Gray-Level Grouping (GLG): An Automatic Method for Optimized Image Contrast Enhancement—Part I: The Basic Method

ZhiYu Chen, *Senior Member, IEEE*, Besma R. Abidi, *Senior Member, IEEE*, David L. Page, *Member, IEEE*, and Mongi A. Abidi, *Member, IEEE*

***Abstract*—Contrast enhancement has an important role in image processing applications. Conventional contrast enhancement techniques either often fail to produce satisfactory results for a broad variety of low-contrast images, or cannot be automatically applied to different images, because their parameters must be specified manually to produce a satisfactory result for a given image. This paper describes a new automatic method for contrast enhancement. The basic procedure is to first group the histogram components of a low-contrast image into a proper number of bins according to a selected criterion, then redistribute these bins uniformly over the grayscale, and finally ungroup the previously grouped gray-levels. Accordingly, this new technique is named gray-level grouping (GLG). GLG not only produces results superior to conventional contrast enhancement techniques, but is also fully automatic in most circumstances, and is applicable to a broad variety of images. An extension of GLG, selective GLG (SGLG), and its variations will be discussed in Part II of this paper. SGLG selectively groups and ungroups histogram components to achieve specific application purposes, such as eliminating background noise, enhancing a specific segment of the histogram, and so on. The extension of GLG to color images will also be discussed in Part II.**

***Index Terms*—Contrast enhancement, gray-level grouping, histogram, quality measure.**

I. Introduction and Related Work

CONTRAST enhancement has an important role in image processing applications. Numerous contrast enhancement techniques exist in literature, such as gray-level transformation based techniques (e.g., logarithm transformation, power-law transformation, piecewise-linear transformation, etc.) and histogram processing techniques (e.g., histogram equalization (HE), histogram specification, etc.) [1]. Conventional contrast enhancement techniques generally yield satisfactory results if the proper technique is selected for a given application along with the proper processing parameters. However, conventional contrast enhancement techniques often fail in producing satisfactory results for a broad range of low-contrast images, such as images characterized by the fact that the amplitudes of their histogram components are very high at one or several locations on the grayscale, while they are very small, but not zero, in the rest of the grayscale. This makes it difficult to increase the image contrast by simply stretching its histogram or by using simple gray-level transformations. The high amplitude of the histogram components corresponding to the image background also often prevents the use of the HE techniques, which could cause a washed-out effect on the appearance of the output image and/or amplify the background noise. Figs. 1 and 4 show examples of low-contrast images and the results of treating them with conventional contrast enhancement techniques.

Fig. 1(a) shows an original low-contrast image of the Mars moon, Phobos, and Fig. 2(a) its histogram. Fig. 1(b) is the result of HE, exhibiting a washed-out appearance which is not acceptable for many applications. The cause for the washed-out appearance is that the left half of the grayscale on the histogram of the equalized image is simply empty, as shown in Fig. 2(b). Fig. 1(c) is the resulting image of histogram specification, and Fig. 2(c) its histogram, which is better than the HE result, but still has an unsatisfactory appearance. More importantly, one major disadvantage of the histogram specification technique is that the desired histogram of the resulting image has to be specified manually, and this precludes the technique from being applied automatically. The manually specified desired histogram used in the treatment is depicted in Fig. 3 [1]. Fig. 4(a) shows a low-contrast X-ray image of luggage. Its HE result in Fig. 4(b) also has a washed-out look.

Numerous advanced histogram-based contrast enhancement techniques have been developed, but most of them are derivatives of conventional techniques (e.g., HE, etc.) [2]–[25], such as bi-histogram equalization (BHE), block-overlapped histogram equalization, multi-scale adaptive histogram equalization, shape preserving local histogram modification, and so on. The mean brightness of histogram-equalized image is always the middle gray-level regardless of the input mean, and this is undesirable in certain applications where brightness preservation is necessary. This characteristic of HE may also lead to a washed-out appearance, amplified noise or other annoying artifacts in the resulting image. BHE was proposed to preserve the brightness by separating the input image's histogram into two parts based on its mean—one ranges from the minimum gray level to the mean gray level, the other from the mean to the maximum. The

Manuscript received February 1, 2005; revised August 26, 2005. This work was supported in part by the DOE University Research Program in Robotics under Grant DOE-DE-FG02-86NE37968, in part by the DOD/TACOM/NAC/ARC Program R01-1344-18, in part by the FAA/NSSA Program R01-1344-48/49, and in part by the Office of Naval Research under Grant N000143010022. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Hassan Foroosh.

The authors are with the Electrical and Computer Engineering Department, University of Tennessee, Knoxville, TN 37996-2100 USA (e-mail: zychen@utk.edu; besma@utk.edu; dpage@utk.edu; abidi@utk.edu).

Digital Object Identifier 10.1109/TIP.2006.875204

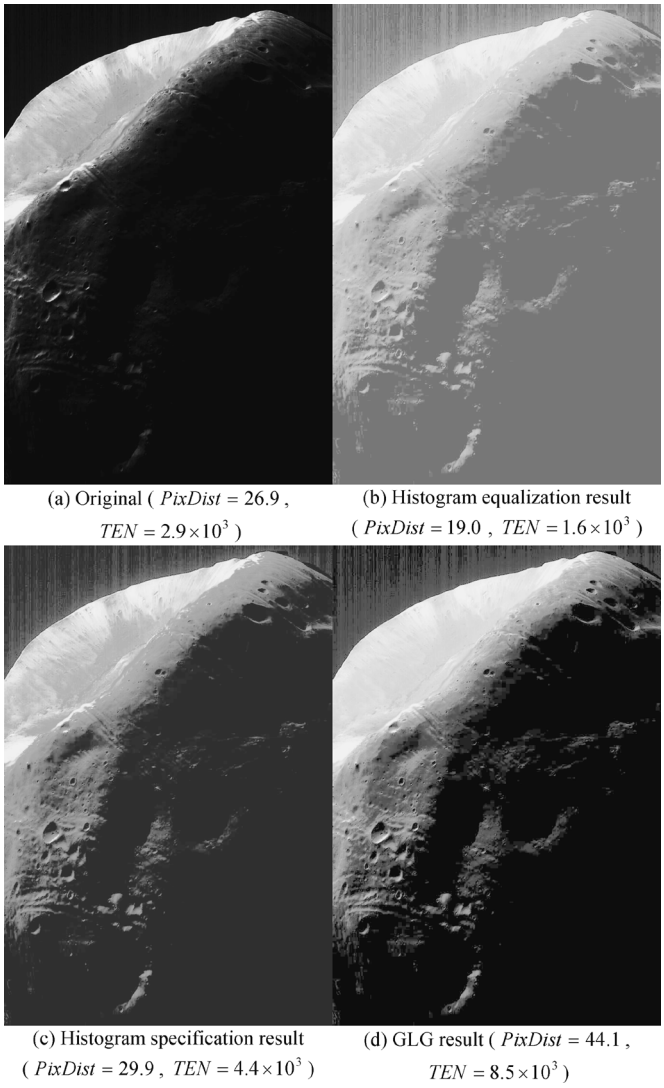


Fig. 1. Mars moon—Phobos. (a) Low-contrast original image. (b) Result of HE, which has a washed-out appearance. (c) Result of histogram specification, though better than the HE result, still has an unsatisfactory appearance. Moreover, this technique is not automatic; the desired histogram profile must be manually specified. (d) Result of gray-level grouping, has a crisper look. The result is produced fully automatically ($PixDist$ and TEN are quality measures that will be discussed in Section III). (Original image courtesy of Dr. R. C. Gonzalez [1]).

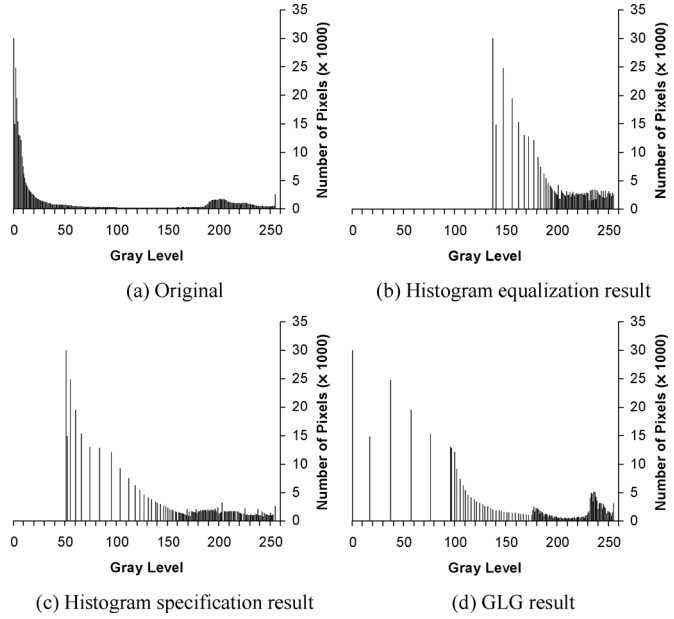


Fig. 2. Histograms of the images of Phobos in Fig. 1. (a) Histogram of the low-contrast original image. (b) Result of HE. Nearly half of the grayscale is wasted. (c) Result of histogram specification. The desired histogram profile must be manually specified. (d) Result of gray-level grouping. The grayscale has been utilized fully and efficiently (the leftmost component in the histograms is the largest peak whose actual amplitude is 3.67×10^5. It is truncated so that the rest of the histograms can be displayed on a proper scale).

