Fast Hue and Range preserving Histogram Specification. Theory and New Algorithms for Color Image Enhancement

Mila Nikolova, Gabriele Steidl

▶ **To cite this version:**

Mila Nikolova, Gabriele Steidl. Fast Hue and Range preserving Histogram Specification. Theory and New Algorithms for Color Image Enhancement. 2013. hal-00871144v1

HAL Id: hal-00871144

https://hal.science/hal-00871144v1

Preprint submitted on 13 Oct 2013 (v1), last revised 6 Nov 2014 (v4)

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Fast Hue and Range Preserving Histogram Specification: Theory and New Algorithms for Color Image Enhancement

Mila Nikolova, and Gabriele Steidl

Abstract

Color image enhancement is a complex and challenging task in digital imaging with abundant applications. Preserving the hue (the dominant color ingredient) of the input image is crucial in a wide range of situations. We propose a simple image enhancement methodology where the hue is conserved and the range (the gamut) of the R, G, B channels is optimally preserved. In our setup, the intensity channel of the input image is transformed into an intensity image whose histogram matches a specified, well-behaved target histogram, using the fast strict ordering algorithm [1]. We derive a new color assignment approach that preserves the hue and the range in an optimal way. We analyze our algorithms in terms of their chromaticity improvement and compare them with the unique and quite popular histogram based hue and range preserving algorithm of Naik and Murthy [2]. Numerical tests confirm our theoretical results and show that our algorithms performs much better than the Naik and Murthy algorithm. In spite of its simplicity, it gives quite often better results than the state-of-the art automatic color enhancement algorithm (ACE) [15].

Index terms – color image enhancement, scaling and shifting methods, hue preserving, gamut problem, exact histogram specification, color perception.

I. Introduction

This paper assists to the tremendous progress in digital color imaging and display technology. In spite of the important amount of research, color perception and color appearance are still open problems. The demand for fast efficient algorithms improving the color content of digital images has increased dramatically. The applications of color image improvement are abundant. They concern for example digital cameras and mobile phone cameras, medical imaging, video, post-production industry, restoration of old pictures and movies.

Typically, color images are stored and viewed using three components (channels): red (R), green (G) and blue (B). In this paper we aim to design color image enhancement methods in the RGB space sharing three important features, namely *hue and range (gamut) preservation* and *low computational complexity*. The hue of an image provides the dominant color ingredient that one really perceives, e.g., red, orange, magenta, yellow and so on. The hue has the nice property of being invariant under changes of direction and intensity of the incident light [3]. Thus, by preserving the hue and enhancing the lightness, the perceived colorfulness of the obtained image can be improved according to the "Hunt effect". Hence *preserving the hue* of an input image appears to be crucial. An example where the hue is modified is shown in Fig. 1, middle. *The range/gamut*

Fig. 1. Histogram equalization (HE). Left: Original image "onion.png" (Matlab IPT image credits notice). Middle: HE to each color channel independently – the colors are disturbed. Right: Enhancement in three steps following [4]: RGB to HSI transform, HE of the intensity channel, then HSI to RGB transform. Here 36.1 % of the pixels have values in $(255, 443.5]$; they are clipped back on $[0, 255]$ which yields too many yellow pixels.

preservation is only seldom addressed in papers on image enhancement. Each color channel in a digital image can only take a limited number, say L, of integer values, e.g., $L = 256$ for 8-bit coding. If the enhancement method produces larger or smaller values these were usually clipped back to the boundary of $[0, L-1]$ which creates an unrealistic change in the hue, see Fig. 1,

M. Nikolova is with CMLA – CNRS, ENS Cachan, 61 av. President Wilson, 94235 Cachan Cedex, France (email: nikolova@cmla.ens-cachan.fr).
Her work was supported partly by the "FMJH Program Gaspard Monge in optimization and operation research", and by the support to this program from EDF.
G. Steidl is with the Dept. of Mathematics, University of Kaiserslautern, 67663 Kaiserslautern, Germany (email: steidl@mathematik.uni-kl.de).

right. Finally, a *low computational complexity* is a general desire for algorithms which is particularly important when dealing with "mega-pixel" images taken by commercial cameras, resources in hardware implementations and extensions to video.

We do not look for fully automatic image enhancement algorithms. The dream of automatic color image enhancement faces (at least) two major limits:

- We are living in a "global" world where images cross the earth in seconds. But the chemistry forming the color receptors in human eye vary among the populations [5]. Therefore, color appearance that is fine in Germany is bad in China, and vice-versa. Have a look at Fig. 5 and decide which image you prefer. For other examples, one can also check "color meaning" on internet. Further, subjective criteria are of paramount importance [4].
- Image improvement/enhancement is always driven by an application: the user needs to assess with greater clarity *specific* visual information determined by his/her purpose [4]. For "Elle", images must be piecewise smooth and well enhanced; no skin problems should be suspected, whereas these are the most important facts for a dermatologist. If pictures have to be published in "International Herald Tribune", rough details are important; a criminologist does not care of the scene appearance but wish to have details in dark areas; see, e.g., Fig. 2.

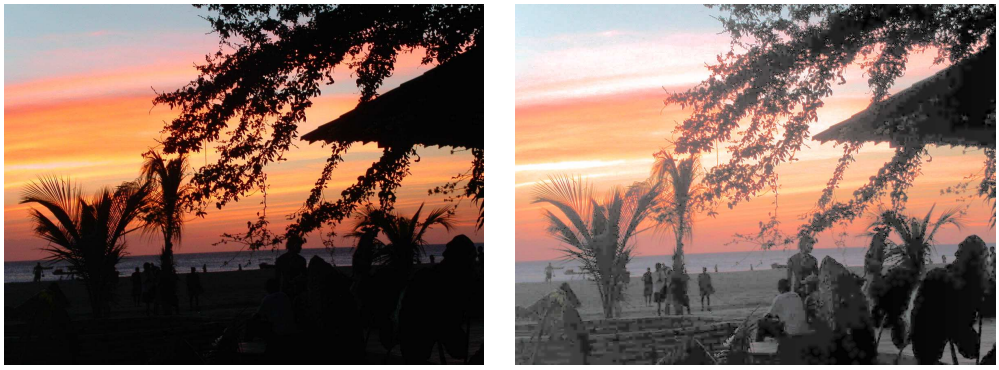

Fig. 2. Image enhancement is application dependent. Left: Original image which is rather dark but gives a nice impression of the sunset. Right: Enhanced image by HE of the intensity channel and generalization to RGB channels by Alg. V presented in this paper. The image looks artificial, but the persons in the dark can be much more clearly seen.

Here we focus on histogram based methods. The selection of a suitable target histogram enables the user's needs to be satisfied. Moreover we wish to conceive quite fast algorithms. In order to achieve our goals, we propose fast algorithms composed of two stages:

(a) the intensity channel of the input RGB image is matched to a specified histogram which gives us the target intensity image;
(b) new (optimal) strategies are developed to compute the RGB color values corresponding to the target intensity image, where the hue and gamut constraints are satisfied.

We comment these stages next.

Stage (a). Exact histogram specification, also known as histogram matching of single-valued (gray-valued) images aims to transform an input image to an output image which exactly fits a prescribed target histogram. Histogram equalization (HE) is a particular case of HS where the target histogram is uniform. Usually HE leads to unnatural structures in images and should not be the method of choice. The construction of a "good" target histogram is a topic on its own which is *not* addressed in this paper. A common approach, presented in [6], is to take the histogram from a well exposed *example* image. Once the target histogram is chosen one needs a reliable histogram matching method. Since a digital image has in general much less pairwise different gray values than number of pixels, HS is an ill posed problem. The clue to ensuring exact HS is to obtain a *meaningful total strict ordering* of all pixels in the input digital image. The role of exact ordering is illustrated in Fig. 4. Currently, the method in [1] which is based on the specialized variational approach in [7] provides the best pixel ordering algorithm in terms of quality, speed and memory needs.

Stage (b). The *extension of histogram methods to color images* is a quite complex task. The histogram of a gray-value image is 1-D while the histogram of a color image is 3-D which gives rise to an under-determined problem. For instance, one can easily check that applying HE to each color channel independently changes the color appearance (the hue) of the image, see Fig. 1 middle. Also, color images that do not respect the range constraints are often produced, see Fig. 1 right. As input histogram we will use the histogram of the image intensity. Fig. 3 shows results of color image enhancement methods which differ only by their input histogram definitions. As central result of this paper we propose a general and optimal hue and range preserving color assignment methodology.

Related work. Since the inaugural paper [9] providing a variational formulation of histogram modification methods, *variational and PDE-based approaches* for color image enhancement were recently proposed; see, e.g., [10], [11], [12],

(a) Original (b) HS of luma (c) HS of green

Fig. 3. Top: Original image (a) and enhanced images (b,c) by Alg. IV in this paper with target histogram due to [8] with parameter $\mu = 0.9$ in (32). The enhancement procedure differs only by the choice of the *input* histogram. The histogram of the luma, see Remark 2, is used in (b) and the histogram of the green channel only in (c). The result in (c) is surprisingly good. In contrast, replacing the green channel by the red or blue one results in hard defects (not shown here). Bottom: The zoom into the images shows that using the histogram of the green channel introduces some artefacts; the left lower part of image (c) is too red and the upper red fan shows some unpleasant pixels.

[13]. These methods provide flexible tools to incorporate various knowledge on human perceptual phenomena. For instance, prompt account can be taken on the fact that contrast is perceived locally while overall appearance is global. Usually they involve user-defined parameters and the numerical cost is heavy compared to histogram-based methods. An automatic color equalization (ACE) has been proposed in [14] where functionals and parameters were fixed by taking close account for human vision facts and by tests. This algorithm was reformulated in a variational framework and successfully extended to image enhancement in [10]. A fast automatic implementation of the method in [10] was developed in [15].

Next we summarize the main approaches via *histogram modification* of color images following a chronological order. Since the suitably normalized histogram of an image is also the *empirical* probability distribution of its pixel values, a statistical vocabulary is used in many papers. In [16] a 3-D color histogram in the RGB color space was proposed for HE; the resultant images present an excessive brightness for bright pixels, see [17]. A method that preserves both the hue and the range (gamut) constraints was inaugurated by Naik and Murthy in [2]. Even though this article did not show color image applications, this is a state-of-the-art method applied in many articles; see, e.g., [18], [17], [19]. We shall analyze in detail this method in the main body of our paper. As to the choice of the color space, some methods operate directly in the true RGB while others operate in transform color spaces, e.g., LHS, HSI, YIQ, HSV, etc. When processing is done in a transform color space, coming back to the original RGB space typically generates a gamut problem, as cautioned in [2]. Beyond the additional numerical cost, a post-processing in RGB is then needed (often realized using [2]). Gray-value grouping was tentatively extended to color HE in [20]. In [21], a new definition of the histogram of a color image was introduced whose cumulative distribution function (cdf) is the product of the marginal cdf's of each color channel. Then the color values are increased / decreased by the same amount iteratively. This work was refined by the author in a later article [22]. Another approach, developed in [18], is to work in the HSI space where the hue and the saturation are equalized and then processed using probability smoothing. All pixels in the RGB space that present gamut problem are corrected using [2]. A generic brightness preserving dynamic histogram equalization scheme, composed of five steps, was proposed in [23]. This scheme was applied to color images in several ways, including transforms into other color spaces. The work in [17] demonstrates that the methods in [18], [21], [16] based on higher dimensional histogram definition, increase the brightness of the image and cannot fit the prescribed uniform histogram. The main conclusion is that only the 1-D histogram of the intensity channel can be considered for equalization. The new color values are then computed using the algorithm in [2]. The latter method was recently improved in[24]. In order to avoid the excessive contrast enhancement due to histogram equalization, Arici et al. [8] propose several techniques which will be also applied in this paper. There are also many histogram based techniques where the enhancement function is an S-type, or power, or logarithmic transform; see e.g. [25], [26], [27], [28], [19], among others. In particular, the approach in [26] is based on mathematical models for color perception and is automatic. Unfortunately, there is no clear algorithm nor tests on color images.

Contributions. We propose a general affine model for fast hue and range preserving image enhancement in the RGB space. The well-known scaling and shifting methods, *always considered for HE*, appear as quite particular cases of our model. A flexibility of our approach is that we use various target histograms for the intensity channel in order to satisfy particular needs and to avoid the drawbacks of HE. We establish necessary and sufficient corrections of the main affine model that ensure an *optimal* range preservation. These findings give rise to our Algorithm III where each enhanced pixel is computed by a simple explicit formula. Range preservation has been seldom taken into account in imaging methods. An exception is the scaling algorithm proposed in by Naik and Murthy in [2] in the context of HE and first tested on color images in [17]. Two simple but important instances of Algorithm III are the Multiplicative Algorithm IV and the Additive Algorithm V. We show how the outcome of Algorithm III can be faithfully approximated as a convex combination of the images obtained using Algorithm IV and Algorithm V, which can be quite useful in practice.

The enhancement performances of our algorithms and the algorithm of Naik and Murthy [2] are analyzed in terms of the chromaticity improvement they provide. In all cases, our algorithms clearly outperform the algorithm in [2] which has been used in many papers to solve the gamut problem. Our color assignment algorithms are simple and fast. The Additive Algorithm V is the fastest because for most of the pixels only additions are done. The other algorithms have the same complexity as the algorithm in [2].

All numerical tests that we have performed corroborate our theoretical results. Last (but not the least) our algorithms give quite often better visual results than the ACE [15] which is considered as the state-of-the-art algorithm.

Outline. In Section II we recall basic facts on the HSI color space and hue plus range preserving color assignment. Also, we sketch our histogram specification method based on a strict ordering algorithm proposed in our report [1]. Section III extends our histogram specification method to color images taking both hue and range preservation into account. Section IV evaluates our algorithms analytically in terms of saturation as well as qualitativelyand compares them with the Naik-Murthy algorithm [2]. Section V presents numerical results for color image enhancement with various target histograms and in comparison to the ACE algorithm. Finally, we draw conclusions and points for future work in Section VI. The proofs of all statements are given in the Appendix.

