

Article

# Local Contrast-Based Pixel Ordering for Exact Histogram Specification

Kohei Inoue , Naoki Ono and Kenji Hara

Department of Media Design, Faculty of Design, Kyushu University, 4-9-1 Shiobaru, Minamiku, Fukuoka 815-8540, Japan

\* Correspondence: k-inoue@design.kyushu-u.ac.jp

**Abstract:** Histogram equalization is one of the basic image processing tasks for contrast enhancement, and its generalized version is histogram specification, which accepts arbitrary shapes of target histograms including uniform distributions for histogram equalization. It is well known that strictly ordered pixels in an image can be voted to any target histogram to achieve exact histogram specification. This paper proposes a method for ordering pixels in an image on the basis of the local contrast of each pixel, where a Gaussian filter without approximation is used to avoid the duplication of pixel values that disturbs the strict pixel ordering. The main idea of the proposed method is that the problem of pixel ordering is divided into small subproblems which can be solved separately, and then the results are merged into one sequence of all ordered pixels. Moreover, the proposed method is extended from grayscale images to color ones in a consistent manner. Experimental results show that the state-of-the-art histogram specification method occasionally produces false patterns, which are alleviated by the proposed method. Those results demonstrate the effectiveness of the proposed method for exact histogram specification.

**Keywords:** exact histogram equalization; exact histogram specification; Gaussian filter; contrast enhancement; pixel ordering



**Citation:** Inoue, K.; Ono, N.; Hara, K. Local Contrast-Based Pixel Ordering for Exact Histogram Specification. *J. Imaging* **2022**, *8*, 247. <https://doi.org/10.3390/jimaging8090247>

Academic Editors: Rafael Huertas, Samuel Morillas and Raimondo Schettini

Received: 29 June 2022

Accepted: 8 September 2022

Published: 10 September 2022

**Publisher's Note:** MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



**Copyright:** © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

Histograms represent the distribution of pixel values in images, and can offer various image statistics [1] to us, i.e., histograms have useful information for image processing applications such as image enhancement, compression, and segmentation [2–4]. Therefore, histograms are the basis for numerous spatial domain processing techniques [5], which include intensity transformation as a special case where the smallest neighborhood of size  $1 \times 1$  is used. Histogram equalization, which theoretically transforms an input image into the corresponding output image having a uniform histogram, is one of the intensity transformation techniques for contrast enhancement [5]. More generally, the method for generating an output image that has a specified target histogram is called histogram specification or histogram matching [5], in which histogram equalization is included as a special case [6]. It is pointed out that histogram equalization may cause over enhancement [7]. In such situations, histogram specification will be a good candidate for substitute methods because it accepts any shape of histograms. In this paper, although we would like to use a Gaussian distribution constantly for the target histogram in histogram specification to simplify our explanation, other choices such as bimodal or multimodal distribution may work well.

For digital images that are composed of pixels with quantized values, it is usual that the number of pixels in an image is greater than the number of possible quantized pixel values. Therefore, there are a large number of pixels that share the same quantized pixel value in a digital image, which means that exact histogram specification for digital images is an ill-posed problem [8]. For the effective utilization of all possible quantized pixel

values and the suppression of the occurrence of false contours, a number of research works have been conducted for more than four decades since Hall published his work on discrete distribution transformation [9]. In addition to these technical needs, it is our genuine curiosity to make and see exactly histogram-equalized or specified images because the solution of the ill-posed problem is not unique. A promising approach to exact histogram specification is to obtain a meaningful strict ordering of all pixels in an image. Once we obtain the ordering of pixels, we can assign a limited number of pixel values to the ordered pixels so that the resultant histogram coincides with a specified target histogram.

The simplest way of pixel ordering is random ordering [10]. However, the random ordering is not a meaningful ordering. Coltuc et al. [11] described the process of exact histogram specification when a strict ordering is given, and proposed an ordering method, which is called the local mean method (LM), based on a filter bank of multiple moving average filters. Wan and Shi [12] proposed a wavelet-based exact pixel ordering algorithm (WA) for both exact histogram specification and image enhancement, which takes into account not only local mean intensity values, but also local edge information. Nikolova and Steidl [13] proposed a fast ordering algorithm for exact histogram specification based on a variational approach, which is equivalent to an iterative nonlinear filtering, and demonstrated that their variational ordering method outperforms both LM [11] and WA [12]. They also applied their algorithm to hue and range preserving enhancement of color images [14], where the histogram of an intensity image is exactly equalized or specified, and then an affine transform is applied to each color in a hue and range preserving manner.

In this paper, we propose a local contrast-based pixel ordering method for exact histogram specification, which computes the local contrast of each pixel by using a Gaussian filter [15] without approximation, and compare the proposed method with the state-of-the-art method by Nikolova and Steidl [13]. Furthermore, the proposed method for grayscale images is extended to that for color images in a consistent manner. Experimental results demonstrate that Nikolova and Steidl's method can produce false patterns in flat areas of images, which are alleviated by the proposed method. For color images, it is experimentally shown that the proposed method outputs exactly histogram-equalized or specified images with closer hue to original images than that obtained by a separable method that utilizes Nikolova and Steidl's method for exact histogram equalization and specification of each color channel. These results demonstrate the effectiveness of the proposed method compared with the state-of-the-art method.

The rest of this paper is organized as follows: Section 2 provides a brief review of some related works. Section 3 briefly summarizes conventional histogram equalization. Section 4 summarizes conventional histogram specification and Nikolova and Steidl's exact histogram specification [13]. Section 5 proposes a local contrast-based exact histogram specification method, which is extended to that for color images. Section 6 shows experimental results, where the proposed method is compared with the state-of-the-art methods by Nikolova and Steidl [13] and Ramos et al. [16]. Finally, Section 7 concludes this paper.

## 2. Related Work

In this section, we review recent related works on histogram equalization and specification. Recently, Pallavi et al. [17] reviewed state-of-the-art techniques for image enhancement, in which the techniques of histogram equalization are also included such as contrast limited adaptive histogram equalization (CLAHE) [18] and brightness preserving bi-histogram equalization (BBHE) [19]. Trongtirakul and Agaian [20] proposed a weighted histogram equalization using entropy of probability density function, which outperformed related methods including a low-rank regularized retinex model (LR3M) [21]. More recently, Zhang et al. [22] proposed an unsupervised low-light image enhancement method via a histogram equalization prior (HEP) based on the observation that the feature maps of histogram equalization enhanced image and the ground truth are similar.

Hussain et al. [23] proposed a locally transformed histogram-based technique for dark image enhancement, which does not get affected from the over-enhancement problem. Balado [24] examined an optimum exact histogram specification problem and the inverse problem, and presented their closed-form performance analyses. Ramos et al. [16] proposed two algorithms for histogram specification and quantile transformation of data without local information. As one of the state-of-the-art algorithms for histogram specification, we implemented the first algorithm of Ramos et al. in this paper as described below. However, their algorithm does not produce an output image having the histogram being exactly identical to the target one.

The above review of recent related works draws a conclusion that Nikolova and Steidl’s algorithm [13] is still one of the state-of-the-art algorithms for exact histogram specification.

### 3. Histogram Equalization

Let  $F = [f_{ij}]$  be a grayscale image for  $i = 1, 2, \dots, m$  and  $j = 1, 2, \dots, n$ , where  $f_{ij} \in \{0, 1, \dots, L\}$  denotes the pixel value at the position  $(i, j)$  on the image plane with  $m$  rows and  $n$  columns. The maximum pixel value  $L$  is typically given by  $L = 2^8 - 1 = 255$  in an 8-bit grayscale. Then, the histogram of the pixel values in  $F$  is given by  $\mathbf{h} = [h_0, h_1, \dots, h_L]$  whose  $k$ th element is computed by

$$h_k = \sum_{i=1}^m \sum_{j=1}^n \delta_{k, f_{ij}} \tag{1}$$

for  $k = 0, 1, \dots, L$ , where  $\delta_{k, f_{ij}}$  denotes the Kronecker delta function defined by  $\delta_{k, f_{ij}} = 0$  for  $k \neq f_{ij}$ , and 1 for  $k = f_{ij}$ . Note that the histogram  $\mathbf{h}$  is not normalized, i.e.,  $\sum_{k=0}^L h_k = mn \neq 1$ . The corresponding cumulative histogram is given by  $\mathbf{H} = [H_0, H_1, \dots, H_L]$  whose  $l$ th element is defined by

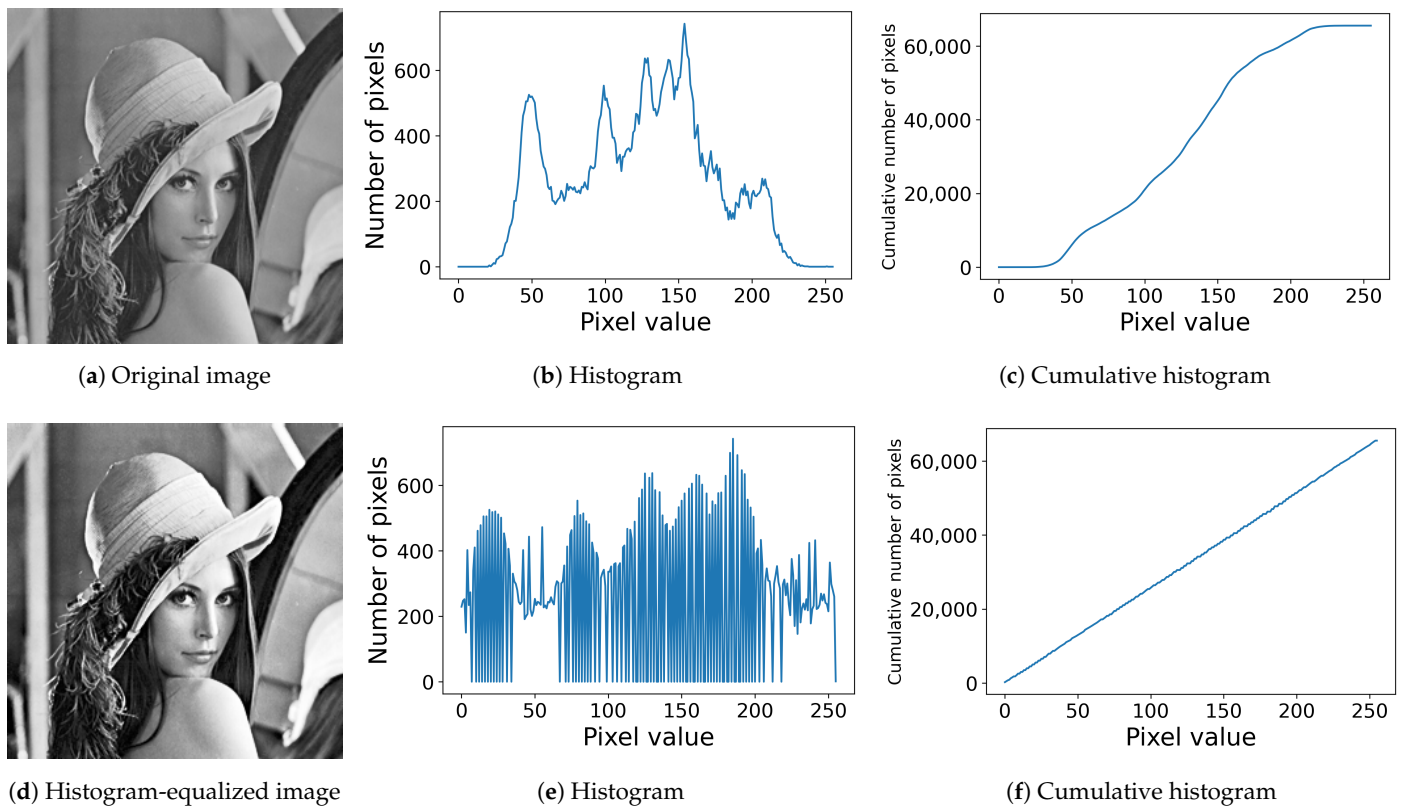
$$H_l = \sum_{k=0}^l h_k \tag{2}$$

for  $l = 0, 1, \dots, L$ , and recursively computed by  $H_l = H_{l-1} + h_l$  for  $l = 1, 2, \dots, L$  with  $H_0 = h_0$ . Histogram equalization converts a pixel value  $f_{ij}$  into

$$f_{ij}^{HE} = \phi_h^{HE}(f_{ij}) := \text{round} \left( L \frac{H_{f_{ij}} - H_{f_{\min}}}{H_{f_{\max}} - H_{f_{\min}}} \right), \tag{3}$$

where  $f_{\min} = \min_{i,j} \{f_{ij}\}$ ,  $f_{\max} = \max_{i,j} \{f_{ij}\}$ , and the ‘round’ operator rounds a given argument toward the nearest integer, and  $\phi_h^{HE}$  denotes the function of converting pixel values in histogram equalization.

Figure 1 shows an example of histogram equalization, where an original image in Figure 1a is converted into the image in Figure 1d, and their histograms and cumulative ones are shown in Figure 1b,c and Figure 1e,f, respectively, where the horizontal axis of each graphic denotes the pixel value ranging from 0 to 255, and the vertical axis denotes the (cumulative) number of pixels for each (cumulative) histogram. As shown in Figure 1e, the resultant histogram is not equalized actually, but the cumulative one is linearized as shown in Figure 1f. That is, to be more precise, the conventional histogram equalization is a cumulative histogram linearization.