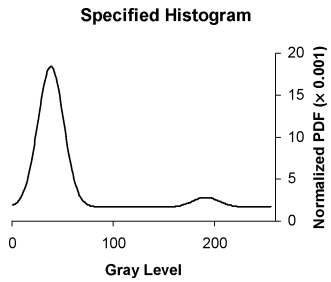
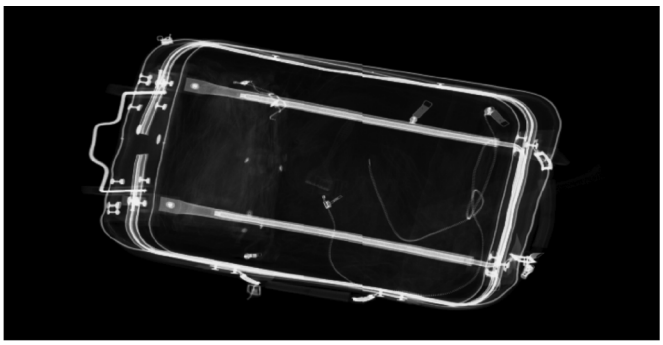


Fig. 3. Manually specified desired histogram profile used to produce the histogram specification result in Fig. 1(c).

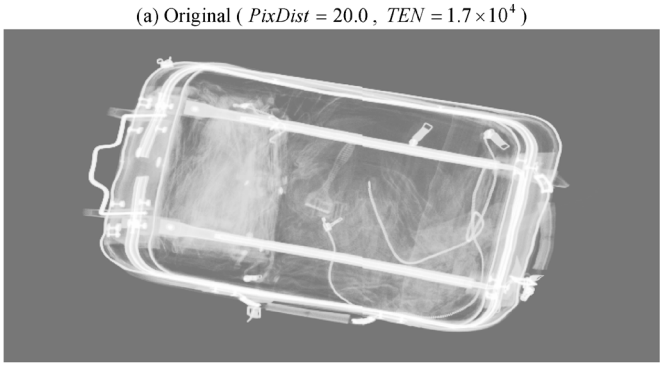
two histograms are then equalized independently [2]. Equal area dualistic subimage histogram equalization (DSIHE) is similar to BHE except that DSIHE separates the histogram at the median gray level—the gray level with cumulative probability equal to 0.5 instead of the mean [3]. These two techniques usually outperform the basic HE technique. However, they have the same limitations of HE and cannot enhance some images well, as they still perform the HE operation in each grayscale segment, just limiting the drawbacks of HE within each grayscale segment.

The global HE method cannot adapt to local brightness features of the input image because it uses histogram information over the whole image. This fact limits the contrast-stretching ratio in some parts of the image, and causes significant contrast losses in the background and other small regions. To overcome this limitation, some local histogram-equalization methods have been developed. A natural extension of global histogram equalization is termed adaptive histogram equalization (AHE), which divides the input image into an array of subimages, each subimage is histogram-equalized independently, and then the processed subimages are fused together with bilinear interpolation [4].

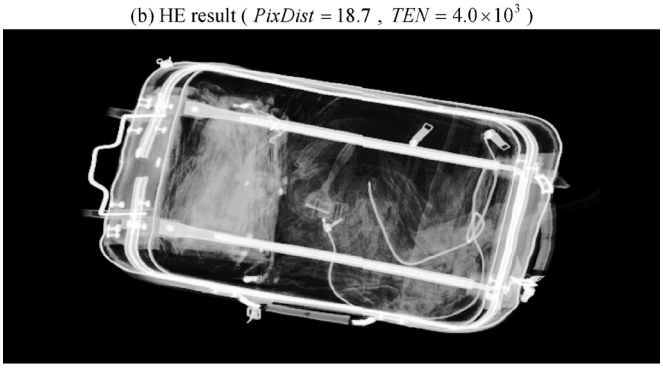
Another local method is called block-overlapped histogram equalization [5], in which a rectangular subimage of the input image is first defined, a histogram of that block is obtained, and then its histogram-equalization function is determined. Thereafter, the center pixel of the block is histogram equalized using this function. The center of the rectangular block is then moved to the adjacent pixel and the histogram equalization is repeated. This procedure is repeated pixel by pixel for all input pixels. Since local histogram equalization must be performed for all pixels in the entire image, the computational complexity of this method is very high. Instead of using rectangular blocks, shape preserving local histogram modification employs connected components and level-sets for contrast



(a) Original ($PixDist = 20.0$, $TEN = 1.7\times10^4$)



(b) HE result ($PixDist = 18.7$, $TEN = 4.0\times10^3$)



(c) GLG result ($PixDist = 43.6$, $TEN = 2.0\times10^4$)

Fig. 4. X-ray image of luggage. (a) Low-contrast original image. (b) Result of HE, which has an unsatisfactory appearance. (c) Result of gray-level grouping, has a sharper look. The result is produced fully automatically.

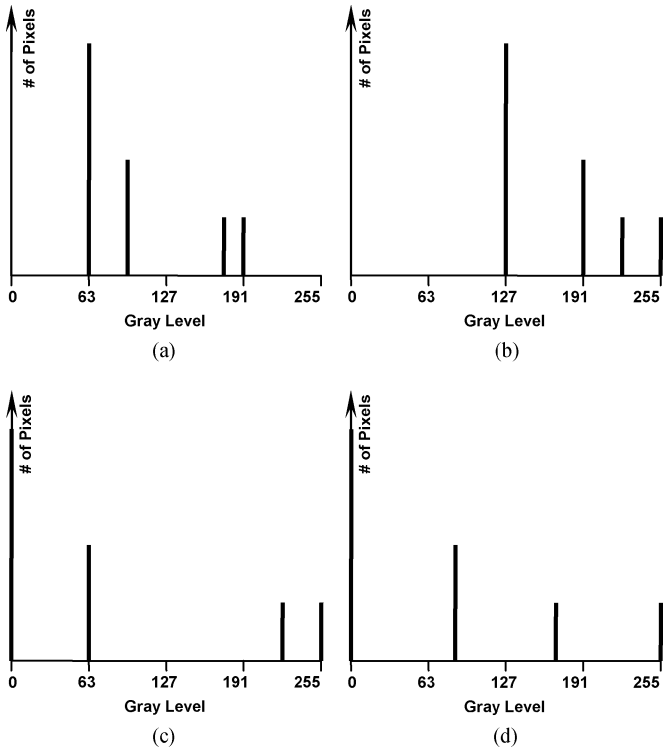


Fig. 5. Histograms of a virtual low-contrast image treated by different enhancement methods. (a) Original. (b) Result of histogram equalization. Half of the grayscale is wasted. (c) Result of linear contrast stretching. Contrast enhancement is not strong for histogram components which are originally very close to each other. (d) Optimal histogram of the enhanced image.

enhancement [6]. Multiscale adaptive histogram equalization [11] and other multiscale contrast enhancement techniques [7], [22] use multiscale analysis to decompose the image into subbands, and apply corresponding enhancement techniques to the high-frequency subband, and then combine the enhanced high-frequency subband with the low-frequency subband to reconstruct the output image.

The above mentioned advanced contrast enhancement techniques usually outperform conventional techniques. However, they still have limitations and cannot handle certain classes of images well and/or are not fully automatic methods.

Our motivation is to develop a new contrast enhancement technique which not only produces better results, but is also general and can be automatically applied to a broad variety of images. This paper introduces a new histogram-based optimized contrast enhancement technique called gray-level grouping (GLG). The basic procedure of this technique is to first group the histogram components of a low-contrast image into a proper number of groups according to a certain criterion, then redistribute these groups of histogram components uniformly over the grayscale so that each group occupies a grayscale segment of the same size as the other groups, and finally ungroup the previously grouped gray-levels.

In the next section, the principle and algorithm of the gray-level grouping technique will be described. The computational complexity of this method and comparison with a few of conventional methods using a benchmark quality measure will be discussed in Section III. An extension of the basic GLG to local contrast enhancement approaches, adaptive gray-level grouping (A-GLG), will be presented in Section IV.

II. Basic Gray-Level Grouping (GLG)

Although piecewise-based contrast stretching or histogram specification might be able to yield satisfactory results if the proper processing parameters are selected for the image to be enhanced, they are not general techniques and cannot be automatically applied to other images, since the histogram profile varies from image to image.

Before introducing the new technique of gray-level grouping, we first revisit several conventional contrast enhancement techniques, analyze their shortcomings, and try to overcome them in developing the new method.

Fig. 5(a) illustrates the histogram of a virtual low-contrast image. The histogram contains four components, which are clustered in the central part of the grayscale. The amplitude

of the second component is half that of the leftmost component, and the right two components are half the second one. Fig. 5(b) shows the result of performing histogram equalization on Fig. 5(a). Due to the highest amplitude of the leftmost component, the resulting histogram is shifted toward the right side of the grayscale. The left half of the grayscale is empty and this produces a washed-out appearance in the output image. The objective of histogram equalization is to achieve a uniform histogram. However, this can be achieved only on continuous histograms. For digital images and their discrete histograms, histogram equalization simply cannot redistribute the histogram components uniformly in most cases, because of the continuous nature of the technique.