II. Preliminaries

Here we recall the HSI color model, discuss the hue and range preservation task (including the Naik-Murthy algorithm [2]) and briefly describe the exact HS method proposed in [1] which is part of our color image enhancement algorithms.

A. HSI Color Space

Usually there is a high correlation between the RGB components and designing colors based on the RGB cube is quite hard. The HSI (hue (H), saturation (S), intensity (I)) color model and its variants as HSL, HSB and HSV are more intuitive because the intensity information is separated from the chromatic components (hue and saturation).

Let $w = (w_r, w_g, w_b)$, be an RGB image where w_r, w_g and w_b are its red, green and blue channels, respectively, of size $M \times N$. We reorder each color channel columnwise into a vector of size $n := MN$ and address the pixels by the index set $\mathbb{I}_n := \{1, \cdots, n\}$. Thus $w_c \in \mathbb{R}^n$ for any $c \in \{r, g, b\}$. Given an RGB image w, its *intensity* is defined by

$$f(w) = \frac{1}{3}(w_r + w_g + w_b), \tag{1}$$

where this and the following expressions are meant componentwise, i.e. for any $i \in \mathbb{I}_n$. Because of the optics of the human eye, the intensity channel has the highest spatial resolution [5].

The *hue* of w, expressed in degrees, is defined by, see e.g.,[29], $H(w) = 0$ if $w_r = w_g = w_b$ and otherwise

$$H(w) := \begin{cases} \theta & \text{if } w_b \le w_g, \\ 360 - \theta & \text{if } w_b > w_g, \end{cases} \tag{2}$$

where

$$\theta := \arccos \frac{\frac{1}{2}((w_r - w_g) + (w_r - w_b))}{((w_r - w_g)^2 + (w_r - w_b)(w_g - w_b))^{\frac{1}{2}}}.$$

Note that the denominator of θ can be rewritten as $(\frac{1}{2}((w_r - w_g)^2 + (w_r - w_b)^2 + (w_g - w_b)^2))^{\frac{1}{2}}$.

Finally, the *saturation* of an image w reads as

$$S(w) := \begin{cases} 1 - \dfrac{\min\{w_r, w_g, w_b\}}{f(w)} & \text{if } f(w) > 0, \\ 0 & \text{if } f(w) = 0\ . \end{cases} \tag{3}$$

B. Hue and Range Preservation

Range preservation is a hard and mandatory constraint for all digital imaging devices (e.g., screens, printers, an so on). We denote by L the number of digits that can be displayed. Thus, $w_c \in \{0, \cdots, L-1\}$ for any $c \in \{r, g, b\}$. A transformed version $\widehat{w}$ of w can be correctly displayed only if

$$\widehat{w}_c[i] \in [0, L-1] \quad \forall\, i \in \mathbb{I}_n \quad \forall\, c \in \{r, g, b\}\ . \tag{4}$$

Otherwise, the obtained images are modified according to the visualization device which is quite a random ad-hoc option. For an illustration, see Fig. 1 right. We recall that the gamut problem is not considered in many references on imaging.

From (2) it is easy to see that the hue of the modified image $\widehat{w}$ is preserved if the RGB channels at each pixel are transformed by the same *affine transform*

$$\widehat{w}_c[i] = a[i]w_c[i] + b[i], \quad c \in \{r, g, b\}, \tag{5}$$

where the constants $a[i]$ and $b[i]$ have to be chosen for any $i \in \mathbb{I}_n$. It is also clear that affine transforms are not the unique family of transforms that preserve the hue; finding another family of transforms, wide enough, is another problem.

For $a[i] = 0$, the transform (5) amounts to an *additive transform*, usually called *shifting*. For $b[i] = 0$, it is a *linear/multiplicative* transform known as *scaling*. Both scaling and shifting have been introduced in [30], [31]. In general, the result of (5) does not fulfill the range constraint (4).

The gamut problem was examined by Naik and Murthy in [2] in the scaling case $b[i] = 0$ and $a[i] := \widehat{f}[i]/f[i]$, where $\widehat{f}$ is a target intensity. It frequently appears that $\widehat{f}[i]/f[i] > 1$ in which case the range constraint (4) *might* not be guarantied. In such case the authors propose to avoid the *potential* problem by switching from the RGB color space to the CMY (Cyan $= L-1-$R, Magenta $= L-1-$G, Yellow $= L-1-$B) space and then to transform back into RGB. This correction step reads for all $c \in \{r, g, b\}$ as

$$\widehat{w}_c[i] = L - 1 - \frac{L-1-\widehat{f}[i]}{L-1-f[i]}(L-1-w_c[i]) \quad \text{if} \quad \frac{\widehat{f}[i]}{f[i]} > 1.$$

This formula is equivalent to $\widehat{w}_c[i] = \frac{L-1-\widehat{f}[i]}{L-1-f[i]}(w_c[i] - f[i]) + \widehat{f}[i]$, which is easier to use in what follows. So the algorithm in [2] has the following equivalent formulation:

Algorithm I: (**Naik and Murthy [2]**)

1) Compute the intensity f of w and the target intensity $\widehat{f}$.
2) For $i \in \mathbb{I}_n$ compute

$$\begin{aligned} &\text{(i)} \qquad \widehat{w}_c[i] := \frac{\widehat{f}[i]}{f[i]}\, w_c[i] && \text{if} \quad \frac{\widehat{f}[i]}{f[i]} \le 1, \\ &\text{(ii)} \qquad \widehat{w}_c[i] := \frac{L-1-\widehat{f}[i]}{L-1-f[i]}(w_c[i] - f[i]) + \widehat{f}[i] && \text{if} \quad \frac{\widehat{f}[i]}{f[i]} > 1. \end{aligned} \tag{6}$$

This algorithm was applied in many papers as, e.g., in [18], [17], [19] to avoid gamut problem. The threshold set to 1 is only a "security" value. Observe that in step (i), the intensity cannot be increased and the saturation remains constant. The intensity of $\widehat{w}[i]$ can be increased only in step (ii) at the price of an important decrease in saturation.

Example 1: For instance, let $w[i] = (10,\ 40,\ 100)$. One has $f[i] = 50$ and $w[i]$ is quite a colorful pixel since its saturation is $S(w[i]) = 0.8$ (i.e. not far from 1). If the target histogram requires that $\widehat{f}[i] = 100$, case (ii) shall be applied which yields $\widehat{w}[i] = (69.8,\ 92.4,\ 137.8)$; the intensity of this pixel is increased as required, but its saturation is only $S(w[i]) = 0.3$. If $\widehat{f}[i] = 230$, case (ii) holds yet again and yields $\widehat{w}[i] = (225.1,\ 228.8,\ 236.1)$; this pixel is almost gray valued since $S(w[i]) = 0.02$ is close to zero.

C. Histogram Specification and Strict Ordering

We consider the single-valued intensity image f of a color image w with RGB values in $\{0, \cdots, L-1\}$. For 8-bit images we have $L = 256$. Then $3f$ has values in $\{0, \cdots, 3(L-1)\}$. We are interested in exact HS to a target histogram $\widehat{h} = (\widehat{h}_1, \ldots, \widehat{h}_L)$ for gray values $\{0, \ldots, L-1\}$, i.e., $\widehat{h}[k] = \sharp\{i \in \mathbb{I}_n \mid \widehat{f}[i] = k-1\}$, $k = 1, \ldots, L$, where $\sharp$ stands for cardinality. If the pixels values of our input image $3f$ could be strictly ordered, exact histogram specification can be easily done by dividing the corresponding ordered list of indices into L groups and assigning gray value 0 to the first $\widehat{h}_1$ pixels, gray value 1 to the second $\widehat{h}_2$ pixels and so on until gray value $L-1$ to the last $\widehat{h}_L$ pixels. Therefore a faithful strict ordering algorithm is of crucial interest. In contrast to the ordering algorithms in [32] and [33], the specialized variational method in [7] provides a much more faithful ordering and requires much less storage. Moreover it can deal with large size images commonly taken by commercial cameras. In [1] we have modified the latter method to become also much faster than all its competitors. The basic idea consists in minimizing a certain smoothed ℓ_1-TV functional by simple *explicit* fixed point iterations with the original image as initialization. After a small number of iterations the "approximate" minimizer has entries which differ (up to very few outliers) pairwise from each other while the ordering of the original pixel values is retained. The ascending ordering of

the pixel in the new image is used as the ordering of the original image. Here is a compact version our ordering algorithm:

Algorithm II: (**Strict ordering by [1] and HS**)

Initialization: $u^{(0)} = 3f$, $(\beta, \alpha) := (0.1, 0.05)$, and target histogram $\widehat{h} = (\widehat{h}_1, \ldots, \widehat{h}_L)$. Choose K (e.g., $K = 6$).

1) For $k = 1, \ldots, K$ compute

$$u^{(k)} = f - \eta^{-1}\big(\beta G^{\mathrm{T}} \eta(G u^{(k-1)})\big).$$

where

$$\eta(t) := \frac{t}{\alpha + |t|} \quad \text{and} \quad \eta^{-1}(y) = \frac{\alpha y}{1 - |y|} \tag{7}$$

and G is a forward difference matrix.

2) Order the values in $\mathbb{I}_n$ according to the corresponding ascending entries of $u^{(K)}$.
3) HS step: divide the obtained ordered list of indices into L groups and assign gray value 0 to the first $\widehat{h}_1$ pixels and so on until gray value $L-1$ to the last $\widehat{h}_L$ pixels. This provides the target intensity $\widehat{f}$.

Comparing with the notation in [1], we have $\eta = \theta'$, where $\theta(t) := |t| - \alpha \log(1 + |t|/\alpha)$. The initialization of the parameters β, α refers to our standard choice used in all our experiments. More precisely, this choice ensures that $y := \beta G^{\mathrm{T}} \eta(G u^{(k-1)})$ fulfills $|y| < 1$ in (7) and that $\|u^{(K)} - 3f\|_\infty < 0.5$, see [34]. In particular this means that if $3f[i] < 3f[j]$ then also $u^{(K)}[i] < u^{(K)}[j]$ so that the pixel ordering of $u^{(K)}$ always respects the pixel ordering of the given integer-valued image $3f$. Moreover $K = 6$ appears to be sufficient to ensure a nearly total strict ordering.

The importance of a meaningful strict ordering of the pixels before applying any HS is illustrated in Fig. 4 in the context of HE using Algorithm I. The Matlab built-in function histeq does not involve strict ordering and the resulting histogram in (d) is not uniform. This entails some bad artifacts emphasized in (b) that are not observed in (c) which is obtained using the strict ordering Algorithm II. The fact that all dark regions are nearly gray valued is due to Algorithm I and will be explained in Subsection IV-B; see also Example 1 and notice that in the case of HE of dark regions we have $f[i] \ll \widehat{f}[i]$.

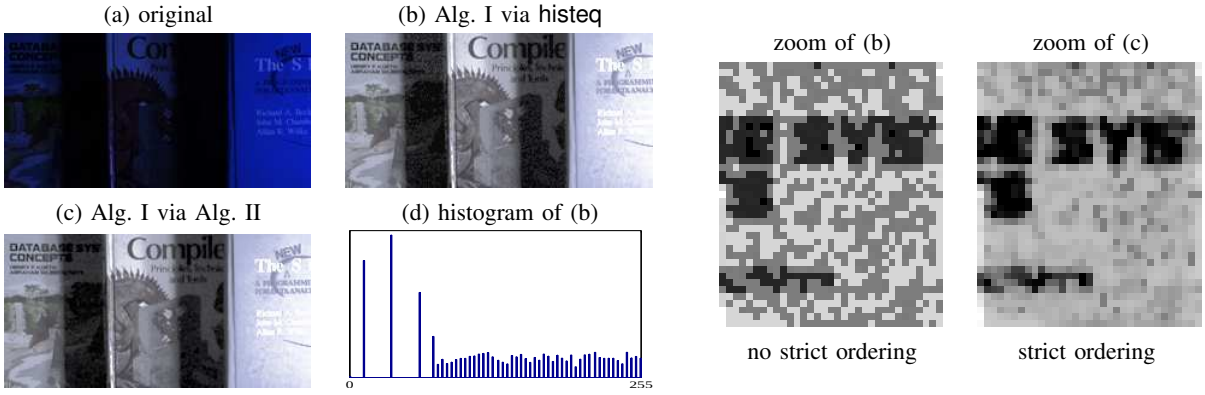

Fig. 4. HE of the dark image in (a) using Algorithm I. In (b), the result is obtained using the Matlab routine histeq. The resulting histogram in (d) is in fact not equalized. In (c), the strict ordering Algorithm II is applied before equalizing the image. The histogram (not shown) is uniform. The zooms show strong artifacts when there is no strict ordering prior to histogram matching.

III. New Affine Histogram Specification Models

In this section we develop our simple affine color enhancement method. First, we compute for a given image w its intensity f defined by (1).

Remark 1: Instead of the intensity f in (1) we could also work with convex combinations

$$f := (\gamma_r w_r + \gamma_g w_g + \gamma_b w_b) \in \mathbb{R}^n, \quad \sum_{i=1}^{3} \gamma_i = 1, \quad \gamma_i \geq 0.$$

The coefficients of γ defining the *luma* are set in proportion to the human perception of each of the red, green and blue colour channels, see [5], [4] and read as $\gamma := (0.299,\ 0.587,\ 0.114)$. These values correspond to the NTSC television standard. Observe that a much higher weight is given to the green channel. This is coherent with our experiment in Fig. 3 where surprisingly good results were achieved by just using the green channel, i.e., $\gamma = (0, 1, 0)$.