**Figure 1.** Histogram equalization: (a) an original image with  $256 \times 256$  pixels; (b) the histogram of the original image; (c) the cumulative histogram of the original image; (d) the histogram-equalized image; (e) the histogram of the image in (d); (f) the cumulative histogram of the image in (d).

#### 4. Histogram Specification

In this section, we summarize conventional histogram specification, which is not an exact method, and then briefly summarize the state-of-the-art method for exact histogram specification by Nikolova and Steidl [13]. After that, we propose a local contrast-based method for exact histogram specification, whose objects to be processed are extended from grayscale images to color images.

##### 4.1. Conventional Histogram Specification

Let  $\hat{h} = [\hat{h}_0, \hat{h}_1, \dots, \hat{h}_L]$  be a target histogram into which we want to convert the histogram  $h$  of the original image  $F$ , where we assume that  $\sum_{k=0}^L \hat{h}_k = \sum_{k=0}^L h_k$  for convenience. Then, we can compute the cumulative histogram  $\hat{H} = [\hat{H}_0, \hat{H}_1, \dots, \hat{H}_L]$  from  $\hat{h}$  in the same manner as above. Theoretically, we can describe the relationship between the target histogram  $\hat{h}$  and the original one  $h$  as

$$f_{ij}^{HS} = (\phi_{\hat{h}}^{HE})^{-1}(f_{ij}^{HE}) = (\phi_{\hat{h}}^{HE})^{-1}(\phi_h^{HE}(f_{ij})), \tag{4}$$

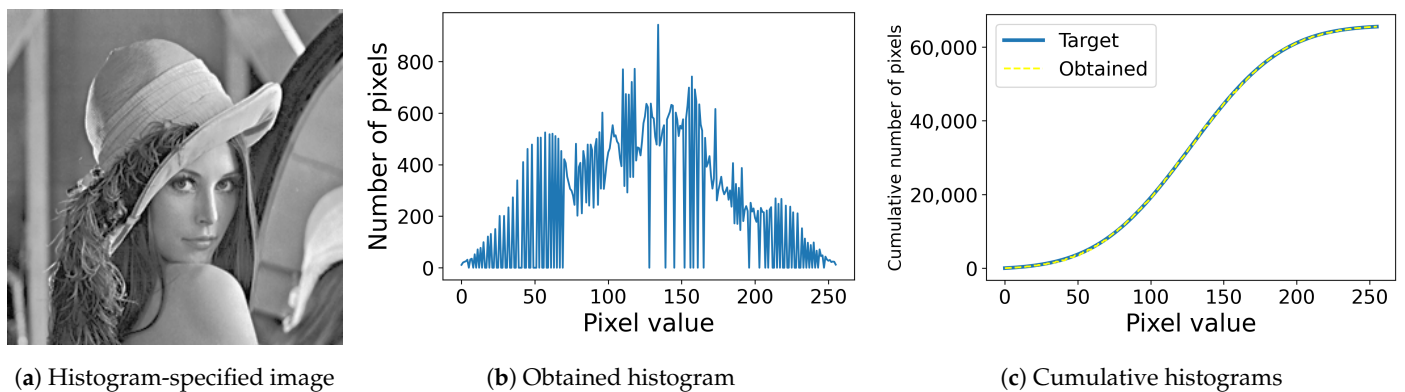
where  $f_{ij}^{HS}$  denotes the output pixel value of histogram specification corresponding to an input pixel value  $f_{ij}$ , and  $(\phi_{\hat{h}}^{HE})^{-1}$  denotes the inverse function of  $\phi_{\hat{h}}^{HE}$  which equalizes the target histogram  $\hat{h}$ . However, in the digital condition, the functions  $\phi_h^{HE}$  and  $\phi_{\hat{h}}^{HE}$  become staircase functions, which are not bijective. Therefore, they have no inverse functions.

A prescription for this situation may be the use of linear interpolation as follows: Assume that  $\phi_h^{HE}(l-1) < \phi_h^{HE}(f_{ij}) \leq \phi_h^{HE}(l)$  for  $l \in \{0, 1, \dots, L\}$ , where we define that  $\phi_h^{HE}(-1) = 0$ . Then, we can determine the output pixel value of histogram specification by

$$f_{ij}^{HS} \approx \text{round} \left( \frac{[\phi_h^{HE}(l) - \phi_h^{HE}(f_{ij})] \max\{l-1, 0\} + [\phi_h^{HE}(f_{ij}) - \phi_h^{HE}(l-1)]l}{\phi_h^{HE}(l) - \phi_h^{HE}(l-1)} \right) \quad (5)$$

corresponding to an input pixel value  $f_{ij}$ .

Figure 2 shows an example of histogram specification, where the resultant image in Figure 2a is obtained with a Gaussian target histogram with mean 127.5 and standard deviation 50 from the original image in Figure 1a. Figure 2b shows the obtained histogram, which is not Gaussian, but the cumulative one fits into the target cumulative histogram as shown in Figure 2c, where a blue solid line denotes the target one, and the yellow dashed line denotes the obtained one. That is, to be more precise, the conventional histogram specification is a cumulative histogram specification.



**Figure 2.** Histogram specification: (a) histogram-specified image computed from the original image in Figure 1a with a Gaussian target histogram; (b) the histogram of the image (a); (c) the target (blue solid line) and obtained (yellow dashed line) cumulative histograms.

#### 4.2. Nikolova and Steidl’s Method

In this subsection, we briefly summarize Nikolova and Steidl’s fast ordering algorithm for exact histogram specification [13]. They proposed a fixed point algorithm that attains the minimizer of fully smoothed  $l_1$ -TV (total variation) functional as follows:

$$J(\mathbf{u}, \mathbf{f}) := \sum_{i=1}^{mn} \theta(u_i - f_i) + \beta \sum_{\kappa=1}^r \theta((G\mathbf{u})_{\kappa}), \quad (6)$$

where  $\beta > 0$ ,  $\mathbf{f} = [f_1, f_2, \dots, f_{mn}]^T = \text{vec}(F) = [f_{11}, f_{21}, \dots, f_{m-1,n}, f_{mn}]^T$  is the vectorization [25] of  $F$  where the superscript  $T$  denotes the matrix transpose [26],  $(G\mathbf{u})_{\kappa}$  denotes the  $\kappa$ th component of  $G\mathbf{u} \in \mathbb{R}^r$  with  $r = 2mn - m - n$ , where  $\mathbb{R}$  denotes the set of real numbers, and  $G$  is a forward difference operator given by

$$G := \begin{pmatrix} I_n \otimes D_m \\ D_n \otimes I_m \end{pmatrix} \in \mathbb{R}^{r, mn}, \quad (7)$$

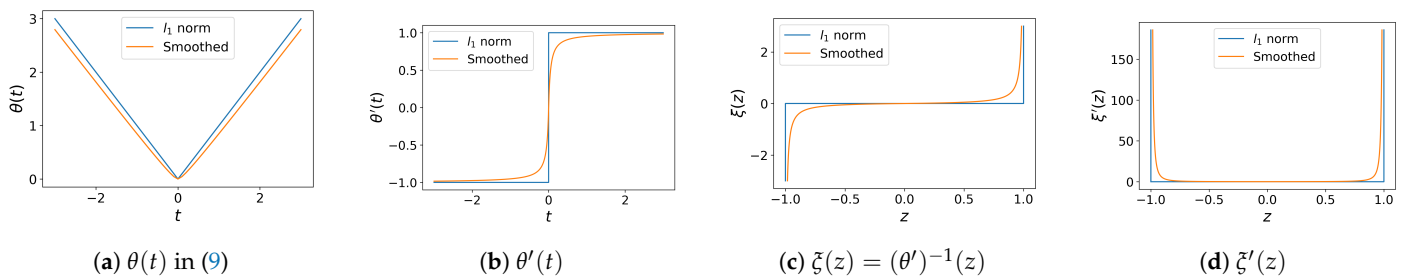
where  $\otimes$  denotes the Kronecker product,  $I_n$  is the  $n \times n$  identity matrix,  $D_n$  denotes the forward difference matrix defined by

$$D_n := \begin{pmatrix} -1 & 1 & & & \\ & -1 & 1 & & \\ & & & \ddots & \\ & & & & -1 & 1 \end{pmatrix} \in \mathbb{R}^{n-1,n}, \tag{8}$$

and the function  $\theta$  is defined by

$$\theta(t) := |t| - \alpha \log\left(1 + \frac{|t|}{\alpha}\right) \tag{9}$$

with a constant  $\alpha > 0$ , which is a smooth approximation of  $l_1$  norm, and has the derivative  $z = \theta'(t) = \frac{t}{\alpha+|t|}$ . The inverse function of  $\theta'$  is defined by  $\zeta(z) := (\theta')^{-1}(z) = \frac{\alpha z}{1-|z|}$ , whose derivative is given by  $\zeta'(z) = \frac{\alpha}{(1-|z|)^2}$ . These functions are illustrated in Figure 3, where blue lines denote the functions for  $l_1$  norm, and orange lines denote their smooth approximations.



**Figure 3.** Nikolova and Steidl’s smooth approximation of  $l_1$  norm: (a)  $l_1$  norm denoted by a blue line is smoothly approximated by  $\theta(t)$  denoted by an orange line; (b) the derivatives of  $l_1$  norm and  $\theta(t)$ ; (c) the inverse functions of  $l_1$  norm and  $\theta'(t)$ , which is denoted by  $\zeta(z)$ ; (d) the derivatives of the inverse functions.

From  $\frac{\partial J}{\partial u_i} = \theta'(u_i - f_i) + \beta \sum_{\kappa=1}^r \theta'((Gu)_\kappa) G_{\kappa,i} = 0$  where  $G_{\kappa,i}$  means the  $(\kappa, i)$ th element of  $G$ , we have  $u_i = f_i - \zeta(\beta \sum_{\kappa=1}^r \theta'((Gu)_\kappa) G_{\kappa,i})$  for  $i = 1, 2, \dots, mn$ , which are combined into a vector equation as  $\mathbf{u} = \mathbf{f} - \zeta(\beta G^T \theta'(G\mathbf{u}))$ , where the functions  $\theta'$  and  $\zeta$  are applied to each element of their arguments. This is a fixed point equation for  $\mathbf{u}$  which gives rise to a fixed point algorithm for minimizing  $J(\mathbf{u}, \mathbf{f})$  as described in Algorithm 1.

**Algorithm 1** Fully smoothed  $l_1$ -TV minimization algorithm [13]

**Require:** a vectorized image  $\mathbf{f} = [f_1, f_2, \dots, f_{mn}]^T$ , parameters  $\beta > 0$  and  $\alpha > 0$  used in functions  $\zeta$  and  $\theta'$ , the number of iterations  $T > 0$

**Ensure:** the minimizer  $\mathbf{u}^{(T)}$  of a fully smoothed  $l_1$ -TV functional  $J(\mathbf{u}, \mathbf{f})$  in (6)

- 1: Initialize the variable  $\mathbf{u}$  as  $\mathbf{u}^{(0)} \leftarrow \mathbf{f}$
- 2: **for**  $t \leftarrow 1, 2, \dots, T$  **do**
- 3:      $\mathbf{u}^{(t)} \leftarrow \mathbf{f} - \zeta(\beta G^T \theta'(G\mathbf{u}^{(t-1)}))$
- 4: **end for**

In this algorithm, we set  $\alpha = 0.05$ ,  $\beta = 0.1$  and  $T = 5$  according to Nikolova and Steidl [13]. The obtained image from  $\mathbf{u}^{(T)}$  with the input image  $\mathbf{f}$  in Figure 1a is shown in Figure 4a. This procedure for computing  $\mathbf{u}^{(T)}$  from  $\mathbf{f}$  can be viewed as a nonlinear filter which reduces the quantization noise in  $\mathbf{f}$  slightly. Therefore, the image in Figure 4a is very close to that in Figure 1a. The pixel values in  $\mathbf{u}^{(T)}$  can be ordered in a strict way with a high probability [8].

Assume that the pixel values in  $\mathbf{u}^{(T)} = [u_1^{(T)}, u_2^{(T)}, \dots, u_{mn}^{(T)}]^T$  are ordered in ascending order:  $u_{\lambda_1}^{(T)} < u_{\lambda_2}^{(T)} < \dots < u_{\lambda_{mn}}^{(T)}$  where  $\lambda_i$  for  $i = 1, 2, \dots, mn$  denote the ordered indices, and the cumulative histogram  $\hat{H} = [\hat{H}_0, \hat{H}_1, \dots, \hat{H}_L]$  is computed from a given target histogram  $\hat{h}$ . Then, if  $\hat{H}_{l-1} < \iota \leq \hat{H}_l$  with  $\hat{H}_{-1} = 0$  for an index  $\iota$  and a pixel value  $l$ , then the corresponding output pixel value is given by  $f_{i_l, j_l}^{NS} = l$ , where  $(i_l, j_l)$  denotes the pixel position corresponding to the ordered index  $\lambda_l$ , i.e.,  $(\lambda_l - 1) = (j_l - 1)m + (i_l - 1)$  with  $0 \leq (i_l - 1) < m$  is satisfied. This procedure is repeated for  $\iota = 1, 2, \dots, mn$  to obtain Nikolova and Steidl’s output image  $F^{NS} = [f_{ij}^{NS}]$ . Figure 4b shows the output image with the same Gaussian target histogram as Figure 2, and Figure 4c shows the target and obtained histograms, where we can see that they have the same Gaussian shape, which demonstrates the exactness of Nikolova and Steidl’s method.