Fig. 5(c) shows the result of performing linear contrast stretch on Fig. 5(a). The resulting histogram spreads over the full range of grayscale, so there is no washed-out appearance in the output image. However, it can be noted that the right two histogram components are still quite close to each other, so are the left two. As a result, the contrast enhancement in some regions of the output image is not the strongest. Since the left two histogram components are far away from the right two, they might be over-contrasted in the output image. Therefore, the contrast enhancement in the resulting image might also be unbalanced. Furthermore, linear contrast stretching is generally not an automatic method, since a piecewise-linear transformation function need to be manually specified to achieve satisfactory results.

In order to overcome the above shortcomings, the components of the desired histogram of the optimal contrast enhancement result should spread over the full range of the grayscale, with the bins being away from each other as far as possible. Fig. 5(d) shows the desired histogram which produces the strongest contrast enhancement. It can be noted that the four histogram components are uniformly spread over the entire grayscale, and are evenly spaced from each other.

The objectives of developing this new technique are as follows.

- Like histogram equalization, the basic objective of the new technique is still to achieve a uniform histogram, but for discrete histograms, in the sense that the histogram components are redistributed uniformly over the grayscale.
- Utilize the grayscale more *efficiently*; conventional contrast enhancement techniques such as histogram equalization are likely to leave too much empty space on the grayscale and cause under or over-contrast.
- Spread the components of histogram over the grayscale in a *controllable* and/or *efficient* way.
- Treat the histogram components in different parts of the grayscale *differently* if necessary, in order to satisfy specific contrast enhancement purposes. This objective will lead to an extension of the basic GLG technique—selective gray-level grouping (SGLG), which will be introduced in Part II of this paper.
- The new technique should be *general*, and be able to treat various kinds of images *automatically*.

The basic principle and procedure of this new technique are explained below.

- Group the histogram components into a proper number of gray-level *bins* according to their amplitudes, in order to initially reduce the number of gray bins. Therefore, empty gray levels can be created on the grayscale, allowing the redistribution of the histogram components in the next step. Furthermore, this grouping operation results in a set of gray-level bins whose amplitudes are close to each other, allowing a quasi-uniform distribution of the histogram components in the next step.
- Redistribute these groups of histogram components *uniformly* over the grayscale, so that each group occupies a grayscale segment of the same size as the other groups, and the concentrated histogram components spread out and image contrast is increased. The size of the grayscale segment occupied by each gray-level bin is determined from the total number of bins. At the same time, the grayscale is utilized efficiently and the over-contrast problem is also avoided.
- The histogram components in different segments of the grayscale can be grouped using different criteria, so they can be *redistributed differently* over the grayscale to meet specific processing purposes, e.g., certain applications may require different parts of the histogram to be enhanced to different extents. This step is needed only in SGLG, which will be discussed in Part II of the paper.

The algorithm of the basic GLG technique is described as follows, along with a simple example for illustration.

- Let $H_n(k)$ denote the histogram of the original image, with k representing the gray levels on the grayscale $[0, M-1]$. To perform gray-level grouping, first the n nonzero histogram components are assigned to gray-level bins, or gray-level groups, $G_n(i)$

$$G_n(i) = H_n(k) \text{ for } H_n(k) \neq 0 \quad k = 0,1,2,\ldots,M-1; \quad i = 1,2,3,\ldots,n. \tag{1}$$

Fig. 6(a) illustrates the histogram of a virtual low-contrast image, whose gray levels are in the interval [0, 8]. It has $n = 5$ nonzero components and its histogram components are

$$H_5(1) = 6, \quad H_5(3) = H_5(4) = 1, \quad H_5(5) = 4$$
$$H_5(7) = 12, \text{ and } H_5(k) = 0, \quad \text{for } k = 0,2,6,8.$$

After the nonzero histogram components are assigned to gray-level bins, we have

$$G_5(1) = 6, \quad G_5(2) = 1, \quad G_5(3) = 1$$
$$G_5(4) = 4, \text{ and } G_5(5) = 12.$$

- The left and right limits, $L_n(i)$ and $R_n(i)$, of the gray-level interval represented by $G_n(i)$ also need to be recorded. In this first step, the intervals consist of single values, which are the gray-level values, k, of the original histogram components, $H_n(k)$

$$L_n(i) = R_n(i) = k, \text{ for } H_n(k) \neq 0 \quad k = 0,1,2,\ldots,M-1, \quad i = 1,2,3,\ldots,n. \tag{2}$$

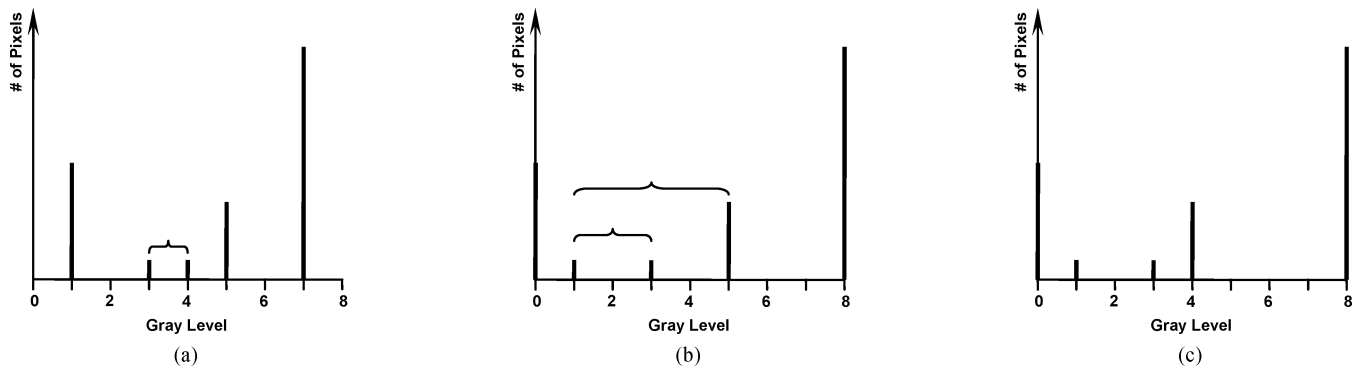


Fig. 6. Illustration of gray-level grouping. (a) Original histogram of a virtual low-contrast image. The bracket indicates the gray levels to be grouped. (b) Histogram after the first gray-level grouping and ungrouping. (c) Histogram after the second gray-level grouping and ungrouping.

In our example, these parameters are

$$L_5(1) = R_5(1) = 1; \quad L_5(2) = R_5(2) = 3$$
$$L_5(3) = R_5(3) = 4; \quad L_5(4) = R_5(4) = 5$$
$$L_5(5) = R_5(5) = 7.$$

- The first occurring smallest $G_n(i)$ is found.

$$a = \min_i G_n(i) \tag{3}$$

and i_a is the group index corresponding to the smallest $G_n(i)$, i.e., a. In our example, $i_a = 2$ and $a = G_5(2) = 1$.

- Grouping is performed in this step. Group $G_n(i_a)$ is merged with the smaller of its two adjacent neighbors, and the gray-level bins $G_n(i)$ are adjusted to create a new set of bins $G_{n-1}(i)$, as follows:

$$G_{n-1}(i) = \begin{cases} G_n(i), & \text{for } i = 1, 2, \ldots, i' - 1 \\ a + b, & \text{for } i = i' \\ G_n(i+1), & \text{for } i = i' + 1, i' + 2, \ldots, n - 1 \end{cases} \tag{4}$$

where

$$b = \min \{G_n(i_a - 1), G_n(i_a + 1)\} \tag{5}$$

and

$$i' = \begin{cases} i_a - 1, & \text{for } G_n(i_a - 1) \le G_n(i_a + 1) \\ i_a, & \text{otherwise.} \end{cases} \tag{6}$$

The left and right limits of the gray-level intervals represented by $G_{n-1}(i)$ also need to be adjusted accordingly

$$L_{n-1}(i) = \begin{cases} L_n(i), & \text{for } i = 1, 2, \ldots, i' \\ L_n(i+1), & \text{for } i = i' + 1, i' + 2, \ldots, n - 1 \end{cases} \tag{7}$$

$$R_{n-1}(i) = \begin{cases} R_n(i), & \text{for } i = 1, 2, \ldots, i' - 1 \\ R_n(i+1), & \text{for } i = i', i' + 1, \ldots, n - 1. \end{cases} \tag{8}$$

In our example, $b = G_5(3) = 1$ and $i' = i_a = 2$. The bracket in Fig. 6(a) indicates the two histogram components to be grouped. The new gray-level bins are

$$G_4(1) = G_5(1) = 6, \quad G_4(2) = a + b = 2$$
$$G_4(3) = G_5(4) = 4 \text{ and } G_4(4) = G_5(5) = 12.$$

The new gray-level intervals represented by the new groups are

$$L_4(1) = R_4(1) = 1; \quad L_4(2) = 3, \quad R_4(2) = 4$$
$$L_4(3) = R_4(3) = 5; \quad L_4(4) = R_4(4) = 7.$$

- Mapping and ungrouping are performed in this step. Now the total number of gray-level bins has been reduced by one. We can start to construct the transformation function $T_{n-1}(k)$, which maps the gray-level values of pixels in the input image to the desired values in the output image. In our method, all gray-level bins are redistributed uniformly over the entire grayscale, the gray levels are mapped to new values, and the combined histogram components are fully or partially uncombined. We first calculate the number of gray levels, N_{n-1}, that each gray-level bin will occupy in the resulting image. With a total number of bins equal to $n - 1$, we have

$$N_{n-1} = \frac{M - 1}{n - 1}. \tag{9}$$

However, if $L_{n-1}(1) = R_{n-1}(1)$, this indicates that the leftmost gray-level bin $G_{n-1}(1)$ contains only one gray level or one histogram component, which usually corresponds to the background, and it will be matched to gray level 0 in the resulting image. To prevent this one histogram component from occupying too many gray levels, we let

$$N_{n-1} = \frac{M - 1}{n - 1 - \alpha} \tag{10}$$

where α is a constant between 0 and 1, and usually assumes a value of 0.8 in our treatments, found through multiple trials to work well with a variety of images.

