Color and Contrast Enhancement by Controlled Piecewise Affine Histogram Equalization

José Luis Lisani[1], Ana Belén Petro[2], Catalina Sbert[3]

[1]University of Balearic Islands (Spain) (`joseluis.lisani@uib.es`)
[2]University of Balearic Islands (Spain) (`anabelen.petro@uib.es`)
[3]University of Balearic Islands (Spain) (`catalina.sbert@uib.es`)

Communicated by Pascal Getreuer   *Demo edited by* Jose-Luis Lisani

Abstract

This paper presents a simple contrast enhancement algorithm based on histogram equalization (HE). The proposed algorithm performs a piecewise affine transform of the intensity levels of a digital image such that the new histogram function will be approximately uniform (as with HE), but where the stretching of the range is locally controlled to avoid brutal noise enhancement. We call this algorithm Piecewise Affine Equalization (PAE). Several experiments show that, in general, the new algorithm improves HE results.

Source Code

The proposed algorithm has been implemented in ANSI C. The source code, the code documentation, and the online demo are accessible from the web page of this article[1].

Keywords: contrast enhancement, histogram modification

1 Overview

Contrast enhancement remains one of the most important issues in image processing. Among all image processing techniques, it is the one that has the strongest impact on image quality. Many contrast enhancement techniques have been introduced to improve the contrast of an image but there seems to be no universal method for all applications, simply because the kind of correction depends on the scene.

The most popular global contrast enhancement is the parameterless global histogram equalization (HE) [1]. HE applies a nonlinear transform to the intensity levels so that the cumulative distribution function $F(x)$ becomes closest to linear, i.e. $F(x) \simeq ax$, with a depending on the dimension of the image and the range of values. The local method proposed by Stark [2] modifies the cumulative distribution function of the histogram to adjust the level of enhancement. Arici et al. [3] propose

---

[1]https://doi.org/10.5201/ipol.2012.lps-pae

a global method to modify the histogram by solving an optimization problem. The new histogram is a weighted average of the input histogram and of the desired uniform one. Probably the most sophisticated local histogram adjustment was proposed by Caselles et al. [4], where the contrast change is made conditionally to connected components of level sets. We call this method local histogram equalization (LHE). In these methods, noise can be exaggeratedly enhanced in some regions.

The proposed algorithm is based on a simple global controlled histogram modification. To that effect, natural bounds are respected to limit the SNR, local degradation, and color attenuation. Contrast enhancement is always the result of a compromise. Indeed, image colors are constrained to belong to a fixed range. Thus, stretching the contrast in one part may result in squeezing it in another. This also explains why methods cannot be fully automatic. The method introduced herewith will be no exception, but will be limited to at most three parameters (in practice only two, as will be justified in the next sections). The first idea for a global method is to perform a piecewise linear transform of the intensity levels such that the new cumulative distribution function will be approximately linear, but where the slope of the contrast change is locally controlled to avoid brutal noise enhancement. We can consider other natural choices (smoother splines, nonlinear contrast changes, partitioning by the main modes of the histogram, ...), but our goal is to limit the slope of the contrast change with a simple construction, and the proposed piecewise affine transformation achieves it. Other choices involve a supplementary work, and the simplicity is not guaranteed, for example, using an optimization technique to find the best partition of the histogram would remove a parameter but the computational cost would be higher.

In reference to the parameters, the slope of the local affine contrast change is typically forced to be no larger than 3. In that way, noise cannot be multiplied by a factor larger than 3. The control from above of the contrast change slope also prevents an excessive enhancement of the quantization noise. When this slope is excessive, the fact that the image is piecewise constant by its quantization is revealed. Some level lines of the image are revealed as though they were contrasted boundaries.

In most images, with the above mentioned bound (3 for the maximum slope) and with a piecewise affine contrast change with 5 sub-intervals, the result is generally close to optimal and does not generate visible artifacts. When the image is very dark, however, other choices may be judicious, such as 2 for the maximum slope and 10 sub-intervals. Indeed, in the dark parts of dark images a factor 3 may reveal undesirable artifacts. Similarly, if the image is strongly degraded by the compression, JPEG blocks may appear by excessive histogram stretch. This explains why no upper default value can be fixed for the maximal slope of the contrast change.

Some software for image editing, as the GNU Image Manipulation Program (GIMP) [5], allows the user to manually edit a contrast change by selecting a few control points in the histogram. This is a technique which is extremely close to the proposed algorithm. Our algorithm is more automatic and it allows to know the exact contrast function applied to the image.

2   Algorithm

Let $u(i,j)$, $i = 0, \ldots n_x - 1$, $j = 0, \ldots, n_y - 1$, be a digital image with values in the range $[0, 255]$. Let $F : [0, 255] \to [0, 1]$ be its cumulative histogram function, that is,

$$F(x) = \frac{|\{(i,j): \ u(i,j) \leq x\}|}{D}$$

where $D = n_x \times n_y$ is the number of pixels.

For fixed N, we consider a regular partition $0 = y_0 < y_1 < \ldots < y_N = 255$ of the interval $[0, 255]$, with $y_k = 255k/N$, $k = 0, 1, \ldots, N$. We define a new partition $x_k = F^{-1}\left(\frac{y_k}{255}\right)$, $k = 0, 1, \ldots, N$.

As the cumulative histogram function may not be strictly increasing, we compute $F^{-1}(z)$ as the minimum value of x at which $F(x) \geq z$.

For each interval of the new partition $I_k = [x_k, x_{k+1}]$ the algorithm constructs a linear transform $T_k(x)$ such that T_k transforms the interval $[x_k, x_{k+1}]$ into $[y_k, y_{k+1}]$. Thus,

$$T_k(x) = y_k + m_k(x - x_k) \quad k = 0, 1, \ldots, N - 1,$$

where for each of these linear transforms, the slope is

$$m_k = \frac{y_{k+1} - y_k}{x_{k+1} - x_k}.$$

If this slope is too small, the corresponding linear transform can compress the histogram, and consequently is at risk of losing too much contrast. On the other hand, if the slope is too large the transform stretches the histogram and this can produce a noise amplification, particularly in the dark zones of the image. To avoid this effect the value of the slopes must be constrained. We shall define two new parameters, $smin$ and $smax$, and the slope as

$$m_k = \begin{cases} \max(m_k, smin) & m_k < 1 \\ \min(m_k, smax) & m_k \geq 1 \end{cases}.$$

These new values of m_k modify the partition of the interval $[0, 255]$ i.e., the points y_k.

This slope-limited piecewise equalization will be referred to as Piecewise Affine Equalization (PAE) in the sequel. This simple algorithm does not seem to be documented in the literature.

A discussion on the optimal values of the parameters N, $smin$ and $smax$ is carried out in the following sections.

The appropriate number of intervals N of the partition depends on the image. When N is large, the result becomes similar to global histogram equalization. The slope parameters are $smin = 0$ and $smax = 3$ by default, since they give sound bounds to contrast loss and SNR degradation. In the next section, these choices are justified.

The main algorithm of the method writes:

Algorithm 1 Intensity transformation on gray scale images

    input : N, $smin$, $smax$, original image (1 channel)
    output: Contrast changed image
    Compute the cumulative distribution function $F(x)$ of the input image
    Compute the partition points $y_k = \frac{255k}{N}$, $k = 0, 1, \ldots, N$
    $x_k = F^{-1}\left(\frac{y_k}{255}\right)$, $k = 0, 1, \ldots, N$
    for $k = 0, 1, \ldots, N - 1$ **do**
        $m_k = \frac{y_{k+1} - y_k}{x_{k+1} - x_k}$
        if $m_k < smin$ **then**
            $m_k = smin$
        if $m_k > smax$ **then**
            $m_k = smax$
        $y_{k+1} = T_k(x_{k+1})$, with $T_k(x) = y_k + m_k(x - x_k)$
        Transform the values $x \in [x_k, x_{k+1}]$ into $y \in [y_k, y_{k+1}]$ by $y = T_k(x)$

Remark that this algorithm is applied on monochrome images. For RGB color images we considered two options: either apply the algorithm independently on each channel, or apply it to the intensity of the color image without modifying the original $R/G/B$ ratio of the pixels. In the first case, the color of the pixels is modified in the process because each RGB channel is transformed by an

affine function with different parameters. In the second case, we first compute the gray level intensity $(I = (R + G + B)/3)$, then this intensity is transformed into I' by the affine transform. Finally, for each pixel, the three color channels are multiplied by I'/I. This process can be summarized by the following pseudocode:

Algorithm 2 Intensity transformation while preserving $R/G/B$ ratios

---

input : The vector of the gray level intensity I, the vector of the transformed intensity I',
RGB channels of the original image (red_in, $green\_in$, $blue\_in$)
output: RGB channels of the output image (red_out, $green\_out$, $blue\_out$)
foreach *pixel i* **do**

$A = \frac{I'[i]}{I[i]}$　　　　　　　　Compute the amplification factor

$red\_out[i] = A \cdot red\_in[i]$　　　　Compute each color channel
$green\_out[i] = A \cdot green\_in[i]$
$blue\_out[i] = A \cdot blue\_in[i]$
if $red\_out[i] > 255$ *or* $green\_out[i] > 255$ *or* $blue\_out[i] > 255$ **then**
　　$B = \max(red\_in[i], green\_in[i], blue\_in[i])$
　　$A = \frac{255}{B}$
　　$red\_out[i] = A \cdot red\_in[i]$
　　$green\_out[i] = A \cdot green\_in[i]$
　　$blue\_out[i] = A \cdot blue\_in[i]$

Other choices for the management of color images are possible, depending on how we define the "color" attribute to be maintained (hue, chroma, R/G/B ratio) and what we want to correct with the algorithm (lightness, brightness, intensity, luma, ...). A discussion about these color correction variants will be published in a later article, and we present hereafter the simplest version.

3 Parameter Tuning

We start by applying the algorithm to gray scale versions of some color images, in order to assess the effect of the contrast change independently from their color content.

In all examples, the original and processed images are shown together with their histograms. The result of global histogram equalization (HE) is also displayed for reference. The scale of the histograms is set to the histogram size of the input image. Black bars in the histograms indicate overflows and truncated bins. In order to reduce quantization artifacts and show smoother densities we display the histograms after zooming the images by a factor of 10 by bilinear interpolation.

