

# CONTRAST ENHANCEMENT BASED ON LAYERED DIFFERENCE REPRESENTATION

Chulwoo Lee, Chul Lee, and Chang-Su Kim

School of Electrical Engineering, Korea University, Seoul, Korea  
E-mails: {wiserein, kayne, changsukim}@korea.ac.kr

## ABSTRACT

A novel contrast enhancement algorithm based on the layered difference representation is proposed in this work. We first represent gray-level differences at multiple layers in a tree-like structure. Then, based on the observation that gray-level differences, occurring more frequently in the input image, should be more emphasized in the output image, we solve a constrained optimization problem to derive the transformation function at each layer. Finally, we aggregate the transformation functions at all layers into the overall transformation function. Simulation results demonstrate that the proposed algorithm enhances images efficiently in terms of both objective quality and subjective quality.

**Index Terms**— Contrast enhancement, layered difference representation, histogram equalization, and constrained optimization.

## 1. INTRODUCTION

In spite of recent advances in imaging technology, a captured scene often fails to preserve image details faithfully or yields poor contrast ratio due to the limited dynamic range. Contrast enhancement techniques can alleviate these problems by increasing contrast ratios and bringing out hidden details. Therefore, contrast enhancement is an essential step in various image processing applications, such as digital photography and visual surveillance.

Conventional contrast enhancement techniques can be categorized into global and local approaches. A global approach derives a single transformation function, which maps input pixel intensities to output pixel intensities, and applies it to all pixels in an entire image. Gamma correction, based on the simple power law, and histogram equalization (HE), which attempts to make the histogram of pixel intensities as uniform as possible, are two popular global contrast enhancement techniques [1]. A local approach, on the other hand, derives and applies the transformation function for each pixel adaptively to the information in neighboring pixel values. However, in general, a local approach demands higher computational complexity and its level of contrast enhancement is harder to control. Therefore, being more stable, global contrast enhancement techniques are more widely used for general purposes than local ones.

Recently, several global techniques have been proposed to overcome the drawbacks of HE, such as contrast overstretching, noise amplification, or contour artifacts in output images. For example, Wang and Ward [2] clamped large histogram values to a certain threshold and then modified the resulting histogram using the power

law. Arici *et al.* [3] constructed the histogram of an input image adaptively to the image characteristics. Their algorithm reduces large histogram values for smooth areas corresponding to background regions, in order to focus on the enhancement of foreground objects. Lee *et al.* [4] employed a logarithm function to reduce large histogram values more effectively, preventing the transformation function from having too steep slopes. Celik and Tjahjadi [5] partitioned an input histogram by modeling the gray-level distribution with a Gaussian mixture model, and then enhanced the contrast in each partition. These recent global contrast enhancement techniques attempt to manipulate the histogram of an input image, but there are other features of the input image that can be effectively used for the contrast enhancement.

In this work, we propose a novel global contrast enhancement algorithm, based on the layered difference representation. We first make an observation that gray-level differences, occurring frequently in the input image, should be amplified in the output image in order to enhance the contrast. We then represent output gray-level differences and the transformation function in a tree-like layered structure. Then, at each layer, we formulate a constrained optimization problem for the contrast enhancement and obtain the difference vector, which is equivalent to the transformation function. Finally, we aggregate the information in the difference vectors at all layers into the overall difference vector. Simulation results demonstrate that the proposed algorithm yields higher image qualities than the conventional contrast enhancement techniques [2, 3].

The rest of this paper is organized as follows. Section 2 describes the proposed algorithm. Section 3 discusses experimental results. Finally, Section 4 concludes the paper.

## 2. PROPOSED ALGORITHM

### 2.1. Layered Difference Representation

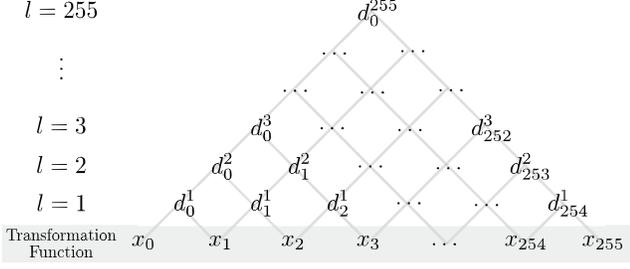
Since the human visual system is more sensitive to gray-level differences between neighboring objects than to absolute gray-levels, contrast enhancement can be achieved by emphasizing the differences. Thus, gray-level differences, which occur frequently in the input image, should be amplified in the output image. We develop a contrast enhancement algorithm based on this observation.