There are four cases to be considered when constructing $T_{n-1}(k)$. For $k = 0, 1, 2, \ldots, M-1$, we have the following.

1) If gray-level k falls inside gray-level bin $G_{n-1}(i)$, and $L_{n-1}(i) \neq R_{n-1}(i)$, this gray level is first mapped onto the right boundary of the gray-level interval assigned to bin $G_{n-1}(i)$, i.e., $[(i-1)N_{n-1}, iN_{N-1}]$, then it is separated from the group by linear rescaling within the assigned gray-level interval. Therefore, its transformation function $T_{n-1}(k)$ is as shown in (11), at the bottom of the page. If $L_{n-1}(1) = R_{n-1}(1)$, constant α prevents the background histogram from occupying too many gray levels.
If $L_{n-1}(i) = R_{n-1}(i)$, i.e., the bin $G_{n-1}(i)$ contains only one gray level, then the transformation function is

$$T_{n-1}(k) = \begin{cases} (i-\alpha)N_{n-1}, & \text{for } L_{n-1}(1) = R_{n-1}(1) \\ iN_{n-1}, & \text{for } L_{n-1}(1) \neq R_{n-1}(1). \end{cases} \tag{12}$$

2) If gray-level k falls between gray-level bin $G_{n-1}(i)$ and $G_{n-1}(i+1)$, then its transformation function is

$$T_{n-1}(k) = \begin{cases} (i-\alpha)N_{n-1}, & \text{for } L_{n-1}(1) = R_{n-1}(1) \\ iN_{n-1}, & \text{for } L_{n-1}(1) \neq R_{n-1}(1). \end{cases} \tag{13}$$

This ensures that $T_{n-1}(k)$ is monotonically increasing along the grayscale, and the gray-level reversal problem will be avoided in the adaptive approach of the GLG method.

3)

$$\text{If } k \leq L_{n-1}(1), \text{ then } T_{n-1}(k) = 0. \tag{14}$$

4)

$$\text{If } k \geq R_{n-1}(n-1), \text{ then } T_{n-1}(k) = M-1. \tag{15}$$

The constructed gray-level transformation function $T_{n-1}(k)$ for $k = 0, 1, 2, \ldots, M-1$ is stored in computer memory. In our example, we let $\alpha = 1$ for simplicity and have $N_4 = (9-1)/(4-1) = 2.67$. The transformed gray levels are

$$T_4(0) = T_4(1) = T_4(2) = 0, \quad T_4(3) = 1, \quad T_4(4) = 3$$
$$T_4(5) = 5, \quad T_4(6) = 5, \quad T_4(7) = T_4(8) = 8.$$

All resulting gray levels are rounded to the closest integer, and the histogram of the resulting image is shown in Fig. 6(b).

- By applying the constructed transformation function $T_{n-1}(k)$ to the histogram $H_n(k)$ of the original image, we obtain the histogram of the processed image $H_{n-1}(k)$. The average distance D_{n-1} between pixels on the grayscale, is used as a criterion to measure the quality of contrast enhancement. This distance is given by the expression

$$D_{n-1} = \frac{1}{N_{pix}(N_{pix}-1)} \sum_{i=0}^{M-2} \sum_{j=i+1}^{M-1} H_{n-1}(i) H_{n-1}(j)(j-i) \quad \text{for } i, j \in [0, M-1] \tag{16}$$

where $[0, M-1]$ is the gray-level range of the grayscale, and N_{pix} is the total number of pixels in the image. This criterion generally applies only to the gray-level grouping technique or similar histogram-based techniques, and may not be used to judge the quality of images treated by other enhancement techniques. A counter example is given here—if we set the mean gray level of a low-contrast image as the threshold, and threshold this image into a black-and-white image, the average distance between pixels on the grayscale of the resulting image will be the maximum that could be achieved theoretically, however, the resulting black-and-white image is obviously unacceptable for purposes of enhancement. However, the GLG process tends to spread the histogram components uniformly over the grayscale, preventing the histogram components from concentrating in particular locations on the grayscale. At the same time, a larger D will keep the histogram components further away from each other for better enhancement. Therefore, we consider the average distance between pixels on the grayscale D as a sound measure of the quality of images enhanced by GLG technique, and consider that the maximal D corresponds to the optimal contrast enhancement. Visual evaluations of multiple images during our testing also confirmed the validity of this measure. This quality measure is essential in the GLG process to achieve the optimal result. It is worth noting that this image contrast criterion, the average distance between pixels on the grayscale, is not inherent to the GLG algorithm, but could be used in other histogram-based algorithms (especially histogram equalization) as well. However, we suggest that this criterion be used with caution. The use of this criterion and a well-known criterion on images in this paper will be discussed in Section III.

In some cases (e.g., the background occupies a large percentage area in the image), in order to achieve the optimal result, the gray levels corresponding to the image background may be excluded when calculating D_{n-1}. For

$$T_{n-1}(k) = \begin{cases} \left(i - \alpha - \frac{R_{n-1}(i)-k}{R_{n-1}(i)-L_{n-1}(i)}\right) N_{n-1} + 1, & \text{for } L_{n-1}(1) = R_{n-1}(1) \\ \left(i - \frac{R_{n-1}(i)-k}{R_{n-1}(i)-L_{n-1}(i)}\right) N_{n-1} + 1, & \text{for } L_{n-1}(1) \neq R_{n-1}(1) \end{cases} \tag{11}$$