**Figure 4.** Nikolova and Steidl’s exact histogram specification: (a) the output  $\mathbf{u}^{(T)}$  of Algorithm 1 is visualized as an 8-bit grayscale image; (b) the output image; (c) Gaussian target (blue solid line) and obtained (yellow dashed line) histograms.

**5. Proposed Method**

In this section, we propose a local contrast-based pixel ordering method for exact histogram specification. We first describe the method for grayscale images, and then it is extended to that for color images. We also give a complexity analysis of the proposed method. The main idea of our approach is that the problem of pixel ordering can be divided into 256 subproblems for an 8-bit grayscale image. We solve the downsized subproblems separately to have 256 groups of ordered pixels, and then concatenate them to have a sequence of all ordered pixels.

*5.1. Local Contrast-Based Exact Histogram Specification for Grayscale Images*

As demonstrated above, we can transform the histogram of a digital image into the specified one, when all pixels in the image are ordered in a strict and faithful way [13]. In most cases, pixels in a digital image take a limited number of discrete values, e.g., 256 values are available for 8-bit grayscale images, which is smaller than the number of pixels in the image, e.g.,  $m \times n = 256 \times 256 = 65,536 \gg 256$  for the above image in Figure 1a. Therefore, many pixels share the same pixel value with each other in the image, and the ordering of the pixels with the same values becomes an ill-posed problem [8]. However, it is fortunate for us that we have a rough ordering of 256 groups of pixels in the image, which means that the entire pixel ordering problem can be divided into 256 subproblems that can be solved separately. In this subsection, we propose a method for ordering pixels in each group on the basis of the local contrast of each pixel, and then the separately ordered subgroups are finally merged into an entire pixel ordering.

The first step of the proposed method is Gaussian filtering without approximation, which means that the weighted mean of all pixels is outputted at each pixel, where the weights are given by a Gaussian function as follows:

$$f_{ij}^{GF} = \frac{\sum_{k=1}^m \sum_{l=1}^n \exp\left(-\frac{(i-k)^2 + (j-l)^2}{2\sigma^2}\right) f_{kl}}{\sum_{k=1}^m \sum_{l=1}^n \exp\left(-\frac{(i-k)^2 + (j-l)^2}{2\sigma^2}\right)} = \frac{\sum_{k=1}^m \exp\left(-\frac{(i-k)^2}{2\sigma^2}\right) \sum_{l=1}^n \exp\left(-\frac{(j-l)^2}{2\sigma^2}\right) f_{kl}}{\sum_{k=1}^m \exp\left(-\frac{(i-k)^2}{2\sigma^2}\right) \sum_{l=1}^n \exp\left(-\frac{(j-l)^2}{2\sigma^2}\right)} \quad (10)$$

for  $i = 1, 2, \dots, m$  and  $j = 1, 2, \dots, n$ , where  $\sigma$  denotes the standard deviation positive constant. Let  $F^{GF} = [f_{ij}^{GF}]$  be an  $m \times n$  matrix which expresses the Gaussian-filtered image of  $F$ . Then, we can compute  $F^{GF}$  by the following matrix operations. The numerator of (10) is computed by

$$N_u = G_L F G_R, \quad (11)$$

where the left matrix  $G_L$  is given by  $G_L = \exp(-D_L \odot D_L / 2\sigma^2)$  with

$$D_L = \mathbf{1}_m \begin{bmatrix} 1 & 2 & \dots & m \end{bmatrix} + \begin{bmatrix} 1 & 2 & \dots & m \end{bmatrix}^T \mathbf{1}_m^T - 2 \begin{bmatrix} 1 & 2 & \dots & m \end{bmatrix} \begin{bmatrix} 1 & 2 & \dots & m \end{bmatrix}, \quad (12)$$

where  $\odot$  denotes the Hadamard product or elementwise product of matrices, and  $\mathbf{1}_m$  denotes the  $m$ -dimensional column vector of ones. Similarly, the right matrix  $G_R$  in (11) is given by  $G_R = \exp(-D_R \odot D_R / 2\sigma^2)$  with

$$D_R = \mathbf{1}_n \begin{bmatrix} 1 & 2 & \dots & n \end{bmatrix} + \begin{bmatrix} 1 & 2 & \dots & n \end{bmatrix}^T \mathbf{1}_n^T - 2 \begin{bmatrix} 1 & 2 & \dots & n \end{bmatrix} \begin{bmatrix} 1 & 2 & \dots & n \end{bmatrix}. \quad (13)$$

On the other hand, the denominator of (10) is computed by

$$D_e = G_L E G_R, \quad (14)$$

where  $E$  denotes the  $m \times n$  matrix of ones. Then, we have the Gaussian-filtered image by  $F^{GF} = N_u \oslash D_e$ , where  $\oslash$  denotes the Hadamard division or elementwise division of matrices.

Next, we define the local contrast (LC) of each pixel by  $d_{ij} := f_{ij} - f_{ij}^{GF}$  which is added to the corresponding pixel as an attribute, and divide the  $mn$  pixels in  $F$  into  $(L + 1)$  groups as  $S_k := \{(f_{ij}, d_{ij}) \mid f_{ij} = k\}$ , where we have that  $h_k = |S_k|$  for  $k = 0, 1, \dots, L$ , i.e.,  $h_k$  is the cardinality of  $S_k$ . Then, we sort the elements of  $S_k$  in the ascending order of  $d_{ij}$  for  $k = 0, 1, \dots, L$ , and concatenate them to have a sequence. As a result, we expect to have the entire pixel ordering  $\mu_i \in \{1, 2, \dots, mn\}$  for  $i = 1, 2, \dots, mn$ , from which the corresponding pixel position  $(i, j_i)$  is given as the integer part  $(j_i - 1)$  and remainder  $(i_i - 1)$  of the division  $(\mu_i - 1) / m$ , i.e.,  $(\mu_i - 1) = (j_i - 1)m + (i_i - 1)$ , and  $\{d_{ij}\}$  is ordered as follows:

$$d_{i_1 j_1} < \dots < d_{i_i j_i} < \dots < d_{i_{mn} j_{mn}}. \quad (15)$$

For exact histogram specification, we would like to have a strict ordering as described in (15). That is, for different pixels with the same pixel value as  $f_{ij} = f_{i'j'}$  with  $i \neq i'$  or  $j \neq j'$ , it is expected that  $d_{ij} \neq d_{i'j'}$  or  $f_{ij}^{GF} \neq f_{i'j'}^{GF}$ . For conventional Gaussian filters with finite kernel sizes such as  $3 \times 3$ ,  $5 \times 5$  and  $7 \times 7$  pixels, the lower bounds of the probability of  $f_{ij}^{GF} = f_{i'j'}^{GF}$  are  $(\frac{1}{256})^{7 \times 7} < (\frac{1}{256})^{5 \times 5} < (\frac{1}{256})^{3 \times 3} \approx 0$ . However, some symmetric patterns may result in the case of  $f_{ij}^{GF} = f_{i'j'}^{GF}$  accidentally. To reduce the probability of such cases, we adopt the Gaussian filter with sufficiently large kernel size as described in (10).

For an input image  $F$  with a target histogram  $\hat{h}$  and its cumulative version  $\hat{H} = [\hat{H}_0, \hat{H}_1, \dots, \hat{H}_L]$ , the proposed method outputs a histogram-specified image  $F^{LC} = [f_{ij}^{LC}]$ , where each pixel value is given by  $f_{i,j}^{LC} = l$  for the index  $\mu_i$  satisfying  $\hat{H}_{l-1} < i \leq \hat{H}_l$  with  $\hat{H}_{-1} = 0$  as well as Nikolova and Steidl's method summarized in Section 4.2. The proposed method is summarized in Algorithm 2.



**Algorithm 2** Local contrast-based exact histogram specification

**Require:** a grayscale image  $F = [f_{ij}]$ , a positive constant  $\sigma$  and a target histogram  $\hat{h}$   
**Ensure:** a histogram-specified image  $F^{LC} = [f_{ij}^{LC}]$

- 1: Compute the Gaussian-filtered image  $F^{GF}$  of  $F$  with a positive constant  $\sigma$
- 2: Compute the local contrast  $D = [d_{ij}]$  by  $D = F - F^{GF}$
- 3: Make  $(L + 1)$  groups  $S_k = \{(f_{ij}, d_{ij}) \mid f_{ij} = k\}$  for  $k = 0, 1, \dots, L$
- 4:  $\iota \leftarrow 1$
- 5: **for**  $k \leftarrow 0, 1, \dots, L$  **do**
- 6:     Sort the elements of  $S_k$  in the ascending order of  $d_{ij}$  as  $d_{i_1, j_1} < \dots < d_{i_u, j_u} < \dots < d_{i_{|S_k|}, j_{|S_k|}}$
- 7:     **for**  $u \leftarrow 1, 2, \dots, |S_k|$  **do**
- 8:          $\mu_u \leftarrow m(j_u - 1) + i_u$
- 9:          $\iota \leftarrow \iota + 1$
- 10:     **end for**
- 11: **end for**
- 12: Compute the target cumulative histogram  $\hat{H} = [\hat{H}_0, \hat{H}_1, \dots, \hat{H}_L]$  from  $\hat{h}$
- 13: **for**  $\iota \leftarrow 1, 2, \dots, mn$  **do**
- 14:     Search for  $l$  satisfying  $\hat{H}_{l-1} < \iota \leq \hat{H}_l$  with  $\hat{H}_{-1} = 0$
- 15:      $f_{i_\iota, j_\iota}^{LC} \leftarrow l$ , where  $(j_\iota - 1)$  and  $(i_\iota - 1)$  are the integer part and remainder of  $(\mu_\iota - 1) / m$
- 16: **end for**

Additionally, we can make an image whose pixel values indicate ordered numbers as  $F^l = [f_{ij}^l]$ , where each pixel value is given by  $f_{i,j}^l = \iota$  for  $\iota = 1, 2, \dots, mn$ . For histogram equalization, the number of pixels to which the same pixel value is assigned is given by  $\bar{h} = \lfloor mn / (L + 1) \rfloor$ , where  $\lfloor x \rfloor$  denotes the floor function that gives the largest integer less than or equal to  $x$  [27]. That is, we make a target histogram  $\hat{h} = [\hat{h}_k]$  as  $\hat{h}_k = \bar{h}$  for  $k = 0, 1, \dots, L$ . However, the truncation by the floor function may cause the shortage of total amount in the histogram:  $\sum_{k=0}^L \hat{h}_k < mn$ . To cover the shortage, if  $Q = mn - \sum_{k=0}^L \hat{h}_k > 0$ , then we add 1 to  $\hat{h}_0, \hat{h}_1, \dots, \hat{h}_{Q-1}$  to have  $\sum_{k=0}^L \hat{h}_k = mn$ . This modified histogram can be inputted to Algorithm 2 as a target histogram for exact histogram equalization.

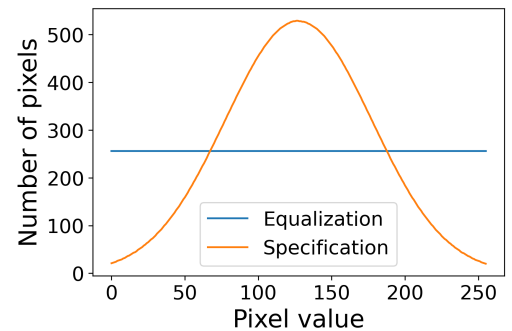
Figure 5 shows the results of the proposed method, where Figure 5a,b show the results of the histogram equalization and specification with  $\sigma = 50$  for the original image in Figure 1a, respectively, and Figure 5c shows their histograms, where we can see that both exact histogram equalization and specification are achieved.



(a) Histogram-equalized image



(b) Histogram-specified image



(c) Equalized and specified histograms

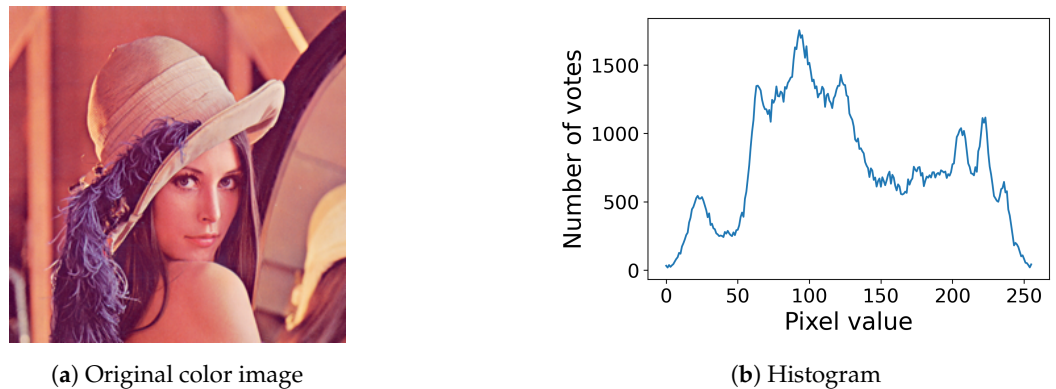
**Figure 5.** Exact histogram equalization and specification: (a) histogram-equalized image of the original image in Figure 1a; (b) the histogram-specified image with a Gaussian target histogram; (c) equalized (blue line) and specified (orange line) histograms.

