Automatic Image Equalization and Contrast Enhancement Using Gaussian Mixture Modeling

Turgay Celik and Tardi Tjahjadi, Senior Member, IEEE

Abstract—In this paper, we propose an adaptive image equalization algorithm that automatically enhances the contrast in an input image. The algorithm uses the Gaussian mixture model to model the image gray-level distribution, and the intersection points of the Gaussian components in the model are used to partition the dynamic range of the image into input gray-level intervals. The contrast equalized image is generated by transforming the pixels’ gray levels in each input interval to the appropriate output gray-level interval according to the dominant Gaussian component and the cumulative distribution function of the input interval. To take account of the hypothesis that homogeneous regions in the image represent homogeneous silences (or set of Gaussian components) in the image histogram, the Gaussian components with small variances are weighted with smaller values than the Gaussian components with larger variances, and the gray-level distribution is also used to weight the components in the mapping of the input interval to the output interval. Experimental results show that the proposed algorithm produces better or comparable enhanced images than several state-of-the-art algorithms. Unlike the other algorithms, the proposed algorithm is free of parameter setting for a given dynamic range of the enhanced image and can be applied to a wide range of image types.

Index Terms—Contrast enhancement, Gaussian mixture modeling, global histogram equalization (GHE), histogram partition, normal distribution.

I. INTRODUCTION

The objective of an image enhancement technique is to bring out hidden image details or to increase the contrast of an image with a low dynamic range [1]. Such a technique produces an output image that subjectively looks better than the original image by increasing the gray-level differences (i.e., the contrast) among objects and background. Numerous enhancement techniques have been introduced, and these can be divided into three groups: 1) techniques that decompose an image into high- and low-frequency signals for manipulation [2], [3]; 2) transform-based techniques [4]; and 3) histogram modification techniques [5]–[16].

Techniques in the first two groups often use multiscale analysis to decompose the image into different frequency bands and enhance its desired global and local frequencies [2]–[4]. These techniques are computationally complex but enable global and local contrast enhancement simultaneously by transforming the signals in the appropriate bands or scales. Furthermore, they require appropriate parameter settings that might otherwise result in image degradations. For example, the center-surround Retinex [2] algorithm was developed to attain lightness and color constancy for machine vision applications. The constancy refers to the resilience of perceived color and lightness to spatial and spectral illumination variations. The benefits of the Retinex algorithm include dynamic range compression and color independence from the spatial distribution of the scene illumination. However, this algorithm can result in “halo” artifacts, particularly in boundaries between large uniform regions. Moreover, “graying out” can occur, in which the scene tends to change to middle gray.

Among the three groups, the third group received the most attention due to their straightforward and intuitive implementation qualities. Linear contrast stretching (LCS) and global histogram equalization (GHE) are two widely utilized methods for global image enhancement [1]. The former linearly adjusts the dynamic range of an image, and the latter uses an input-to-output mapping obtained from the cumulative distribution function (CDF), which is the integral of the image histogram. Since the contrast gain is proportional to the height of the histogram, gray levels with large pixel populations are expanded to a larger range of gray levels, whereas other gray-level ranges with fewer pixels are compressed to smaller ranges. Although GHE can efficiently utilize display intensities, it tends to over enhance the image contrast if there are high peaks in the histogram, often resulting in a harsh and noisy appearance of the output image. LCS and GHE are simple transformations, but they do not always produce good results, particularly for images with large spatial variation in contrast. In addition, GHE has the undesired effect of overemphasizing any noise in an image.

In order to overcome the aforementioned problems, local-histogram-equalization (LHE)-based enhancement techniques have been proposed, e.g., [5] and [6]. For example, the LHE method [6] uses a small window that slides through every image pixel sequentially, and only pixels within the current position of the window are histogram equalized; the gray-level mapping for enhancement is done only for the center pixel of the window. Thus, it utilizes local information. However, LHE sometimes causes over enhancement in some portion of the image and enhances any noise in the input image, along with the image features. Furthermore, LHE-based methods produce undesirable checkerboard effects.