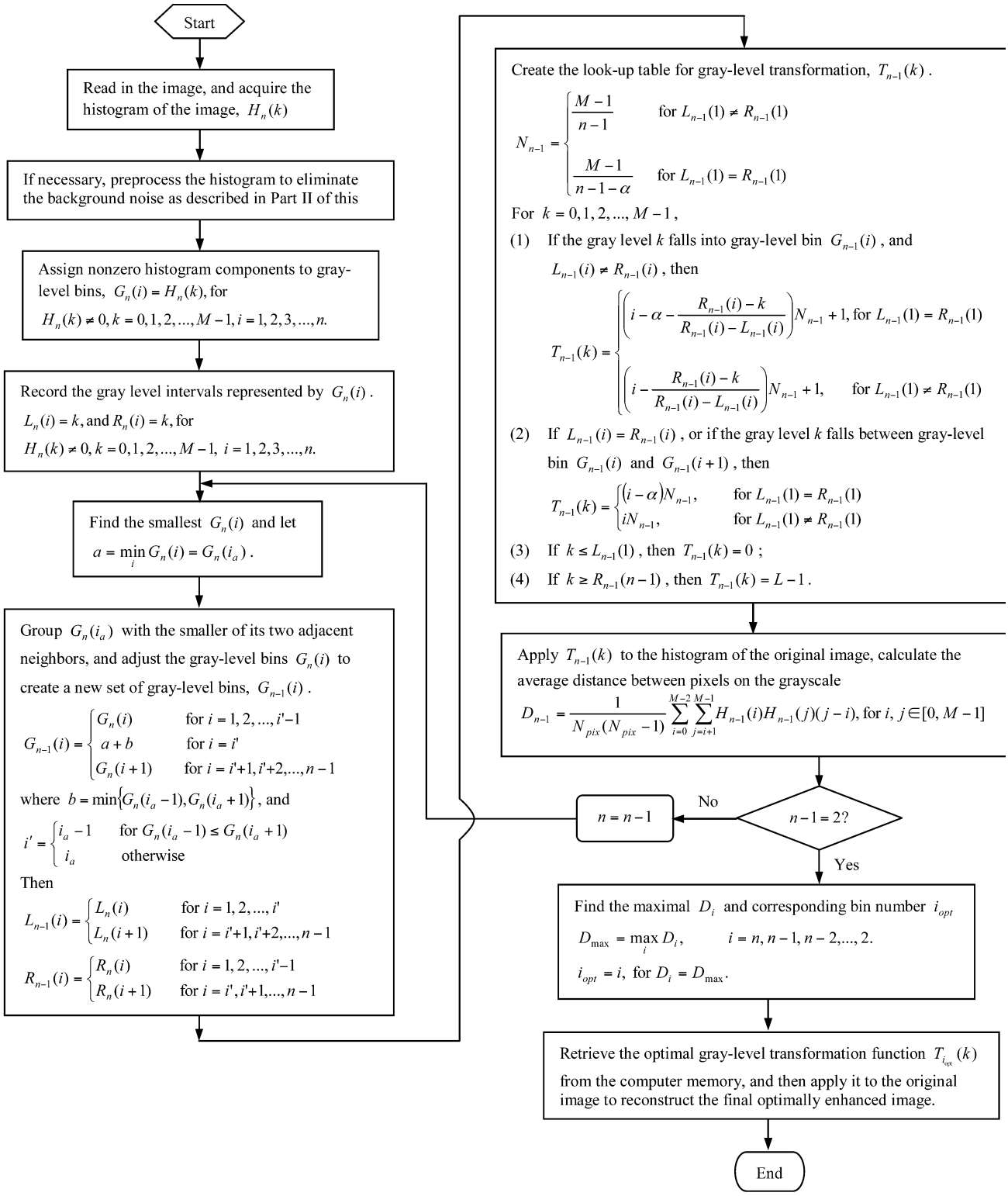


Fig. 7. Flow chart of the optimized gray-level grouping algorithm.

many images, the histogram components corresponding to the background are the highest and distinct in the histogram profile. Therefore, the approximate area of the background can be calculated automatically by summing the amplitudes of the histogram components of the background, given that the algorithm is notified by the user that the input image has a large-area background. If the background occupies a percentage area in the image larger than a user specified threshold (e.g., 40%), the background gray levels are then excluded when calculating D_{n-1}.

In our example, D_5 of Fig. 6(a) is 1.36, and D_4 of Fig. 6(b) is 1.87.

To determine the optimal number of gray-level bins that will lead to the optimal contrast enhancement, we need to

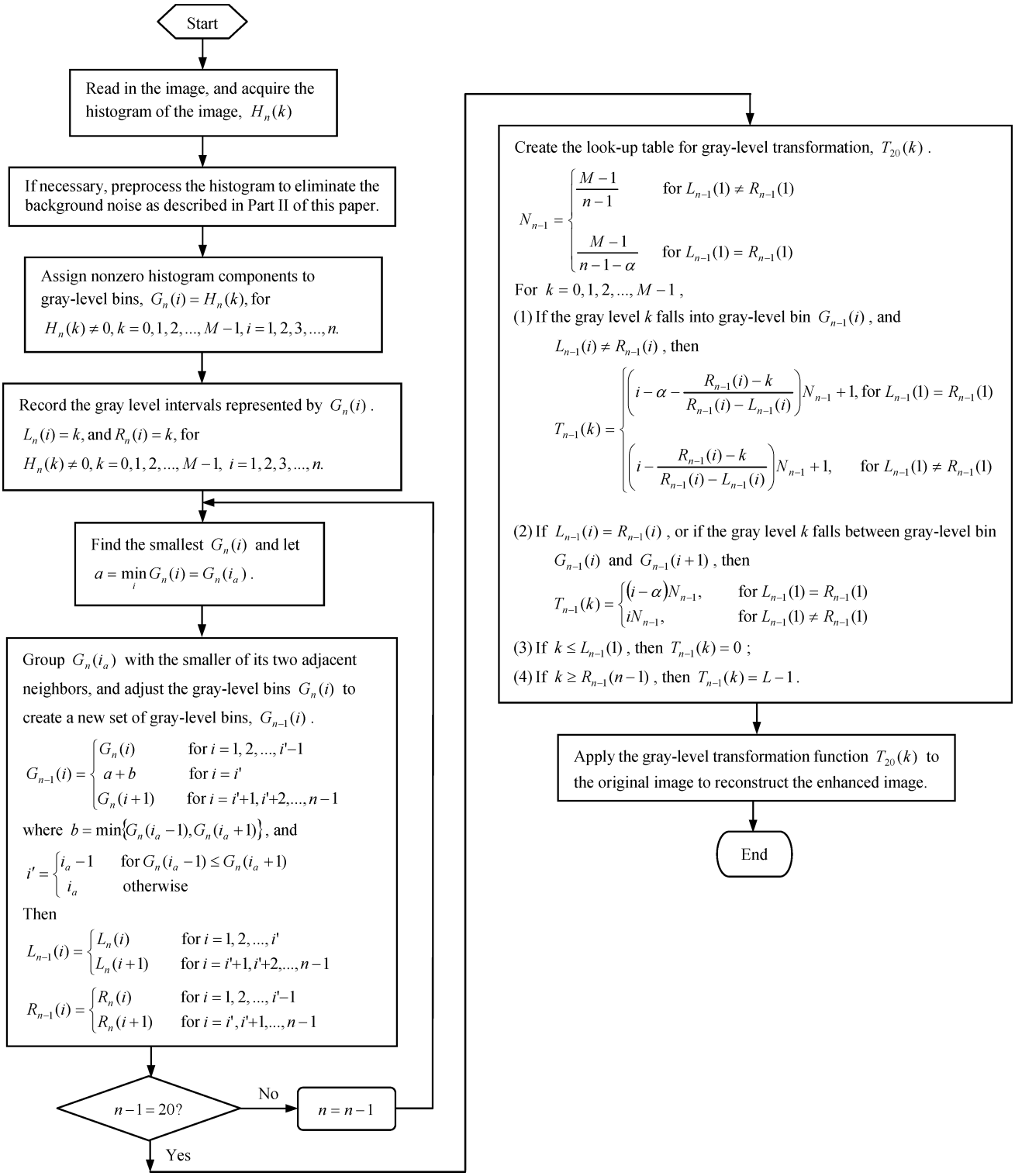
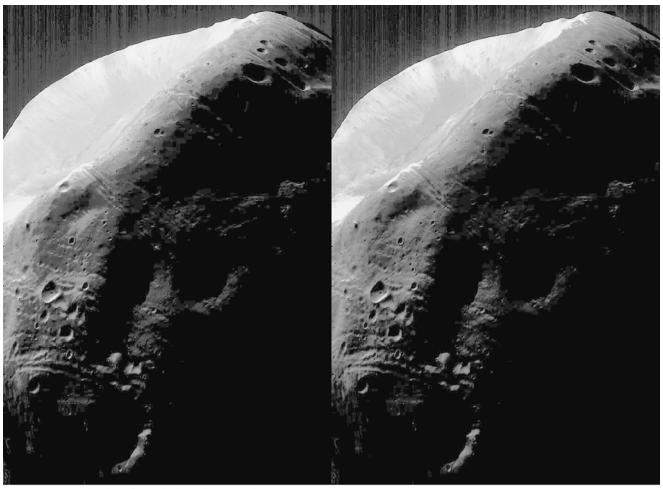


Fig. 8. Flow chart of the FGLG algorithm, which groups the original gray-level bins into a default number of bins, 20, executes much faster than the optimized GLG, and has comparable results.

repeat the above procedure and group the histogram components into all possible numbers from n to 2 (there is no need to group all histogram components into one bin since the histogram will be the same as the original after it is ungrouped), and calculate the average distance between pixels on the grayscale, D_i, for each set of bins. The maximal D_i will lead to the corresponding optimum number i_{opt} for gray-level bins

$$D_{\max} = \max_i D_i, \qquad \text{for } i = n, n-1, n-2, \ldots, 2 \tag{17}$$

$$i_{\text{opt}} = i, \qquad \text{for } D_i = D_{\max}. \tag{18}$$



(a) Optimal GLG result (*PixDist* = 44.1 , $TEN = 8.5\times10^3$) (b) Fast GLG result (*PixDist* = 42.6 , $TEN = 8.1\times10^3$)

Fig. 9. Comparison of GLG results using different gray-level bin numbers. Both images are satisfactory. (a) GLG result of the Phobos image with the optimal bin number of 4, found through the iterative process. (b) FGLG result of the Phobos image with the default bin number of 20.