5.2. Extension to Color Images

In this subsection, we extend the above proposed method for grayscale images to color images. Let  $C = [c_{ij}]$  be an RGB color image, where  $c_{ij} = [c_{ij1}, c_{ij2}, c_{ij3}]$  denotes the RGB color vector of the  $(i, j)$ th pixel in  $C$ , and  $c_{ijk} \in \{0, 1, \dots, L\}$  for  $k = 1, 2$  and  $3$ . Then, we make a histogram into which all RGB values in  $C$  are united as follows:

$$h^C = [h_0^C, \dots, h_l^C, \dots, h_L^C], \quad h_l^C = \sum_{i=1}^m \sum_{j=1}^n \sum_{k=1}^3 \delta_{l, c_{ijk}} \quad \text{for } l = 0, 1, \dots, L. \quad (16)$$

For example, Figure 6 shows an RGB color image and its histogram defined in (16).



**Figure 6.** Color image and its histogram: (a) an RGB color image with  $256 \times 256$  pixels; (b) the histogram of the color image (a).

The first step of our exact histogram specification is Gaussian filtering applied to each color channel in  $C$ . Let  $C^{GF} = [c_{ij}^{GF}]$  be the resultant image, where  $c_{ij}^{GF} = [c_{ij1}^{GF}, c_{ij2}^{GF}, c_{ij3}^{GF}]$  denotes the RGB color vector of each pixel in  $C^{GF}$ . Then, we compute the LC as  $D^C = [d_{ij}^C] = C - C^{GF}$  where  $d_{ij}^C = [d_{ij1}^C, d_{ij2}^C, d_{ij3}^C] = c_{ij} - c_{ij}^{GF}$ , each element of which is added to the corresponding element of the corresponding pixel as an attribute:  $(c_{ijk}, d_{ijk}^C)$  for  $i = 1, 2, \dots, m$ ;  $j = 1, 2, \dots, n$  and  $k = 1, 2, 3$ , and divide them into  $(L + 1)$  groups as  $S_l := \{(c_{ijk}, d_{ijk}^C) \mid c_{ijk} = l\}$ , where we have that  $h_l^C = |S_l|$  for  $l = 0, 1, \dots, L$ . Then, we sort the elements of  $S_l$  in the ascending order of  $d_{ijk}^C$  for  $l = 0, 1, \dots, L$ , and concatenate them to have a sequence. As a result, we expect to have the entire element ordering  $\eta_l \in \{1, 2, \dots, 3mn\}$  for  $l = 1, 2, \dots, 3mn$ , each of which is connected with the corresponding pixel position  $(i, j)$  and channel number  $k_l$  by satisfying

$$d_{i_1, j_1, k_1}^C < \dots < d_{i_l, j_l, k_l}^C < \dots < d_{i_{3mn}, j_{3mn}, k_{3mn}}^C. \quad (17)$$

Let  $\hat{h}^C$  be a target histogram with the cumulative one  $\hat{H}^C = [\hat{H}_0^C, \hat{H}_1^C, \dots, \hat{H}_L^C]$ . Then, the histogram-specified image  $C^{LC} = [c_{ij}^{LC}]$  is given by  $c_{ij}^{LC} = [c_{ij1}^{LC}, c_{ij2}^{LC}, c_{ij3}^{LC}]$  with  $c_{i, j, k_l}^{LC} = l$  for the index  $\eta_l$ , satisfying  $\hat{H}_{l-1}^C < \iota \leq \hat{H}_l^C$  with  $\hat{H}_{-1}^C = 0$ . This procedure for color images is summarized in Algorithm 3.

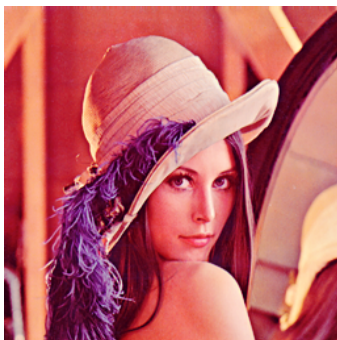
Figure 7 shows the results of the proposed method for color images, where Figure 7a,b show the results of the histogram equalization and specification with  $\sigma = 50$  for the original color image in Figure 6a, respectively, and Figure 7c shows their histograms, where we can also see that both exact histogram equalization and specification are achieved.

**Algorithm 3** Local contrast-based exact histogram specification for color images

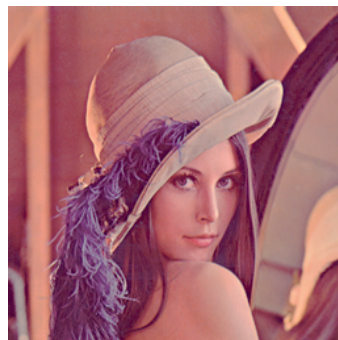
**Require:** a color image  $C = [c_{ij}]$ , a positive constant  $\sigma$  and a target histogram  $\hat{h}^C$

**Ensure:** a histogram-specified image  $C^{LC} = [c_{ij}^{LC}]$

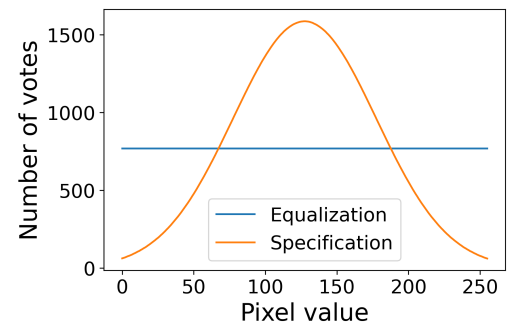
- 1: Compute the Gaussian-filtered image  $C^{GF}$  of  $C$  with a positive constant  $\sigma$
- 2: Compute the local contrast  $D^C = [d_{ij}^C]$  by  $D^C = C - C^{GF}$  where  $d_{ij}^C = [d_{ij1}^C, d_{ij2}^C, d_{ij3}^C]$
- 3: Make  $(L + 1)$  groups  $S_l = \{(c_{ijk}, d_{ijk}^C) \mid c_{ijk} = l\}$  for  $l = 0, 1, \dots, L$
- 4:  $\iota \leftarrow 1$
- 5: **for**  $l \leftarrow 0, 1, \dots, L$  **do**
- 6:     Sort the elements of  $S_l$  in the ascending order of  $d_{ijk}^C$  as  $d_{i_1, j_1, k_1}^C < \dots < d_{i_u, j_u, k_u}^C < \dots < d_{i_{|S_l|}, j_{|S_l|}, k_{|S_l|}}^C$
- 7:     **for**  $u \leftarrow 1, 2, \dots, |S_l|$  **do**
- 8:          $\eta_u \leftarrow mn(k_u - 1) + m(j_u - 1) + i_u$
- 9:          $\iota \leftarrow \iota + 1$
- 10:     **end for**
- 11: **end for**
- 12: Compute the target cumulative histogram  $\hat{H} = [\hat{H}_0, \hat{H}_1, \dots, \hat{H}_L]$  from  $\hat{h}$
- 13: **for**  $\iota \leftarrow 1, 2, \dots, 3mn$  **do**
- 14:     Search for  $l$  satisfying  $\hat{H}_{l-1} < \iota \leq \hat{H}_l$  with  $\hat{H}_{-1} = 0$
- 15:      $c_{i_\iota, j_\iota, k_\iota}^{LC} \leftarrow l$ , where  $i_\iota, j_\iota$  and  $k_\iota$  are related to  $\eta_\iota$  by  $\eta_\iota = mn(k_\iota - 1) + m(j_\iota - 1) + i_\iota$
- 16: **end for**



(a) Histogram-equalized image



(b) Histogram-specified image



(c) Equalized and specified histograms

**Figure 7.** Exact histogram equalization and specification for a color image: (a) histogram-equalized image of the original color image in Figure 6a; (b) the histogram-specified image with a Gaussian target histogram; (c) equalized (blue line) and specified (orange line) histograms.

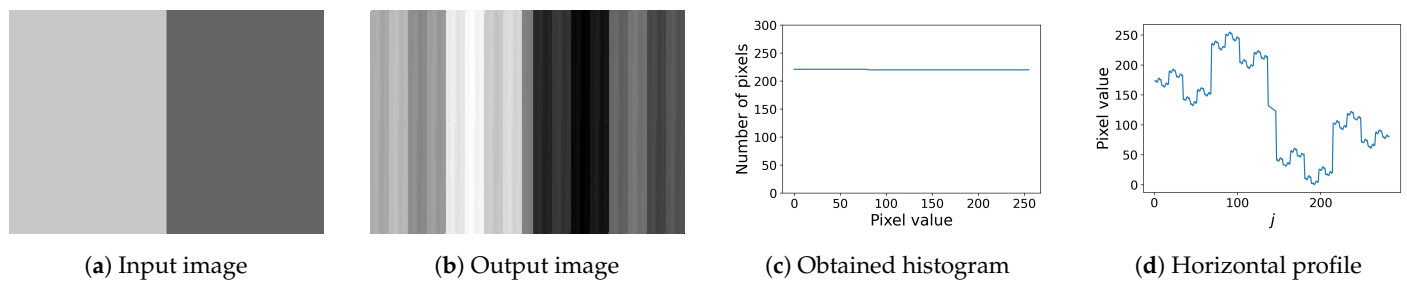
5.3. Complexity Analysis

In this subsection, we analyze the complexity of the proposed method in Algorithm 2. Let  $N = mn$  be the input size. The computation of Gaussian filtering in line 1 is described as matrix calculations. The time complexity of the numerator (11) is estimated as  $mn^2 + m^2n = mn(m + n) = N(m + n) \approx N^{3/2}$ , which is the same as that of the denominator (14). The Hadamard division requires  $N$  elementwise divisions. Then, the order of time complexity of Gaussian filtering is  $O(N^{3/2})$ . The next main procedure is sort in line 6, which is repeated  $L + 1$  times. Assume that the cardinality  $|S_l|$  is  $\frac{N}{L+1}$  on average. Then, the time complexity of the sorting procedure is estimated as  $(L + 1) \times \frac{N}{L+1} \log \frac{N}{L+1} = N[\log N - \log(L + 1)]$  from which we have  $O(N \log N)$ . The time complexity of the remaining procedures is bounded by  $O(N)$ . Consequently, the time complexity of the proposed method is given by  $O(N^{3/2})$ .

The space complexities of images  $F$ ,  $F^{GF}$ ,  $D$ , and  $F^{LC}$  are  $O(N)$ , and that of  $(L + 1)$  groups  $S_k$  for  $k = 0, 1, \dots, L$  in line 3 is also  $O(N)$ . Histogram arrays  $\hat{h}$  and  $\hat{H}$  require  $O(L + 1)$ . Selecting the largest one, we can see that the space complexity of the proposed method is  $O(N)$ .

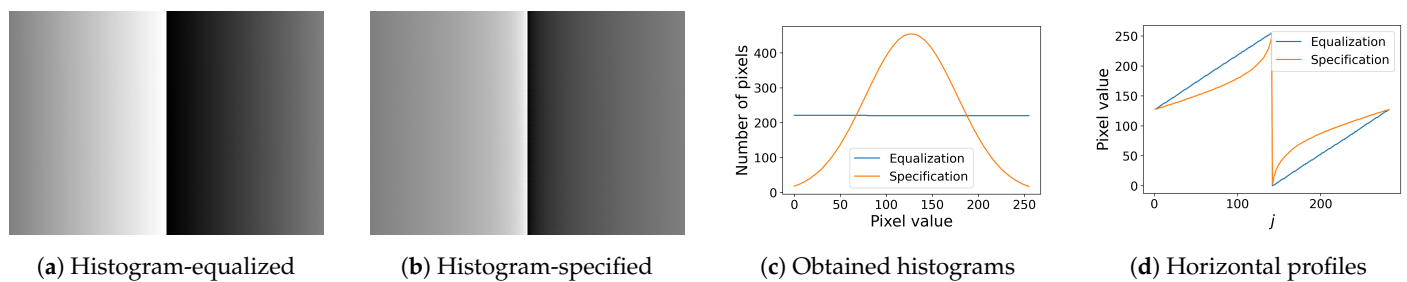
### 6. Experimental Results

In this section, we show experimental results in comparison with Nikolova and Steidl’s method [13]. First, we show the results with a synthetic input image shown in Figure 8a. Nikolova and Steidl’s method summarized in Section 4.2 outputs the image in Figure 8b, where we can see that vertical stripes appear in the flat areas in the original image. As a result, Nikolova and Steidl’s method achieves the equalized histogram shown in Figure 8c. Figure 8d shows the horizontal profile of the output image in Figure 8b, where the vertical and horizontal axes denote the pixel value and the column index  $j$ , respectively. Although the pixel values fluctuate with small steps, the pixel values on the left side are larger than that on the right side as well as the original image.