Our next goal is to transform w into an image $\widehat{w}$ having the following properties:

(a) Intensity fit: $\widehat{f} = \frac{1}{3}(\widehat{w}_r + \widehat{w}_g + \widehat{w}_b)$.
(b) Hue preservation: the hue of $\widehat{w}$ and w coincide.

(c) Range preservation: $0 \leq \widehat{w}_c \leq L-1$, $c \in \{r,\, g,\, b\}$.

From (2), the hue it is preserved if the RGB channels at each pixel are transformed by the same affine transform

$$\widehat{w}_c[i] = a[i]w_c[i] + b[i], \quad c \in \{r,\, g,\, b\}. \tag{8}$$

Summing up over c in (8) we see that property (a) is fulfilled if and only if

$$\widehat{f}[i] = a[i]f[i] + b[i] \quad \Leftrightarrow \quad b[i] = \widehat{f}[i] - a[i]f[i]. \tag{9}$$

Therefore the affine model (8) obeys (a) if and only if

$$\widehat{w}_c[i] = a[i](w_c[i] - f[i]) + \widehat{f}[i], \quad c \in \{r,\, g,\, b\}. \tag{10}$$

We will use for all $i \in \mathbb{I}_n$ the magnitudes

$$\begin{array}{rcl} M[i] & := & \max\{w_c[i] : c \in \{r,\, g,\, b\}\}, \\ m[i] & := & \min\{w_c[i] : c \in \{r,\, g,\, b\}\}. \end{array} \tag{11}$$

Remark 2: Since f is a convex combination of all w_c, $c \in \{r,\, g,\, b\}$, we have

$$0 \leq m[i] \leq f[i] \leq M[i] \leq L-1\ .$$

It is clear that $M[i] = f[i]$ if and only if $w_r[i] = w_g[i] = w_b[i]$. Similarly, $m[i] = f[i]$ if and only if $w_r[i] = w_g[i] = w_b[i]$. Then (10) entails for $c \in \{r,\, g,\, b\}$ that

$$f[i] \in \big\{\, m[i],\, M[i]\big\} \quad \Rightarrow \quad \widehat{w}_c[i] = \widehat{f}[i]. \tag{12}$$

In these particular cases all requirements (a), (b) and (c) are trivially satisfied. Conversely,

$$f[i] \notin \big\{\, m[i],\, M[i]\big\} \quad \Rightarrow \quad f[i] > 0.$$

In the general case when $f[i] \notin \{m[i], M[i]\}$, one has to fix the relationship between $a[i]$ and $b[i]$.

Two particular instances of (8), mentioned in subsection II-B, are the following:

– *Scaling*: For $b[i] = 0$, one obtains from (9) that model (10) reads as

$$\widehat{w}_c[i] = \frac{\widehat{f}[i]}{f[i]} w_c[i], \quad c \in \{r,\, g,\, b\}. \tag{13}$$

– *Shifting*: For $a[i] = 1$, model (10) becomes

$$\widehat{w}_c[i] = w_c[i] - f[i] + \widehat{f}[i], \quad c \in \{r,\, g,\, b\}.$$

Both models are in general not range preserving, i.e., they do not fulfill (c).

A. Affine Algorithm with Optimal Range Preservation

Our affine model (10) can also be seen as a convex combinations of the shifting and scaling models for some $\lambda \in [0, 1]$:

$$\begin{aligned} \widehat{w}_c[i] &= \lambda\, \frac{\widehat{f}[i]}{f[i]} w_c[i] + (1-\lambda)\,(w_c[i] - f[i] + \widehat{f}[i]) \\ &= \left(\lambda \frac{\widehat{f}[i]}{f[i]} + 1 - \lambda\right)\,(w_c[i] - f[i]) + \widehat{f}[i]. \end{aligned} \tag{14}$$

Clearly, for $\lambda = 1$ we obtain the scaling method and for $\lambda = 0$ the shifting model. The result of our model (14) does in general not fulfill (c). Next we modify this model to obtain *an optimal range preservation*. We say that a pixel $\widehat{w}[i]$ has

- an *upper gamut* problem if $\max\{\widehat{w}_c[i] : c \in \{r,\, g,\, b\}\} > L-1$;
- a *lower gamut* problem if $\min\{\widehat{w}_c[i] : c \in \{r,\, g,\, b\}\} < 0$.

We will treat these gamut problems in an optimal way in the following sense: *When a pixel $\widehat{w}[i]$ computed according to* (14) *has a gamut problem, the* optimal correction *is to replace its value by the* nearest *color value that belongs to the range, and saties (a) and (b).*

Algorithm I corresponds to $\lambda = 1$ in our affine model (14). It never yields a gamut problem at the expense of the threshold set to one in (6); an important drawback is explained in Example 1 and in subsection IV-B. We recall that the gamut problem is not considered in most of the references in imaging.

a) Upper gamut problem in our affine model.: Assume that $\max\{\widehat{w}_c[i] \ : \ c \in \{r,\, g,\, b\}\} > L-1\}$ for some $i \in \mathbb{I}_n$. By (14) and Remark 2 this means that

$$\frac{\lambda \widehat{f}[i] + (1-\lambda) f[i]}{f[i]}\, M[i] + (1-\lambda)(\widehat{f}[i] - f[i]) > L-1. \tag{15}$$

For definiteness, let $k \in \{r,\, g,\, b\}$ be such that $w_k[i] = M[i]$. Then, $\widehat{w}_k[i] > L-1$ and the best correction of this pixels, satisfying (c), is clearly to choose $a[i]$ in (10) so that $\widehat{w}_k[i] = L-1$. Equivalently,

$$L-1 = a[i](M[i] - f[i]) + \widehat{f}[i]. \tag{16}$$

From Remark 2 we know that

$$M[i] - f[i] > 0,$$

so $a[i]$ in (16) is well defined and positive, that is

$$a[i] = \frac{L-1-\widehat{f}[i]}{M[i]-f[i]} > 0.$$

The modified color values of pixel i are then given by

$$\widehat{w}_c[i] = \frac{L-1-\widehat{f}[i]}{M[i]-f[i]}(w_c[i] - f[i]) + \widehat{f}[i], \quad c \in \{r,g,b\}. \tag{17}$$

For the color channel such that $w_k[i] = M[i]$ we have indeed $\widehat{w}_k[i] = L-1$.

Proposition 1: If (15) is fulfilled, then $\widehat{w}_c[i]$ in (17) satisfies

$$0 \le \widehat{w}_c[i] \le L-1, \quad c \in \{r,\, g,\, b\}.$$

Therefore, the color modification in (17) solves the upper gamut problem without yielding a lower gamut problem.

b) Lower gamut problem in our affine model.: First we mention that this problem can obviously not appear for the multiplicative model (13), i.e. for $\lambda = 1$ in (14). Let $\lambda \in [0,1)$. Assume that $\min\{\widehat{w}_c[i] : c \in \{r,\, g,\, b\}\} < 0$ for some $i \in \mathbb{I}_n$. Using the definition of $m[i]$ in (11) this is equivalent to

$$\frac{\lambda \widehat{f}[i] + (1-\lambda) f[i]}{f[i]}\, m[i] + (1-\lambda)(\widehat{f}[i] - f[i]) < 0. \tag{18}$$

Let $k \in \{r,\, g,\, b\}$ be such that $w_k[i] = m[i]$. The presence of lower gamut at i means that the additive scheme has produced $\widehat{w}_k[i] < 0$. The most natural correction using (10) of the color value of this pixel is to set $\widehat{w}_k[i] = 0$. This is equivalent to find $a[i]$ such that

$$0 = a[i]\ (m[i] - f[i]) + \widehat{f}[i]. \tag{19}$$

From Remark 2, if (18) is true, then

$$f[i] - m[i] > 0.$$

Consequently, $a[i]$ in (19) is well defined and reads as

$$a[i] = \frac{\widehat{f}[i]}{f[i]-m[i]}.$$

Therefore the corrected color value at i is given by

$$\widehat{w}_c[i] = \frac{\widehat{f}[i]}{f[i]-m[i]}(w_c[i] - f[i]) + \widehat{f}[i], \quad c \in \{r,g,b\}, \tag{20}$$

where we recall that for the color channel such that $w_k[i] = m[i]$ we obtain $\widehat{w}_k[i] = 0$.

Proposition 2: If (18) is fulfilled, then $\widehat{w}_c[i]$ in (20) satisfies

$$0 \le \widehat{w}_c[i] \le L-1 \quad c \in \{r,\, g,\, b\}.$$

Hence the color correction in (20) solves the lower gamut problem and do not introduce an upper gamut problem.

Using Propositions 1 and 2, the optimal range-preserving approximation of our affine model (14) can be computed in one iteration where all pixels in the input image are modified only once. The algorithm is described below.

Algorithm III: (**Optimal Affine Range-Preserving Color Enhancement**)
1. Compute the intensity f of w and the target intensity $\widehat{f}$ using our fast sorting algorithm.
2. For $i \in \mathbb{I}_n$ compute

$$\begin{aligned} M[i] &:= \max\{w_c[i] : c \in \{r,\, g,\, b\}\} \\ m[i] &:= \min\{w_c[i] : c \in \{r,\, g,\, b\}\} \end{aligned}$$

2.1 If $f[i] \in \{M[i], m[i]\}$, then $\widehat{w}_c[i] = \widehat{f}[i]$, $c \in \{r,\, g,\, b\}$.
2.2 If $f[i] \neq 0$, compute

$$a[i] = \lambda\, \frac{\widehat{f}[i]}{f[i]} + (1-\lambda) \quad \text{and} \quad \begin{cases} G_m^\lambda := a[i](m[i] - f[i]) + \widehat{f}[i], \\ G_M^\lambda := a[i](M[i] - f[i]) + \widehat{f}[i], \end{cases} \tag{21}$$

and for all $c \in \{r,\, g,\, b\}$:

$$\begin{aligned} &\text{(i)} && \widehat{w}_c[i] := a[i]\,(w_c[i] - f[i]) + \widehat{f}[i] && \text{if} \quad G_m^\lambda[i] \geq 0 \ \text{ and } \ G_M^\lambda[i] \leq L-1, \\ &\text{(ii)} && \widehat{w}_c[i] := \frac{L-1-\widehat{f}[i]}{M[i]-f[i]}(w_c[i] - f[i]) + \widehat{f}[i] && \text{if} \quad G_M^\lambda[i] > L-1, \\ &\text{(iii)} && \widehat{w}_c[i] := \frac{\widehat{f}[i]}{f[i]-m[i]}(w_c[i] - f[i]) + \widehat{f}[i] && \text{if} \quad G_m^\lambda[i] < 0. \end{aligned} \tag{22}$$

B. Multiplicative, Additive algorithms and their Combinations

For $\lambda \in \{0, 1\}$ Algorithm III yields two simple range preserving scaling and shifting algorithms called Multiplicative and Additive Algorithm, respectively. Observing that

$$\begin{aligned} G_m^0[i] &= m[i] - f[i] + \widehat{f}[i], \\ G_M^0[i] &= M[i] - f[i] + \widehat{f}[i], \\ G_M^1[i] &= \tfrac{\widehat{f}[i]}{f[i]} M[i] \end{aligned}$$

these algorithms read as follows:

Algorithm IV: (**Multiplicative Color Enhancement**)
1. Compute the intensity f of w and the target intensity $\widehat{f}$ using Algorithm II.
2. For $i \in \mathbb{I}_n$ compute $M[i] := \max\{w_c[i] : c \in \{r,\, g,\, b\}\}$.
2.1 If $f[i] = M[i]$, then $\widehat{w}_c[i] = \widehat{f}[i]$, $c \in \{r,\, g,\, b\}$.
2.2 If $f[i] \neq 0$, compute

$$G_M^1[i] = \frac{\widehat{f}[i]}{f[i]} M[i]$$

and for all $c \in \{r,\, g,\, b\}$:

$$\begin{aligned} &\text{(i)} && \widehat{w}_c[i] := \frac{\widehat{f}[i]}{f[i]}\, w_c[i] && \text{if} \quad G_M^1[i] \leq L-1, \\ &\text{(ii)} && \widehat{w}_c[i] := \frac{L-1-\widehat{f}[i]}{M[i]-f[i]}(w_c[i] - f[i]) + \widehat{f}[i] && \text{if} \quad G_M^1[i] > L-1. \end{aligned}$$

Algorithm V: (**Additive Color Enhancement**)
1. Compute the intensity f of w and the target intensity $\widehat{f}$ using Algorithm II.
2. For $i \in \mathbb{I}_n$ compute

$$\begin{aligned} M[i] &:= \max\{w_c[i] : c \in \{r,\, g,\, b\}\} \\ m[i] &:= \min\{w_c[i] : c \in \{r,\, g,\, b\}\} \end{aligned}$$

2.1 If $f[i] \in \{M[i], m[i]\}$, then $\widehat{w}_c[i] = \widehat{f}[i]$, $c \in \{r,\, g,\, b\}$.
2.2 If $f[i] \neq 0$, compute

$$G_m^0[i] = m[i] - f[i] + \widehat{f}[i] \quad \text{and} \quad G_M^0[i] = M[i] - f[i] + \widehat{f}[i]$$

and for all $c \in \{r,\, g,\, b\}$:

$$\begin{aligned}
&\text{(i)} && \widehat{w}_c[i] := w_c[i] - f[i] + \widehat{f}[i] && \text{if} \quad G_m^0[i] \geq 0 \ \text{ and } \ G_M^0[i] \leq L-1,\\
&\text{(ii)} && \widehat{w}_c[i] := \frac{L-1-\widehat{f}[i]}{M[i]-f[i]}(w_c[i]-f[i]) + \widehat{f}[i] && \text{if} \quad G_M^0[i] > L-1,\\
&\text{(iii)} && \widehat{w}_c[i] := \frac{\widehat{f}[i]}{f[i]-m[i]}(w_c[i]-f[i]) + \widehat{f}[i] && \text{if} \quad G_m^0[i] < 0.
\end{aligned}$$

Instead of applying our Affine Algorithm III for some $\lambda \in [0,1]$, we can compute the images $\widehat{w}^{\,+}$ by the Additive Algorithm V and $\widehat{w}^{\,\times}$ by the Multiplicative Algorithm IV and build their convex combination for the same $\lambda \in [0,1]$:

$$\widetilde{w}_c := \lambda \widehat{w}_c^{\,\times} + (1-\lambda)\widehat{w}_c^{\,+} \quad \forall c \in \{r,\, g,\, b\}. \tag{23}$$

Clearly, $\widetilde{w}_c$ in (23) fulfills again all conditions (a)-(c). Then by sliding λ we can easily move between the two models. An illustration of the influence of λ is seen in Fig. 5.