In our example, we continue to group the gray-level bins. This time bin $G_4(2)$ and $G_4(3)$ will be grouped as indicated by the bracket in Fig. 6(b), and the new set of gray-level bins are

$$G_3(1) = G_4(1) = 6$$
$$G_3(2) = G_4(2) + G_4(3) = 2 + 4 = 6$$
$$G_3(3) = G_4(4) = 12.$$

Their boundaries are

$$L_3(1) = R_3(1) = 1; \quad L_3(2) = 3, \quad R_3(2) = 5$$
$$L_3(3) = R_3(3) = 7.$$

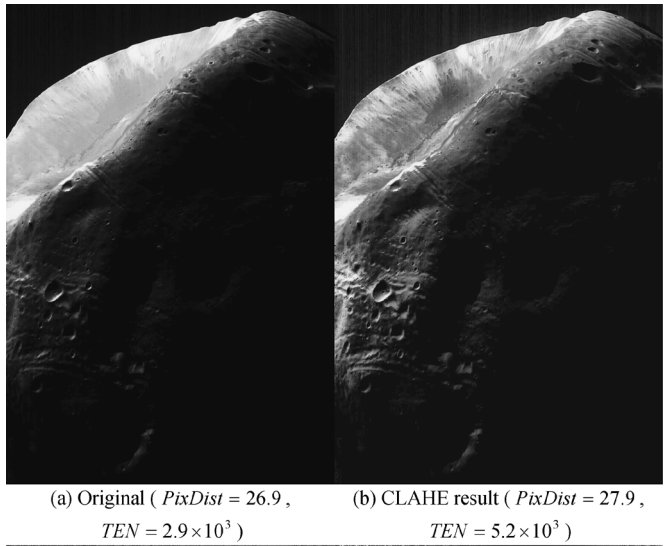
Then $N_3 = 8/(3-1) = 4$ and the new transformed gray levels are

$$T_3(0) = T_3(1) = T_3(2) = 0, \quad T_3(3) = 1, \quad T_3(4) = 3$$
$$T_3(5) = 4, \quad T_3(6) = 4, \quad T_3(7) = T_3(8) = 8.$$

The resulting histogram is shown in Fig. 6(c). The average distance between pixels on the grayscale $D_3 = 1.90$ is larger than D_5 and D_4.

- To obtain the final optimally enhanced image, we retrieve the optimal gray-level transformation function $T_{i_{\text{opt}}}(k)$ from computer memory, and then apply it to the original image.

Fig. 7 illustrates the flow chart of the optimized gray-level grouping algorithm. Figs. 1(d) and 2(d) show the result of applying this technique to the Phobos image and the resulting histogram, respectively. Fig. 4(c) shows the GLG result of the X-ray luggage image. Since the background area percentage is quite large in the X-ray image, the background is excluded



(a) Original (*PixDist* = 26.9 , $TEN = 2.9\times10^3$) (b) CLAHE result (*PixDist* = 27.9 , $TEN = 5.2\times10^3$)

(c) Global-GLG result (*PixDist* = 44.1 , $TEN = 8.5\times10^3$) (d) Adaptive-GLG result (*PixDist* = 41.1 , $TEN = 2.2\times10^4$)

Fig. 10. Mars moon—phobos. (a) Low-contrast original image. (b) Result of CLAHE. (c) Result of global gray-level grouping. (d) Result of A-GLG. Its contrast enhancement is the strongest (original image courtesy of Dr. R. C. Gonzalez [1]).

when calculating the average distance between pixels on the grayscale, in order to achieve the strongest enhancement in the resulting image of Fig. 4(c). It is obvious that the GLG results are better than those of histogram equalization and histogram specification. Furthermore, the new method is fully automatic.

It should be pointed out that in Fig. 1(d), background noise in the upper left corner has been amplified in the basic GLG result. This effect also exists in the results of HE and histogram specification, as shown in Fig. 1(b) and (c). This problem can be solved and will be addressed in Part II of this paper.

Although we have described an approach for finding the optimal number of gray-level groups, it has been found that the quality of the resulting images is not very sensitive to the total number of gray-level bins if this number is below 150, because the ungrouping of grouped gray levels results in similar spacing between high-amplitude histogram components. Therefore, we can use a default value for the total number of gray-level groups, e.g., 20, which has been found to yield satisfactory results in a

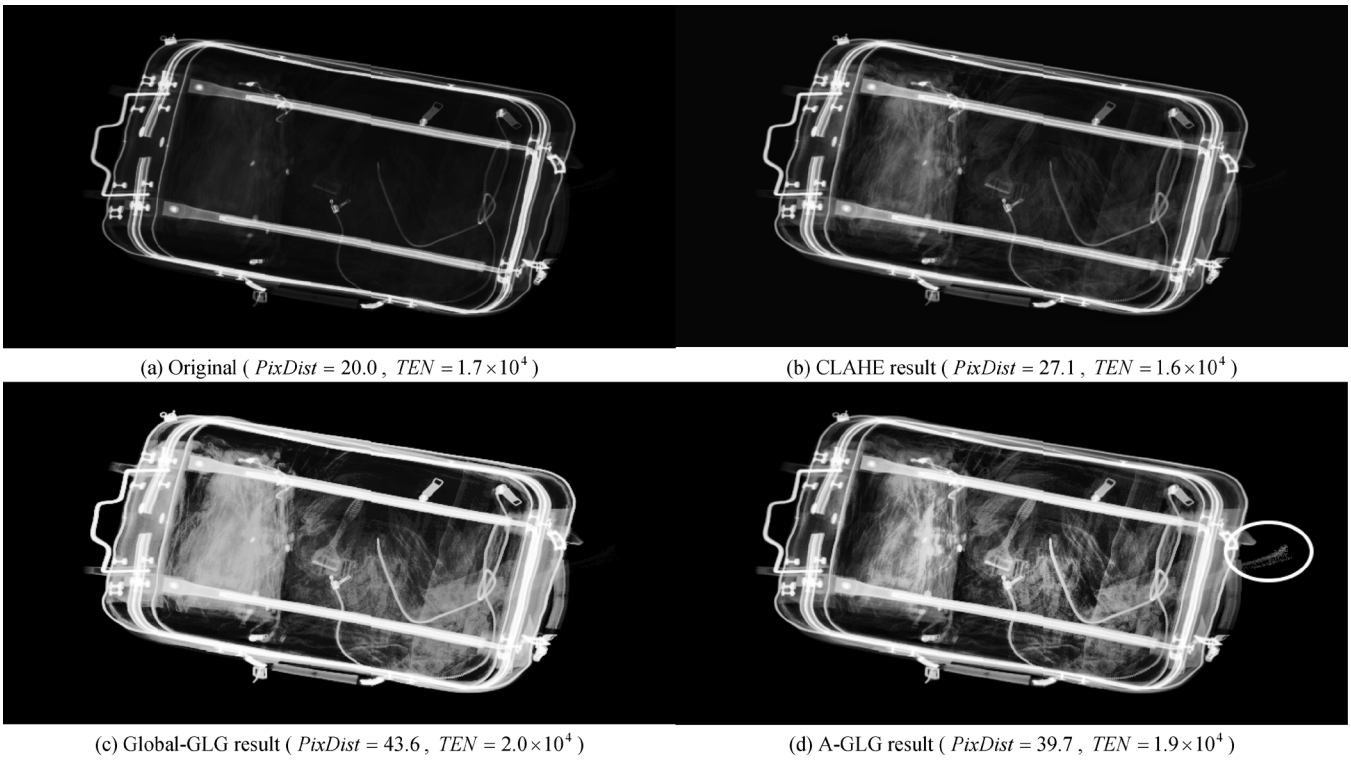


Fig. 11. X-ray image of luggage. (a) Original image. (b) Result of CLAHE. (c) Result of global GLG. (d) Result of A-GLG. Not only its contrast enhancement is the strongest, but also some objects that are invisible in the original image and the CLAHE result become readily apparent (e.g., the tip of the stripe to the right of the luggage, as shown by the white oval in the image). (The Tenengrad value of the global GLG result for this image is a little higher than that of the adaptive GLG result, because some light regions in conjunction with the surrounding dark background in the global GLG result exhibit larger gradient magnitudes, which lead to a larger Tenengrad value, but the adaptive GLG result apparently has a higher local contrast enhancement).

large number of experiments and saves on iterations and computational expenses significantly. The choice of this number is also based on the fact that it is comparable to the number of gray shades that the human eye can discern, which is a couple of dozens. Without constructing the transformation function and calculating the average distance between pixels on the grayscale for each set of gray-level bins, this algorithm executes much faster (more than 3 orders of magnitude faster for 8-bit images) than the optimized GLG, so it is called fast gray-level grouping (FGLG). Fig. 8 illustrates the flow chart of FGLG.

Fig. 9(a) and (b) shows the comparison of the results of treating the Phobos image by GLG with two different numbers of gray-level bins. Fig. 9(a) is the result using the optimal number of bins of four given by (18), and Fig. 9(b) is the result using the default number of bins of 20. It can be seen that there is not much difference in the two images, and both images are satisfactory.

III. Computational Complexity and Quality Measure

The computational complexity of the GLG technique is basically determined by the number of comparison operations for finding the smallest gray-level groups and the number of multiplication and/or division operations for calculating the gray-level transformation functions $T_{n-1}(k)$.

The number of comparison operations for finding the smallest gray-level group among L groups is $O(L)$. Since the smallest gray-level group needs to be found for all possible sets of groups in the optimal GLG process, the computational complexity for finding the smallest gray-level groups is $O(L^2)$, where L is the total number of gray levels on the grayscale. For each set of gray-level groups, the number of multiplication and/or division operations for calculating the gray-level transformation function is $O(L)$. Since this also needs to be performed on all possible sets of gray-level groups in the optimal GLG process, the computational complexity for calculating gray-level transformation functions in the optimal GLG process is $O(L^2)$. However, since the gray-level transformation function is calculated only once in the fast GLG process, its computational complexity for multiplication operations is $O(L)$. As comparison, the computational complexity of the HE algorithm is $O(L)$.