**Figure 8.** Nikolova and Steidl’s exact histogram equalization: (a) input image with  $200 \times 282$  pixels, where the dark and bright areas have pixel values 100 and 200, respectively. (b) vertical stripes are generated in the output image; (c) the obtained histogram confirms that Nikolova and Steidl’s method exactly equalizes the histogram; (d) horizontal profile in (b) shows the fluctuation of pixel values, but it preserves the original ordering in (a).

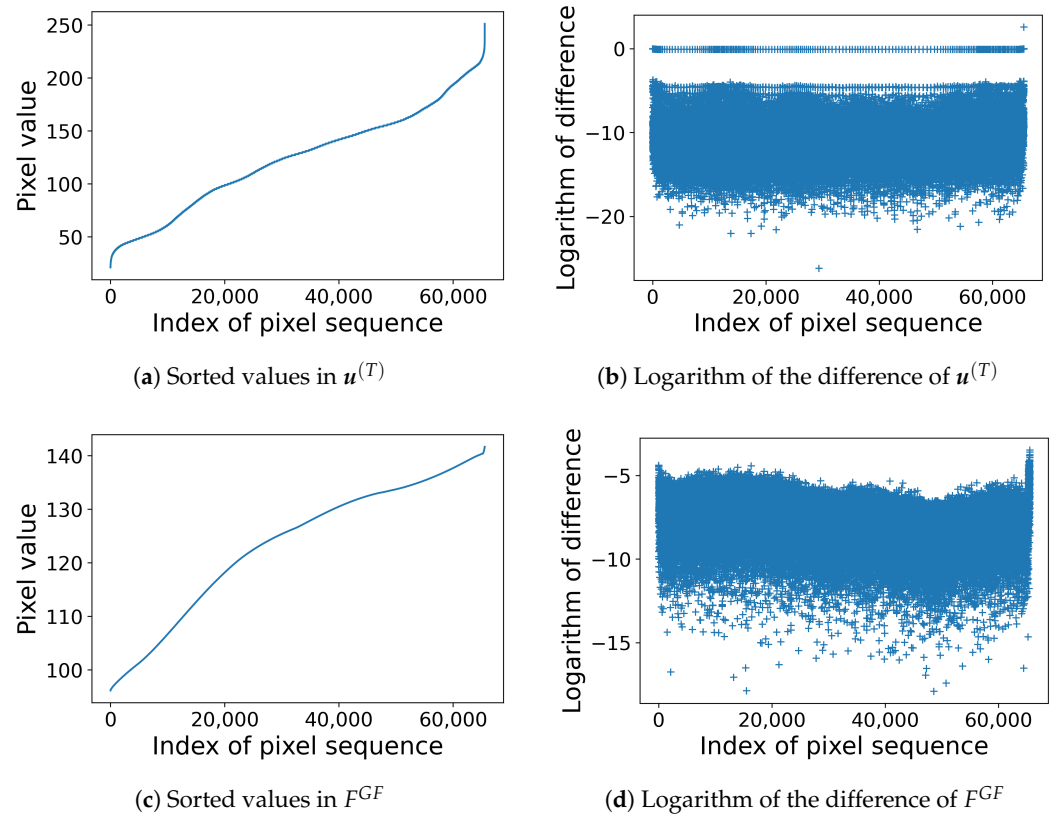
Figure 9 shows the results of the proposed method described in Section 5, where Figure 9a,b show the output images of exact histogram equalization and specification with the Gaussian target histogram used in the previous section. Their histograms are shown in Figure 9c. The proposed method enhances the edge in the center of the input image in Figure 8a, and does not produce the pseudo-strips observed in Figure 8b that are graphically confirmed by the horizontal profiles in Figure 9d.



**Figure 9.** Exact histogram equalization and specification by the proposed method: (a) histogram-equalized image has the enhanced central edge; (b) histogram-specified image also has the enhanced edge; (c) histograms of the output images in (a,b) are equalized and specified to be Gaussian, respectively; (d) horizontal profiles in (a,b) have no fluctuation observed in Figure 8d.

Next, we compare the ability in strict ordering by Nikolova and Steidl’s and the proposed methods. Figure 10 compares the filter outputs used in the two methods with the input image in Figure 1a. Nikolova and Steidl’s filter in Algorithm 1 outputs  $u^{(T)}$ , the values in which are sorted in ascending order in Figure 10a where the vertical and horizontal axes denote the pixel value and index, respectively. The differences between the neighboring pixel values in Figure 10a are very small, and therefore, we take the logarithm of them as shown in Figure 10b. Similarly, we sort the values of the output of the Gaussian filter in (10) as shown in Figure 10c, and take the logarithm of the difference values in Figure 10c

as shown in Figure 10d, where we can see that the minimum value is greater than that in Figure 10c, which is preferable for strict ordering because the larger the difference is, the clearer the ordering is. The minimum values of the differences without taking logarithm are  $4.23 \times 10^{-12}$  and  $1.66 \times 10^{-8}$  for Figure 10b,d, respectively.



**Figure 10.** Comparison of filter outputs  $u^{(T)}$  and  $F^{GF}$ : (a) The values in the output  $u^{(T)}$  in Algorithm 1 visualized in Figure 4a are sorted in ascending order. (b) The values of  $\log(u_{\lambda_{i+1}}^{(T)} - u_{\lambda_i}^{(T)})$  for  $i = 1, 2, \dots, mn - 1$  are plotted. (c) The values of the output of the Gaussian filter in (10) with  $\sigma = 50$  are sorted in ascending order. (d) The logarithm of the difference of the sorted values in (c) are plotted.

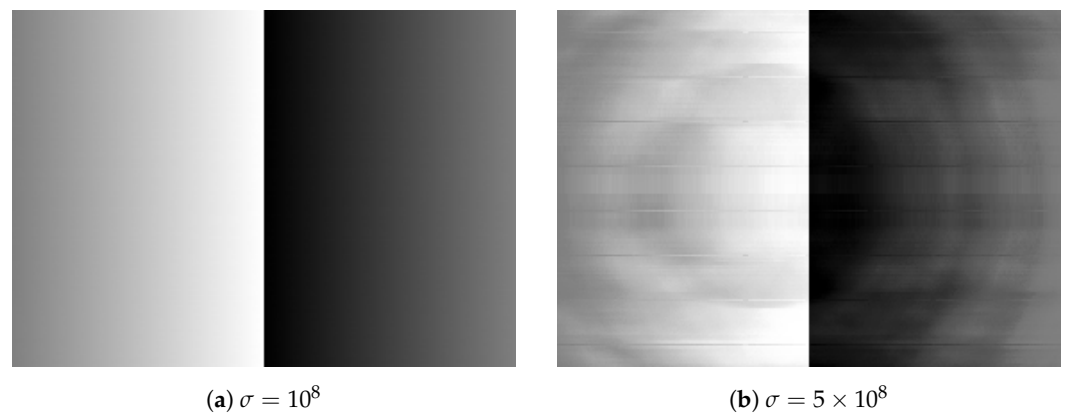
The proposed method has a parameter  $\sigma$  in (10), the effect of which is investigated in Figure 11a, where the vertical and horizontal axes denote the minimum difference in the sorted output values of the Gaussian filter and the parameter  $\sigma$ , respectively, for the input image in Figure 1a. In this figure, although we cannot find clear tendency of the minimum difference as a function of  $\sigma$ , the obtained difference values are larger than that of  $u^{(T)}$  given by Algorithm 1 that exemplifies the insensitivity of the proposed method to  $\sigma$ . Figure 11b,c show the results of histogram equalization with  $\sigma = 1$  and 70, respectively. These images demonstrate that the proposed method is insensitive to the values of parameter  $\sigma$ .

We investigate the limitation of the proposed method on the value of  $\sigma$ . Figure 12 shows the results of histogram equalization for the input image in Figure 8a, where Figure 12a,b correspond to  $\sigma = 10^8$  and  $5 \times 10^8$ , respectively. Figure 12a is similar to Figure 9a. However, in Figure 12b, we observe an artifact. As a result, we estimate that the limitation of the value of  $\sigma$  exists between  $10^8$  and  $5 \times 10^8$ , and recommend using the value of  $\sigma$  smaller than  $10^8$ .

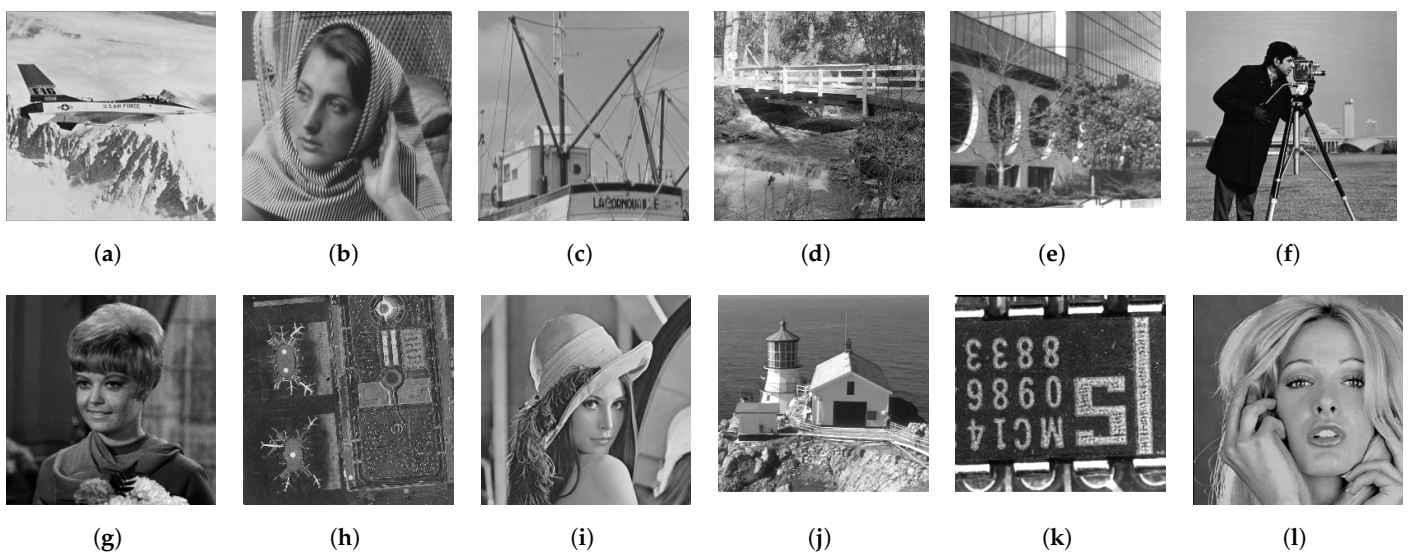
Figure 13 shows twelve grayscale images in the Standard Image Data-Base (SIDBA) [28], which are enhanced by Nikolova and Steidl’s exact histogram equalization as shown in Figure 14, where the contrast of all images is enhanced well because the histogram of each image is exactly equalized.



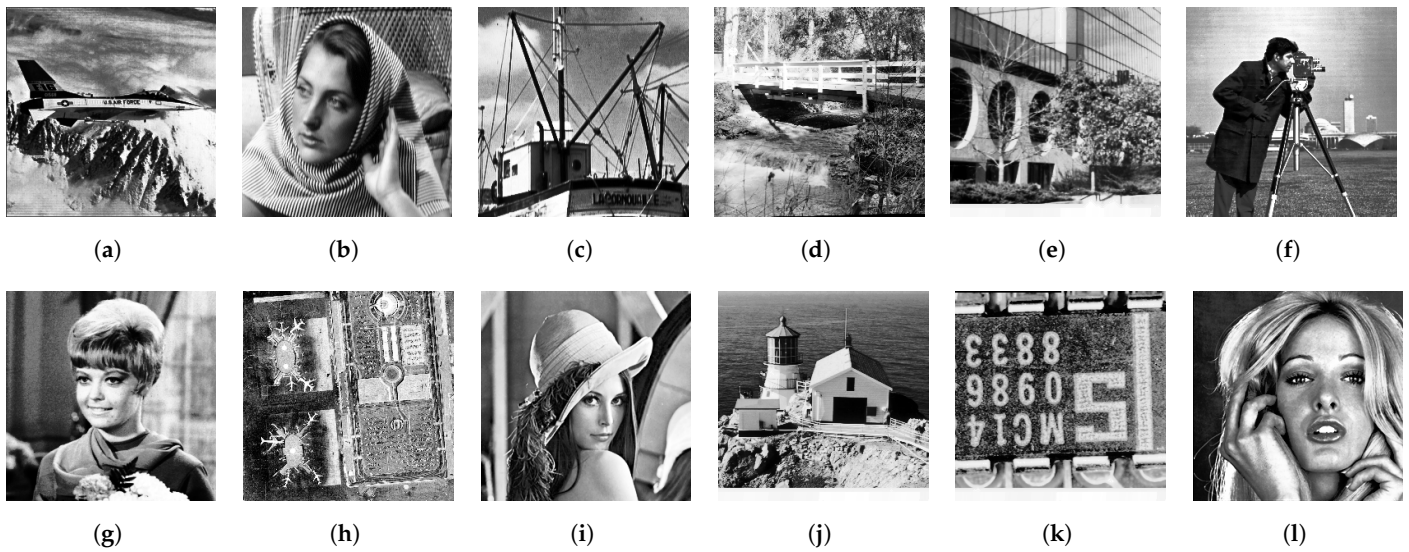
**Figure 11.** Insensitivity to parameter  $\sigma$ : (a) the minimum difference in the sorted values of  $F^{GF}$  as a function of  $\sigma$ . The positive values mean that there are no pixels with the same pixel value in Gaussian-filtered image  $F^{GF}$ , from which we have a strict ordering; (b) histogram equalization with  $\sigma = 1$ ; (c) histogram equalization with  $\sigma = 70$ .