The contrast change between the original and processed images is also displayed. In the figures, the meaning of the colors is as follows: the linear transforms T_k, described in the *Algorithm* section, are displayed in blue and the partition points (x_k, y_k) are marked as blue dots. The horizontal axis represents original values and the vertical axis new values. The identity transform is displayed with a red line for reference. Finally, a green curve represents the "empirical" transform, T_e, between the images. That is, $y = T_e(x) = F_2^{-1}(F_1(x))$, where x and y are the original and final values and F_1, F_2 are the cumulative histogram functions of the original and processed images, respectively. As the cumulative histogram function may not be strictly increasing, we compute $F_2^{-1}(z)$ as the minimum value of x at which $F_2(x) \geq z$. Remark that when T_e and T_k coincide, only the blue curve is displayed.

An extended version of these examples is available in the web page of this article[2].

---
[2]https://doi.org/10.5201/ipol.2012.lps-pae

3.1 Number of Subintervals of the Partition, N

In the first examples (figures 1 to 3) we do not limit the slope of the linear transform T_k (that is, we set $smin = 0$, $smax = 255$). We test the algorithm with increasing values of N (2, 3 and 5) and we compare the results with HE.

Remark that when the slope is not bounded, N controls how the input gray levels will be mapped in the output image: with $N = 2$ the darker $1/2$ of the image values are linearly mapped from 0 to 128 and the brighter $1/2$ from 128 to 255; with $N = 3$ the darker $1/3$ of the image values are linearly mapped from 0 to 85, the brighter $1/3$ from 170 to 255, and the intermediate values from 85 to 170; and so on. Moreover, we observe that in this case, PAE is just a piecewise linear approximation of the HE transform. In fact, when $N \geq 5$, PAE and HE are very similar and exhibit the same saturation problems. In particular, in figure 2 we test the algorithm with large values of N (10, 100, 1000) and show that the results are identical to HE.

There is no a priori optimal choice for N since the best value depends on the image. For instance, in figure 1 $N = 2$ seems to give a better balance in the contrast of dark and bright regions than higher values of N, while in figure 3 $N = 5$ gives the best result.

3.2 Minimum Slope, $smin$

If we just limit the minimum value of the slope (for example, we set $smin = 0.5$ or $smin = 1$) we observe that some flat regions in the HE transform are replaced by steepest regions in PAE. As a result, some values that were squeezed by HE are better contrasted with our algorithm. However, it must be remarked that this improvement is hardly perceived, since the number of pixels with values in flat zones of the HE transform is small[3].

Moreover, when we force the slope to be at least equal to $smin$ in the flat zone, we decrease the dynamic range of the output for the values above the flat zone (see figure 4). As a consequence, some values become excessively saturated, while HE gives better contrasted results. This can be seen in the wing of the swan in figure 5 (both with $smin = 0.5$ and $smin = 1$). For this reason it is better to keep $smin$ as low as possible. We have decided to set $smin = 0$ in all our experiments.

3.3 Maximum Slope, $smax$

The maximum allowed slope of the linear transforms T_k controls the degree of saturation allowed in the output image. In the following examples we display different results of the PAE algorithm with increasing values of $smax$ (2, 3 and 5). We compare the results with HE. In all the tests we have fixed $N = 5$ and $smin = 0$.

The optimal value of $smax$ depends on the input image. For instance, in figure 6 it seems that $smax = 2$ provides a good compromise between contrast enhancement and saturation, while in figure 7 the best results are obtained with $smax = 5$.

In figure 8 we test the effect of limiting the maximum slope $smax$ while increasing N. In figure 2 we have seen that if the slope is not limited ($smax = 255$) PAE and HE are identical when N is large. Here we repeat the experiment with $smax = 3$. We observe that the three results of PAE are almost identical but different from HE since the limitation of the slope prevents an excessive enhancement of the dark zones.

---

[3] Assuming that histogram equalization is performed in the range $[0, 255]$, the HE transform is $y = 255F(x/255)$, where x and y are the input and output values, respectively, and F is the cumulative histogram function of the input image. Consider the case where $x_2 > x_1$. Since $F(x)$ is a non-decreasing function we will have $y_2 \geq y_1$. If it turns out that $y_2 \approx y_1$ (flat region in the transform function) it means that $F(x_2/255) \approx F(x_1/255)$, that is, the number of pixels with value in the range $(x_1, x_2]$ is small.
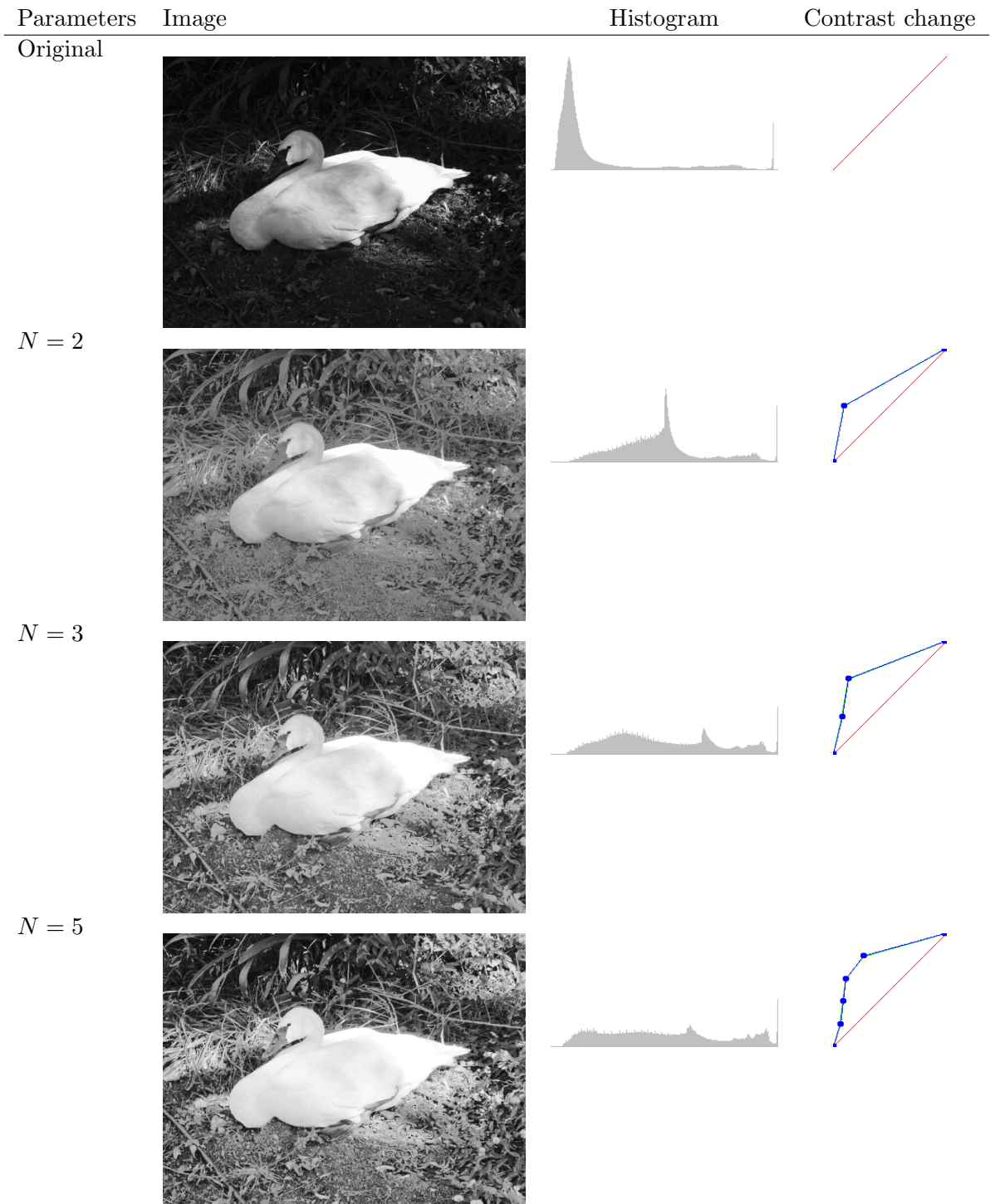
Parameters	Image	Histogram	Contrast change
Original			
$N = 2$			
$N = 3$			
$N = 5$			

Figure 1: "Swan" example, with increasing values of N ($smin = 0$, $smax = 255$)

