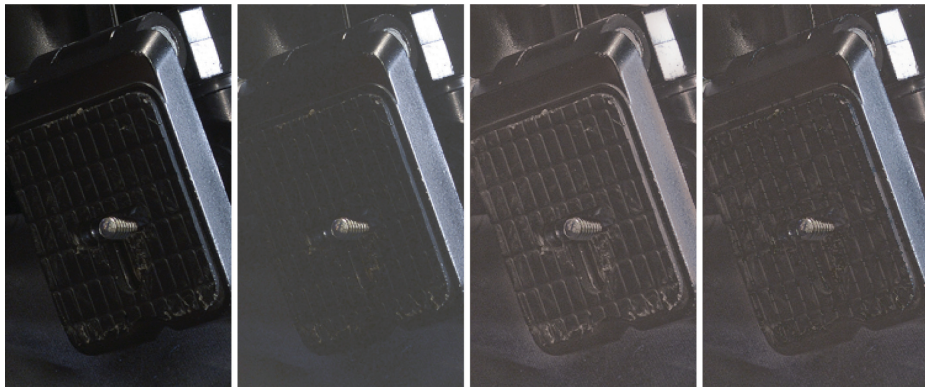


Fig. 3. A close-up of one of the rows in Figure 2: original, SGCK [21], variational GMA [16], and proposed solution with $\kappa = 2$, $\alpha_{min} = 0.8$ and $N_{it} = 100$.

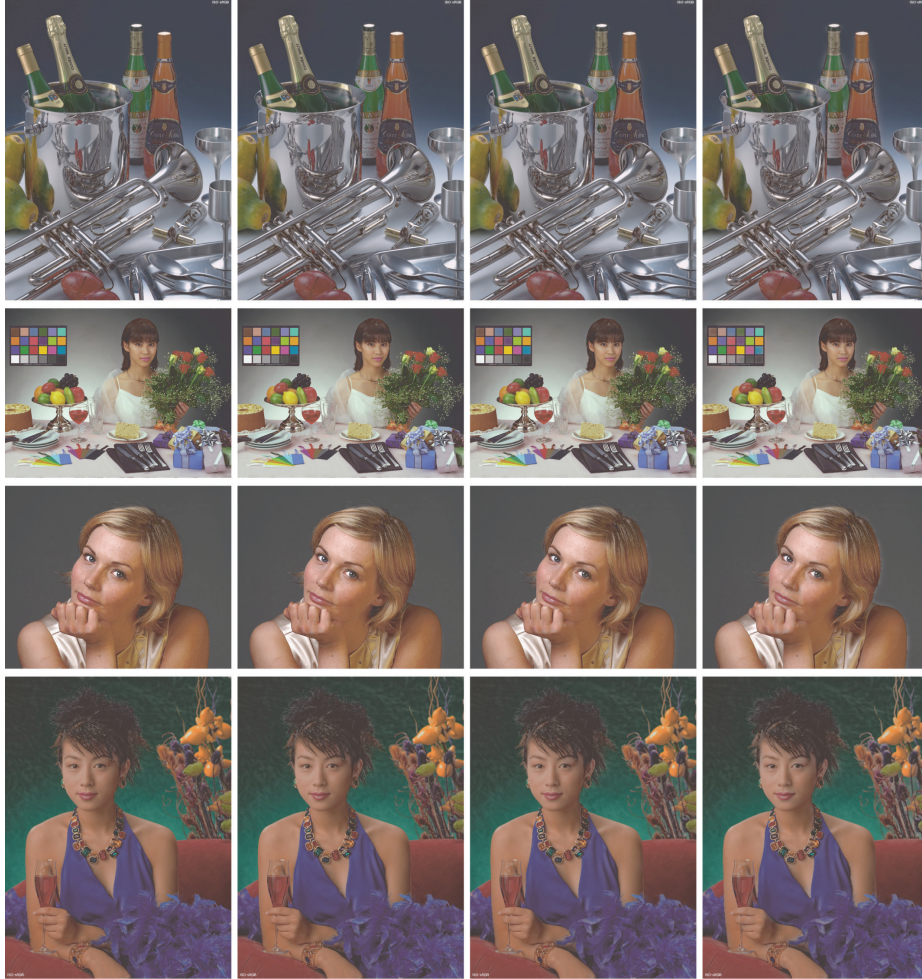


Fig. 4. Effect of changing the κ value in the algorithm. The κ values are, from left to right, 0.5, 1, 2, and 5. For all of them, $N_{it} = 100$ and $\alpha_{min} = 0.8$.