For the sake of notational simplicity, let us consider the typical 8-bit imaging system, *i.e.*, the maximum gray-level is 255. Let  $\mathbf{x} = [x_0, x_1, \dots, x_{255}]^T$  denote the transformation function that maps gray-level  $k$  in the input image to gray-level  $x_k$  in the output image [4]. To derive a desirable transformation function, we introduce the layered difference representation. Suppose that a pair of adjacent pixels in the input image have gray-levels  $k$  and  $k + l$ . Then, they are mapped to gray-levels  $x_k$  and  $x_{k+l}$  in the output image. The difference variable  $d_k^l$  at layer  $l$  is then defined as

$$d_k^l = x_{k+l} - x_k \quad \text{for } 0 \leq k \leq 255 - l. \quad (1)$$

---

This work was supported partly by the Global Frontier R&D Program on <Human-centered Interaction for Coexistence> funded by the National Research Foundation of Korea grant funded by the Korean Government(MEST) (NRF-M1AXA003-2011-0031648), and partly by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science and Technology (No. 2012-0000916).



**Fig. 1.** The transformation function  $\mathbf{x} = [x_0, x_1, \dots, x_{255}]^T$  and the difference variables  $d_k^l$ 's in a tree-like pyramidal structure.

Notice that the input gray level difference  $l$  between  $k$  and  $k + l$  is mapped to the output gray-level difference  $d_k^l$ .

Let  $h(k, k + l)$ , which is referred to as the 2D histogram of gray-level pairs, denotes the number of pairs of adjacent pixels with values  $k$  and  $k + l$  in the input image. Different types of neighborhood can be employed, but the simple 4-adjacency is used in this work. A large  $h(k, k + l)$  indicates that the gray-level pair  $(k, k + l)$  appears frequently in the input image. In such a case, the difference variable  $d_k^l$  should be large, so that those frequent pairs have a large gray-level difference in the output image and thus the image contrast is enhanced. In other words,  $d_k^l$  should be proportional to  $h(k, k + l)$  for the contrast enhancement. Therefore, at each layer, we have the desirable relationship

$$d_k^l = \kappa_l \times h(k, k + l), \quad (2)$$

where  $\kappa_l$  is a normalizing constant at layer  $l$ .

Our objective is to design the transformation function  $\mathbf{x}$  that satisfies the system of equations in (1) and (2). However, this system of equations does not have an exact solution, and we develop an algorithm to provide an approximate solution instead. Before describing the algorithm, let us state two elementary properties of this system of equations. Fig. 1 illustrates the transformation function  $\mathbf{x}$  and the difference variables  $d_k^l$ 's in a tree-like pyramidal structure.<sup>1</sup>

First, the transformation function  $\mathbf{x}$  can be completely determined by the difference variables  $d_k^1$ 's at layer 1. Specifically,

$$x_k = \sum_{i=0}^{k-1} d_i^1 \quad \text{for } 1 \leq k \leq 255, \quad (3)$$

and  $x_0 = 0$  since the darkest level in the input image should be mapped to the darkest level in the output to preserve the dynamic range. Therefore, we can solve the system of equations in terms of  $d_k^1$ 's and then reconstruct the transformation function using (3).

Second, the difference variables  $d_k^l$ 's at higher layers can be also expressed by those  $d_k^1$ 's at layer 1. From (1), we have

$$d_k^l = \sum_{i=k}^{k+l-1} d_i^1. \quad (4)$$

In other words, in the tree-like structure in Fig. 1, each difference variable  $d_k^l$  at layer  $l$  is equal to the sum of its descendants at layer 1. Thus, the difference variables  $d_k^l$ 's have a strong structure across layers. On the other hand, the 2D histogram values,  $h(k, k + l)$ 's, do not have such a strong structure. Thus, we cannot satisfy the proportional relationships in (2) exactly, and should obtain an approximate solution.

<sup>1</sup>It is not a tree, since two parents cannot have a common child in a tree.