Parameters	Image	Histogram	Contrast change
Original			
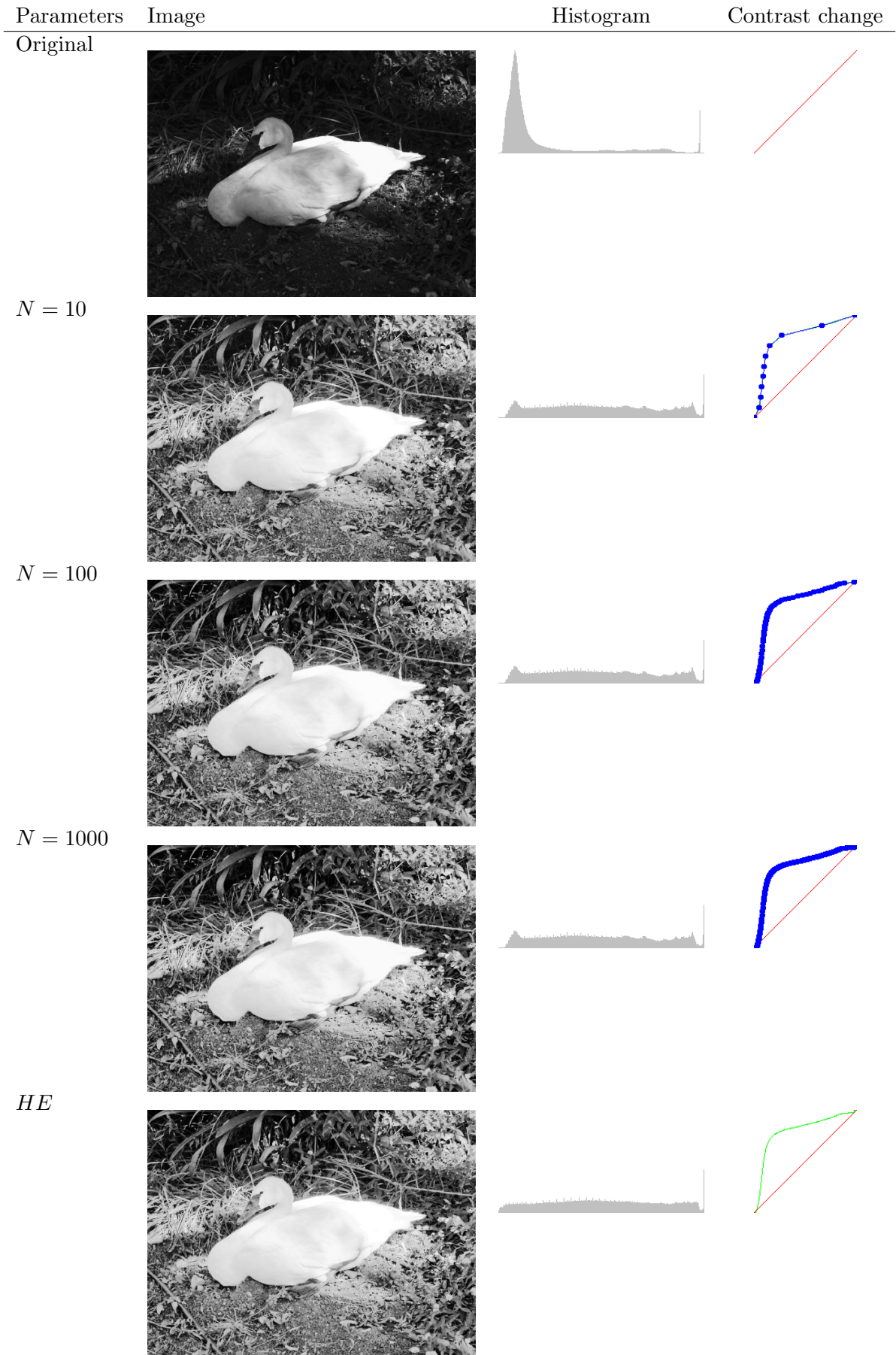
Figure 2: "Swan" example, with large values of N ($smin = 0$, $smax = 255$)

Parameters	Image	Histogram	Contrast change
Original			
$N = 2$			
$N = 3$			
$N = 5$			
HE			
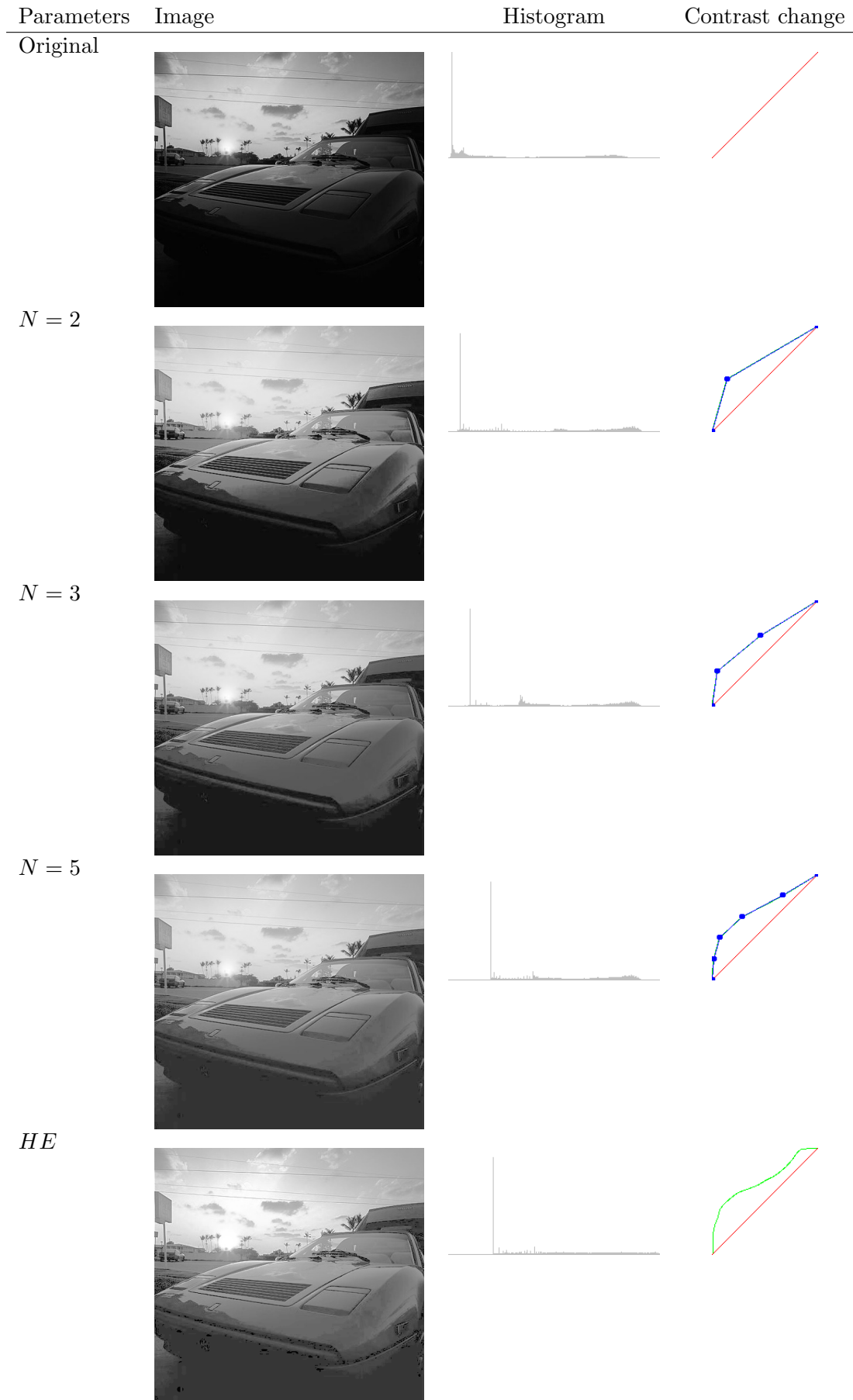
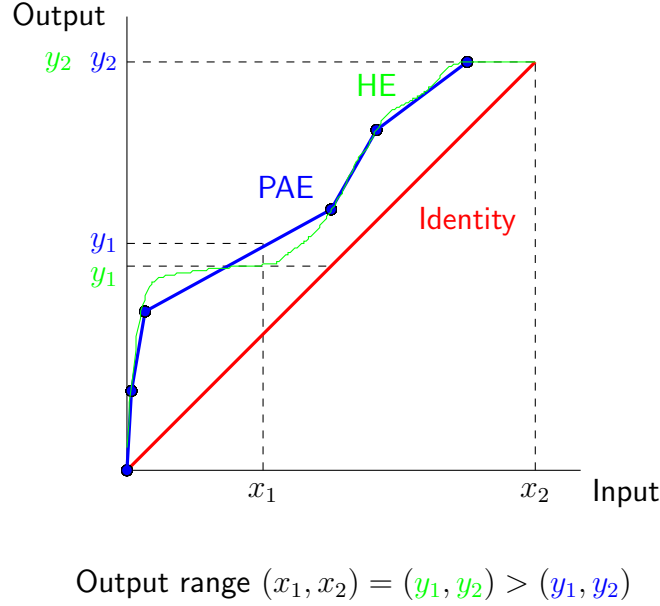Figure 3: "Ferrari" example, with increasing values of N ($smin = 0$, $smax = 255$)

Figure 4: Comparison of PAE (N=5, smin=0.5, smax=255) and HE transforms

3.4   Color Images

When dealing with RGB color images, we considered two options for the application of the algorithm: either to process the channels independently, or to process the intensity component (i.e. the average of R, G and B) and then compute the output RGBs from the new intensity value, with the restriction that the ratios $R/G/B$ must be kept.
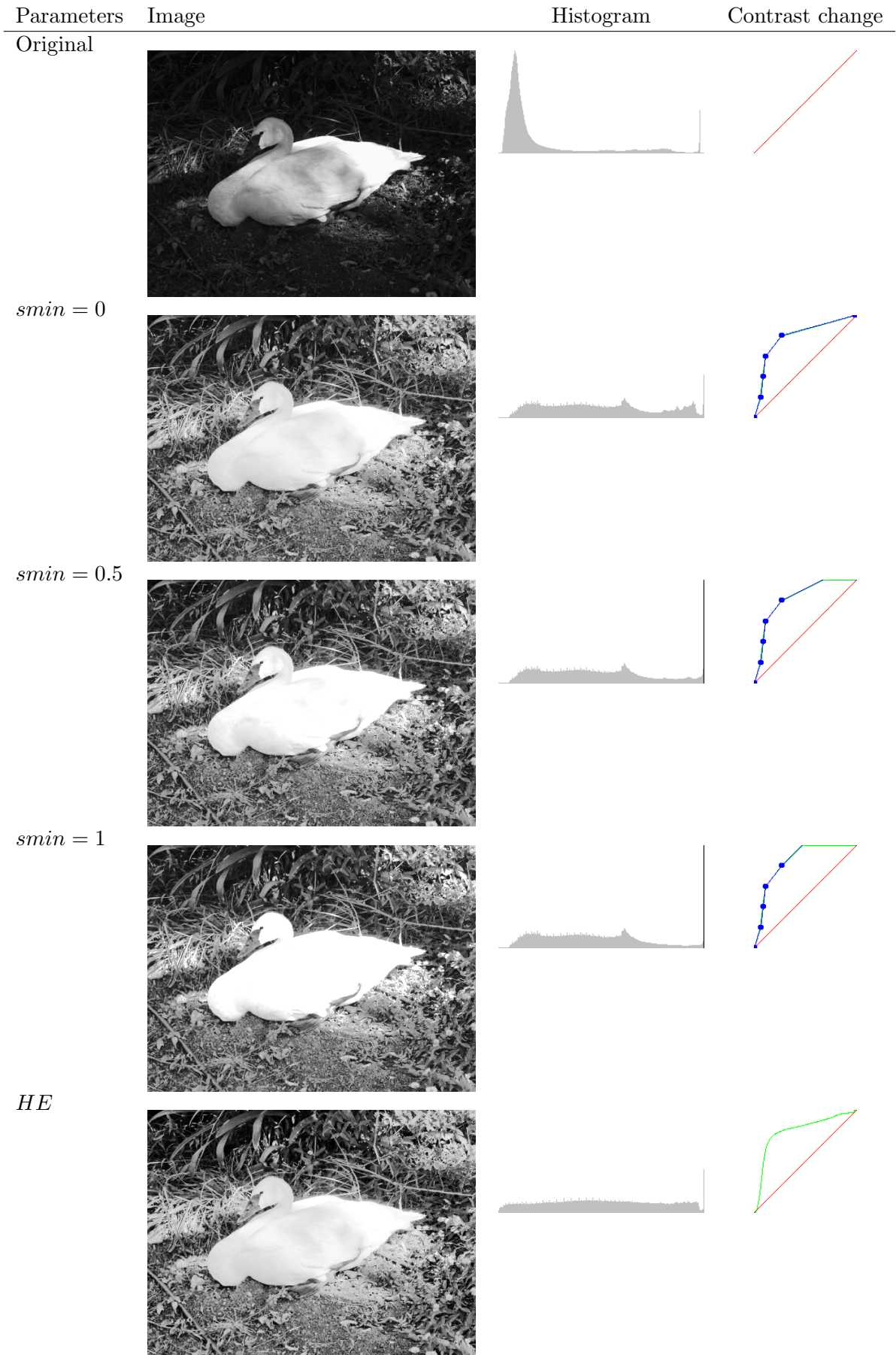
In the first case, the original image colors are modified (i.e. their chromatic components, Hue and Saturation, are changed). In particular, with HE we achieve both contrast enhancement and white balance, since the maximum value of each channel is stretched to 255. With PAE, due to the limitation in the slope, we do not get a white balanced image, but a close approximation, closer as $smax$ increases (see figure 9).