Next we explain the relationship between Algorithm III and (23). Here the sets

$$\begin{aligned}
\mathcal{U}(\lambda) &:= \{i \in \mathbb{I}_n : G_M^\lambda[i] > L-1\},\\
\mathcal{L}(\lambda) &:= \{i \in \mathbb{I}_n : G_m^\lambda[i] < 0\}
\end{aligned} \tag{24}$$

which correspond to the upper gamut step (ii) and the lower gamut step (iii) in Algorithm III, respectively, play an important role. They obey the property stated next.

Proposition 3: The sets $\mathcal{U}(\lambda)$ and $\mathcal{L}(\lambda)$ defined in (24) fulfill $\mathcal{L}(1) = \varnothing$ and

$$\mathcal{U}(\lambda_1) \subseteq \mathcal{U}(\lambda_2), \quad \mathcal{L}(\lambda_1) \supseteq \mathcal{L}(\lambda_2), \quad 0 \leq \lambda_1 < \lambda_2 \leq 1.$$

In particular, (24) yields

$$\begin{aligned}
\mathcal{U}(1) &= \left\{i \in \mathbb{I}_n : \tfrac{\widehat{f}[i]}{f[i]} M[i] > L-1\right\},\\
\mathcal{U}(0) &= \left\{i \in \mathbb{I}_n : \widehat{f}[i] - f[i] + M[i] > L-1\right\},\\
\mathcal{L}(0) &= \left\{i \in \mathbb{I}_n : \widehat{f}[i] - f[i] + m[i] < 0\right\}.
\end{aligned} \tag{25}$$

From Proposition 3 one has $\mathcal{U}(0) \subseteq \mathcal{U}(1)$. The notation in (25) enables Algorithms IV and IV to be restated as if follows:
– Multiplicative algorithm $(\lambda = 1)$

$$\begin{aligned}
&\text{(i)} && \widehat{w}_c[i] := \tfrac{\widehat{f}[i]}{f[i]}\,(w_c[i]-f[i]) + \widehat{f}[i] && \text{if} \quad i \in \mathbb{I}_n \setminus \mathcal{U}(1),\\
&\text{(ii)} && \widehat{w}_c[i] := \tfrac{L-1-\widehat{f}[i]}{M[i]-f[i]}(w_c[i]-f[i]) + \widehat{f}[i] && \text{if} \quad i \in \mathcal{U}(1).
\end{aligned} \tag{26}$$

– Additive algorithm $(\lambda = 0)$

$$\begin{aligned}
&\text{(i)} && \widehat{w}_c[i] := (w_c[i]-f[i]) + \widehat{f}[i] && \text{if} \quad i \in \mathbb{I}_n \setminus \{\mathcal{U}(0) \cup \mathcal{L}(0)\},\\
&\text{(ii)} && \widehat{w}_c[i] := \tfrac{L-1-\widehat{f}[i]}{M[i]-f[i]}(w_c[i]-f[i]) + \widehat{f}[i] && \text{if} \quad i \in \mathcal{U}(0),\\
&\text{(iii)} && \widehat{w}_c[i] := \tfrac{\widehat{f}[i]}{f[i]-m[i]}(w_c[i]-f[i]) + \widehat{f}[i] && \text{if} \quad i \in \mathcal{L}(0).
\end{aligned} \tag{27}$$

The relation between $\widetilde{w}_c$ and the outcome $\widehat{w}$ of our Algorithm III with the same λ is described in the following proposition.

Proposition 4: Let $\widehat{w}$ be the obtained by the Algorithm III and $\widetilde{w}$ by (23) for the same $\lambda \in [0,1]$. Then it holds for $i \in \mathbb{I}_n \setminus \left\{\mathcal{U}(1) \setminus \mathcal{U}(0) \cup \mathcal{L}(0)\right\}$ that $\widetilde{w}_c[i] = \widehat{w}_c[i]$.

Note that the "exceptional set" $\mathcal{U}(1) \setminus \mathcal{U}(0)$ contains in general much less pixels than $\mathcal{U}(1)$. The sets $\mathcal{U}(1) \setminus \mathcal{U}(0)$ and $\mathcal{L}(0)$ are usually small for reasonable target intensity images $\widehat{f}$. If we wish to see the enhancement results $\widehat{w}$ for various $\lambda \in [0,1]$, Proposition 4 justifies to compute instead the corresponding $\widetilde{w}$ by (23).

IV. Comparison of the algorithms

A. Saturation Properties

Here we analyse the saturation of images enhanced by our methods and the Naik-Murthy algorithm. By the definition of the saturation in (3), we have the following result.

Fig. 5. Enhancement of the image "couple" by our Algorithm III with concave histogram (parameters $(l, r) = (0.9, 0.1)$). Original image (top left) and enhancement with $\lambda = 0,\ \frac{1}{4},\ \frac{1}{2},\ \frac{3}{4},\ 1$. The corresponding values $\mathcal{U}(\lambda)$ (in percent of all image pixels) are given by 1.09, 2.20, 3.62, 5.74, 8.70 and the sets $\mathcal{L}(\lambda)$ are empty. A closely similar visual result will be obtained by forming the convex combinations according to (23). While the $\widehat{w}^{\times}$ is very colorful, its additive counterpart $\widehat{w}^{+}$ appears to be rather gray. All nuances between these images can be obtained by building their convex combinations.

Proposition 5: Let $S(w[i])$ and $S(\widehat{w}[i])$ denote the saturation of pixel i in the input image w and the image $\widehat{w}$ obtained by our Algorithm III, respectively. If $f[i] \in \{m[i], M[i]\}$ we have $S(\widehat{w}[i]) = 0$. Otherwise the saturation is given by

$$\begin{aligned}
&\text{(i)} && S(\widehat{w}[i]) = S(w[i]) \left(\lambda + (1-\lambda)\frac{f[i]}{\widehat{f}[i]}\right) && \text{if} \quad i \in \mathbb{I}_n \setminus \{\mathcal{U}(\lambda) \cup \mathcal{L}(\lambda)\},\\
&\text{(ii)} && S(\widehat{w}[i]) = S(w[i])\, \frac{f[i]}{\widehat{f}[i]}\, \frac{L-1-\widehat{f}[i]}{M[i]-f[i]} && \text{if} \quad i \in \mathcal{U}(\lambda),\\
&\text{(iii)} && S(\widehat{w}[i]) = 1 && \text{if} \quad i \in \mathcal{L}(\lambda).
\end{aligned}$$

To clarify the comparison, all magnitudes relevant to Algorithm I (Naik and Murthy) hold the superscript $\bullet$, those relevant to Algorithms IV (Multiplicative) and V (Additive) have the superscripts $\times$ and $+$, respectively. In particular we obtain for the Multiplicative and Additive algorithms:

– Algorithm IV (Multiplicative)

$$\begin{aligned}
&\text{(i)} && S(\widehat{w}^{\times}[i]) = S(w[i]) && \text{if} \quad i \in \mathbb{I}_n \setminus \mathcal{U}(1),\\
&\text{(ii)} && S(\widehat{w}^{\times}[i]) = S(w[i])\, \tfrac{f[i]}{\widehat{f}[i]}\, \tfrac{L-1-\widehat{f}[i]}{M[i]-f[i]} && \text{if} \quad i \in \mathcal{U}(1).
\end{aligned} \tag{28}$$

– Algorithm V (Additive)

$$\begin{aligned}
&\text{(i)} && S(\widehat{w}^{+}[i]) = S(w[i])\, \tfrac{f[i]}{\widehat{f}[i]} && \text{if} \quad i \in \mathbb{I}_n \setminus (\mathcal{U}(0) \cup \mathcal{L}(0)),\\
&\text{(ii)} && S(\widehat{w}^{+}[i]) = S(\widehat{w}^{\times}[i]) && \text{if} \quad i \in \mathcal{U}(0),\\
&\text{(iii)} && S(\widehat{w}^{+}[i]) = 1 && \text{if} \quad i \in \mathcal{L}(0).
\end{aligned} \tag{29}$$

Let us denote

$$\mathcal{V} := \left\{ i \in \mathbb{I}_n \ : \ \frac{\widehat{f}[i]}{f[i]} > 1 \right\}. \tag{30}$$

If $i \in \mathcal{U}(1)$ then by (25) one has $\frac{\widehat{f}[i]}{f[i]} > \frac{L-1}{M[i]} \geq 1$. From (25) and Remark 2 we find that if $i \in \mathcal{L}(0)$, then $\frac{\widehat{f}[i]}{f[i]} < 1 - \frac{m[i]}{f[i]} < 1$. Therefore,

$$\mathcal{V} \supseteq \mathcal{U}(1) \quad \text{and} \quad \mathcal{L}(0) \subset \mathbb{I}_n \setminus \mathcal{V}. \tag{31}$$

Using the notation in (30), case (i) in Algorithm I (Naik-Murthy) holds for any $i \in \mathbb{I}_n \setminus \mathcal{V}$ and step (ii) holds for any $i \in \mathcal{V}$. The saturation of images enhanced by applying the Naik - Murthy Algorithm I is given by the following proposition.

Proposition 6: Let $S(w[i])$ and $S(\widehat{w}^{\bullet}[i])$ denote the saturation of pixel i in the input image w and the image $\widehat{w}^{\bullet}$ obtained by the Naik-Murthy Algorithm I, respectively. Then the saturation of the enhanced image reads

$$\text{(i)} \qquad S(\widehat{w}^{\bullet}[i]) = S(w[i]) \qquad\qquad \text{if} \quad i \in \mathbb{I}_n \setminus \mathcal{V},$$

$$\text{(ii)} \qquad S(\widehat{w}^{\bullet}[i]) = S(w[i]) \, \tfrac{f[i]}{\widehat{f}[i]} \, \tfrac{L-1-\widehat{f}[i]}{L-1-f[i]} \quad \text{if} \quad i \in \mathcal{V}.$$

Remark 3: Let us note that all inclusions in Proposition 3 and in (31) are strict, except for a negligible set of images. E.g., in (31) we find $\mathcal{U}(1) = \mathcal{V}$ if and only if $M[i] = L-1$ for all $i \in \mathcal{V}$.

Using Proposition 5 and Proposition 6, the saturation that Algorithms IV, V and I provide can be rigorously compared.

(a) Let $i \in \mathbb{I}_n \setminus \mathcal{V}$. Then

$$S(w[i]) = S(\widehat{w}^{\times}[i]) = S(\widehat{w}^{\bullet}[i]) \leq S(\widehat{w}^{+}[i]),$$

where the last inequality becomes an equality only for $\widehat{f}[i] = f[i]$. Beyond this case, only the Additive Algorithms V increases the saturation; but since $\widehat{f}[i] < f[i]$, i.e. the output intensity is decreased, the visual effect might not be clearly seen (recall the Hunt effect).

(b) Let $i \in \mathcal{V} \setminus \mathcal{U}(1) = \{i \in \mathbb{I}_n : 1 < \frac{\widehat{f}[i]}{f[i]} \leq \frac{L-1}{M[i]}\}$. Then

$$S(w[i]) = S(\widehat{w}^{\times}[i]) > S(\widehat{w}^{+}[i]) > S(\widehat{w}^{\bullet}[i])$$

and $S(\widehat{w}^{\bullet}[i])$ decreases faster than $S(\widehat{w}^{+}[i])$ when $\widehat{f}[i]$ increases because

$$\frac{S(\widehat{w}^{\bullet}[i])}{S(\widehat{w}^{+}[i])} = \frac{M[i]-f[i]}{L-1-f[i]} < 1.$$

(c) Let $i \in \mathcal{U}(1)$. Then

$$S(w[i]) > S(\widehat{w}^{\times}[i]) \geq S(\widehat{w}^{\bullet}[i]),$$

where the equality is reached if and only if $M[i] = L-1$. But for most of the pixels one has $M[i] < L-1$. Further,

$$S(w[i]) > S(\widehat{w}^{+}[i]) \geq S(\widehat{w}^{\bullet}[i]),$$

where the equality holds if and only if $M[i] = L-1$ and $i \in \mathcal{U}(0)$. So the inequality is strict for most of the pixels.

In all cases, the images enhanced using Algorithm I have the weakest saturation. On $\mathbb{I}_n \setminus \mathcal{V}$, where the intensity is decreased, the Additive Algorithm V gives a better saturation than the Multiplicative Algorithm IV. On $\mathcal{V} \setminus \mathcal{U}(1)$ the Multiplicative Algorithm IV gives rise to a better saturation than the Additive Algorithm V.

B. Qualitative comparison

We begin with a simple but instructive example where the image has a constant color, or equivalently $w \in \mathbb{R}^{1\times 1\times 3}$.