## 2.2. Intra-Layer Optimization

Given the histogram  $h(k, k + l)$  of the input image, we should decide the difference variables  $d_k^l$ 's at layer 1, which satisfy the system of equations (2) and (4). However, as mentioned previously, the system of equations does not have an exact solution due to the strong structure of  $d_k^l$ 's across layers. Therefore, we first ignore the inter-layer dependencies, and solve the system of equations at each layer separately. In other words, for each fixed layer index  $l$ , we obtain and solve the reduced system of equations (2) and (4). We then aggregate the separate solutions for distinct layers to form the overall solution, as will be described in Section 2.3.

Let us describe how to choose the normalizing constant  $\kappa_l$  in (2) first. By summing up the equalities in (2) for all possible  $k$ 's at layer  $l$ , we have

$$\kappa_l = \frac{\sum_{k=0}^{255-l} d_k^l}{\sum_{k=0}^{255-l} h(k, k + l)}. \quad (5)$$

The numerator  $\sum_k d_k^l$  can be approximated as follows. At layer 1,  $\sum_k d_k^1 = x_{255} = 255$ , since the maximum gray-level in the input should be mapped to the maximum gray-level in the output. At layer 2, since  $d_k^2 = d_k^1 + d_{k+1}^1$  from (4), we have

$$\begin{aligned} \sum_k d_k^2 &= d_0^1 + 2 \times (d_1^1 + \dots + d_{253}^1) + d_{254}^1 \\ &\simeq 2 \times (d_0^1 + \dots + d_{253}^1) \\ &\simeq 2 \times 254, \end{aligned} \quad (6)$$

by approximating the difference variables  $d_k^1$ 's to their average values, which are equal to 1. Similarly,  $\sum_k d_k^l \simeq l \cdot (256 - l)$ . Then, from (2) and (5), we have

$$d_k^l = l \cdot (256 - l) \cdot \frac{h(k, k + l)}{\sum_{i=0}^{255-l} h(i, i + l)}. \quad (7)$$

At each layer  $l$ , we compute  $d_k^l$ 's from the input histogram by (7). Also, notice that, since each  $d_k^l$  is the sum of  $d_k^1$ 's in (4), we can form the linear equation

$$\mathbf{A}_l \mathbf{y}_l = \mathbf{d}_l, \quad (8)$$

where  $\mathbf{A}_l \in \mathbb{R}^{(256-l) \times 255}$  is a binary matrix composed of 0 and 1,  $\mathbf{y}_l = [d_0^l, d_1^l, \dots, d_{254}^l]^T$  is the difference vector to be determined, and  $\mathbf{d}_l = [d_0^l, d_1^l, \dots, d_{255-l}^l]^T$  is the column vector, the elements of which are computed from (7). For instance, at layer 2, the linear equation  $\mathbf{A}_2 \mathbf{y}_2 = \mathbf{d}_2$  is given by

$$\begin{bmatrix} 1 & 1 & 0 & \dots & 0 & 0 & 0 \\ 0 & 1 & 1 & \dots & 0 & 0 & 0 \\ 0 & 0 & 1 & \dots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & 1 & 1 & 0 \\ 0 & 0 & 0 & \dots & 0 & 1 & 1 \end{bmatrix} \begin{bmatrix} d_0^1 \\ d_1^1 \\ d_2^1 \\ \vdots \\ d_{253}^1 \\ d_{254}^1 \end{bmatrix} = \begin{bmatrix} d_0^2 \\ d_1^2 \\ d_2^2 \\ \vdots \\ d_{252}^2 \\ d_{253}^2 \end{bmatrix}. \quad (9)$$

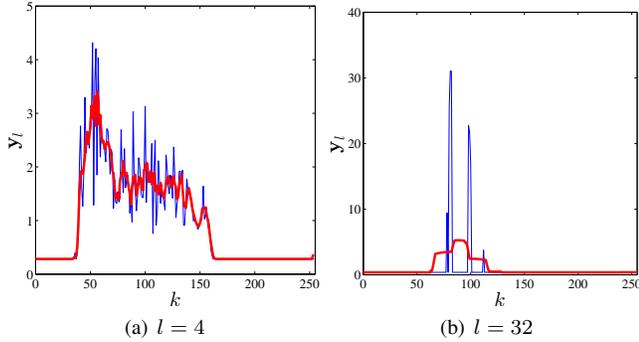
In this way, at each layer  $l$ , we obtain the difference vector  $\mathbf{y}_l$  by solving the constrained optimization problem,

$$\text{minimize} \quad \|\mathbf{A}_l \mathbf{y}_l - \mathbf{d}_l\|^2 \quad (10)$$

$$\text{subject to} \quad \mathbf{y}_l \succeq \mathbf{0}, \quad (11)$$

$$\mathbf{1}^T \mathbf{y}_l = 255, \quad (12)$$

where  $\mathbf{1}$  denotes the column vector, all elements of which are 1. Also,  $\mathbf{a} \succeq \mathbf{0}$  means that all elements in vector  $\mathbf{a}$  are greater than or