**Figure 12.** Histogram equalization by the proposed method with relatively large values of  $\sigma$ : (a) No artifact is generated with  $\sigma = 10^8$ . (b) An artifact is generated with  $\sigma = 5 \times 10^8$ .



**Figure 13.** Grayscale images in the SIDBA image dataset [28]: (a) Airplane, (b) Barbara, (c) Boat, (d) Bridge, (e) Building, (f) Cameraman, (g) Girl, (h) Lax, (i) Lenna, (j) Lighthouse, (k) Text, (l) Woman.

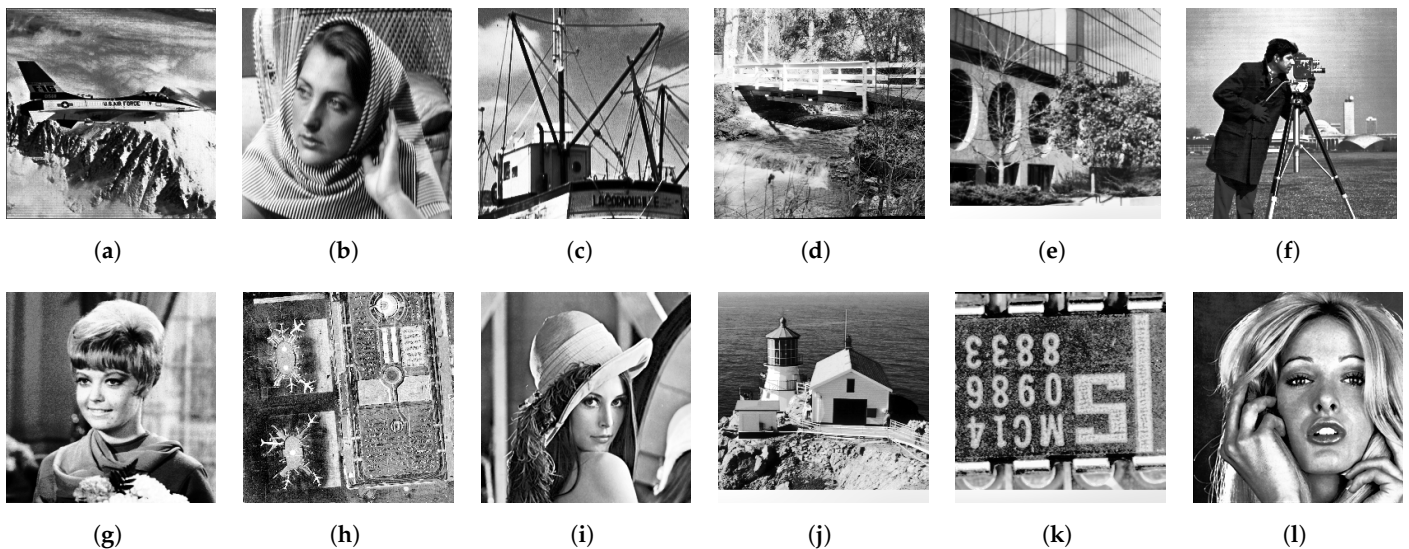


**Figure 14.** Nikolova and Steidl’s exact histogram equalization: (a) Airplane, (b) Barbara, (c) Boat, (d) Bridge, (e) Building, (f) Cameraman, (g) Girl, (h) Lax, (i) Lenna, (j) Lighthouse, (k) Text, (l) Woman.

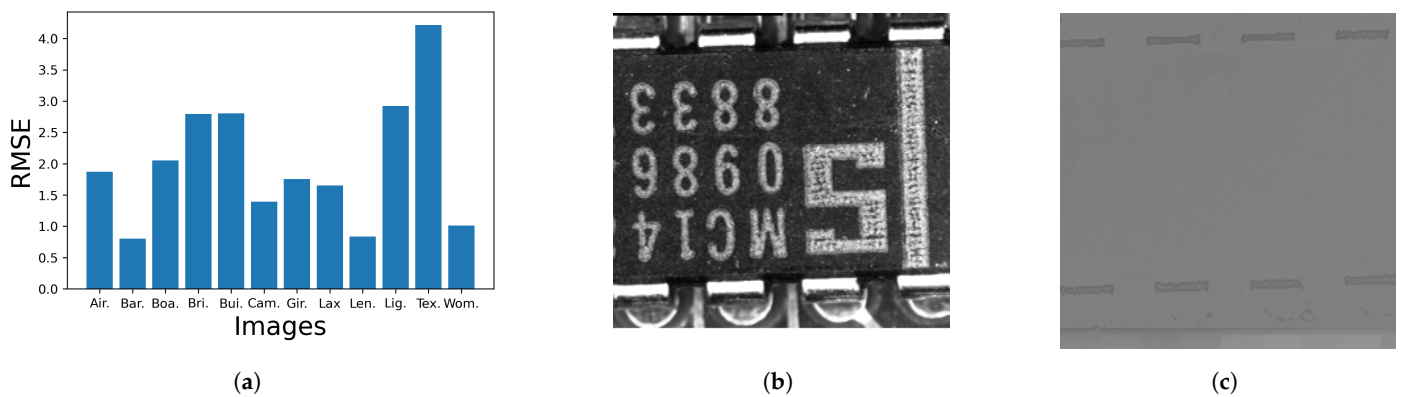
Similarly, Figure 15 shows the results by the proposed method, which also equalizes the histogram of each image exactly. Therefore, the output images of both methods are similar to each other.

Figure 16a shows the root mean squared error (RMSE) between Nikolova and Steidl’s output image and the corresponding output image by the proposed method, where “Text” image shown in (b) get the largest RMSE value, but the difference is small as visualized in (c), where error-free pixels have the neutral gray value 127.

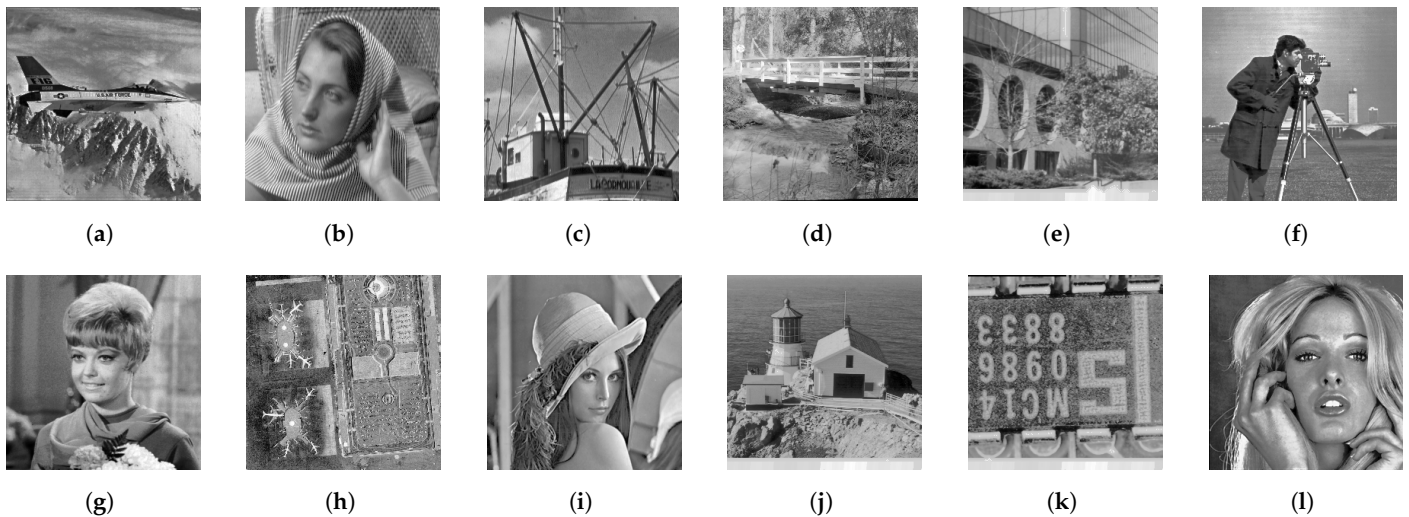
We also compare the results of exact histogram specification with the Gaussian target histogram, which are shown in Figures 17 and 18 for Nikolova and Steidl’s and the proposed methods, respectively. In Figure 17, we observe that the obtained contrast is not so high as that of Figure 14 because the Gaussian target histogram suppresses the numbers of extremely dark and bright pixels, and assigns many pixel values to middle-range gray pixels. As a result, the dynamic range of gray pixels is boosted relatively. The output images in Figure 18 have a similar tendency to that in Figure 17 because the proposed method also obtains an exact Gaussian target histogram as well as Nikolova and Steidl’s method.



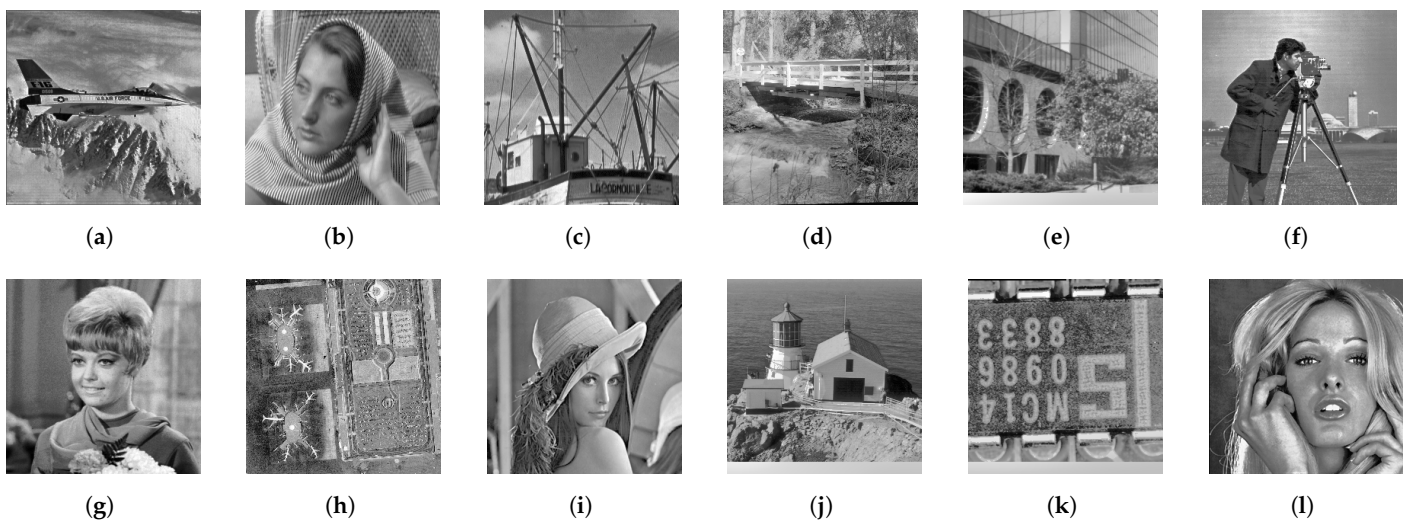
**Figure 15.** The proposed exact histogram equalization: (a) Airplane, (b) Barbara, (c) Boat, (d) Bridge, (e) Building, (f) Cameraman, (g) Girl, (h) Lax, (i) Lenna, (j) Lighthouse, (k) Text, (l) Woman.



**Figure 16.** Comparison between Figures 14 and 15: (a) RMSEs of twelve grayscale images; (b) The “Text” image obtains the largest RMSE value in (a); (c) Difference from the neutral gray indicates the difference between Nikolova and Steidl’s and the proposed methods. (a) Root mean squared error; (b) Text (identical to Figure 13k); (c) Difference image.