If the intensity component is processed, ideally the original colors are kept, since their chromaticity (understood as the $R/G/B$ ratio) is preserved. In practice, however, it is not possible to achieve an arbitrary (between 0 and 255) intensity value for any fixed $R/G/B$ ratio, since for some intensity value some of the channels will become saturated. For instance, consider the case of a pixel with $RGB = (30, 50, 130)$. Its intensity is $I = (30 + 50 + 130)/3 = 70$. Let's assume that PAE transforms this intensity value to 140. The corresponding transformed RGB should be $(30 \times 140/70, 50 \times 140/70, 130 \times 140/70) = (60, 100, 260)$, but since there is saturation in the B component the actual output will be $(59, 98, 255)$ (the integer RGB color with maximum intensity that better preserves the original $R/G/B$ ratio without saturation). And the output intensity (that should be 140 according to PAE) is actually $(59 + 98 + 255)/3 = 137.3$.

For these reasons, when PAE is applied to the intensity component of a color image, the original $R/G/B$ ratios are indeed preserved, but the transform that relates the input and output images does not always coincide with the theoretical PAE transform. This can be observed by comparing the green (actual transform) and blue (PAE transform) curves in the "contrast change" image of the displayed results.

Figures 9 and 10 illustrate the differences between independent and $R/G/B$-preserving contrast enhancement, with constant N ($N = 5$) and increasing values of $smax$.

Figures 11 to 14 show the results of applying PAE to the "Swan" and "Ferrari" color images, with

Parameters	Image	Histogram	Contrast change
Original			
$smin = 0$			
$smin = 0.5$			
$smin = 1$			
HE			

Figure 5: "Swan" example, with increasing values of $smin$ ($N = 5$, $smax = 255$)

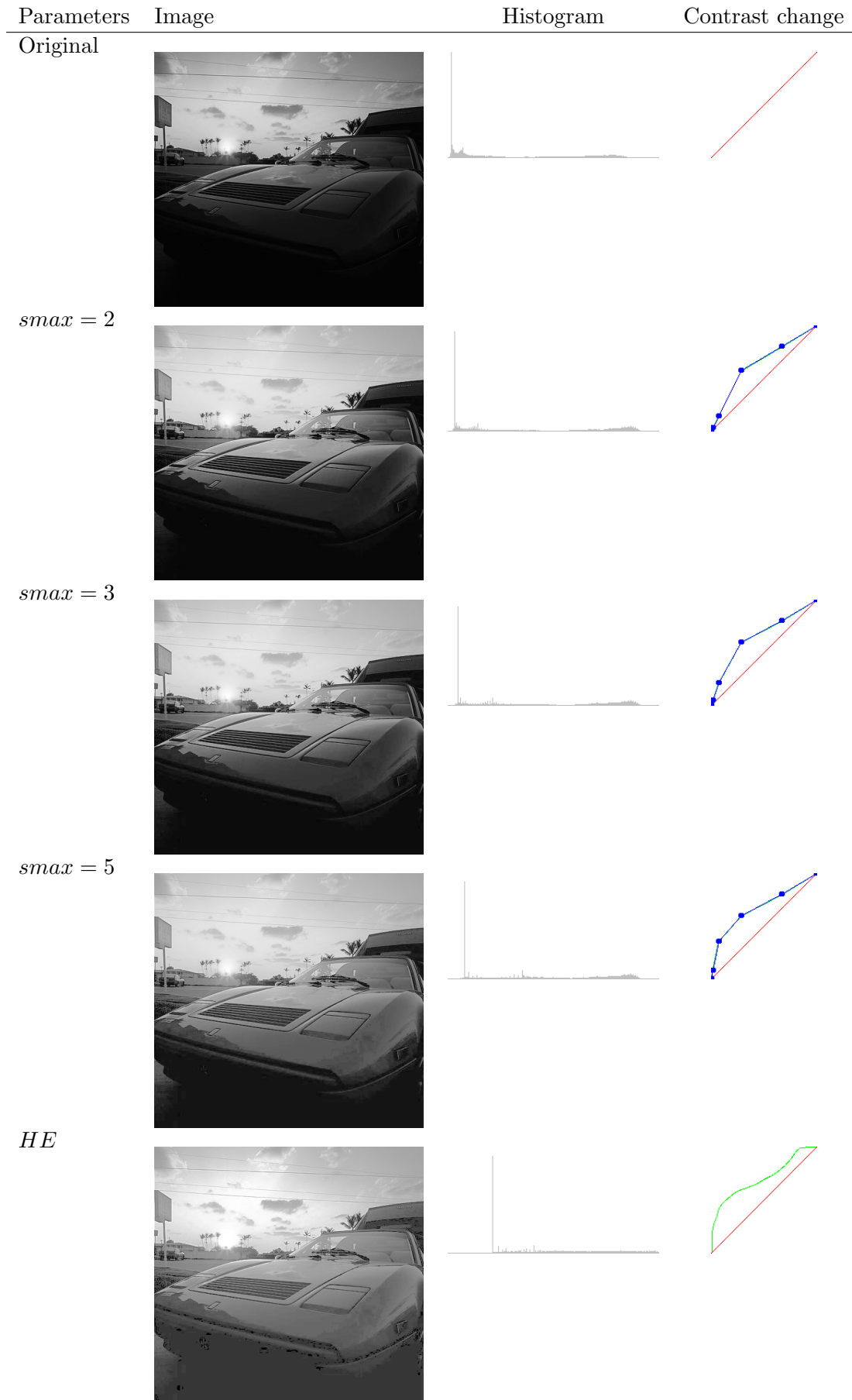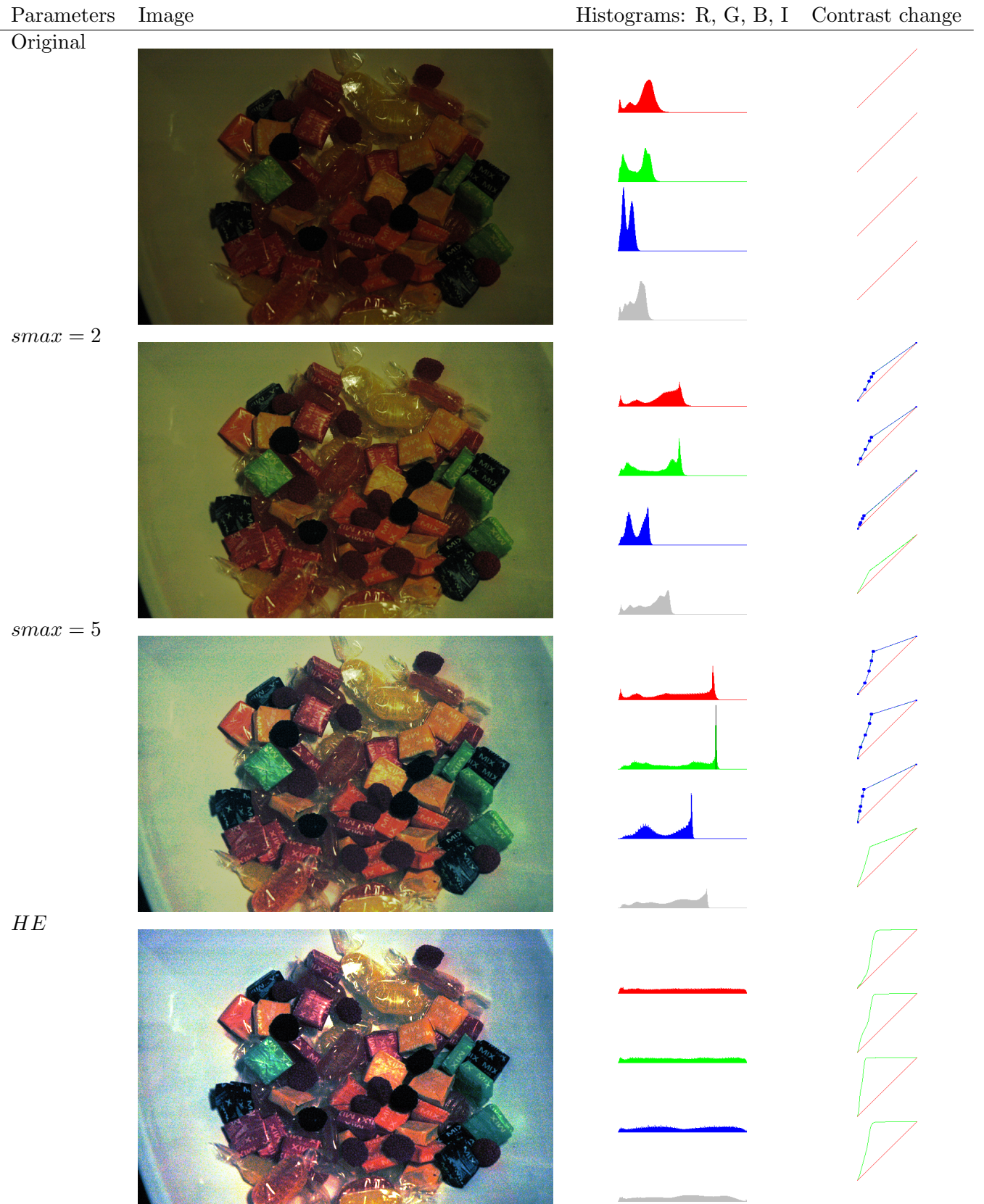| Parameters	Image	Histogram	Contrast change
Original			
$smax = 2$			
$smax = 3$			
$smax = 5$			
HE			
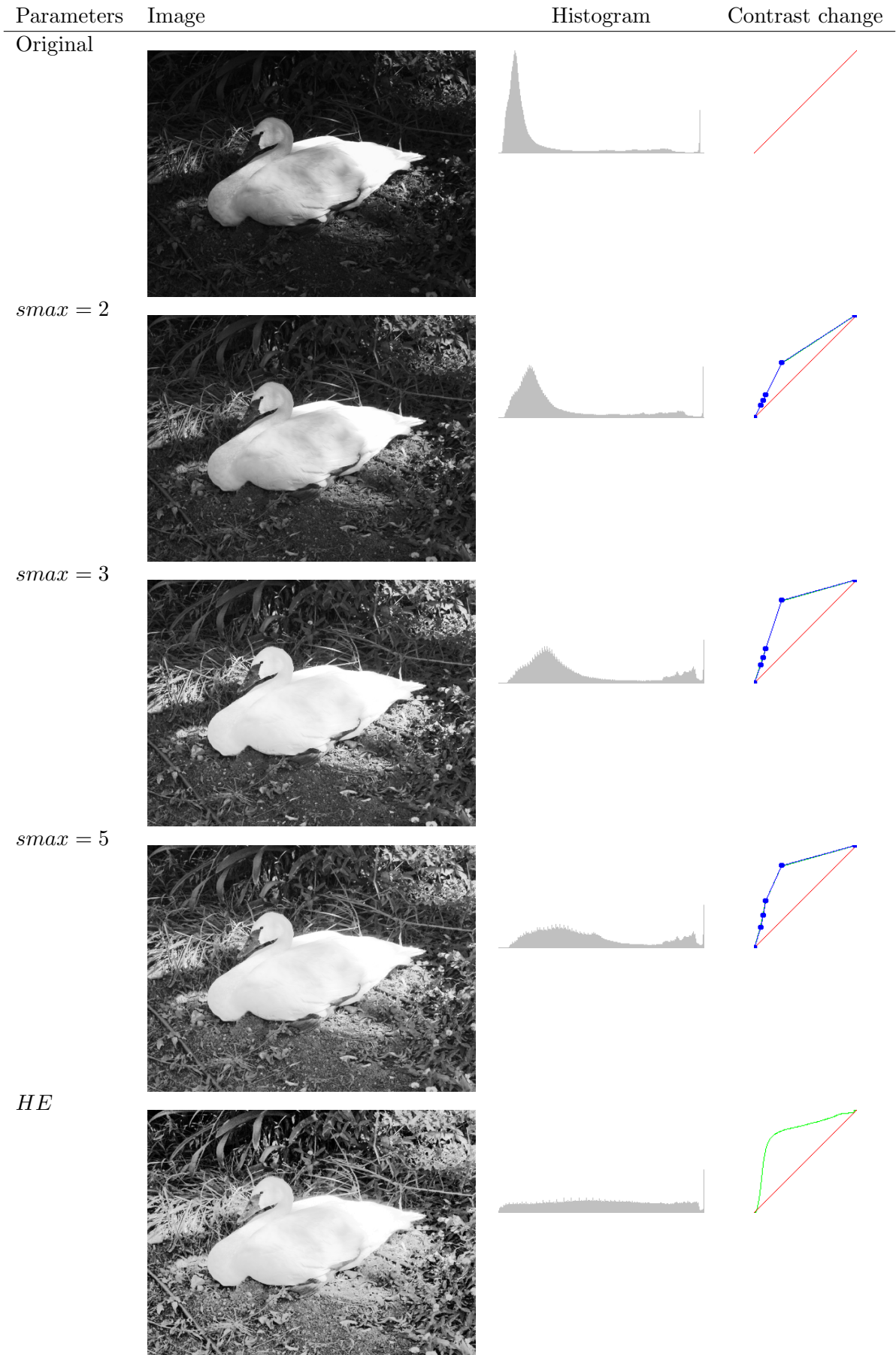
Figure 6: "Swan" example, with increasing values of $smax$ ($N = 5$, $smin = 0$)