**Fig. 2.** The difference vector  $\mathbf{y}_l$  (solid) is convolved with a Gaussian kernel to yield the smoothed version  $\tilde{\mathbf{y}}_l$  (dashed).

equal to 0. The inequality in (11) and the equality in (12) are the constraints to guarantee that the transformation function is monotonically increasing and preserves the dynamic range, respectively. Since (10) is a quadratic cost function of  $\mathbf{y}_l$ , the optimization problem can be efficiently solved by various techniques. In this work, we employ the active-set method [6] to obtain the difference vector  $\mathbf{y}_l$  at each layer  $l$ .

### 2.3. Inter-Layer Aggregation

By performing the intra-layer optimization at each layer, we obtain 255 difference vectors  $\mathbf{y}_l$ ,  $1 \leq l \leq 255$ , respectively. We aggregate the information in these difference vectors to form an overall difference vector  $\bar{\mathbf{y}}$ .

Before the aggregation, we first smooth each  $\mathbf{y}_l$ . The smoothing is necessary, since  $h(k, k+l)$  in (7) generally becomes sparser as  $l$  gets larger. Thus, at a large  $l$ , most elements in  $\mathbf{d}_l$  are zero, which also leads to a sparse solution  $\mathbf{y}_l$  consisting of most zeros and a few high peaks. Fig. 2 shows exemplar  $\mathbf{y}_l$ 's at  $l = 4$  and 32. Note that, at  $l = 32$ , most elements are zero except for the two high peaks. This causes artifacts, such as contrast overstretching and noise amplification, as in the conventional HE. Therefore, to reduce these artifacts, we convolve  $\mathbf{y}_l$  with a Gaussian kernel,

$$\tilde{\mathbf{y}}_l = \mathbf{y}_l * \mathbf{g}_l, \quad (13)$$

where  $*$  denotes the convolution operation, and  $\mathbf{g}_l$  is the Gaussian kernel at layer  $l$ . At a high  $l$ , the kernel size should be big to effectively remove a sudden peak value and obtain a smooth difference vector. Thus, in this work, we set both the size and the variance of the Gaussian kernel to be  $l$ . In Fig. 2, the dashed curves depict the smoothed difference vectors.

Next, we aggregate  $\tilde{\mathbf{y}}_l$ 's at all layers to yield the overall difference vector  $\bar{\mathbf{y}} = [\bar{d}_0^1, \bar{d}_1^1, \dots, \bar{d}_{254}^1]^T$ . Note that we obtain  $\tilde{\mathbf{y}}_l$  from the histogram  $h(k, k+l)$  of pixel pairs, whose gray-level differences are  $l$ . It is hence reasonable to assume that the reliability of  $\tilde{\mathbf{y}}_l$  is proportional to  $s_l = \sum_k h(k, k+l)$ , which is the total number of pixel pairs with gray-level differences  $l$ . Therefore, we set  $\bar{\mathbf{y}}$  to be the weighted average of  $\tilde{\mathbf{y}}_l$ 's, given by

$$\bar{\mathbf{y}} = \sum_{l=\mu}^{255} \log\left(\frac{s_l}{10^\rho} + 1\right) \times \tilde{\mathbf{y}}_l \quad (14)$$

where  $\rho$  and  $\mu$  are user-controllable parameters. Since  $s_l$  tends to be exponentially decreasing, we use the logarithm function with the

**Table 1.** Objective assessment of contrast enhancement.

	Input	WTHE	WAHE	Proposed
AMBE	-	6.482	7.647	<b>5.208</b>
DE	<b>2.154</b>	2.144	2.135	2.113
EME	9.878	9.485	9.265	<b>12.811</b>
PixDist	26.315	33.164	32.694	<b>34.125</b>

parameter  $\rho$  to take into account the information at high layers, as well as that at low layers, in the aggregation. Moreover, at very low  $l$ 's, although the histogram values  $h(k, k+l)$ 's are high, those observation data are often corrupted due to noise components in the input image and hence are not reliable. Thus, by setting the parameter  $\mu$ , we discard the first few  $\tilde{\mathbf{y}}_l$ 's in the aggregation step. In this work, we fix  $\rho$  and  $\mu$  to 10 and 3, respectively.