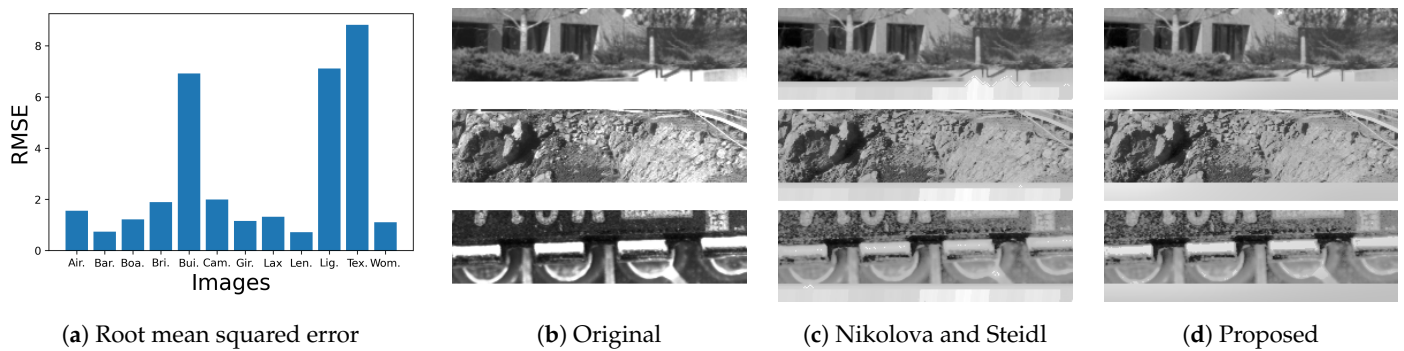
**Figure 17.** Nikolova and Steidl’s exact histogram specification: (a) Airplane, (b) Barbara, (c) Boat, (d) Bridge, (e) Building, (f) Cameraman, (g) Girl, (h) Lax, (i) Lenna, (j) Lighthouse, (k) Text, (l) Woman.



**Figure 18.** The proposed exact histogram specification: (a) Airplane, (b) Barbara, (c) Boat, (d) Bridge, (e) Building, (f) Cameraman, (g) Girl, (h) Lax, (i) Lenna, (j) Lighthouse, (k) Text, (l) Woman.



Figure 19a shows the RMSE between the corresponding output images in Figures 17 and 18, where three images “Bui.,” “Lig.” and “Tex.” have relatively large RMSE values. We can find the differences at the bottom of those images as shown in Figure 19b, where the bottom areas of “Bui.,” “Lig.” and “Tex.” are arranged from top to bottom. The corresponding results by Nikolova and Steidl’s and the proposed methods are shown in Figure 19c and Figure 19d, respectively, where we can see artifacts in (c) similar to that in Figure 8b. On the other hand, Figure 19d has no such artifacts.

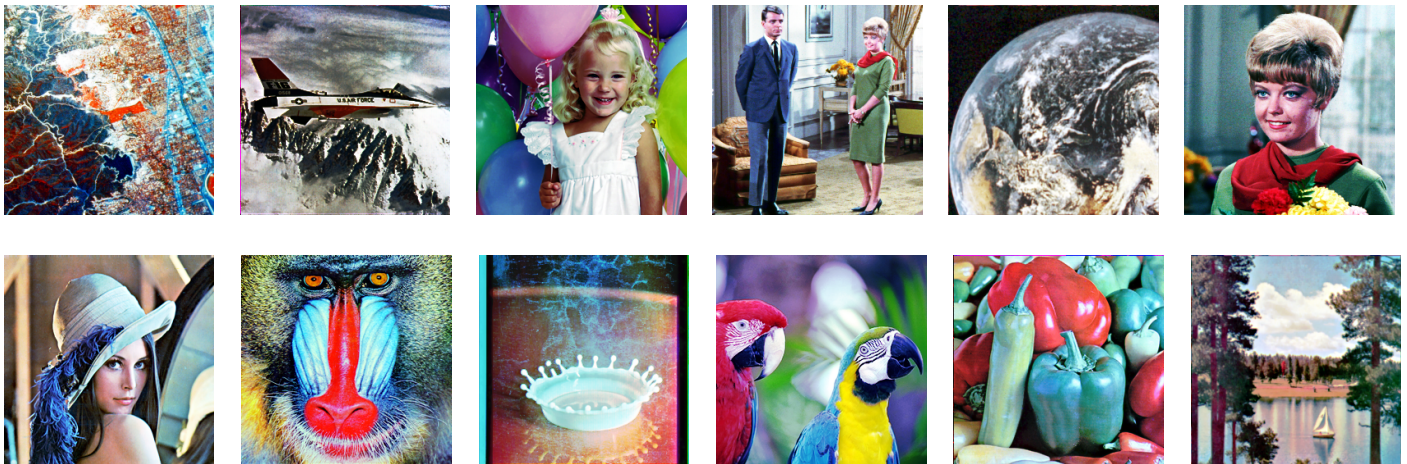


**Figure 19.** Comparison between Figures 17 and 18: (a) RMSEs of twelve grayscale images; (b) The bottom areas of the original “Bui.,” “Lig.” and “Tex.” images are arranged from top to bottom; (c) the same parts of the results by Nikolova and Steidl’s method; (d) the same parts of the results by the proposed methods.

Figure 20 shows twelve color images in the SIDBA image dataset [28], which are enhanced by a separable histogram equalization that applies Nikolova and Steidl’s exact histogram equalization algorithm to each color channel separately as shown in Figure 21, where we can see that the contrast of every image is enhanced, but the hue has changed from the original images in Figure 20.

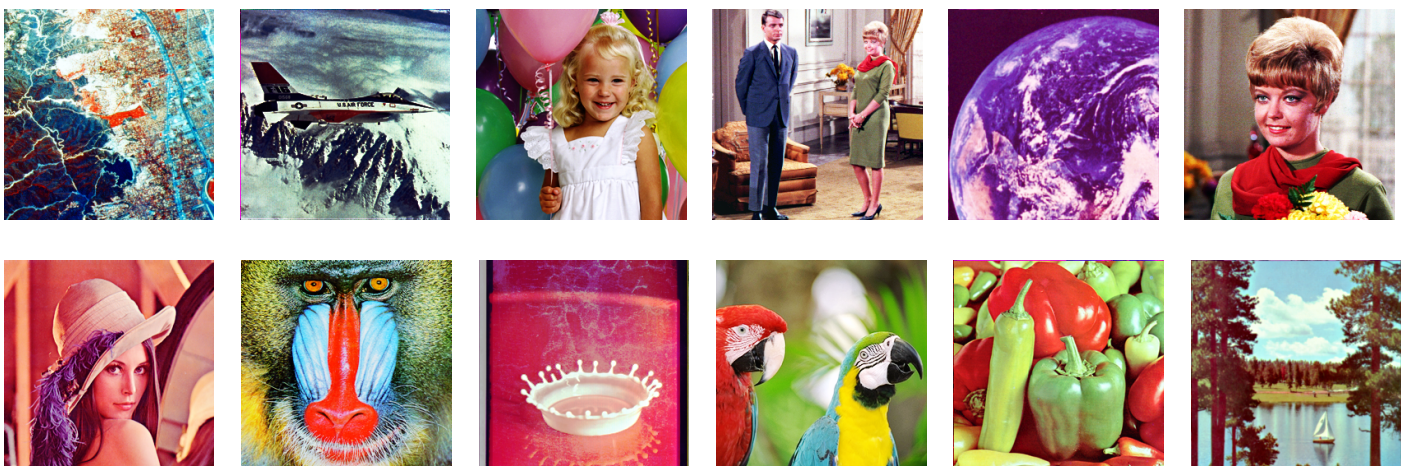


**Figure 20.** Color images in the SIDBA image dataset [28]: (a) Aerial, (b) Airplane, (c) Balloon, (d) Couple, (e) Earth, (f) Girl, (g) Lenna, (h) Mandrill, (i) Milkdrop, (j) Parrots, (k) Peppers, (l) Sailboat.



**Figure 21.** Results of separable histogram equalization with Nikolova and Steidl’s exact histogram equalization algorithm.

Figure 22 shows the results from the proposed histogram equalization method described in Section 5.2, where we can see that the hue of each image is closer to the original one than that in Figure 21.



**Figure 22.** Results of the proposed exact histogram equalization.

Figure 23 shows the RMSE of hue of each output image from the corresponding original image, where the vertical and horizontal axes denote the RMSE value and the images, respectively, and the blue and orange bars denote the separable and proposed methods, from which we observe that the proposed method preserves the original hue better than the separable method.

Figure 24 shows the results of a separable histogram specification that applies Nikolova and Steidl’s exact histogram specification algorithm to each color channel separately. Similarly to the above results of histogram equalization in Figure 21, here we can see the hue change from the original images.

On the other hand, Figure 25 shows the results by the proposed histogram specification method, where we can also see that the hue of each image is closer to the original one than that in Figure 24.

Figure 26 shows the RMSE of hue of each output image from the corresponding original image, from which we also observe that the proposed method preserves the original hue better than the separable method.

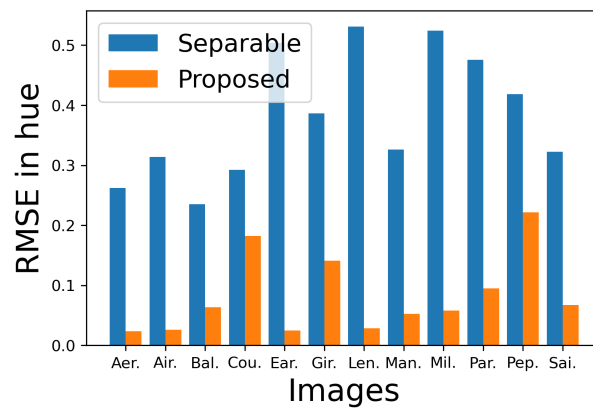


Figure 23. Comparison of the RMSE of hue of the images in Figures 21 and 22 from the original images.

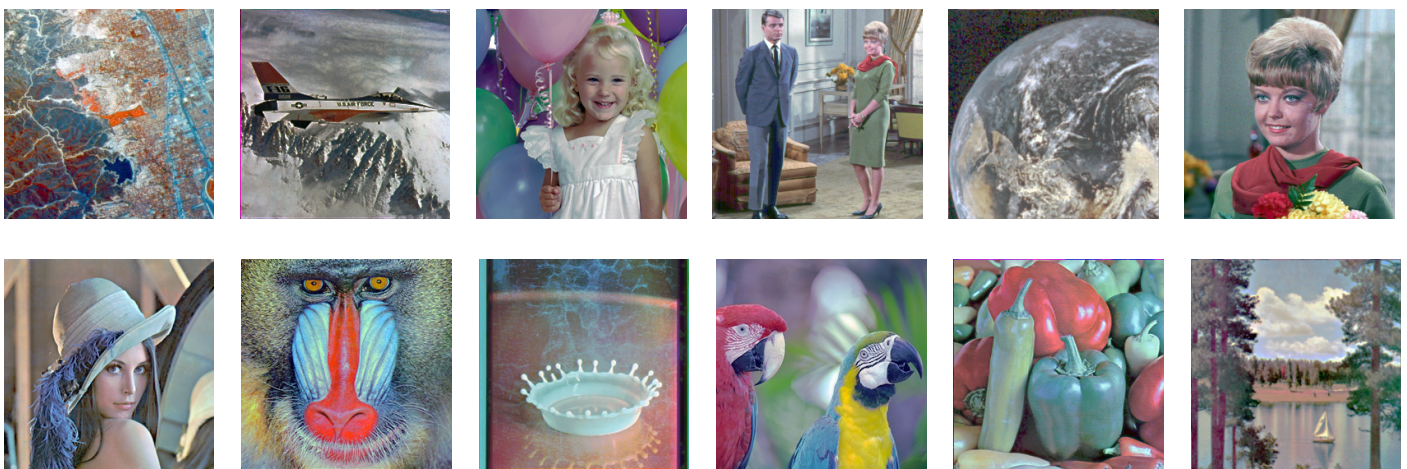


Figure 24. Results of the separable histogram specification with Nikolova and Steidl’s exact histogram specification algorithm.

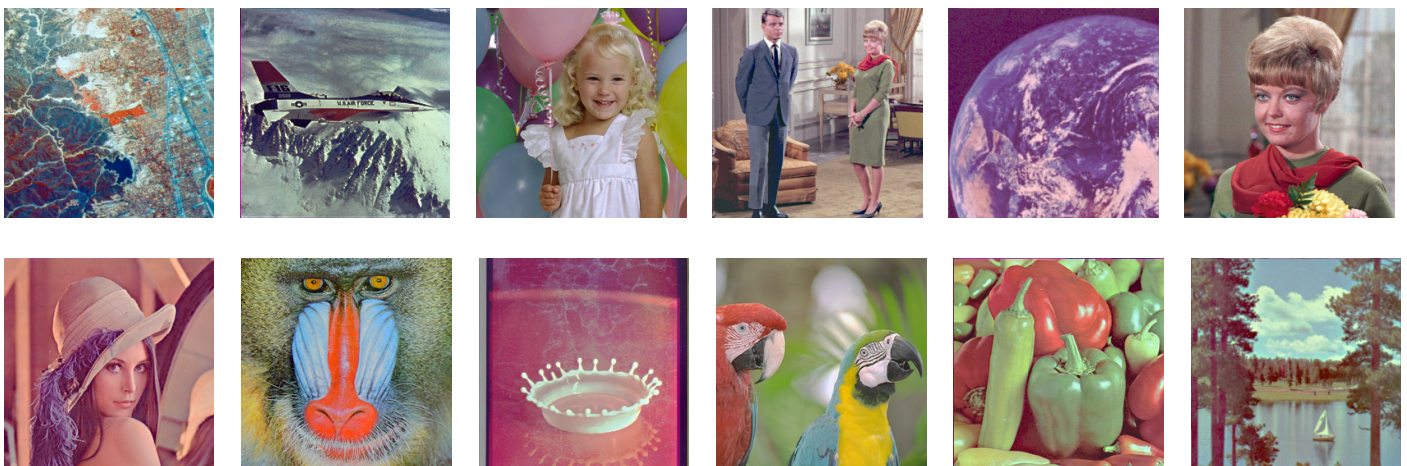


Figure 25. Results of the proposed exact histogram specification.