In order to evaluate the competitiveness of the GLG method against existing contrast enhancement techniques, we used the most well-known benchmark image sharpness measure, the Tenengrad criterion [26], [27], to compare the results of the GLG method and the conventional methods studied in this paper. The Tenengrad criterion is based on gradient magnitude maximization. It is considered one of the most robust and functionally accurate image quality measures [27]. The Tenengrad value of an image I is calculated from the gradient $\nabla I(x,y)$ at each pixel (x,y), where the partial derivatives are obtained by a high-pass filter, e.g., the Sobel operator, with the convolution kernels i_x and i_y. The gradient magnitude is given as

$$S(x,y) = \sqrt{(i_x * I(x,y))^2 + (i_y * I(x,y))^2} \qquad (19)$$

and the Tenengrad criterion is formulated as

$$\mathrm{TEN} = \sum_x \sum_y S(x,y)^2, \qquad \text{for } S(x,y) > T \tag{20}$$

where T is a threshold. The image quality is usually considered higher if its Tenengrad value is larger.

We calculated the Tenengrad values (TEN) of all images in this article, and listed them in the corresponding figure captions. It is noted that the images processed with the GLG technique have significantly larger Tenengrad values, which indicates that the GLG method is superior to the conventional techniques compared to in this paper. This result agrees with the visual evaluation by the human eye. It is also worth noting that the Tenengrad criterion indicates that the optimal GLG result is better than the fast GLG result, as shown in Fig. 9.

In the previous section, we proposed an image contrast measure, the average distance between pixels on the grayscale. We also calculated the values of this criterion ($PixDist$) for all images in this article and listed them in the corresponding figure captions. It can be seen that, this criterion generally agrees well with the benchmark Tenengrad measure in evaluating image contrast. It also should be noted that, the $PixDist$ values of the adaptive GLG (A-GLG) results described in Section IV usually do not agree with the perceived image contrasts, because the adaptive GLG process significantly changes the global histogram profile of the image, and therefore makes the comparison of the $PixDist$ values of the global GLG and adaptive GLG results meaningless. This is one of the situations in which the $PixDist$ criterion should not be used.

IV. Adaptive Gray-Level Grouping (A-GLG)

By analogy to the well-established conventional technique of AHE [4], or contrast-limited adaptive histogram equalization (CLAHE), GLG also has its adaptive counterparts—A-GLG, or CLA-GLG. In the A-GLG or CLA-GLG method, the image is first divided into an array of subimages (usually an 8×8 array), each subimage is treated with the GLG method, and all treated subimages are merged together by bilinear interpolation to generate the processed whole image.

The algorithm of adaptive GLG (A-GLG) technique is described as the following.

1) Divide the original image into an $M \times N$ array of subimages, and process all subimages with the GLG algorithm to obtain their optimal GLG gray-level transformation functions, i.e., $T_{i,j}(k)$, for $i = 1, 2, \ldots, M$, $j = 1, 2, \ldots, N$, and $k = 0, 1, \ldots, L-1$. Here, $L-1$ represents the highest gray-level value on the grayscale.
2) Create an intermediate $(M+1) \times (N+1)$ array of gray-level transformation functions $A_{i,j}(k)$ for $i = 1, 2, \ldots, M+1$, $j = 1, 2, \ldots, N+1$, and $k = 0, 1, \ldots, L-1$, as follows.
   a) For the four corner components

$$\begin{aligned} A_{1,1}(k) &= T_{1,1}(k), \quad A_{1,N+1}(k) = T_{1,N}(k) \\ A_{M+1,1}(k) &= T_{M,1}(k), \quad A_{M+1,N+1}(k) = T_{M,N}(k). \end{aligned} \tag{21}$$

   b) For the boundary components

$$A_{i,j}(k) = \begin{cases} T_{i,j}(k), & \text{for } T_{i,j-1}(k) = L-1 \\ T_{i,j-1}(k), & \text{for } T_{i,j}(k) = L-1 \\ \left(T_{i,j-1}(k) + T_{i,j}(k)\right)/2, & \text{otherwise} \end{cases} \tag{22}$$

   for $i = 1, M+1$, $j = 2, 3, \ldots, N$, and $k = 0, 1, \ldots, L-1$, and

$$A_{i,j}(k) = \begin{cases} T_{i,j}(k), & \text{for } T_{i-1,j}(k) = L-1 \\ T_{i-1,j}(k), & \text{for } T_{i,j}(k) = L-1 \\ \left(T_{i-1,j}(k) + T_{i,j}(k)\right)/2, & \text{otherwise} \end{cases} \tag{23}$$

   for $i = 2, 3, \ldots, M$, $j = 1, N+1$, and $k = 0, 1, \ldots, L-1$.
   c) For the interior components

$$A_{i,j}(k) = \frac{1}{p} \sum_{m,n} T_{m,n}(k) \qquad \text{for } T_{m,n}(k) \neq L-1 \tag{24}$$

   where $m = i-1, i$, $n = j-1, j$, and $p = 4$, 3, 2, or 1, which equals to the number of operands in the numerator. The above equation applies to $i = 2, 3, \ldots, M$, $j = 2, 3, \ldots, N$, and $k = 0, 1, \ldots, L-1$.

   This step is an averaging process to balance the contrast of adjacent subimages in the final output image. If gray-level k in the original image is mapped to gray-level $L-1$ by $T_{i,j}(k)$, it is considered as background and therefore excluded from the averaging process.
3) Perform bilinear interpolation to reconstruct the final output image. For each original subimage $I_{i,j}(x,y)$, function $k = I_{i,j}(x,y)$ returns the gray-level value k of the pixel at subimage coordinate, (x,y), for $x = 1, 2, \ldots, h_{i,j}$, $y = 1, 2, \ldots, w_{i,j}$, where $h_{i,j}$ and $w_{i,j}$ are the height and width of the corresponding subimage, respectively. The bilinearly-interpolated output subimage $O_{i,j}(x,y)$ is given by the following expression:

$$\begin{aligned} &O_{i,j}(x,y) \\ &= \frac{1}{(h_{i,j}+1)(w_{i,j}+1)} \\ &\quad \times [(h_{i,j}+1-x)\left[(w_{i,j}+1-y)A_{i,j}(k) + yA_{i,j+1}(k)\right] \\ &\qquad + x\left[(w_{i,j}+1-y)A_{i+1,j}(k) + yA_{i+1,j+1}(k)\right]] \end{aligned} \tag{25}$$

   for $x = 1, 2, \ldots, h_{i,j}$, $y = 1, 2, \ldots, w_{i,j}$, and $k = I_{i,j}(x,y)$.

   The final processed whole image is obtained by stitching the array of output subimages together.

Fig. 10(c) shows the A-GLG result of the Phobos image with comparison to the CLAHE result and global GLG result. It can be seen that the A-GLG result is bviously better than CLAHE and global GLG results.

AHE is often not applicable to many images such as images with large areas of dark background. In that case, because of its adaptive nature, AHE will turn the dark background into white, and cause undesirable artifacts. Therefore, CLAHE is more applicable than AHE. However, A-GLG does not have

this problem, and will automatically keep the dark background black.

Fig. 11(c) shows the A-GLG result of the X-ray image of luggage with comparison to the CLAHE result and global GLG result. It can be seen that not only the contrast of the A-GLG result is better than that of the CLAHE result (e.g., the razor in the bag is brighter and clearer), but also some objects that are invisible in the original image and the CLAHE result become readily apparent (e.g., the tip of the stripe to the right of the luggage, as shown by the white oval in the image).

V. Conclusion

We have developed a new automatic contrast enhancement technique. GLG is a general and powerful technique, which can be conveniently applied to a broad variety of low-contrast images and generates satisfactory results (more examples will be given in Part II of this paper). The GLG technique can be conducted with full automation at fast speeds and outperforms conventional contrast enhancement techniques. The benchmark image quality measure, Tenengrad criterion, indicates that the GLG results are superior to those of conventional techniques studied in this paper. The optimized GLG algorithm generally can process an image within a few seconds on a personal computer (PC), and the FGLG algorithm can process an image on the time scale of millisecond on a PC. The basic GLG method also provides a platform for various extensions of this technique, such as selective gray-level grouping (SGLG), (S)GLG with preprocessing steps for eliminating image background noises, (S)GLG on color images, and so on. All these variations extend the capability of the basic GLG technique, and will be discussed in Part II of this paper.

Acknowledgment

The authors would like to thank Y. Yao for help with the Tenengrad criterion, and Dr. A. Koschan and Dr. A. Gribok for their insightful and valuable suggestions.

References

[1] R. C. Gonzalez and R. E. Woods, *Digital Image Processing*, 2nd ed. Englewood Cliffs, NJ: Prentice-Hall, 2002, ISBN: 0-201-18075-8.

[2] Y.-T. Kim, "Enhancement using brightness preserving bi-histogram equalization," *IEEE Trans. Consumer Electron.*, vol. 43, no. 1, pp. 1–8, Feb. 1997.

[3] Y. Wan, Q. Chen, and B.-M. Zhang, "Image enhancement based on equal area dualistic sub-image histogram equalization method," *IEEE Trans Consumer Electron.*, vol. 45, no. 1, pp. 68–75, Feb. 1999.