where the constant is chosen such that $\max_{x,y} (|\nabla p(x,y)|/\kappa)^2 \gg 1$ in order to stop the diffusion over sharp edges. This results in the following scalar equation:

$$\nabla \cdot \left(\frac{\nabla(p_s(x,y) - p(x,y))}{1 + (|\nabla p(x,y)|/\kappa)^2} \right) = 0. \quad (8)$$

This equation is discretised using the finite difference method with homogeneous boundary conditions, and iterated using the steepest decent method with N_{it} iterations, in analogy with [16]. This gives a solution for $\alpha_s(x,y)$ which can be inserted in Equation (3). However, like in Reference [16], the resulting images tend to preserve details at the cost of losing saturation, since no fidelity term is

in used (see, e.g., [20]). Therefore, in analogy with conventional gamut mapping algorithms [1], we set a limitation to how small α_s can be. That is, the constraint in Equation (4) is replaced with $\alpha_s(x, y) \in [\alpha_{min}, 1]$ for a suitable choice of α_{min} .

3 Results and Discussion

Figure 2 shows the result of gamut mapping some images with the algorithm proposed in the previous section. For comparison, the images were also mapped using pure clipping, the SCGK algorithm [21], and the previously proposed variational scheme [16]. We see that the proposed algorithm preserves more of the details present in the original image than the SCGK algorithm without introducing the artificial looking edges and haloes produced by the previously proposed method [16]. It should be noted, though, that the number of iterations for the previously proposed algorithm was set too high ($N_{it} = 100$) in order to be directly comparable to the current proposal. This is much higher than suggested in [16], and exaggerates the problems of that algorithm. A close-up of one of the images is shown in Figure 3.

3.1 The κ Parameter

Figure 4 illustrates the behaviour of the algorithm for various choices of κ . The κ parameter controls the anisotropy of the diffusion. The smaller the value of κ , the more sensitive the diffusion coefficient (7) is to the local variations in the image. Intuitively, this should mean that the κ parameter should be set as small as possible. However, the resulting images become unnaturally sharp for too small κ values. Further, for very low κ values, the algorithm becomes too sensitive to noise in the image. Finally, smaller κ increases the risk of numerical instabilities. A close-up is shown in Figure 5.

On the other side, if κ is set to a large value, the diffusion coefficient will be insensitive to edges. This results in diffusion across the edges in the image, with



Fig. 5. A close-up of one of the rows in Figure 4. The κ values are, from left to right, 0.5, 1, 2, and 5. For all of them, $N_{it} = 100$ and $\alpha_{min} = 0.8$.



Fig. 6. The effect of the α_{min} parameter. The values are, from left to right: 0.7, 0.8 and 0.9. For all of them, $N_{it} = 100$ and $\kappa = 2$.



Fig. 7. Close-ups of one row in Figure 6. The α_{min} parameter values are, from left to right: 0.7, 0.8 and 0.9. For all of them, $N_{it} = 100$ and $\kappa = 2$.