Finally, we obtain the transformation function  $\mathbf{x}$  from the overall difference vector  $\bar{\mathbf{y}}$  via

$$x_k = \sum_{i=0}^{k-1} \bar{d}_i^1 \quad \text{for } 1 \leq k \leq 255. \quad (15)$$

## 3. EXPERIMENTAL RESULTS

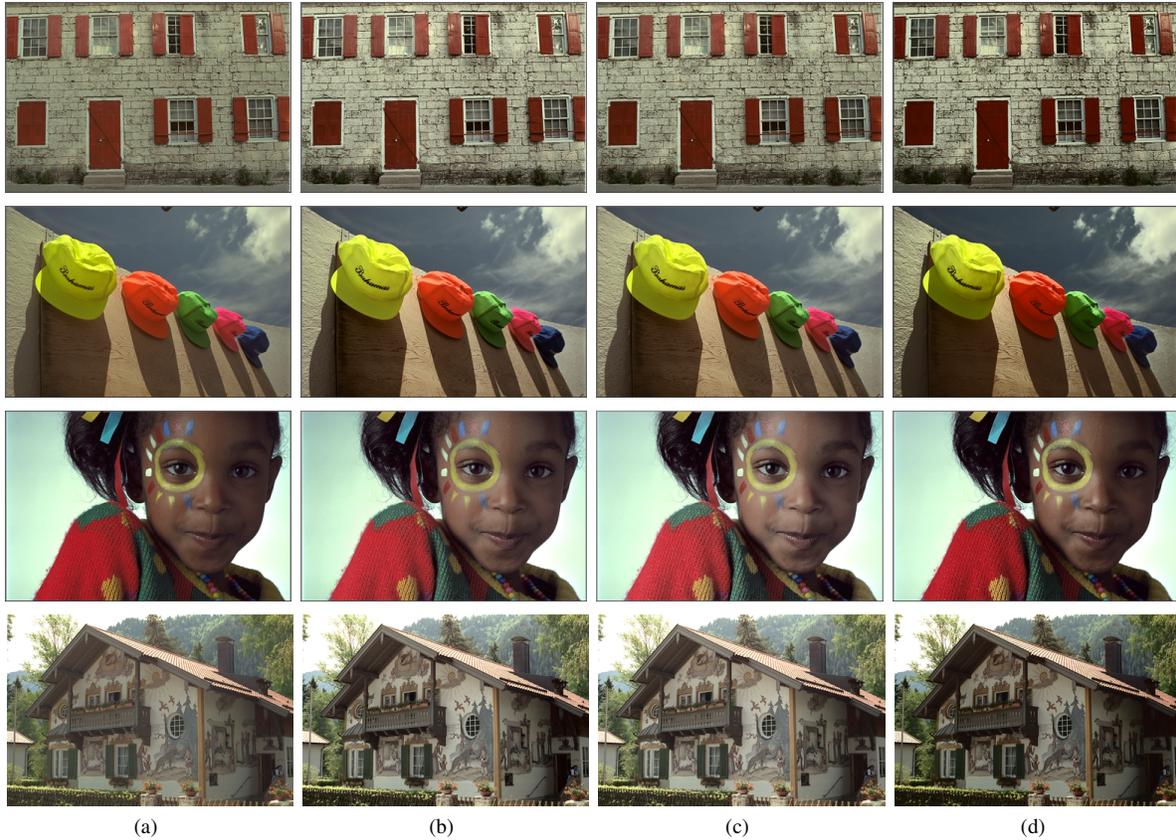
We evaluate the performance of the proposed algorithm on 27 test images from the Kodak Lossless True Color Image Suite<sup>2</sup> and the USC-SIPI database<sup>3</sup>. We compare the proposed algorithm with the conventional methods in [2] and [3], which are referred to as weighted thresholded histogram equalization (WTHE) and weighted approximated histogram equalization (WAHE), respectively. Parameters  $r$  in WTHE and  $g$  in WAHE is fixed to 0.3 and 1.5, respectively, to achieve the best subjective qualities on the test images.