Parameters	Image	Histogram	Contrast change
Original			
$smax = 2$			
$smax = 3$			
$smax = 5$			
HE			

Figure 7: "Ferrari" example, with increasing values of $smax$ ($N = 5$, $smin = 0$)

Parameters	Image	Histogram	Contrast change
Original			
$N = 10$			
$N = 100$			
$N = 1000$			
HE			
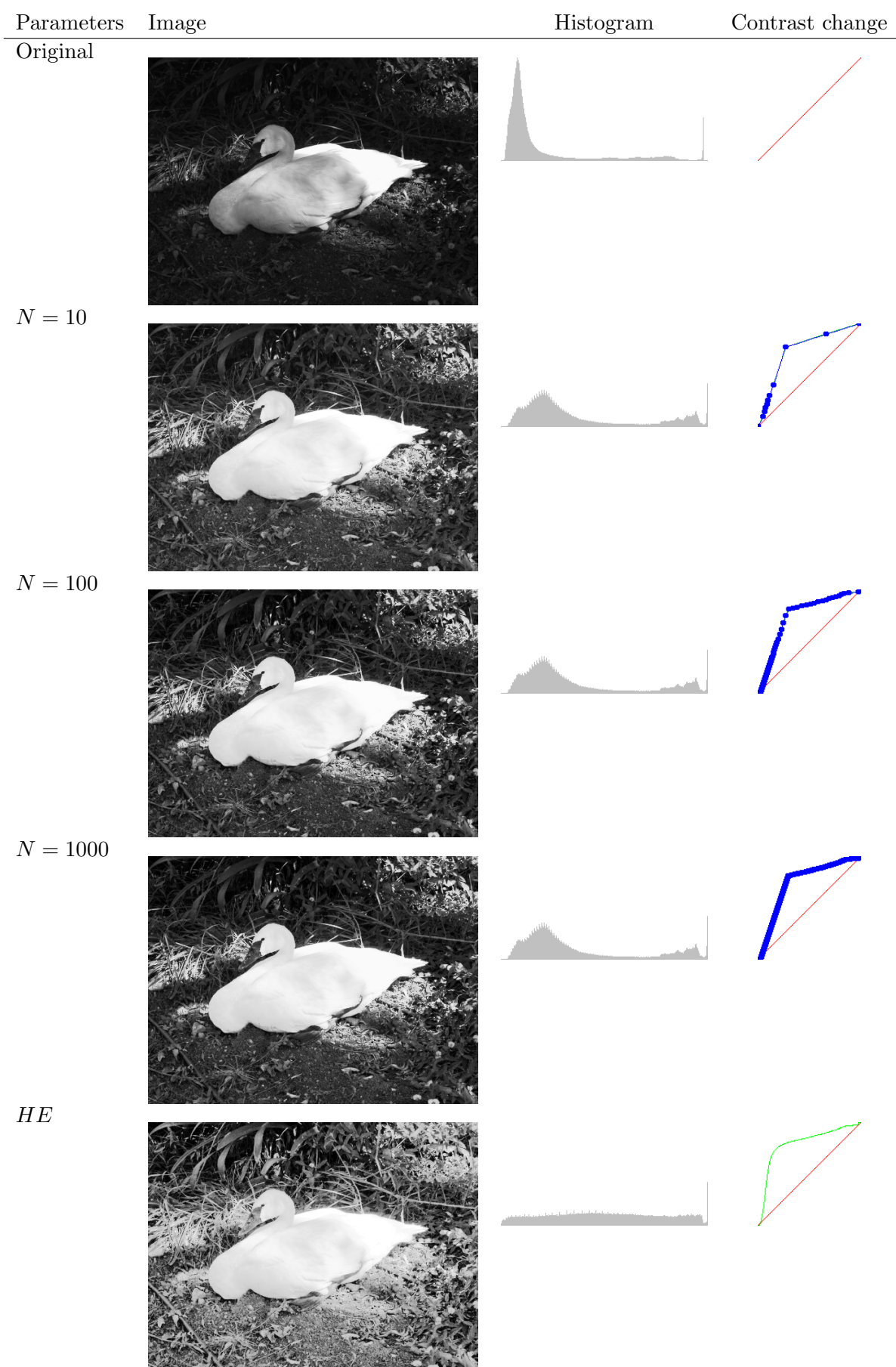
Figure 8: "Swan" example, with $smax = 3$ and large values of N ($smin = 0$)

Parameters	Image	Histograms: R, G, B, I	Contrast change

Original

$smax = 2$

$smax = 5$

HE

Figure 9: "Candies" example, independent processing of RGB channels ($N = 5$, $smin = 0$)