Histogram specification (HS) [1] is a method that uses a desired histogram to modify the expected output-image histogram.
However, specifying the output histogram is not a straightforward task as it varies from image to image. The dynamic HS (DHS) [7] generates the specified histogram dynamically from the input image. In order to retain the original histogram features, DHS extracts the differential information from the input histogram and incorporates extra parameters to control the enhancement such as the image original value and the resultant gain control value. However, the degree of enhancement achievable is not significant.

Some research works have also focused on improving histogram-equalization-based contrast enhancement such as mean preserving bihistogram equalization (BBHE) [8], equal-area dualistic subimage histogram equalization (DSIHE) [9], and minimum mean-brightness (MB) error bihistogram equalization (MMBEBHE) [10]. BBHE first divides the image histogram into two parts with the average gray level of the input-image pixels as the separation intensity. The two histograms are then independently equalized. The method attempts to solve the brightness preservation problem. DSIHE uses entropy for histogram separation. MMBEBHE is the extension of BBHE, which provides maximal brightness preservation. Although these methods can achieve good contrast enhancement, they also generate annoying side effects depending on the variation in the gray-level distribution [7]. Recursive mean-separate histogram equalization [11] is another improvement of BBHE. However, it is also not free from side effects. Dynamic histogram equalization (DHE) [12] first smoothes the input histogram by using a 1-D smoothing filter. The smoothed histogram is partitioned into subhistograms based on the local minima. Prior to equalizing the subhistograms, each subhistogram is mapped into a new dynamic range. The mapping is a function of the number of pixels in each subhistogram; thus, a subhistogram with a larger number of pixels will occupy a bigger portion of the dynamic range. However, DHE does not place any constraint on maintaining the MB of the image. Furthermore, several parameters are used, which require appropriate setting for different images.

Optimization techniques have been also employed for contrast enhancement. The target histogram of the method, i.e., brightness-preserving histogram equalization with maximum entropy (BPHEME) [13], has the maximum differential entropy obtained using a variational approach under the MB constraint. Although entropy maximization corresponds to contrast stretching to some extent, it does not always result in contrast enhancement [14]. In the flattest HS with accurate brightness preservation (FHSABP) [14], convex optimization is used to transform the image histogram into the flattest histogram, subject to a MB constraint. An exact HS method is used to preserve the image brightness. However, when the gray levels of the input image are equally distributed, FHSABP behaves very similar to GHE. Furthermore, it is designed to preserve the average brightness, which may produce low contrast results when the average brightness is either too low or too high. In histogram modification framework (HMF), which is based on histogram equalization, contrast enhancement is treated as an optimization problem that minimizes a cost function [15]. Penalty terms are introduced in the optimization in order to handle noise and black/white stretching. HMF can achieve different levels of contrast enhancement through the use of different adaptive parameters. These parameters have to be manually tuned according to the image content to achieve high contrast. In order to design a parameter-free contrast enhancement method, genetic algorithm (GA) is employed to find a target histogram that maximizes a contrast measure based on edge information [16]. We call this method contrast enhancement based on GA (CEBGA). CEBGA suffers from the drawbacks of GA-based methods, namely, dependence on initialization and convergence to a local optimum. Furthermore, the mapping to the target histogram is scored by only maximum contrast, which is measured according to average edge strength estimated from the gradient information. Thus, CEBGA may produce results that are not spatially smooth. Finally, the convergence time is proportional to the number of distinct gray levels of the input image.

The aforementioned techniques may create problems when enhancing a sequence of images, when the histogram has spikes, or when a natural-looking enhanced image is required. In this paper, we propose an adaptive image equalization algorithm that is effective in terms of improving the visual quality of different types of input images. Images with low contrast are automatically improved in terms of an increase in the dynamic range. Images with sufficiently high contrast are also improved but not as much. The algorithm further enhances the color quality of the input images in terms of color consistency, higher contrast between foreground and background objects, larger dynamic range, and greater details in image contents. The proposed algorithm is free from parameter setting. Instead, the pixel values of an input image are modeled using the Gaussian mixture model (GMM). The intersection points of the Gaussian components are used in partitioning the dynamic range of the input image into input gray-level intervals. The gray levels of the pixels in each input interval are transformed according to the dominant Gaussian component and the CDF of the interval to obtain the contrast-equalized image.