Example 2: We apply our Algorithms IV, V and I to two different images composed out of one pixel namely a dark one $w_{\text{dark}} = (25, 48, 32)$ and a brighter one $w_{\text{light}} = (80, 172, 108)$ having the same hue but different intensities $f_{\text{dark}} = 35$ and $f_{\text{light}} = 120$. In Figs. 6 and 7, the input pixels w are shown on the top row, while the next rows detail the results of the algorithms w.r.t. the target intensity $\widehat{f} \in \{0, \ldots, 255\}$ given on the x-axis. By (25) we see that the pixel belongs to $\mathcal{U}(1)$ for $\widehat{f} > \frac{(L-1)f}{M} =: f_{\mathcal{U}(1)}$, to $\mathcal{U}(0)$ for $\widehat{f} > (L-1) - M + f =: f_{\mathcal{U}(0)}$, to $\mathcal{L}(0)$ for $\widehat{f} < f - m =: f_{\mathcal{L}(0)}$ and to $\mathcal{V}$ for $\widehat{f} > f := f_{\mathcal{V}}$. The corresponding values for our dark and light image are given in the following table:

	$f_{\mathcal{U}(1)}$	$f_{\mathcal{U}(0)}$	$f_{\mathcal{L}(0)}$	$f_{\mathcal{V}}$
w_{dark}	185.9	242	10	35
w_{light}	177.9	203	40	120

Fig. 6 deals with the dark input pixel w_{dark}. The Multiplicative Algorithm IV, step (i) is applied for $\widehat{f} \in [0, 185.9]$. All color values are multiplied by $\widehat{f}/f$, where $\widehat{f}/f > 1$ for $\widehat{f} > 35$ which yields a clear increase of the distance between all color channels. The third row shows a pleasant enhancement of the dark input pixel. By (28) the input saturation is preserved. Additive Algorithm V, step (i) is performed for $\widehat{f} \in [10,\ 242]$. In this step all color values are increased by the same amount $\widehat{f} - f$. Since the input pixel is quite dark, the values w_c, $c \in \{r, g, b\}$ and f are relatively close to each other. For this reason, all color channels $\widehat{w}_c^{+}$ remain close to each other. On $[10,\ 35)$ we have $f > \widehat{f}$ so by (29) $S(\widehat{w}^{+}) > S(w)$ and $S(\widehat{w}^{+})$ continuously decreases from 1 to $S(w) = 0.29$. On $[35,\ 242]$, $S(\widehat{w}^{+})$ decreases from $S(w) = 0.29$ to 0.04 according to $S(w)f/\widehat{f}$. This explains why the colors on the third row remain quite dark, compared to Algorithm IV. For Algorithm I, case (i) holds only for $\widehat{f} \in [0, 35]$, where the input saturation is preserved. If $\widehat{f} > 35$, step (ii) is performed and $S(\widehat{w}^{\bullet})$ decreases much faster than in Algorithm V. As a consequence, on $(35, 255]$ the enhanced colors tend to be nearly equal and the obtained color values are nearly gray, see the third row in the figure. Fig. 7 shows the performance for the brighter input pixel w_{bright}. The Multiplicative Algorithm IV, (i) is applied for $\widehat{f} \in [0, 177.9]$. The input saturation is preserved. The Additive Algorithm V,

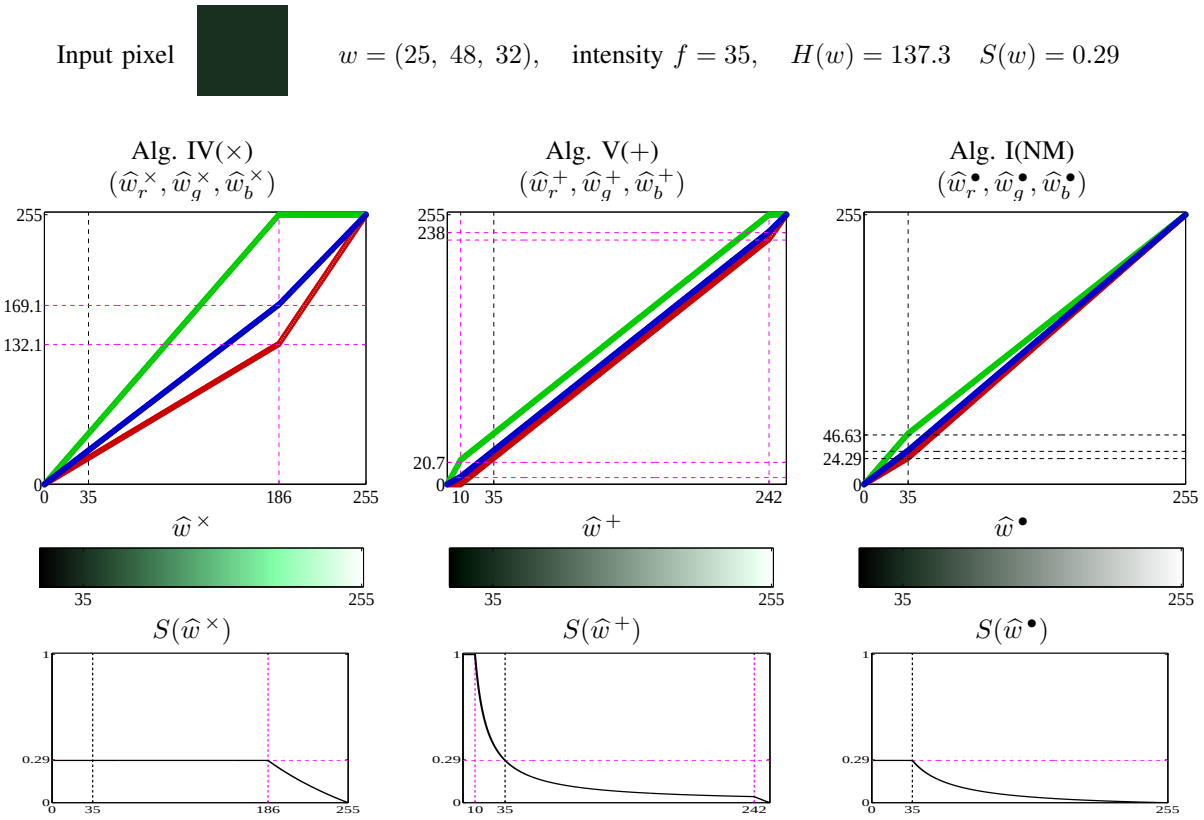

Fig. 6. Enhancement of a quite dark pixel shown in the first row. Second and third rows: the output intensity $\widehat{f}$ is on the x-axis and the plots depict the results of Algorithms IV, V and I. The second row specify the value of each color channel. The third row shows the resulting color $\widehat{w}$ w.r.t. $\widehat{f}$ and the last row plots the saturation of the output pixel as a function of $\widehat{f}$.

(i) holds for $\widehat{f} \in [40, 203]$. On $[40,\ 120]$ the recovered saturation decreases from 1 to $S(w) = 0.33$ and on $(120, 203]$ it slowly decreases to $0.6S(w)$. In Algorithm I, step (i) holds for $\widehat{f} \leq f = 120$ where the input saturation is unchanged. Step (ii) is applied for $\widehat{f} \in (120, 255]$ – the interval is not so large as in Fig. 6 and the saturation decreases much less fast to zero. On the 3rd row one sees that the colors obtained with all the three algorithms are quite similar.

Remark 4: From Fig. 6 one sees that if a dark pixel has a wrong hue (e.g. due to compression or printing artifacts, noise, color cast, and so on), the Multiplicative Algorithm IV may amplify significantly the intensity of this wrong color. If the input image contains a lot of such pixels, the Additive Algorithm V could be a more judicious choice, or a convex combination of both solutions as proposed in Subsection III-B.

Our conclusions drawn in Subsection IV-A and from Example 2 are tested on two images, "bungalow" (underexposed) and "flower" (slightly lustreless). The distributions of $\widehat{f}[i]/f[i]$ for these two images is very different – the first one ranges on $[0,\ 18]$ and the second on $[0,\ 1.22]$. Roughly speaking, "bungalow" mimics the phenomena explained for Fig. 6 and "flower" – those relevant to Fig. 7.

V. Numerical Results

In this section we compare our Additive and Multiplicative algorithms IV and V with the Naik-Murthy algorithm I and the automatic color enhancement algorithm (ACE) [15]. ACE was introduced by Gatta, Rizzi, and Marini in [35] with further improvements in a series of papers. This method is based on modeling several low level mechanisms of the human visual system. The direct computation of ACE for an n-pixel image costs $\mathcal{O}(n^2)$ arithmetic operations. A faster algorithm reducing the cost to $\mathcal{O}(n \log n)$ operations was given by Bertalmio, Caselles, Provenzi, and Rizzi in [10]. This sophisticated method is often considered as state-of-the-art method for color image enhancement. In this paper we use the fast implementation of the ACE algorithm in [15] available online at `http://demo.ipol.im/demo/g_ace/`.

The next Subsection V-A briefly comments on our choices of target histograms. We want to emphasize that the search for target histograms is not the content of this paper. Instead we will use rather simple recommendations. Then, in Subsection V-B, various enhancement examples demonstrate the very good performance of our algorithms.

A. Target Histograms

Our enhancement results depend (up to some degree) on the choice of the target histogram. We have seen in the introduction that HE is far away from being suitable in many cases. Various target histograms have been proposed in the literature. Among

Input pixel  $w = (80,\ 172,\ 108)$, intensity $f = 120$, $H(w) = 137.3$, $S(w) = 0.33$

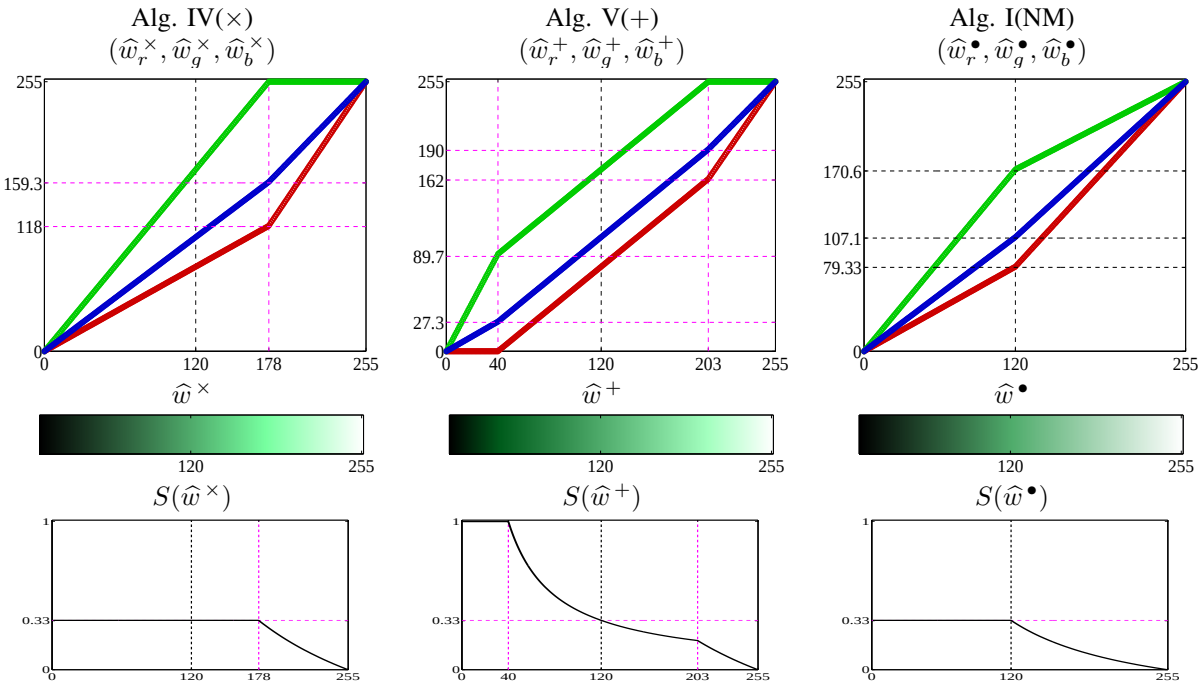

Fig. 7. Enhancement of a quite bright pixel shown in the first row. Second and third rows: the output intensity $\widehat{f}$ is on the x-axis and the plots depict the results of Algorithms IV, V and I. The second row specify the value of each color channel. The third row shows the resulting color $\widehat{w}$ w.r.t. $\widehat{f}$ and the last row plots the saturation of the output pixel as a function of $\widehat{f}$.

them we evoke the brightness preserving maximum entropy construction [36], the piecewise linear adjustment [37] and the mix of both, and the piecewise maximum entropy model [27]. These receipts were improved by the target histograms designed in [8] which we apply in this paper. A common approach, formulated in the textbook [6], is to take the histogram of a well exposed *example* image. Commercials in photography and image processing software (e.g., Photoshop) recommend bell-shaped histograms since they correspond to well exposed pictures. Combining their advices with the approach in [6], we use some bell-shaped target histograms, such as concave and Gaussian shaped functions.

Let $h_{\text{target}} : [0, L-1] \to \mathbb{R}_+$ be the shape of a desired histogram. We recall that the corresponding target histograms $\widehat{h}_{\text{target}}$ is normalized so that

$$\widehat{h}_{\text{target}}(x) = \frac{n\, h_{\text{target}}(x)}{\sum_{x=0}^{L-1} h_{\text{target}}(x)}, \quad \forall\, x \in \{0, \cdots, L-1\}.$$

The shapes of our concave and Gaussian histograms depend on two user defined (and easy to select) parameters,

$$l := h_{\text{target}}(0) \in [0,1] \quad \text{and} \quad r := h_{\text{target}}(L-1) \in [0,1]$$

and are fixed so that $\max_{[0,L-1]} h_{\text{target}}(x) = 1$.

(a) Concave function

$$h_{\text{conc}}(x) = ax^2 + bx + c, \quad x \in [0, L-1],$$

where the coefficients a, b and c read for $l \in [0,1)$ as
$b = -\frac{2(l-1)}{L-1}\left(1 + \sqrt{\frac{r-1}{l-1}}\right)$, $a = \frac{b^2}{4(l-1)}$, $c = l$, and take the values $b = 0$, $a = \frac{r-1}{(L-1)^2}$, $c = l$ for $l = 1$.