First, we assess the enhancement performance objectively using four quality metrics: absolute mean brightness error (AMBE) [3], discrete entropy (DE) [3], measure of enhancement (EME) [3], and PixDist [7]. Table 1 lists the average performance on the 27 test images. AMBE measures the absolute difference between input and output means. A lower value implies that the corresponding algorithm well preserves the mean brightness of an input image. Note that the proposed algorithm incurs the lowest brightness change. DE measures the amount of information in an image, where a higher value indicates that the image contains more randomness and variations. The proposed algorithm has a poor rank for DE. However, since DE is calculated from the probability mass function of gray-levels, it can not faithfully assess the enhancement of image structures. EME approximates the average contrast in an image by dividing the image into non-overlapping blocks, computing a score based on the minimum and maximum gray-levels in each block, and averaging them. Since the proposed algorithm enhances local details efficiently, it yields the highest EME score. PixDist computes the average mutual intensity difference for all pixel pairs in an image. The proposed algorithm provides the best performance in terms of the PixDist measure as well.

Next, we compare the qualities of output images subjectively. In Fig. 3, only the result images on ‘‘Windows,’’ ‘‘Hats,’’ ‘‘Girl,’’ and ‘‘House’’ are presented because of the page limitation. The original images look dull due to low contrast ratios. WTHE is based on HE, but it sets the limit on an input histogram to avoid contrast

<sup>2</sup><http://r0k.us/graphics/kodak/>

<sup>3</sup><http://sipi.usc.edu/database/>



**Fig. 3.** Contrast enhancement results on the test images “Windows,” “Hats,” “Woman,” “Girl,” and “House”: (a) Original input images, (b) WTHe [2], (c) WAHE [3], and (d) the proposed algorithm.

overstretching. We see that it provides satisfactory results without annoying artifacts. WAHE exploits the spatial variance information to enhance the contrast on textured regions more efficiently than on homogeneous regions. Thus, it enhances object details by sacrificing background details. On the other hand, the proposed algorithm uses the occurrence frequency of gray-level differences to enhance the output image contrast. Therefore, the proposed algorithm improves the contrast especially on repeated patterns or local details, such as cracks on the bricked wall in “Windows,” wrinkles and letters on the hats in “Hats,” and tree leaves and wall patterns in “House.” As a result, we see that the proposed algorithm offers clear image structures and vivid colors by improving local details.

#### 4. CONCLUSIONS

We proposed a novel contrast enhancement algorithm using the layered difference representation, in which the statistical information of gray-level differences between neighboring pixels in an input image is exploited to control output gray-level differences. We derived the relationship between difference variables and image contrast and formulated the contrast enhancement as a constrained optimization problem. By solving the problem at each layer, the proposed algorithm obtains the difference vector. Then, the proposed algorithm aggregates the difference vectors at all layers into the overall difference vector. Simulation results demonstrated that the proposed algorithm provides better image qualities than the conventional methods.

#### 5. REFERENCES

- [1] R. C. Gonzalez and R. E. Woods, *Digital Image Processing*, Prentice Hall, 3rd edition, 2007.
- [2] 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.
- [3] 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, Sept. 2009.
- [4] C. Lee, C. Lee, Y.-Y. Lee, and C.-S. Kim, “Power-constrained contrast enhancement for emissive displays based on histogram equalization,” *IEEE Trans. Image Process.*, vol. 21, no. 1, pp. 80–93, Jan. 2012.
- [5] T. Celik and T. Tjahjadi, “Automatic image equalization and contrast enhancement using Gaussian mixture modeling,” *IEEE Trans. Image Process.*, vol. 21, no. 1, pp. 145–156, Jan. 2012.
- [6] P. Gill, W. Murray, and M. Wright, *Numerical Linear Algebra and Optimization*, vol. 1, Redwood City, Calif.: Addison-Wesley, 1991.
- [7] Z. Y. Chen, B. R. Abidi, D. L. Page, and M. A. Abidi, “Gray-level grouping (GLG): An automatic method for optimized image contrast enhancement-Part I: The basic method,” *IEEE Trans. Image Process.*, vol. 15, no. 8, pp. 2290–2302, Aug. 2006.