This paper is organized as follows: Section II presents the proposed automatic image equalization algorithm and the necessary background related to the GMM fit of the input image data. Section III presents the subjective and quantitative comparisons of the proposed method with several state-of-the-art enhancement techniques. Section IV concludes this paper.

II. PROPOSED ALGORITHM

Let us consider an input image \( X = \{x(i, j) | 1 \leq i \leq H, 1 \leq j \leq W\} \) of size \( H \times W \) pixels, where \( x(i, j) \in \mathbb{R} \). Let us assume that \( X \) has a dynamic range of \([x_{\text{min}}, x_{\text{max}}]\), where \( x(i, j) \in [x_{\text{min}}, x_{\text{max}}] \). The main objective of the proposed algorithm is to generate an enhanced image \( Y = \{y(i, j) | 1 \leq i \leq H, 1 \leq j \leq W\} \), which has a better visual quality with respect to \( X \). The dynamic range of \( Y \) can be stretched or tightened into interval \([y_{\text{min}}, y_{\text{max}}]\), where \( y(i, j) \in [y_{\text{min}}, y_{\text{max}}] \), \( y_{\text{min}} < y_{\text{max}} \) and \( y_{\text{min}}, y_{\text{max}} \in \mathbb{R} \).

A. Modeling

A GMM can model any data distribution in terms of a linear mixture of different Gaussian distributions with different parameters. Each of the Gaussian components has a different mean, standard deviation, and proportion (or weight) in the
mixture model. A Gaussian component with low standard deviation and large weight represents compact data with a dense distribution around the mean value of the component. When the standard deviation becomes larger, the data is dispersed about its mean value. The human eye is not sensitive to small variations around dense data but is more sensitive to widely scattered fluctuations. Thus, in order to increase the contrast while retaining image details, dense data with low standard deviation should be dispersed, whereas scattered data with high standard deviation should be compacted. This operation should be done so that the gray-level distribution is retained. In order to achieve this, we use the GMM to partition the distribution of the input image into a mixture of different Gaussian components.

The gray-level distribution \( p(x) \), where \( x \in \mathbf{X} \), of the input image \( \mathbf{X} \) can be modeled as a density function composed of a linear combination of \( N \) functions using the GMM [17], i.e.,

\[
p(x) = \sum_{n=1}^{N} p(u_n)p(x|u_n)
\]

(1)

where \( p(x|u_n) \) is the \( n \)th component density and \( P(u_n) \) is the prior probability of the data points generated from component \( u_n \) of the mixture. The component density functions are constrained to be Gaussian distribution functions, i.e.,

\[
p(x|u_n) = \frac{1}{\sqrt{2\pi\sigma_{u_n}^2}} \exp\left(-\frac{(x - \mu_{u_n})^2}{2\sigma_{u_n}^2}\right)
\]

(2)

where \( \mu_{u_n} \) and \( \sigma_{u_n}^2 \) are the mean and the variance of the \( n \)th component, respectively. Each of the Gaussian distribution functions satisfies the following constraint:

\[
\int_{-\infty}^{\infty} p(x|u_n)dx = 1.
\]

(3)

The prior probabilities are chosen to satisfy the following constraints:

\[
\sum_{n=1}^{N} P(u_n) = 1 \quad \text{and} \quad 0 \leq P(u_n) \leq 1.
\]

(4)

A GMM is completely specified by its parameters \( \boldsymbol{\theta} = \{P(u_n), \mu_{u_n}, \sigma_{u_n}^2\}_{n=1}^{N} \). The estimation of the probability distribution function of an input-image data \( x \) reduces to finding the appropriate values of \( \boldsymbol{\theta} \). In order to estimate \( \boldsymbol{\theta} \), maximum-likelihood-estimation techniques such as the expectation maximization (EM) algorithm [18] have been widely used. Assuming the data points \( \mathbf{X} = \{x_1, x_2, \ldots, x_T \times W\} \) are independent, the likelihood of data \( \mathbf{X} \) is computed by

\[
\mathcal{L}(\mathbf{X}; \boldsymbol{\theta}) = \prod_{kl} p(x_k; \boldsymbol{\theta})
\]