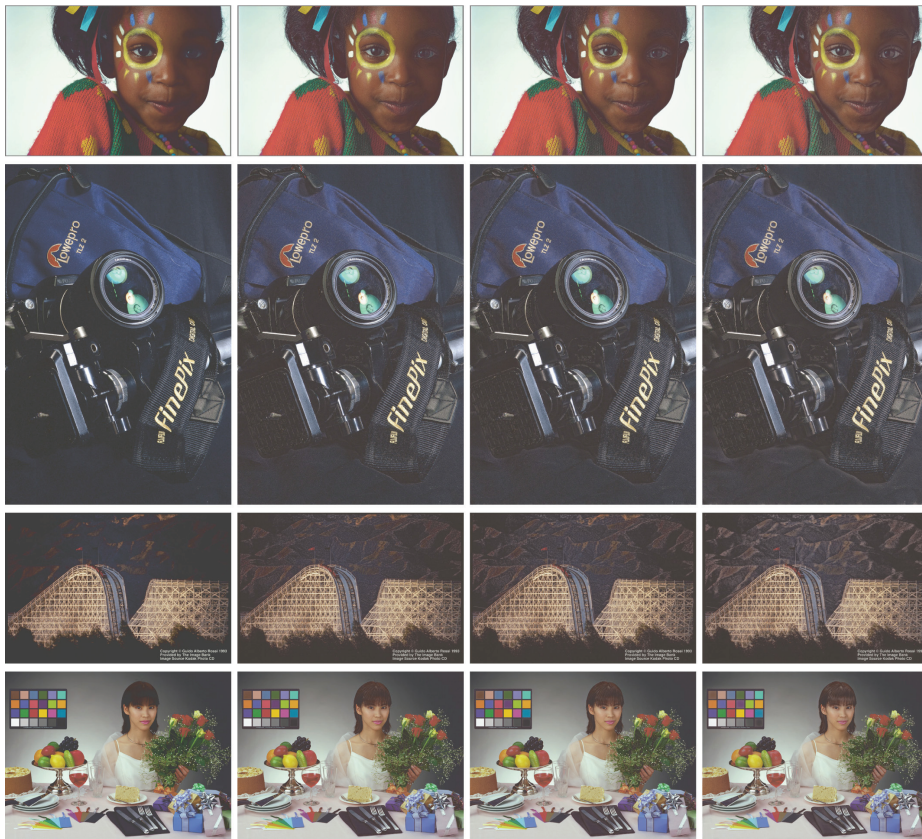


Fig. 8. The effect of the N_{it} parameter. The values are, from left to right: 0, 10, 20, and 100. For all of them, $\alpha_{min} = 0.8$ and $\kappa = 2$.

halo artefacts as the consequence. If $\alpha_{min} = 0$, the algorithm reduces to that proposed in Reference [16] in the limit $\kappa \rightarrow \infty$. In our experiments, the best performance is found for values in the range $\kappa \in [1, 2]$.

3.2 The α_{min} Parameter

The α_{min} parameter limits the amount of compression. Thus, similar to the SGCK algorithm [21], the parameter is a tradeoff between preservation of details and preservation of saturation. An $\alpha_{min} = 0$, represents no preservation of saturation while for an, $\alpha_{min} = 1$, the algorithm reduces to clipping. The most convincing results are obtained by choosing values around 0.8, cf. Figure 6 with close-ups in Figure 7.

3.3 The Number of Iterations

Figure 8 shows how the result of our algorithm develops with the number of iterations. Due to the anisotropic diffusion, the algorithm converges slower than the one proposed in Reference [16]. After 100 iterations, the result is quite stable for most images. The most noticeable changes are, however, observed in the first iterations. Thus, for practical implementations, when speed is called for, 10–20 iterations should be sufficient. Already after 10 iterations, the result resemble that presented in [4], which is, according to Dugay et al. [22] a state-of-the-art algorithm. Thus, the algorithm is very fast, the complexity of each iteration being $O(N)$ for an image with N pixels.

4 Conclusion

The efficient variational spatial colour gamut mapping algorithm suggested in Reference [16] has been refined by the introduction of anisotropic diffusion. As a consequence, the problem of halo creation is strongly reduced, or even totally removed. The problem of too strong desaturation has been solved by introducing a limit to how much the colours can be compressed as an additional constraint in the equations.

To reduce the complexity of the algorithm, we performed the calculations of the gradient and the resulting compression map on the luminance image only. This simplification holds for all colour regions apart from those known as isoluminance, i.e. neighboring colours that have the same luminance values. Thus the algorithm will not recover details that are lost in the conversion between the colour gamut mapped image and its greyscale version.

The resulting algorithm shows promising results for a broad range of images for the same choice of parameters, $\alpha_s = 0.8$ and $\kappa = 2$.