The above experimental results demonstrate the effectiveness of the proposed local contrast-based pixel ordering method for exact histogram equalization and specification compared with the state-of-the-art method by Nikolova and Steidl [13]. A drawback of Nikolova and Steidl’s algorithm observed in Figure 8b, where a stripe pattern occurred on flat areas, may be a newfound phenomenon. The proposed method can avoid the occurrence of such false patterns.

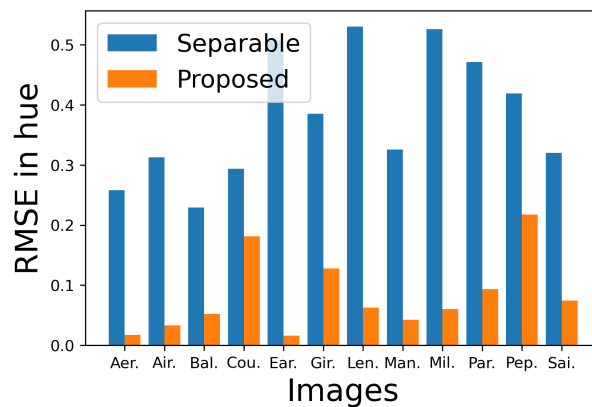


Figure 26. Comparison of the RMSE of hue of the images in Figures 24 and 25 from the original images.

In the above experiments, although we used a Gaussian target histogram for histogram specification constantly, we can choose other candidates such as multimodal distributions without any problems.

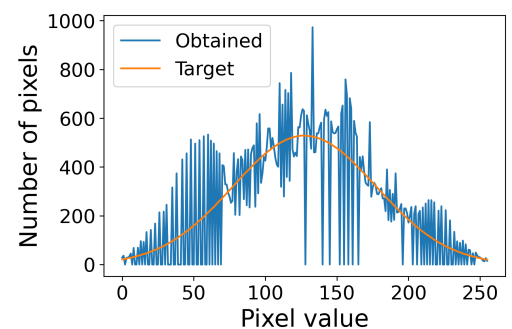
For the implementation of Nikolova and Steidl’s algorithm described in Algorithm 1, we need to use sparse matrices even if the input image is not so large, e.g., the original image in Figure 1a with  $256 \times 256$  pixels because the difference operator  $G$  in (7) becomes a very large matrix with  $r \times mn \approx 8.56 \times 10^9$  elements. On the other hand, the proposed method can be implemented without sparse matrices.

Additionally, we implemented Ramos’s recent algorithm for histogram specification [16]. Figure 27 shows the result of Ramos’s histogram specification, where Figure 27a shows the output image computed from the input image in Figure 1a with the Gaussian target histogram with mean 127.5 and standard deviation 50. Figure 27b shows the obtained histogram of the output image in Figure 27a in a blue line with the target histogram in the orange line. This figure shows that Ramos’s algorithm is not an exact histogram specification method, and the obtained histogram is similar to that of the conventional histogram specification shown in Figure 2b. On the other hand, the proposed method gives exactly the same histogram as a prescribed target histogram as shown in Figure 2c. In Ramos’s formulation of  $l^p$  norm minimization, we set  $p = \infty$ . For details, please refer to Ramos’s original paper [16]. For the implementation of Ramos’s algorithm, we need to use sparse matrices with about  $mn \times L$  elements as well as Nikolova and Steidl’s one.

A limitation of the proposed method is that users need to prepare a target histogram for histogram specification. However, it may be difficult to know the optimal shape of the histogram for a given image in advance. The estimation of suitable target histograms for given images will be a subject to be considered in the future.



(a) Original color image



(b) Histogram

Figure 27. Histogram specification by Ramos’s algorithm [16]: (a) histogram specified image; (b) obtained and target histograms.

## 7. Conclusions

In this paper, we proposed a method for ordering pixels in an image for exact histogram specification, and compared it with the state-of-the-art algorithm by Nikolova and Steidl [13]. We divided the problem of pixel ordering into small subproblems which can be solved memory-efficiently. This idea makes the problem of exact histogram specification more tractable than ever. The proposed method uses the Gaussian filter without approximation instead of the nonlinear filter used in Nikolova and Steidl's algorithm. We use the local contrast defined with the Gaussian-filtered image as a key to ordering pixels. As a result, it was experimentally demonstrated that we could avoid the occurrence of false patterns that were observed in the results obtained by Nikolova and Steidl's algorithm.

We also extended the proposed method for grayscale images to that for color images. Experimental results showed that the proposed method keeps the hue of the original images better than conventional separable methods of histogram equalization and specification for color images.

Our future work will include the development of a method for making a preferable target histogram for histogram specification depending on a given input image.

**Author Contributions:** Conceptualization, K.I.; methodology, K.I.; software, K.I.; validation, N.O. and K.H.; formal analysis, K.I.; investigation, K.I.; resources, K.I.; data curation, K.I.; writing—original draft preparation, K.I.; writing—review and editing, N.O. and K.H.; visualization, K.I.; supervision, K.H.; project administration, K.I.; funding acquisition, K.I. All authors have read and agreed to the published version of the manuscript.

**Funding:** This work was supported by JSPS KAKENHI Grant No. JP21K11964.

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** SIDBA, the Standard Image Data-BaSe can be downloaded from [http://www.ess.ic.kanagawa-it.ac.jp/app\\_images\\_j.html](http://www.ess.ic.kanagawa-it.ac.jp/app_images_j.html) (accessed on 21 June 2022).

**Conflicts of Interest:** The authors declare no conflict of interest.

### Abbreviations

The following abbreviations are used in this manuscript:

LM	Local mean method
WA	Wavelet-based exact pixel ordering algorithm
TV	Total variation
SIDBA	The Standard Image Data-BaSe
RMSE	Root mean squared error

## References

1. Pouli, T.; Reinhard, E.; Cunningham, D.W. *Image Statistics in Visual Computing*, 1st ed.; A. K. Peters, Ltd.: Natick, MA, USA, 2013.
2. Naik, S.K.; Murthy, C.A. Hue-preserving color image enhancement without gamut problem. *IEEE Trans. Image Process.* **2003**, *12*, 1591–1598. [[CrossRef](#)] [[PubMed](#)]
3. Rahman, M.A.; Fazle Rabbi, M.M.; Rahman, M.M.; Islam, M.M.; Islam, M.R. Histogram modification based lossy image compression scheme using Huffman coding. In Proceedings of the 2018 4th International Conference on Electrical Engineering and Information & Communication Technology (iCEEICT), Dhaka, Bangladesh, 13–15 September 2018; pp. 279–284. [[CrossRef](#)]
4. Qin, K.; Xu, K.; Liu, F.; Li, D. Image segmentation based on histogram analysis utilizing the cloud model. *Comput. Math. Appl.* **2011**, *62*, 2824–2833. [[CrossRef](#)]
5. Gonzalez, R.C.; Woods, R.E. *Digital Image Processing*; Prentice Hall: Upper Saddle River, NJ, USA, 2008; pp. 150–161.
6. Gonzalez, R.C.; Fittes, B.A. Gray-level transformations for interactive image enhancement. *Mech. Mach. Theory* **1977**, *12*, 111–122. [[CrossRef](#)]
7. Raheja, A.; Chawla, R.; Gupta, S.; Vashist, A. Controlling Over Enhancement of Images Using Histogram Equalization Technique. *IOP Conf. Ser. Mater. Sci. Eng.* **2020**, *804*, 012055. Available online: <https://iopscience.iop.org/article/10.1088/1757-899X/804/1/012055> (accessed on 28 June 2022). [[CrossRef](#)]

8. Nikolova, M.; Wen, Y.W.; Chan, R. Exact Histogram Specification for Digital Images Using a Variational Approach. *J. Math. Imaging Vis.* **2013**, *46*, 309–325. [[CrossRef](#)]
9. Hall, E.L. Almost Uniform Distributions for Computer Image Enhancement. *IEEE Trans. Comput.* **1974**, *C-23*, 207–208. [[CrossRef](#)]
10. Rosenfeld, A.; Kak, A.C. *Digital Picture Processing: Volume 1 and 2*; Academic Press: Orlando, FL, USA, 1982; ISBN 0-12-597301-2/0-12-597302-0.
11. Coltuc, D.; Bolon, P.; Chassery, J.-M. Exact histogram specification. *IEEE Trans. Image Process.* **2006**, *15*, 1143–1152. [[CrossRef](#)] [[PubMed](#)]
12. Wan, Y.; Shi, D. Joint Exact Histogram Specification and Image Enhancement Through the Wavelet Transform. *IEEE Trans. Image Process.* **2007**, *16*, 2245–2250. [[CrossRef](#)] [[PubMed](#)]
13. Nikolova, M.; Steidl, G. Fast Ordering Algorithm for Exact Histogram Specification. *IEEE Trans. Image Process.* **2014**, *23*, 5274–5283. [[CrossRef](#)]
14. Nikolova, M.; Steidl, G. Fast Hue and Range Preserving Histogram Specification: Theory and New Algorithms for Color Image Enhancement. *IEEE Trans. Image Process.* **2014**, *23*, 4087–4100. [[CrossRef](#)]
15. Lindeberg, T. Scale-space for discrete signals. *IEEE Trans. Pattern Anal. Mach. Intell.* **1990**, *12*, 234–254. [[CrossRef](#)]
16. Ramos, V.S.; Silveira, L.F.d.Q.; Silveira Júnior, L.G.d.Q. Histogram Specification by Assignment of Optimal Unique Values. *arXiv* **2021**, arXiv:2102.02378.
17. Pallavi, S.; Sannidhan, M.S.; Abhir Bhandary. A Comprehensive Review on Various State-of-the-Art Techniques for Image Enhancement. *Int. J. Eng. Technol.* **2018**, *7*, 86–864. Available online: <https://www.sciencepubco.com/index.php/ijet/article/view/19576> (accessed on 28 June 2022).
18. Yadav, G.; Maheshwari, S.; Agarwal, A. Contrast limited adaptive histogram equalization based enhancement for real time video system. In Proceedings of the 2014 International Conference on Advances in Computing, Communications and Informatics (ICACCI), Delhi, India, 24–27 September 2014; pp. 2392–2397. [[CrossRef](#)]
19. Kim, Y.-T. Contrast enhancement using brightness preserving bi-histogram equalization. *IEEE Trans. Consum. Electron.* **1997**, *43*, 1–8. [[CrossRef](#)]
20. Trongtirakul, T.; Agaian, S. Weighted Histogram Equalization Using Entropy of Probability Density Function. In Proceedings of the 44th Electrical Engineering Conference (EECON 44), Nan, Thailand, 17–19 November 2021; Volume 1, pp. 461–464. Available online: <https://arxiv.org/ftp/arxiv/papers/2111/2111.08578.pdf> (accessed on 28 June 2022).
21. Ren, X.; Yang, W.; Cheng, W.-H.; Liu, J. LR3M: Robust Low-Light Enhancement via Low-Rank Regularized Retinex Model. *IEEE Trans. Image Process.* **2020**, *29*, 5862–5876. [[CrossRef](#)] [[PubMed](#)]
22. Zhang, F.; Shao, Y.; Sun, Y.; Zhu, K.; Gao, C.; Sang, N. Unsupervised Low-Light Image Enhancement via Histogram Equalization Prior. *arXiv* **2021**, arXiv:2112.01766.
23. Hussain, K.; Rahman, S.; Rahman, M.; Khaled, S.M.; Wadud, A.A.; Hossain Khan, M.A.; Shoyaib, M. A histogram specification technique for dark image enhancement using a local transformation method. *IPSJ Trans. Comput. Vis. Appl.* **2018**, *10*, 3. [[CrossRef](#)]
24. Balado, F. Optimum Exact Histogram Specification. In Proceedings of the 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Calgary, AB, Canada, 15–20 April 2018; pp. 1413–1417. [[CrossRef](#)]
25. Vectorization (Mathematics) From Wikipedia, the Free Encyclopedia. Available online: [https://en.wikipedia.org/wiki/Vectorization\\_\(mathematics\)](https://en.wikipedia.org/wiki/Vectorization_(mathematics)) (accessed on 28 June 2022).
26. Weisstein, E.W. “Transpose.” From MathWorld—A Wolfram Web Resource. Available online: <https://mathworld.wolfram.com/Transpose.html> (accessed on 29 May 2022).
27. Weisstein, E.W. “Floor Function.” From MathWorld—A Wolfram Web Resource. Available online: <https://mathworld.wolfram.com/FloorFunction.html> (accessed on 13 June 2022).
28. The Standard Image Data-BASE (SIDBA). Available online: [http://www.ess.ic.kanagawa-it.ac.jp/app\\_images\\_j.html](http://www.ess.ic.kanagawa-it.ac.jp/app_images_j.html) (accessed on 21 June 2022).