(b) Gaussian function

$$h_{\text{Gauss}}(x) = \exp\left(-\frac{(x-c)^2}{s}\right), \quad x \in [0, L-1],$$

where for $l \in (0,1)$ and $r \in (0,1)$ the coefficients c and s are given by $c = \ln l\, \frac{L-1}{\ln l - \ln r}\left(1 - \sqrt{\frac{\ln r}{\ln l}}\right)$, $s = -\frac{c^2}{\ln r}$.

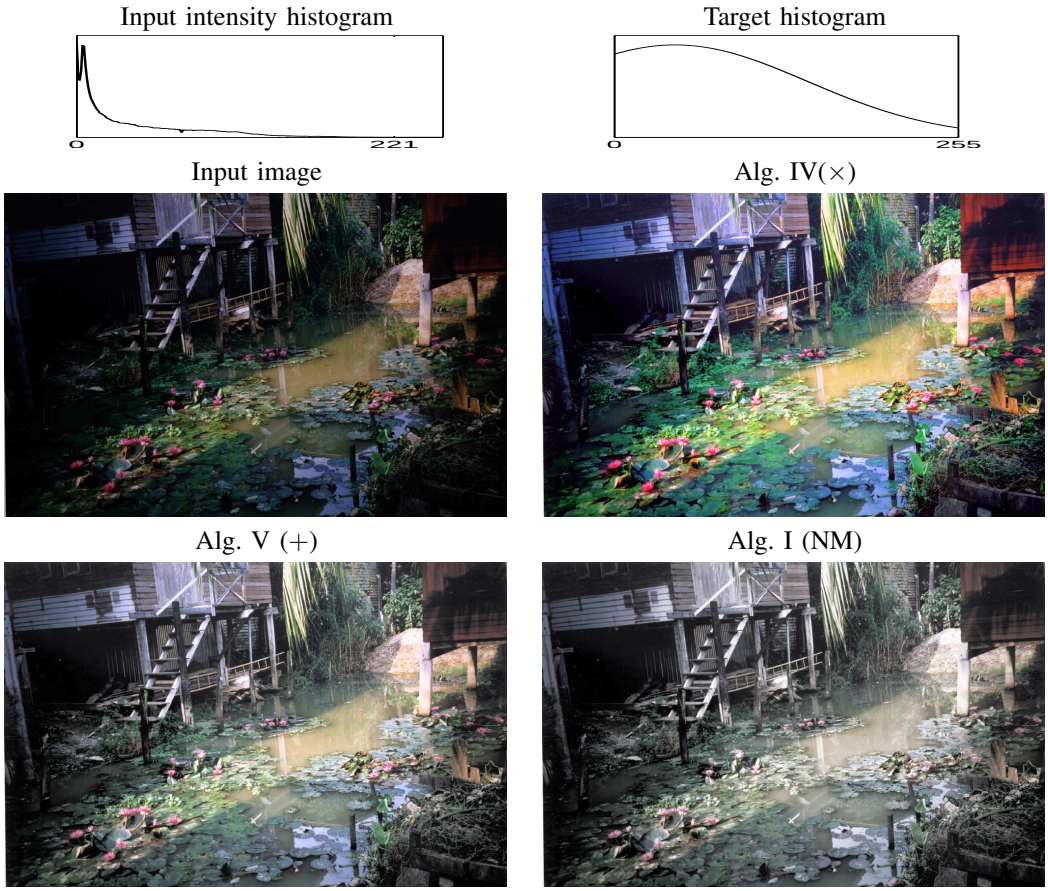

Fig. 8. Original image “bungalow” and enhanced versions.

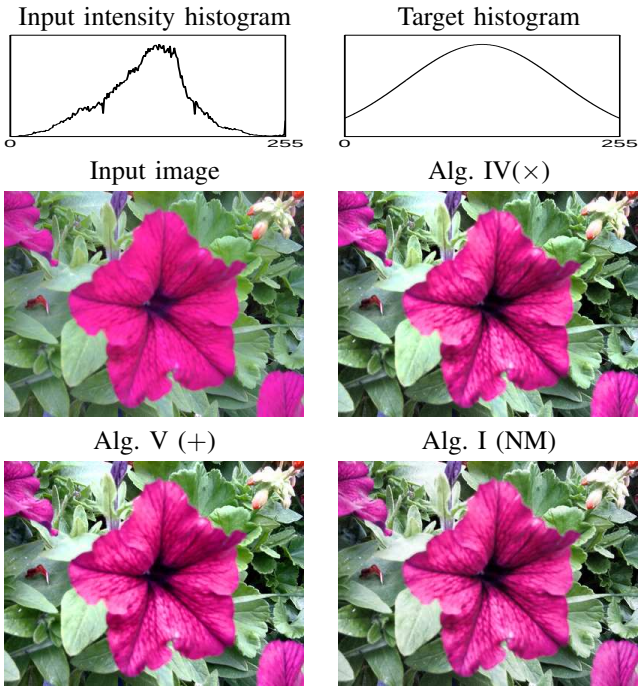

Fig. 9. Original image “flower” and enhanced versions.

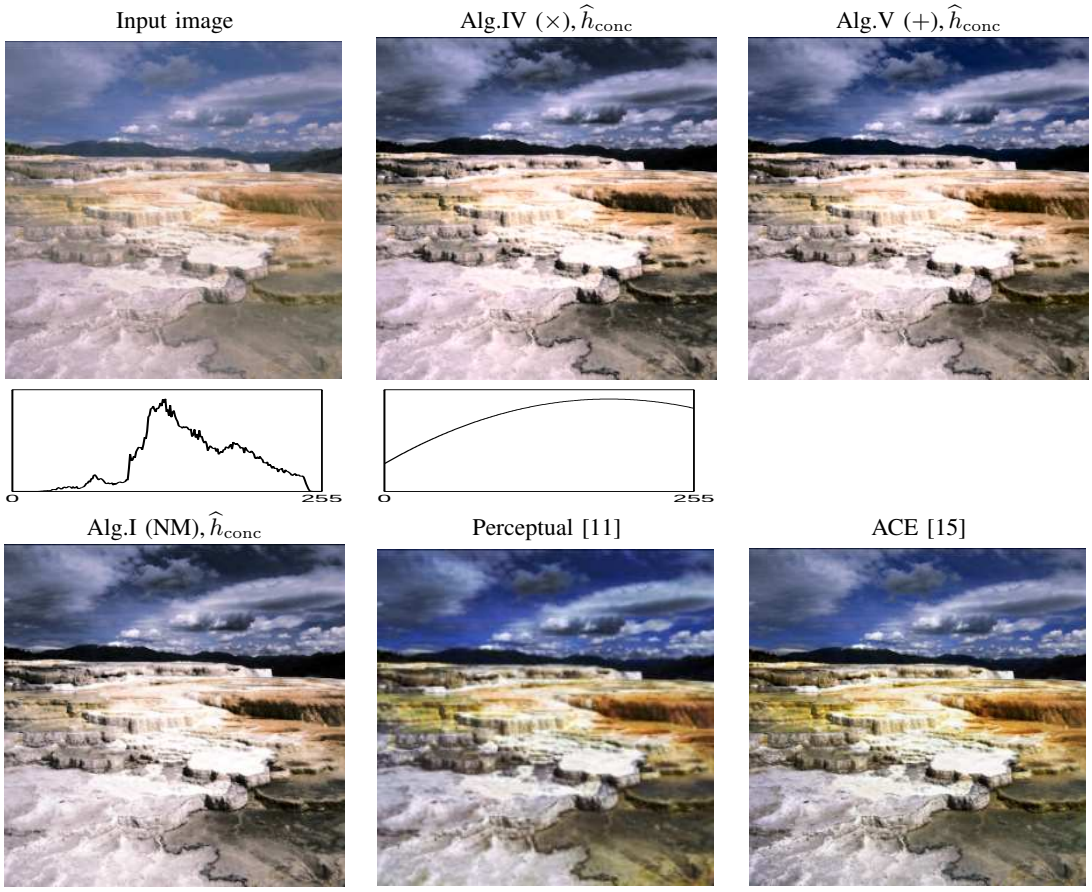

Fig. 10. Top: Input image "islanda" and enhanced images by our Multiplicative and Additive algorithms with concave target histogram, $(l, r) = (0.3, 0.9)$. Middle: Histogram of the luminance of the original image and of the concave target histogram. Bottom: Enhanced images by the Naik-Murthy algorithm with concave target histogram, the perceptual color enhancement method in [11] (Image courtesy of the authors, R. Palma-Amestoy, E. Provenzi, M. Bertalmio, and V. Caselles), and by ACE. The HS based algorithms provide similar good results. The last two perceptually inspired approaches modify the hue which gives a nice, different color content of the image.

(c) Following Arici, Dikbas and Altunbasak [8], we test also target histogram $\widehat{h}_{\rm ADA}$ that mix the histogram of the input image h_f with a uniform histogram denoted by u:

$$\begin{aligned}\widehat{h}_{\rm ADA} &:= \operatorname*{argmin}_v \{\mu\|v - h_f\|_2^2 + \|v - u\|_2^2\} \\ &= \frac{\mu}{1+\mu} h_f + \frac{1}{1+\mu} u. \end{aligned} \tag{32}$$

This basic construction was modified in [8] by a smoothing term which was not useful in our experiments. As another improvement a black and white stretching was proposed which multiplies the above $h(x)$ by a factor smaller than 1 (default $\frac{1}{6}$) if $\widehat{h}_{\rm ADA}(x) < w$ or $\widehat{h}_{\rm ADA}(x) > b$ for some chosen parameters w and b. We have applied this additional stretching only in Fig. 12.

B. Enhancement Examples

In this subsection we compare our Multiplicative and Additive algorithms, denoted by Alg. IV(×) and Alg. V (+) on top of the figures, with the the Naik-Murthy algorithm [2], abbreviated by Alg. I (NM), and the automatic color enhancement method (ACE) in [15]. The histogram specifications in the Naik-Murthy algorithm and in our algorithms were done via our Strict Ordering Algorithm II. The histograms of the intensity of the original image and the target histograms are depicted in the middle row of the figures. The chosen target histograms $\widehat{h}_{\rm conc}, \widehat{h}_{\rm Gauss}, \widehat{h}_{\rm ADA}$ from the previous subsection are given at the top of the figures. The number of pixels which were treated by the lower and upper gamut steps is contained in Table I. This number is very small for our algorithms in all experiments.

The enhancement results for the image "islanda" in Fig. 10 corroborate our discussion in Subsection IV-B, in particular the second case in Example 2.

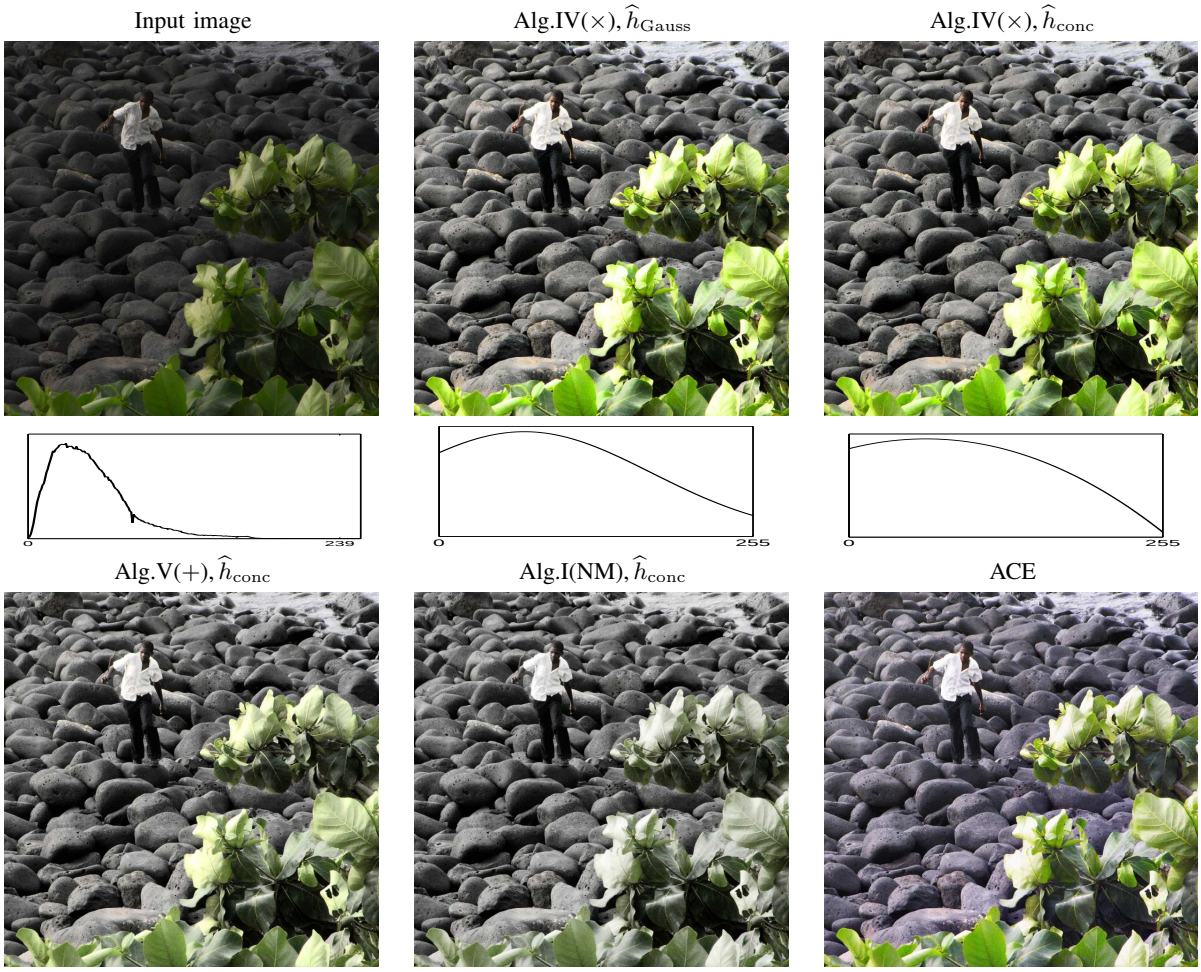

Fig. 11. Top: Original image "boy-on-stones" and enhanced images by our Multiplicative algorithm with Gaussian target histogram ($(l, r) = (0.8, 0.2)$) and concave one ($(l, r) = (0.9, 0.05)$). In spite of the different target histograms both results are very similar. Middle: Histogram of the luminance of the original image and of the Gaussian and concave target histogram. Bottom: Enhanced images by our Additive algorithm and the Naik-Murthy algorithm with concave target histogram, and by ACE. These images are worse than those obtained by the Multiplicative algorithm. However, our Additive method still outperforms the Naik-Murthy approach. The ACE enhanced image has blue color artefacts.