Parameters	Image	Histograms: R, G, B, I	Contrast change
Original			
$smax = 2$			
$smax = 5$			
HE			
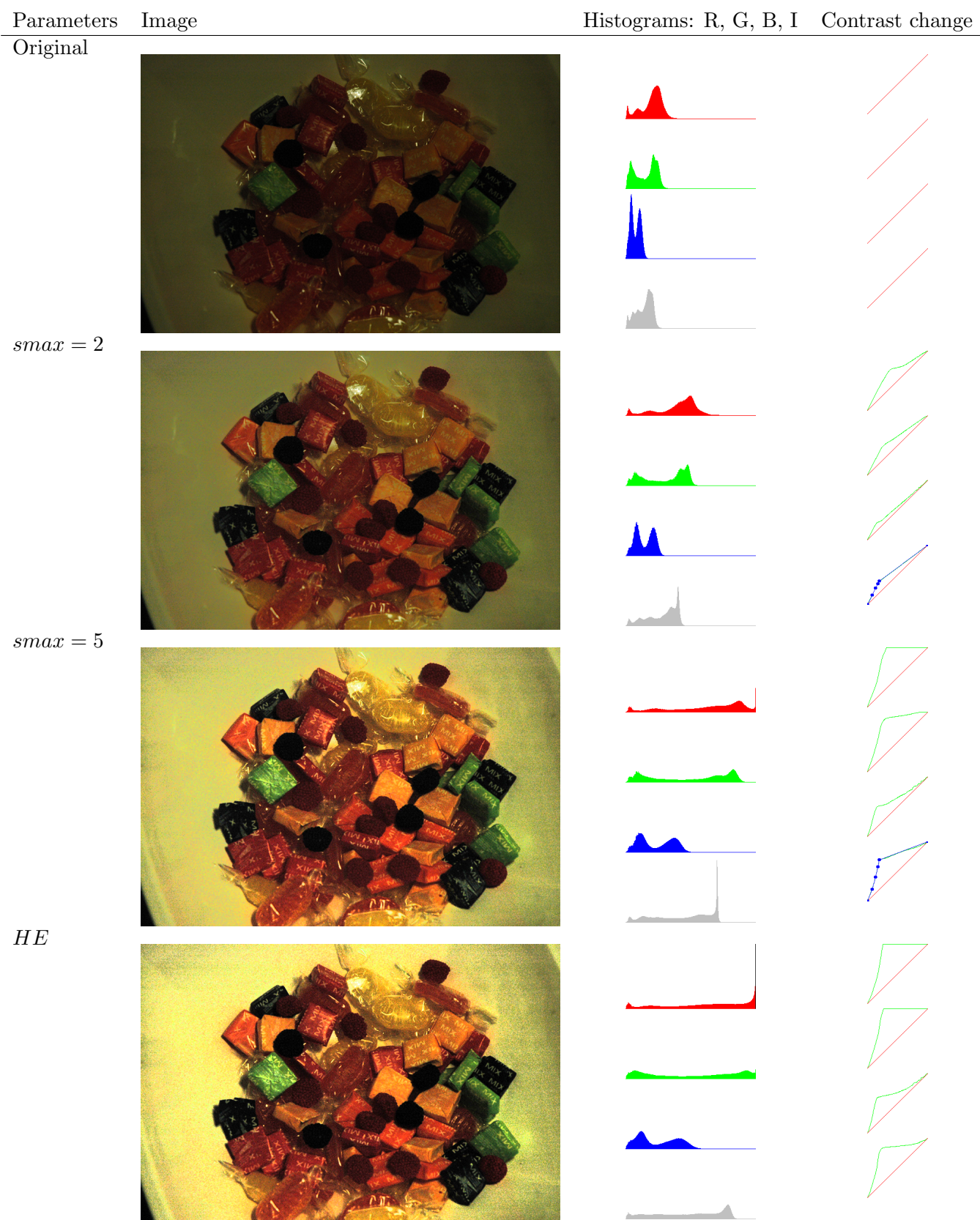
Figure 10: "Candies" example, R/G/B-preserving processing ($N = 5$, $smin = 0$)

257

increasing values of N. The results are compared to HE. In all the examples we use a conservative bound for the slope ($smax = 3$) to prevent the excessive saturation of the colors and the amplification of noise artifacts.

The last example (figures 15 and 16) shows an image where HE performs better than PAE. In this case it is not necessary to limit the slope of the transform since no saturation is produced.

4   Conclusions

The results shown in the previous sections legitimate the proposed PAE algorithm and its default parameters. The choice of a slope-controlled piecewise affine transform (over other image transforms) is justified since it achieves our two main goals, namely, prevention of an excessive contrast enhancement that reveals noise and quantization artifacts, and simplicity. We have also justified that PAE depends basically on two parameters, since the *smin* parameter can be fixed to 0 in all of the experiments.

In general, it has been shown that the results of PAE outperform those of classical global histogram equalization, enhancing dark zones while keeping noise and quantization artifacts under control.

Acknowledgments

Image Credits

References

[1] R. C. Gonzalez and R. E. Woods. *Digital Image Processing. Second edition*, Prentice Hall, 2002. ISBN:0201180758

[2] L. Alex Stark. *Adaptive image contrast enhancement using generalizations of histogram equalization* IEEE Transactions on Image Processing, vol. 9, no. 5, pp. 889–896, 2002. http://dx.doi.org/10.1109/83.841534

[3] T. Arici, S. Dikbas and Y. Altunbasak. *A Histogram Modification Framework and Its Application for Image Contrast Enhancement* IEEE Transactions on Image Processing, vol. 18, no. 9, pp. 1921–1935, 2009. http://dx.doi.org/10.1109/TIP.2009.2021548

[4] V. Caselles, J-L. Lisani, J-M. Morel and G. Sapiro. *Shape preserving local histogram modification*, IEEE Transactions on Image Processing, vol. 8, no. 2, pp. 220–230, 1999. http://dx.doi.org/10.1109/83.743856

---

[4]http://www.nasa.gov/audience/formedia/features/MP_Photo_Guidelines.html

Parameters	Image	Histograms: R, G, B, I	Contrast change

Original

N = 3

N = 5

HE
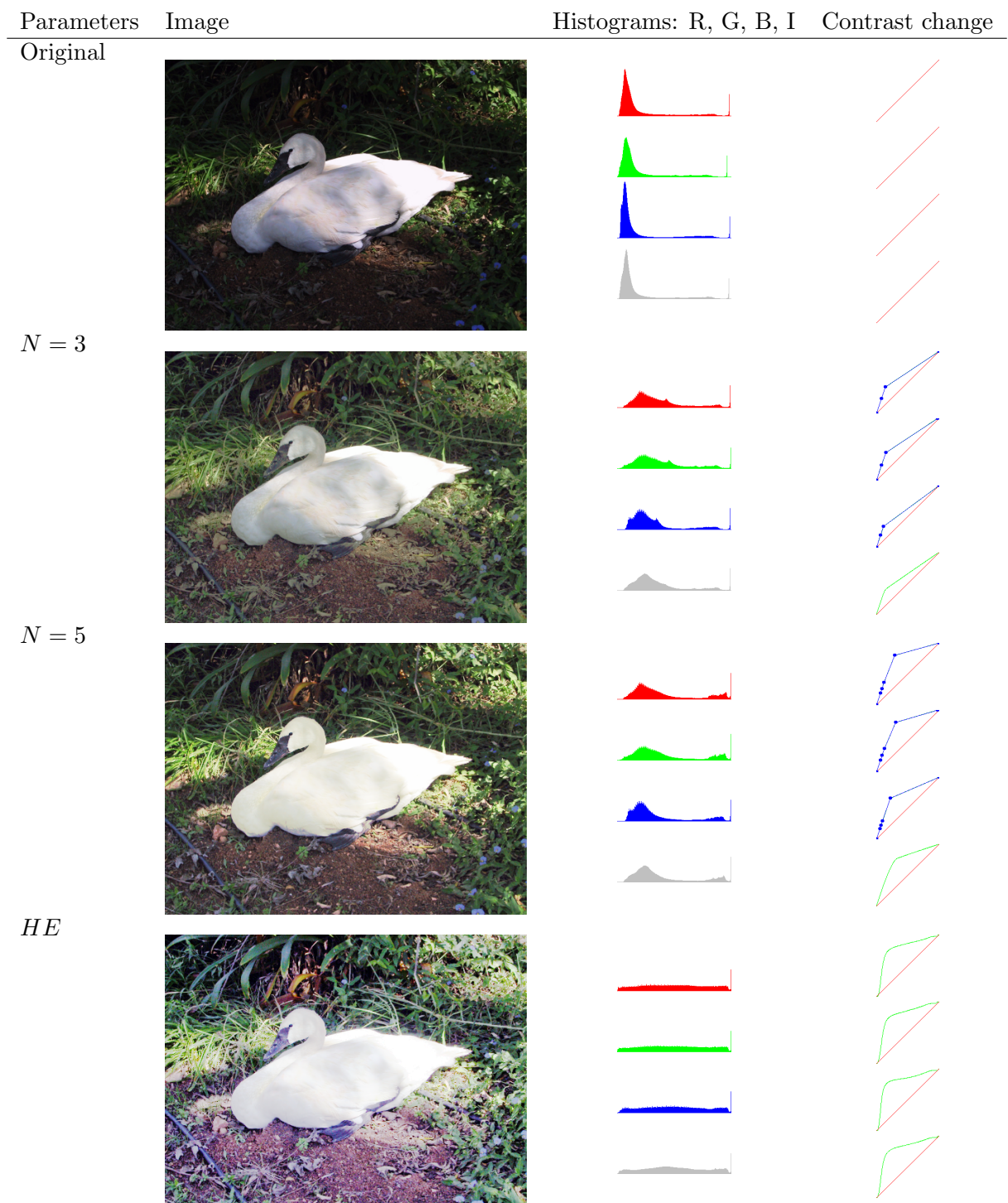
Figure 11: "Swan" example, independent processing of RGB channels ($smin = 0$, $smax = 3$)