[4] S. M. Pizer, E. P. Amburn, J. D. Austin, R. Cromartie, A. Geselowitz, T. Greer, B. H. Romeny, J. B. Zimmerman, and K. Zuiderveld, "Adaptive histogram equalization and its variations," *Comput. Vis., Graph., Image Process.*, vol. 39, pp. 355–368, 1987.

[5] T.-K. Kim, J.-K. Paik, and B.-S. Kang, "Contrast enhancement system using spatially adaptive histogram equalization with temporal filtering," *IEEE Trans. Consumer Electron.*, vol. 44, no. 1, pp. 82–86, Feb. 1998.

[6] V. Caselles, J.-L. Lisani, J.-M. Morel, and G. Sapiro, "Shape preserving local histogram modification," *IEEE Trans. Image Process.*, vol. 8, no. 2, pp. 220–230, Feb. 1999.

[7] D.-C. Chang and W.-R. Wu, "Image contrast enhancement based on a histogram transformation of local standard deviation," *IEEE Trans. Med. Imag.*, vol. 17, no. 4, pp. 518–531, Aug. 1998.

[8] S.-D. Chen and A. R. Ramli, "Contrast enhancement using recursive mean-separate histogram equalization for scalable brightness preservation," *IEEE Trans. Consumer Electron.*, vol. 49, no. 4, pp. 1301–1309, Nov. 2003.

[9] ——, "Minimum mean brightness error bi-histogram equalization in contrast enhancement," *IEEE Trans. Consumer Electron.*, vol. 49, no. 4, pp. 1310–1319, Nov. 2003.

[10] S. Dippel, M. Stahl, R. Wiemker, and T. Blaffert, "Multiscale contrast enhancement for radiographies: laplacian pyramid versus fast wavelet transform," *IEEE Trans. Med. Imag.*, vol. 21, no. 4, pp. 343–353, Apr. 2002.

[11] Y. Jin, L. Fayad, and A. Laine, "Contrast enhancement by multi-scale adaptive histogram equalization," in *Proc. SPIE*, 2001, vol. 4478, pp. 206–213.

[12] J.-Y. Kim, L.-S. Kim, and S.-H. Hwang, "An advanced contrast enhancement using partially overlapped sub-block histogram equalization," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 11, no. 4, pp. 475–484, Apr. 2001.

[13] S.-Y. Kim, D. Han, S.-J. Choi, and J.-S. Park, "Image contrast enhancement based on the piecewise-linear approximation of CDF," *IEEE Trans. Consumer Electron.*, vol. 45, no. 3, pp. 828–834, Aug. 1999.

[14] T.-K. Kim, J.-K. Paik, and B.-S. Kang, "Contrast enhancement system using spatially adaptive histogram equalization with temporal filtering," *IEEE Trans. Consumer Electron.*, vol. 44, no. 1, pp. 82–87, Feb. 1998.

[15] S. C. Matz and R. J. P. de Figueiredo, "A nonlinear technique for image contrast enhancement and sharpening," in *Proc. IEEE Int. Symp. Circuits and Systems*, 1999, vol. 4, pp. 175–178.

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

[17] J. P. Oakley and B. L. Satherley, "Improving image quality in poor visibility conditions using a physical model for contrast degradation," *IEEE Trans. Image Process.*, vol. 7, no. 2, pp. 167–179, Feb. 1998.

[18] S.-C. Pei, Y.-C. Zeng, and C.-H. Chang, "Virtual restoration of ancient chinese paintings using color contrast enhancement and lacuna texture synthesis," *IEEE Trans. Image Process.*, vol. 13, no. 3, pp. 416–429, Mar. 2004.

[19] A. Polesel, G. Ramponi, and V. J. Mathews, "Image enhancement via adaptive unsharp masking," *IEEE Trans. Image Process.*, vol. 9, no. 3, pp. 505–510, Mar. 2000.

[20] F. Russo, "An image enhancement technique combining sharpening and noise reduction," *IEEE Trans. Instrum. Meas.*, vol. 51, no. 4, pp. 824–828, Aug. 2002.

[21] F. Sattar and D. B. H. Tay, "Enhancement of document images using multiresolution and fuzzy logic techniques," *IEEE Signal Process. Lett.*, vol. 6, no. 10, pp. 249–252, Oct. 1999.

[22] J.-L. Starck, F. Murtagh, E. J. Candes, and D. L. Donoho, "Gray and color image contrast enhancement by the curvelet transform," *IEEE Trans. Image Process.*, vol. 12, no. 6, pp. 706–717, Jun. 2003.

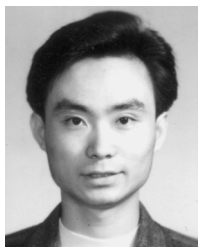
[23] J. A. Stark, "Adaptive image contrast enhancement using generalizations of histogram equalization," *IEEE Trans. Image Process.*, vol. 9, no. 5, pp. 889–896, May 2000.

[24] J. Yang, G. Dong, Y. Peng, Y. Yamaguchi, and H. Yamada, "Generalized optimization of polarimetric contrast enhancement," *IEEE Geosci. Remote Sens. Lett.*, vol. 1, no. 3, pp. 171–174, Jul. 2004.

[25] X. Zong, A. F. Laine, and E. A. Geiser, "Speckle reduction and contrast enhancement of echocardiograms via multiscale nonlinear processing," *IEEE Trans. Med. Imag.*, vol. 17, no. 4, pp. 532–540, Aug. 1998.

[26] E. P. Krotkov, *Active Computer Vision by Cooperative Focus and Stereo*. New York: Springer-Verlag, 1989.

[27] A. Buerkle, F. Schmoeckel, M. Kiefer, B. P. Amavasai, F. Caparrelli, A. N. Selvan, and J. R. Travis, "Vision-based closed-loop control of mobile microrobots for micro handling tasks," *Proc. SPIE*, vol. 4568, Microrobotics and Microassembly III, pp. 187–198, 2001.



ZhiYu Chen (SM'98) received the B.E. degree (with high honors) in engineering physics and the M.S. degree from Tsinghua University, Beijing, China, in 1994 and 1997, respectively. He is currently pursuing the Ph.D. degree in electrical engineering at the University of Tennessee, Knoxville (UTK).

His area of research was plasma science and technology. He has authored or coauthored seven journal articles and numerous conference papers in that field. In late 2003, he switched his research interest to the field of digital image processing and computer vision. He is now with the Imaging, Robotics and Intelligent Systems (IRIS) Laboratory at UTK.

Mr. Chen was a recipient of the IEEE Nuclear and Plasma Sciences Society Graduate Student Award in 2002. He also serves as a Reviewer for IEEE Transactions on Plasma Science.

Besma R. Abidi (M'88–SM'06) received the M.S. degrees in image processing and remote sensing (Hons.) from the National Engineering School of Tunis, Tunis, Tunisia, in 1985 and 1986, respectively, and the Ph.D. degree from The University of Tennessee, Knoxville (UTK), in 1995.

She is a Research Assistant Professor with the Department of Electrical and Computer Engineering, UTK, which she joined in 1998. She occupied a postdoctorate position with the Oak Ridge Institute of Science and Energy, Oak Ridge, TN, and was a Research Scientist at the Oak Ridge National Laboratory from 1998 to 2001. From 1985 to 1988, she was an Assistant Professor at the National Engineering School of Tunis. Her general areas of research are in image enhancement and restoration, sensor positioning and geometry, video tracking, sensor fusion, and biometrics.

Dr. Abidi is a member of SPIE, Tau Beta Pi, Eta Kappa Nu, Phi Kappa Phi, and The Order of the Engineer.

David L. Page (M'91) received the B.Sc. and M.Sc. degrees from Tennessee Technological University, Cookeville, in 1993 and 1995, respectively, and the Ph.D. degree from the University of Tennessee, Knoxville (UTK), in 2003, all in electrical engineering.

He then began work as an Electronics Engineer at the Naval Surface Warfare Center, Dahlgren, VA, where he was involved in distributed computing research. He currently serves as a Research Assistant Professor in the Imaging, Robotics, and Intelligent Systems Laboratory at UTK. His research interests focus on three-dimensional computer vision, with specific emphasis on mesh segmentation, curvature estimation, and object description.

Mongi A. Abidi (M'82) received the M.S. and Ph.D. degrees in electrical engineering from the University of Tennessee, Knoxville (UTK), in 1985 and 1987, respectively.

He is a W. Fulton Professor with the Department of Electrical and Computer Engineering, UTK, which he joined in 1986. His interests include image processing, multisensor processing, three-dimensional imaging, and robotics. He has published over 120 papers in these areas and co-edited the book *Data Fusion in Robotics and Machine Intelligence* (Academic, 1992).

Dr. Abidi received the 1994–1995 Chancellor's Award for Excellence in Research and Creative Achievement and the 2001 Brooks Distinguished Professor Award. He is a member of the Computer Society, Pattern Recognition Society, SPIE, Tau Beta Pi, Phi Kappa Phi, and Eta Kappa Nu honor societies. He also received the First Presidential Principal Engineer Award prior to joining the University of Tennessee.