The following images "boy-on-stones", "cathedral" and "orchid" are photos that the authors could not take correctly and they tried to improve them.

Fig. 11 shows the enhancement results for the "boy-on-stones" image. The original image is too dark, and is related to the first case in Example 2 in Subsection IV-B. The image "cathedral" is much too dark. A comparison of enhancement methods is given in Fig. 12. This is the only image where we apply the target histogram from [8] with a black and white stretching with parameters $w = 30$ and $b = 150$. Finally, the "orchid" image in Fig. 13 which suffers from flashlight effect was successfully enhanced by our methods. Note that the ACE enhance image contains blue color artefacts again.

VI. Conclusions and future work

This work provides the first comprehensive and rigorous presentation of the wide family of histogram specification based affine color assignment models. We have proposed a fast hue and range preserving algorithm. We analyzed the performances of this algorithm and two of its important instances as well as the gamut preserving method in [2].

As usual dealing with a topic creates many open questions we want to answer in our future research. Instead of the intensity of the input image, we can consider other combinations between the RGB channels that are better adapted to human color perception and to the image content. An example with other input histograms than those of the intensity, namely with luma and the green channel was given in Fig. 3. Moreover, it may be useful to take the saturation or chromaticity of the input image into account Finally, a systematic approach to find the target histogram is clearly desirable and topic of future research. Since our algorithms are fast, extensions to video should be envisaged.

VII. Appendix

Proof of Proposition 1: The upper bound follows directly from the choice of $\widehat{w}_c[i]$, $c \in \{r, g, b\}$. It remains to show the lower bound. First we see that (15) implies $\widehat{f}[i] \geq f[i]$, since in case $\widehat{f}[i] < f[i]$ we would get by replacing $\widehat{f}[i]$ by $f[i]$ in

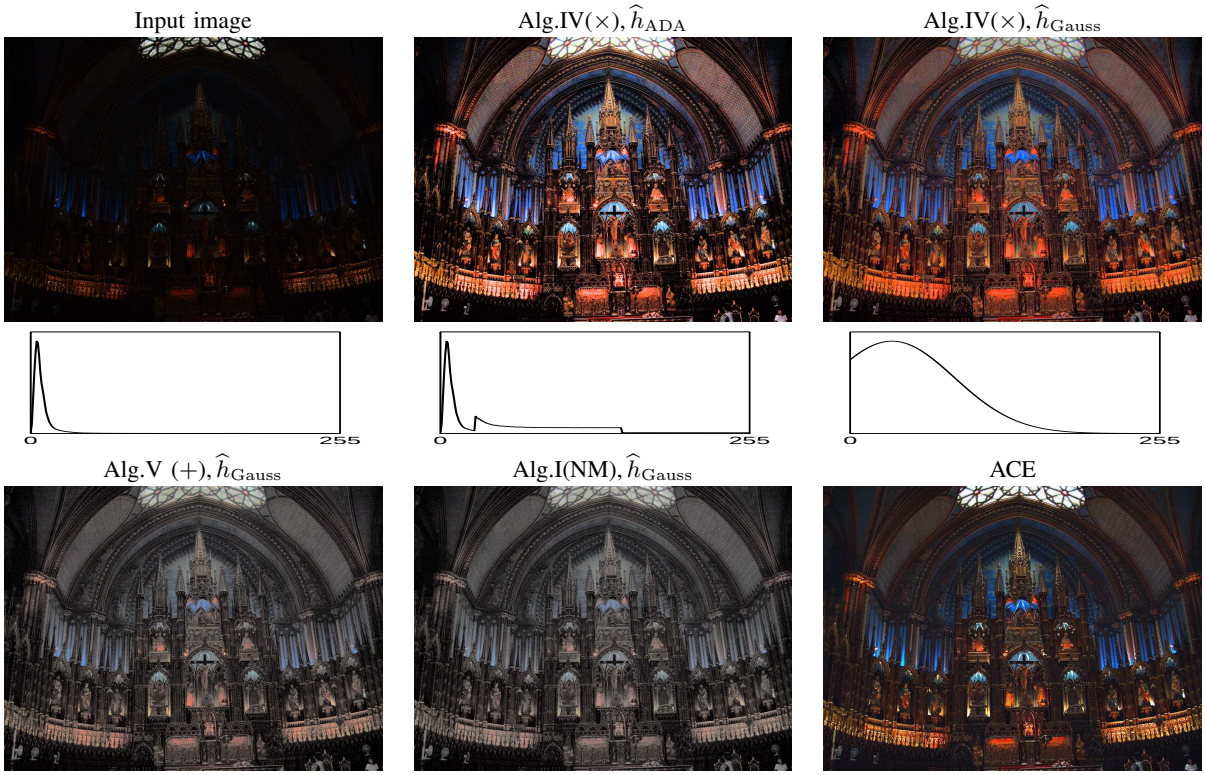

Fig. 12. Top: Original image "cathedral" and enhanced images by our Multiplicative algorithm with target histogram from [8] ($\mu = 0.3$), and Gaussian target histogram $(l, r) = (0.8, 0.0001)$. The methods give different visually acceptable results. Middle: Histogram of the intensity of the original image and of the target histogram obtained by [8] and the Gaussian one. Bottom: Enhanced images by our Additive algorithm and the Naik-Murthy algorithm with Gaussian target histogram, and by ACE. Our Additive algorithm and the Naik-Murthy algorithm give similar (bad) results with a small advantage for the first one. where the first algorithm performs slightly better. ACE produces a nice colorful image.

image	$\sharp\mathcal{U}(1)$	$\sharp\mathcal{U}(0)$	$\sharp\mathcal{L}(0)$	$\sharp\mathcal{V}$
islanda ($\widehat{h}_{\mathrm{conc}}$)	8.64	2.61	7.63	53.96
boy-on-stones ($\widehat{h}_{\mathrm{conc}}$)	3.94	1.56	0.61	93.38
boy-on-stones ($\widehat{h}_{\mathrm{Gauss}}$)	4.76			
cathedral ($\widehat{h}_{\mathrm{Gauss}}$)	1.64	0.00	0.81	97.72
cathedral ($\widehat{h}_{\mathrm{ADA}}$)	5.86			
orchid ($\widehat{h}_{\mathrm{Gauss}}$)	0.20	0.07	1.45	98.02
orchid ($\widehat{h}_{\mathrm{ADA}}$)	3.70			

TABLE I
PERCENTAGE OF PIXELS REQUIRING AN UPPER OR LOWER GAMUT CORRECTION IN FIGS. 10 -13 FOR HS BASED METHODS.

the denominator of the quotient and the second summand in (15) the contradiction $M[i] > L - 1$. Since $M[i] - f[i] > 0$ the lower bound holds true if and only if

$$\begin{aligned}(L-1-\widehat{f}[i])(w_c[i]-f[i])+\widehat{f}[i](M[i]-f[i]) &\geq 0,\\ (L-1-\widehat{f}[i])w_c[i]-(L-1)f[i]+\widehat{f}[i]M[i] &\geq 0\end{aligned}$$

which is clearly fulfilled if

$$\widehat{f}[i]M[i]-(L-1)f[i]\geq 0. \tag{33}$$

By (15) we have

$$\begin{aligned}&\widehat{f}[i]M[i]-(L-1)f[i]\\ &> \widehat{f}[i]M[i]-\left(\lambda\widehat{f}[i]+(1-\lambda)f[i]\right)(M[i]-f[i])-\widehat{f}[i]f[i]\\ &= \left(\widehat{f}[i]-\left(\lambda\widehat{f}[i]+(1-\lambda)f[i]\right)\right)(M[i]-f[i]).\end{aligned}$$

By $\widehat{f}[i] \geq f[i]$ the first factor on the right-hand side is non-negative which is also true for the second one. Thus, (33) is

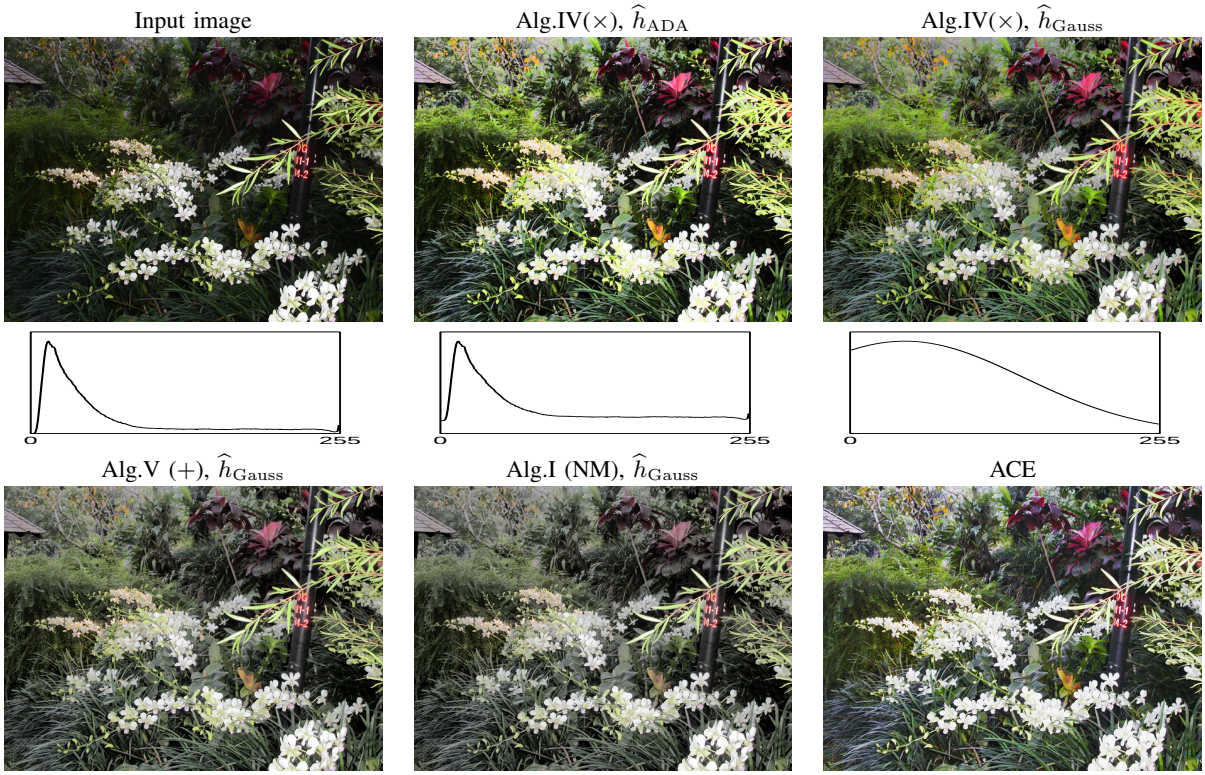

Fig. 13. Top: Original image "orchid" with flashlight effect and enhanced images by our Multiplicative algorithm with target histogram from (c) and Gaussian target histogram ($(l, r) = (0.9, 0.1)$). The first one still overemphasizes the white color in the flowers. This effect is reduced in the second enhanced image. Middle: Histogram of the intensity of the original image and of the target histogram obtained by [8] ($\mu = 1$) and the Gaussian one ($(l, r) = (0.9, 0.1)$). Bottom: Enhanced images by our Additive algorithm and the Naik-Murthy algorithm with the same concave target histogram, and by ACE. As in previous two examples our Additive algorithm and the Naik-Murthy method give too gray images, where the result from the Additive algorithm looks better. The ACE enhanced image contains again blue color artefacts and the flashlight effect on the white flowers is still visible.

satisfied and we are done. ■

Proof of Proposition 2: The lower bound is clear from the construction of $w_c[i]$. We show the upper bound. Developing the expression (20) leads to

$$\begin{aligned} \widehat{w}_c[i] &= \frac{\widehat{f}[i] w_c[i] - \widehat{f}[i]\ f[i] + \widehat{f}[i]\ f[i] - \widehat{f}[i]\ m[i]}{f[i] - m[i]} \\ &= \frac{\widehat{f}[i]}{f[i] - m[i]}\ (w_c[i] - m[i]), \quad c \in \{r, g, b\}, \end{aligned} \tag{34}$$

Using (18), one has

$$\widehat{f}[i]\ \frac{\lambda m[i] + (1-\lambda) f[i]}{f[i]} + (1-\lambda)(m[i] - f[i]) < 0$$

and hence

$$\widehat{f}[i] < \frac{(1-\lambda) f[i]}{\lambda m[i] + (1-\lambda) f[i]} (f[i] - m[i]) < f[i] - m[i].$$

Thus, since $0 < w_c[i] - m[i] < L - 1$, we obtain finally

$$\frac{\widehat{f}[i]}{f[i] - m[i]} (w_c[i] - m[i]) < L - 1.$$

■

Proof of Proposition 3: Let $0 \leq \lambda_1 < \lambda_2 \leq 1$. In the upper gamut case we always have $\widehat{f}[i] > f[i]$. Then $\lambda_1 \widehat{f}[i] + (1 - \lambda_1) f[i] < \lambda_2 \widehat{f}[i] + (1 - \lambda_2) f[i]$, hence $G_M^{\lambda_1}[i] \leq G_M^{\lambda_2}[i]$ and $\mathcal{U}(\lambda_1) \subseteq \mathcal{U}(\lambda_2)$. Similarly, since $\widehat{f}[i] < f[i]$ in the lower gamut case, $\lambda_1 \widehat{f}[i] + (1 - \lambda_1) f[i] > \lambda_2 \widehat{f}[i] + (1 - \lambda_2) f[i]$ so that $G_m^{\lambda_1}[i] \geq G_M^{\lambda_2}[i]$ and hence $\mathcal{L}(\lambda_1) \supseteq \mathcal{L}(\lambda_2)$. ■

Proof of Proposition 4: Let $i \in \mathbb{I}_n \setminus \{\mathcal{U}(1) \cup \mathcal{L}(0)\}$. Then $\widehat{w}_c^{\times}[i]$ is given by (26)(i) and since $\mathcal{U}(0) \subseteq \mathcal{U}(1)$, the value

$\widehat{w}_c^+[i]$ is given by (27)(i). Consequently,

$$\widetilde{w}_c[i] = \lambda\frac{\widehat{f}[i]}{f[i]}\left(w_c[i]-f[i]\right)+(1-\lambda)(w_c[i]-f[i])+\widehat{f}[i] = \widehat{w}_c[i],$$

where the last equality follows by $\mathcal{U}(\lambda)\subseteq\mathcal{U}(1)$ from (21) and Algorithm III(i).