Parameters	Image	Histograms: R, G, B, I	Contrast change
Original			
$N = 3$			
$N = 5$			
HE			
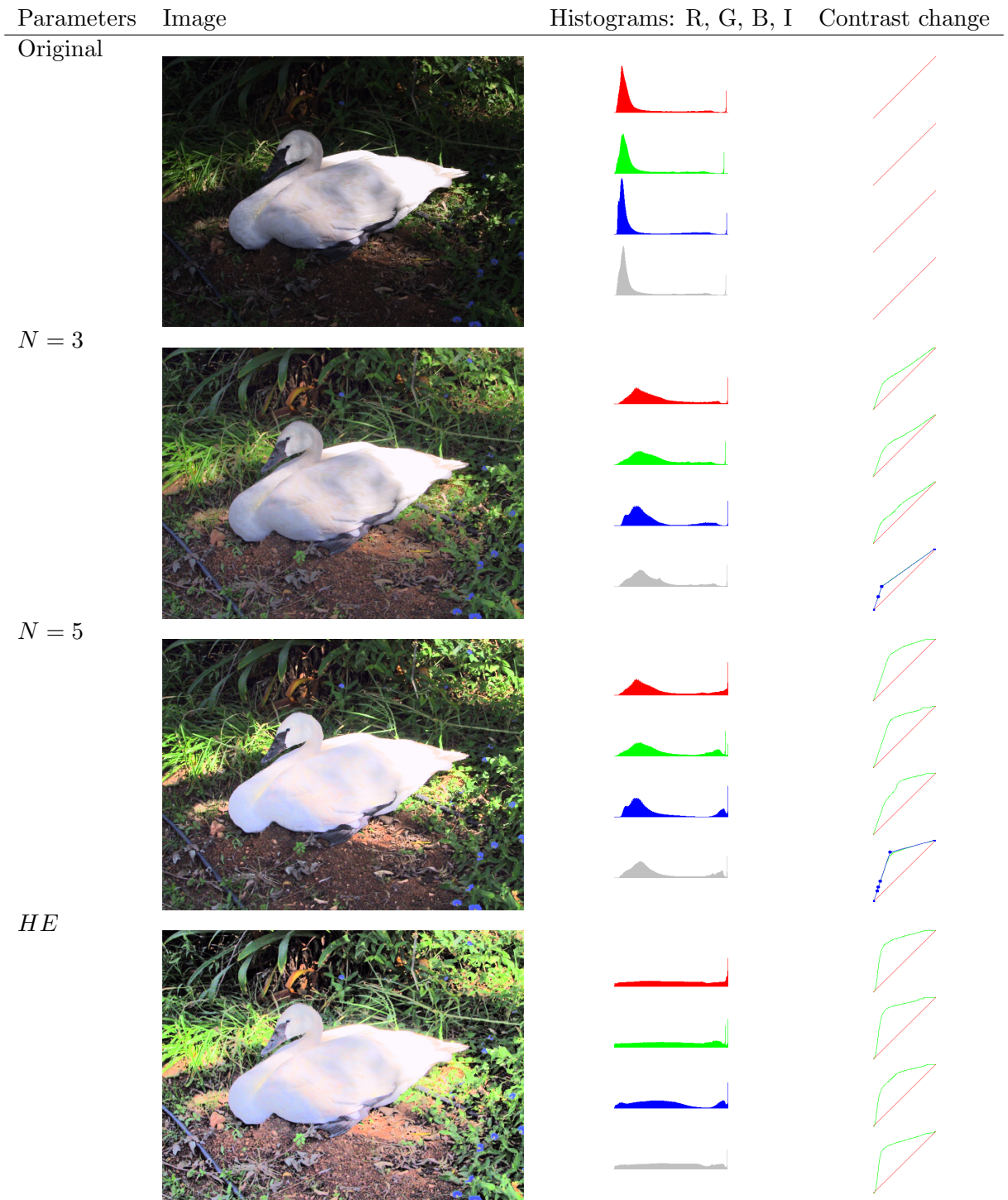
Figure 12: "Swan" example, R/G/B-preserving processing ($smin = 0$, $smax = 3$)

Parameters	Image	Histograms: R, G, B, I	Contrast change

Original

N = 3

N = 5

HE

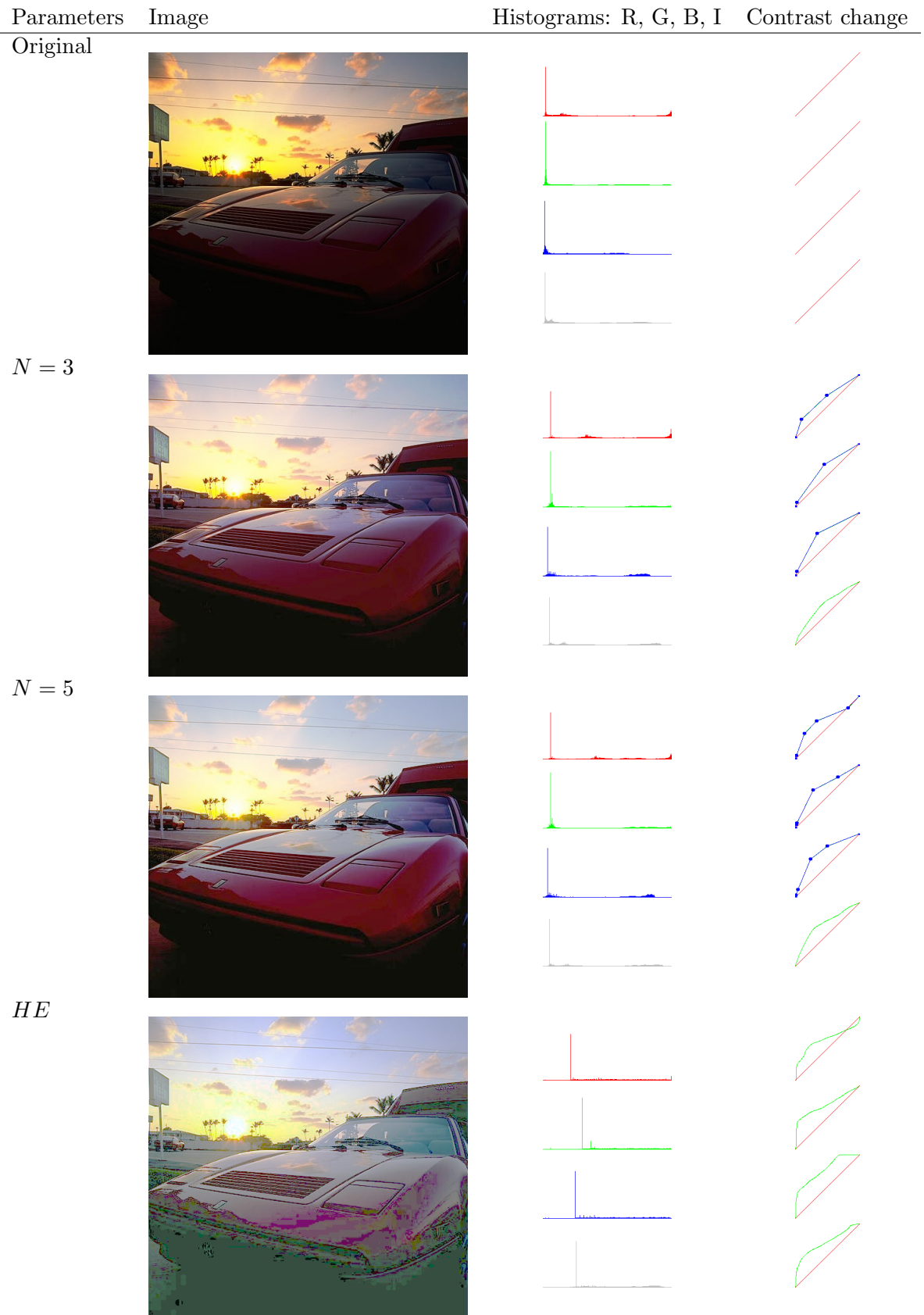

Figure 13: "Ferrari" example, independent processing of RGB channels ($smin = 0$, $smax = 3$)