References

1. Morovič, J., Ronnier Luo, M.: The fundamentals of gamut mapping: A survey. *Journal of Imaging Science and Technology* 45(3), 283–290 (2001)
2. Bala, R., de Queiroz, R., Eschbach, R., Wu, W.: Gamut mapping to preserve spatial luminance variations. *Journal of Imaging Science and Technology* 45(5), 436–443 (2001)

3. Kimmel, R., Shaked, D., Elad, M., Sobel, I.: Space-dependent color gamut mapping: A variational approach. *IEEE Trans. Image Proc.* 14(6), 796–803 (2005)
4. Farup, I., Gatta, C., Rizzi, A.: A multiscale framework for spatial gamut mapping. *IEEE Trans. Image Process.* 16(10) (2007), doi:10.1109/TIP.2007.904946.
5. Giesen, J., Schubert, E., Simon, K., Zolliker, P.: Image-dependent gamut mapping as optimization problem. *IEEE Trans. Image Process.* 16(10), 2401–2410 (2007)
6. Eschbach, R.: Image reproduction: An oxymoron? *Colour: Design & Creativity* 3(3), 1–6 (2008)
7. Land, E.H., McCann, J.J.: Lightness and retinex theory. *Journal of the Optical Society of America* 61(1), 1–11 (1971)
8. McCann, J.J.: A spatial colour gamut calculation to optimise colour appearance. In: MacDonald, L.W., Luo, M.R. (eds.) *Colour Image Science*, pp. 213–233. John Wiley & Sons Ltd., Chichester (2002)
9. Meyer, J., Barth, B.: Color gamut matching for hard copy. *SID Digest*, 86–89 (1989)
10. Morovič, J., Wang, Y.: 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*, Scottsdale, Arizona, pp. 282–287 (2003)
11. Eschbach, R., Bala, R., de Queiroz, R.: Simple spatial processing for color mappings. *Journal of Electronic Imaging* 13(1), 120–125 (2004)
12. Zolliker, P., Simon, K.: Retaining local image information in gamut mapping algorithms. *IEEE Trans. Image Proc.* 16(3), 664–672 (2007)
13. Nakauchi, S., Hatanaka, S., Usui, S.: Color gamut mapping based on a perceptual image difference measure. *Color Research and Application* 24(4), 280–291 (1999)
14. Bakke, A.M., Farup, I., Hardeberg, J.Y.: Evaluation of algorithms for the determination of color gamut boundaries. *Journal of Imaging Science and Technology* 54(5), 050502–050511 (2010)
15. Balasubramanian, R., Dalal, E.: A method for quantifying the color gamut of an output device. In: *Color Imaging: Device-Independent Color, Color Hard Copy, and Graphic Arts II*, San Jose, CA. Proc. SPIE, vol. 3018 (January 1997)
16. Alsam, A., Farup, I.: Colour gamut mapping as a constrained variational problem. In: Salberg, A.-B., Hardeberg, J.Y., Jenssen, R. (eds.) *SCIA 2009*. LNCS, vol. 5575, pp. 109–118. Springer, Heidelberg (2009)
17. Perona, P., Malik, J.: Scale-space and edge detection using anisotropic diffusion. *IEEE Trans. Image Proc.* 12(7), 629–639 (1990)
18. Battiato, S., Gallo, G., Stanco, F.: Smart interpolation by anisotropic diffusion. In: *International Conference on Image Analysis and Processing*, vol. 0, p. 572 (2003)
19. Gali, I., Weickert, J., Welk, M., Bruhn, A., Belyaev, A., Seide, H.-P.: Image compression with anisotropic diffusion. *Journal of Mathematical Imaging and Vision* 31(255–269) (2008)
20. Lin, Z., Islam, S.: An adaptive edge-preserving variational framework for color image regularization. In: *IEEE International Conference on Image Processing, ICIP 2005*, pp. 101–104 (2005)
21. CIE Technical Committee 8-03. Guidelines for the evaluation of gamut mapping algorithms. Technical Report 156, CIE (2003)
22. Dugay, F., Farup, I., Hardeberg, J.Y.: Perceptual evaluation of color gamut mapping algorithms. *Color Research and Application* 33(6), 470–476 (2008)