Let $i\in\mathcal{U}(0)$. Then $\widehat{w}_c^+[i]$ is given by (27)(ii) and since $\mathcal{U}(0)\subseteq\mathcal{U}(1)$, $\widehat{w}_c^\times[i]$ is given by (26)(ii). We have $\widehat{w}_c^+[i]=\widehat{w}_c^\times[i]$ which shows by $\mathcal{U}(0)\subseteq\mathcal{U}(\lambda)$ that $\widetilde{w}_c[i]=\widehat{w}[i]$. This finishes the proof. ■

Proof of Proposition 5: The result in case $f[i]\in\{m[i],M[i]\}$ follows by Step 2.1 of our Algorithm III. The saturation obtained in Step 2.2 is proven for the three cases separately.

(i) Let $i\in\mathbb{I}_n\setminus\{\mathcal{U}(\lambda)\cup\mathcal{L}(\lambda)\}$. Since $\frac{\lambda\widehat{f}[i]+(1-\lambda)f[i]}{f[i]}>0$, we have

$$\begin{aligned}
S(\widehat{w}[i]) &= 1-\frac{1}{\widehat{f}[i]}\min_{c\in\{r,\,g,\,b\}}\left\{\frac{\lambda\widehat{f}[i]+(1-\lambda)f[i]}{f[i]}\left(w_c[i]-f[i]\right)+\widehat{f}[i]\right\}\\
&= -\frac{1}{\widehat{f}[i]}\left(\frac{\lambda\widehat{f}[i]+(1-\lambda)f[i]}{f[i]}\left(\min\{w_r[i],w_g[i],w_b[i]\}-f[i]\right)\right)\\
&= \left(1-\frac{\min\{w_r[i],w_g[i],w_b[i]\}}{f[i]}\right)\left(\lambda+(1-\lambda)\frac{f[i]}{\widehat{f}[i]}\right)\\
&= S(w[i])\left(\lambda+(1-\lambda)\frac{f[i]}{\widehat{f}[i]}\right).
\end{aligned}$$

(ii) Let $i\in\mathcal{U}(\lambda)$. The expression for $\widehat{w}_c[i]$ also reads as:

$$\widehat{w}_c[i] = d[i](w_c[i]-f[i])+\widehat{f}[i] \quad\text{with}\quad d[i]:=\frac{L-1-\widehat{f}[i]}{M[i]-f[i]}.$$

Since $d[i]>0$, we obtain

$$\begin{aligned}
S(\widehat{w}[i]) &= 1-\frac{1}{\widehat{f}[i]}\Big(d[i]\big(\min\{w_r[i]w_g[i],w_b[i]\}-f[i]\big)+\widehat{f}[i]\Big)\\
&= 1-\frac{f[i]}{\widehat{f}[i]}d[i]\frac{\min\{w_r[i],w_g[i],w_b[i]\}}{f[i]}+\frac{f[i]}{\widehat{f}[i]}\,d[i]-1\\
&= S(w[i])\frac{f[i]}{\widehat{f}[i]}d[i] = S(w[i])\frac{(L-1)f[i]-\widehat{f}[i]f[i]}{M[i]\widehat{f}[i]-\widehat{f}[i]f[i]}.
\end{aligned}$$

(iii) Consider that $i\in\mathcal{L}(\lambda)$. Then the assumption follows directly from the definition (3) of the saturation and construction of Algorithm III. The proof is complete. ■

Proof of Proposition 6: The case $\frac{\widehat{f}[i]}{f[i]}\le 1$ follows just as a special case of Proposition 5(i) for $\lambda=1$.

Let $\widehat{f}[i]>f[i]$. Then (6) can be written as

$$\widehat{w}_c[i] = d[i]\left(w_c[i]-f[i]\right)+\widehat{f}[i] \quad\text{with}\quad d[i]:=\frac{L-1-\widehat{f}[i]}{L-1-f[i]}.$$

By the same arguments as in part (ii) of the previous proof we obtain $S(\widehat{w}[i])=S(w[i])\frac{f[i]}{\widehat{f}[i]}\,d[i]$ and we are done. ■

REFERENCES

[1] M. Nikolova and G. Steidl, "Fast sorting algorithm for exact histogram specification", Preprint hal-00870501, 2013.

[2] S. F. Naik and C. A. Murthy, "Hue-preserving color image enhancement without gamut problem", *IEEE Trans. Image Process.*, vol. 12, no. 12, pp. 1591–1598, Dec. 2003.

[3] T. Gevers and A. Smeulders, "Color-based object recognition", *Pattern Recognition*, vol. 32, 1999.

[4] C. Solomon and T. Breckon, *Fundamentals of Digital Image Processing. A Practical Approach with Examples in Matlab*, John Wiley & Sons, UK, 1 edition, 2011.

[5] R. S. Berns, *Billmeyer and Saltzman Principles of Color Technology*, Willey & Sons. Roy S., Ca, 3 edition, 2000.

[6] A. C. Bovik, *Handbook of image and video processing (Communications, Networking and Multimedia)*, Academic Press, Orlando, 2005.

[7] M. Nikolova, Y.-W. Wen, and R. Chan, "Exact histogram specification for digital images using a variational approach", *J. Math. Imaging and Vision*, vol. 46, no. 3, pp. 309–325, Jul. 2013.

[8] T. Arici, S. Dikbas, and Y. Altunbasak, "A histogram modification framework and its application for image contrast enhancement", *IEEE Trans. Image Process.*, vol. 18, no. 9, pp. 1921–1935, Sep. 2009.

[9] V. Caselles, J. L. Lisani, J. M. Morel, and G. Sapiro, "Shape preserving local histogram modification", *IEEE Trans. Image Process.*, vol. 8, no. 2, pp. 220–229, Feb. 1999.

[10] M. Bertalmío, V. Caselles, E. Provenzi, and A. Rizzi, "Perceptual color correction through variational techniques", *IEEE Trans. Image Process.*, vol. 16, no. 4, pp. 1058–1072, Apr. 2007.

[11] R. Palma-Amestoy, E. Provenzi, M. Bertalmio, and V. Caselles, "A perceptually inspired variational framework for color enhancement", *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 31, no. 3, pp. 458–474, 2009.

[12] N. Papadakis, E. Provenzi, and V. Caselles, "A variational model for histogram transfer of color images", *IEEE Trans. Image Process.*, vol. 30, no. 6, pp. 1682–1695, June 2011.

[13] E. Provenzi, *Perceptual Color Correction: A Variational Perspective*, pp. 109–119, Lecture notes in Computer science LNCS 5646. Springer, 2009.

[14] A. Rizzi, C. Gatta, and D. Marini, "A new algorithm for unsupervised global and local color correction", *Pattern Recognit. Lett*, vol. 124, 2003.

[15] P. Gertreuer, "Automatic color enhancement (ACE) and its fast implementation", *Image Processing On Line*, 10.5201/ipol.2012.g-ace, 2012.

[16] P. E. Trahanias and A. Venetsanopoulos, "Color image enhancement through 3-D histogram equalization", in *Proc. 15th IAPR Int. Conf. Pattern Recognit*, 1997, vol. 1, pp. 545–548.

[17] J. H. Han, S. Yang, and B. U. Lee, "A novel 3-D color histogram equalization method with uniform 1-D gray scale histogram", *IEEE Trans. Image Process.*, vol. 20, no. 2, pp. 506–512, Feb. 2011.

[18] N. Bassiou and C. Kotropoulos, "Color image histogram equalization by absolute discounting back-off", *Computer Vision and Image Understanding*, vol. 107, 2007.

[19] G. Singh and T. Rawat, "Color image enhancement by linear transformations solving out of gamut problem", *International Journal of Computer Applications*, vol. 67, no. 14, pp. 28–32, Apr. 2013.

[20] Z. Y. Chen, B. R. Abidi, D. L. Page, and M. A. Abidi, "Gray-Level Grouping (GLG): An automatic method for optimized image contrast enhancementpart i: The basic method", *IEEE Trans. Image Process.*, vol. 15, no. 8, pp. 2290–2302, Aug. 2006.

[21] D. Menotti, L. Najman, A. d. Araújo, and J. Facon, "A fast hue-preserving histogram equalization method for color image enhancement using a bayesian framework", in *Proc. 14th Int.Workshop Syst., Signal Image Process. (IWSSIP)*, 2007, pp. 414–417.

[22] D. Menotti, L. Najman, J. Facon, and A. Albuquerque, "Histogram equalization methods for color image contrast enhancement", *International Journal of Computer Science & Information Technology (IJCSIT)*, vol. 4, no. 5, pp. 243–259, Oct. 2012.

[23] N. Kong and H. Ibrahim, "Color image enhancement using brightness preserving dynamic histogram equalization", *IEEE Trans. Consum. Electron.*, vol. 54, no. 4, pp. 1962–1967, Nov. 2008.

[24] M. Nikolova, "A fast algorithm for exact histogram specification. simple extension to colour images", in *Scale Space and Variational Methods in Computer Vision*. Lecture Notes in Computer Science 7893, Springer, 2013, pp. 174–185.

[25] Q. Wang and R. K. Ward, "Fast image/video contrast enhancement based on weighted thresholded histogram equalization", *IEEE Trans. Consum. Electron.*, vol. 53, no. 2, pp. 757–764, May 2007.

[26] D. Sen and P. Sankar, "Automatic exact histogram specification for contrast enhancement and visual system based quantitative evaluation", *IEEE Trans. Image Process.*, vol. 20, no. 5, pp. 1211–1220, May 2011.

[27] G. Thomas, "A modified version of van-cittert's iterative deconvolution procedure", *IEEE Trans. Acoustics Speech and Signal Processing*, vol. ASSP-29, pp. 938–939, 1981.

[28] C.-L. Chien and D.-C. Tseng, "Color image enhancement with exact HSI color model", *Int. J. of Innovative Computing, Information and Control*, vol. 7, no. 12, pp. 6691–6710, Dec. 2011.

[29] R. Gonzalez and R. Woods, *Digital Image Processing*, Prentice-Hall, New Jersey, 2 edition, 2002.

[30] C. C. Yang and J. J. Rodriguez, "Efficient luminance and saturation processing techniques for bypassing color coordinate transformations", in *Proc. IEEE Int. Conf. on Systems, Man, and Cybernetics*, 1995, vol. 1, pp. 667–672.

[31] C. C. Yang and J. J. Rodriguez, "Saturation clipping in the lhs and yiq color spaces", in *Proc. IS&T/SPIE Int. Symp. Electronic Imaging: Science & Technology Color Imaging: Device-Independent Color, Color Hard Copy, and Graphic Arts*, 1996, vol. 1.

[32] D. Coltuc, P. Bolon, and J.-M. Chassery, "Exact histogram specification", *IEEE Trans. Image Process.*, vol. 15, no. 6, pp. 1143–1152, 2006.

[33] Y. Wan and D. Shi, "Joint exact histogram specification and image enhancement through the wavelet transform", *IEEE Trans. Image Process.*, vol. 16, no. 9, pp. 2245–2250, 2007.

[34] F. Bauss, M. Nikolova, and G. Steidl, "Fully smoothed ℓ_1-TV models: Bounds for the minimizers and parameter choice", *J. Math. Imaging and Vision, (online)*, 2013.

[35] C. Gatta, A. Rizzi, and D. Marini, "ACE:", in *An automatic color equalization algorithm*. First European on Color in Graphics Image and Vision (CGIV02), 2002, pp. 316–320.

[36] C. Wang and Z. Ye, "Brightness preserving histogram equalization with maximum entropy: a variational perspective", *IEEE Trans. Consum. Electron.*, vol. 51, no. 4, pp. 1326–1334, Nov. 2005.

[37] C. M. Tsai and Z. M. Ye, "Contrast enhancement by automatic and parameter-free piecewise linear transformation for color images", *IEEE Trans. Consum. Electron.*, vol. 54, no. 2, pp. 213–219, May 2008.