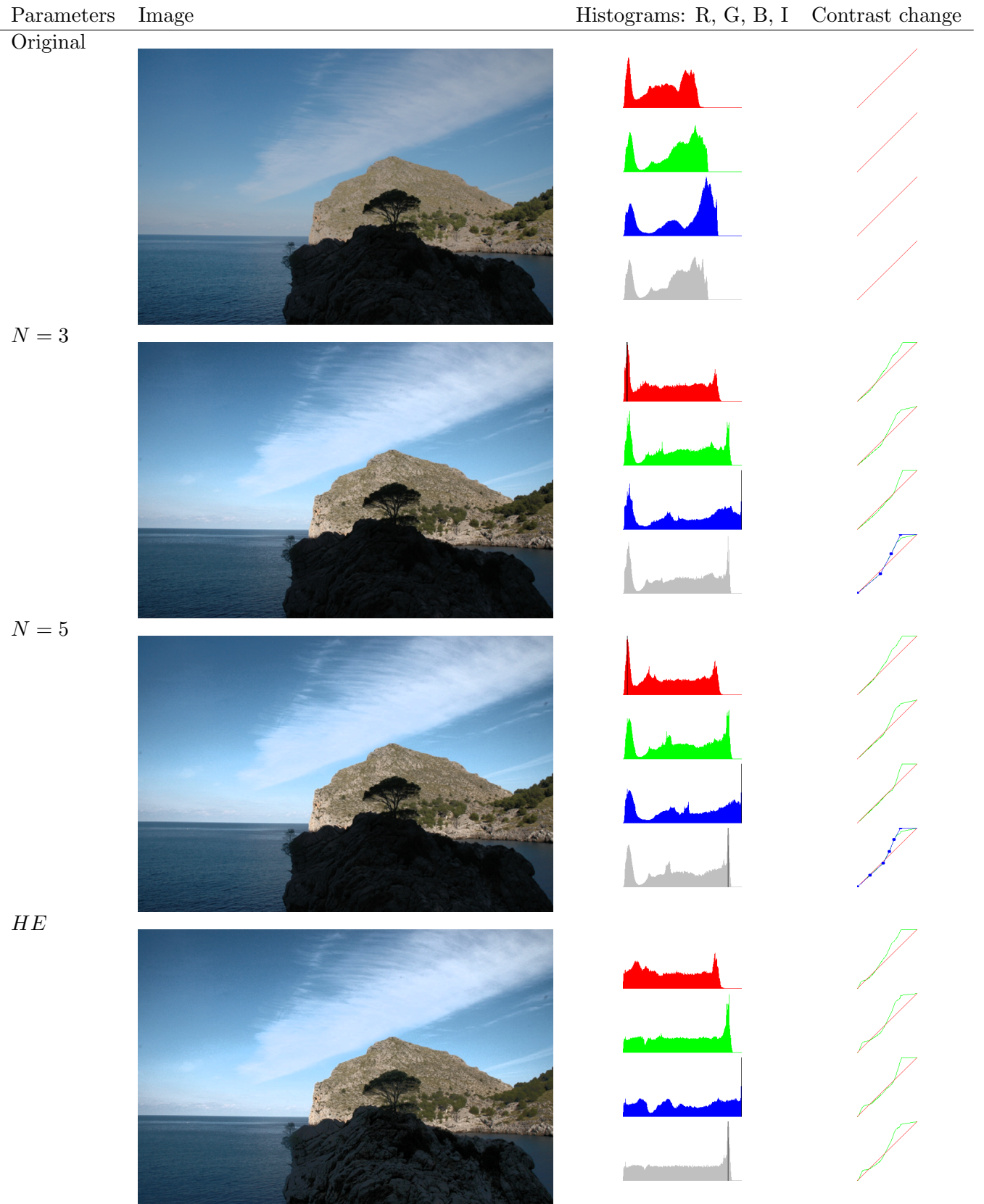| Parameters	Image	Histograms: R, G, B, I	Contrast change
Original			
$N = 3$			
$N = 5$			
HE			
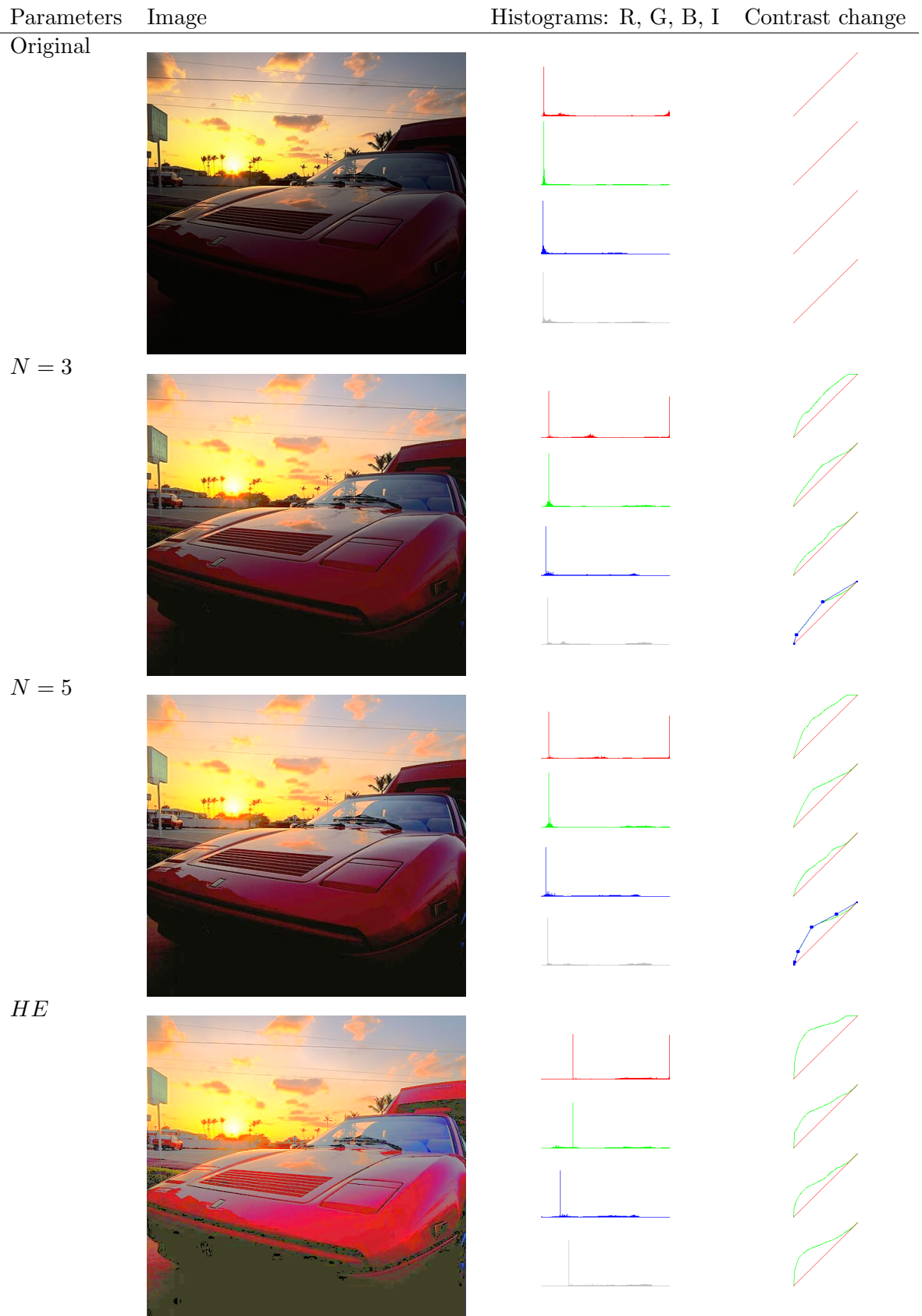
Figure 14: "Ferrari" example, R/G/B-preserving processing ($smin = 0$, $smax = 3$)

Parameters	Image	Histograms: R, G, B, I	Contrast change

Original

$N = 3$

$N = 5$

HE
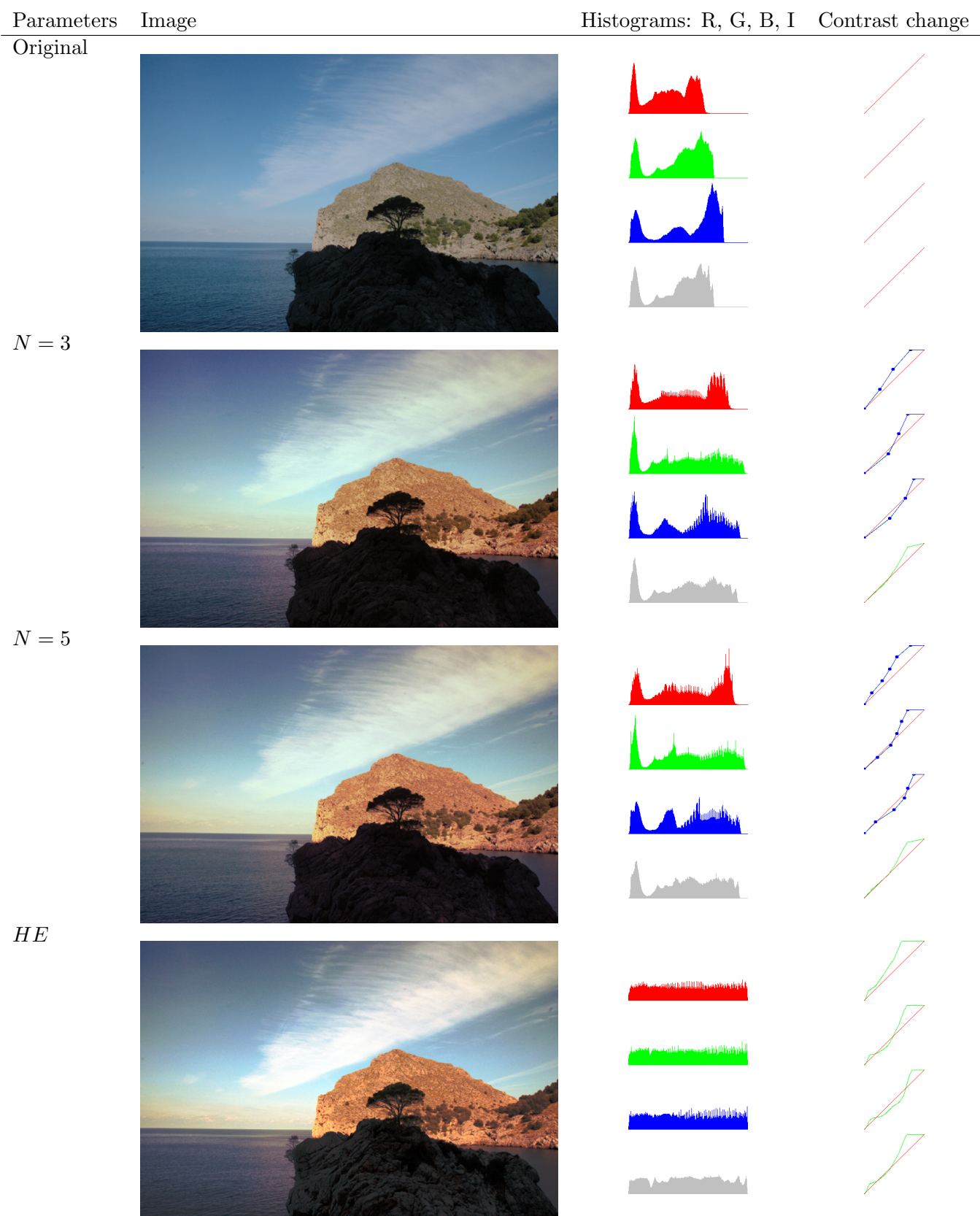
Figure 15: "Calobra" example, independent processing of RGB channels ($smin = 0$, $smax = 3$)

Parameters	Image	Histograms: R, G, B, I	Contrast change

Original

$N = 3$

$N = 5$

HE

Figure 16: "Calobra" example, R/G/B-preserving processing ($smin = 0$, $smax = 3$)

[5] S. Kimball, P. Mattis and the development team *GIMP, GNU Image Manipulation Program.*@2001-2012. http://www.gimp.org/

[6] N. Limare, J-L. Lisani, J-M. Morel, A-B. Petro and C. Sbert. *Simplest color balance*, Image Processing On Line, 2011. http://dx.doi.org/10.5201/ipol.2011.llmps-scb

[7] N. Limare, J-M. Morel, A-B. Petro and C. Sbert. *Retinex Poisson Equation: a Model for Color Perception*, Image Processing On Line, 2011. http://dx.doi.org/10.5201/ipol.2011.lmps_rpe