(5)

given the distribution or, more specifically, the distribution parameters \( \boldsymbol{\theta} \). The goal of the estimation is to find \( \hat{\boldsymbol{\theta}} \) that maximizes the likelihood, i.e.,

\[
\hat{\boldsymbol{\theta}} = \arg\max_{\boldsymbol{\theta}} \mathcal{L}(\mathbf{X}; \boldsymbol{\theta}).
\]

(6)

Instead of maximizing this function directly, the log-likelihood \( L(\mathbf{X}; \boldsymbol{\theta}) = \ln \mathcal{L}(\mathbf{X}; \boldsymbol{\theta}) \equiv \sum_{kl} p(x_k; \boldsymbol{\theta}) \) is used because it is analytically easier to handle.

The EM algorithm starts from an initial guess for the distribution parameters and the log-likelihood is guaranteed to increase on each iteration until it converges. The convergence leads to a local or global maximum, but it can also lead to singular estimates, which is true, particularly for Gaussian mixture distributions with arbitrary covariance matrices. The initialization is one of the problems of the EM algorithm. The selection of initial guess (partly) determines where the algorithm converges or hits the boundary of the parameter space to produce singular meaningless results. Furthermore, the EM algorithm requires the user to set the number of components, and the number is fixed during the estimation process.

The Figueiredo–Jain (FJ) algorithm [19], which is an improved variant of the EM algorithm, overcomes major weaknesses of the basic EM algorithm. The FJ algorithm adjusts the number of components during estimation by annihilating components that are not supported by the data. It avoids the boundary when it annihilates components that are becoming singular. It is also allowed to start with an arbitrarily large number of components, which addresses the initialization of the EM algorithm. The initial guesses for component means can be distributed into the whole space occupied by the training samples, even setting one component for every single training sample. Due to its advantages over EM algorithm, in this paper, we adopt the FJ algorithm for parameter estimation.

Fig. 1(a) and (b) illustrates an input image and its histogram, together with its GMM fit, respectively. The histogram is modeled using four Gaussian components, i.e., \( N = 4 \). The close match between the histogram (shown as rectangular vertical bars) and the GMM fit (shown as solid black line) is obtained using the FJ algorithm. There are three main gray tones in the input image corresponding to the tank, its shadow, and the image background. The other gray-level tones are distributed around the three main tones. However, FJ algorithm results in four Gaussian components \( (N = 4) \) for the mixture model. This is because the gray tone with the highest average gray value corresponding to the image background has a deviation too large for a single Gaussian component to represent it. Thus, it is represented by two Gaussian components, i.e., \( u_3 \) and \( u_4 \), as shown in Fig. 1(b). All intersection points between Gaussian components that fall within the dynamic range of the input
image are denoted by yellow circles, and significant intersection points that are used in dynamic range representation are denoted by black circles. There is only one dominant Gaussian component between two intersection points, which adequately represents the data within this gray-level interval. For instance, the range of the input data within the interval of [35, 90] is represented by the Gaussian component \( w_1 \) (shown as solid blue line). Thus, the data within each interval are represented by a single Gaussian component that is dominant with respect to the other components. The dynamic range of the input image is represented by the union of all intervals.

B. Partitioning

The significant intersection points are selected from all the possible intersections between the Gaussian components. The intersection points between two Gaussian components \( w_m \) and \( w_n \) are found by solving

\[
P(w_m)p(x|w_m) = P(w_n)p(x|w_n)
\]

or equivalently

\[
\frac{(x - \mu_{w_m})^2}{2\sigma_{w_m}^2} + \frac{(x - \mu_{w_n})^2}{2\sigma_{w_n}^2} = \ln \left( \frac{P(w_n)}{P(w_m)} \right) \frac{\sigma_{w_m}}{\sigma_{w_n}}
\]

which results in

\[
a\delta^2 + b\delta + c = 0
\]

where

\[
a = (\sigma_{w_m}^2 - \sigma_{w_n}^2),
\]

\[
b = 2(\mu_{w_m}\sigma_{w_m}^2 - \mu_{w_n}\sigma_{w_n}^2),
\]

\[
c = (\mu_{w_m}\sigma_{w_m}^2 - \mu_{w_n}\sigma_{w_n}^2) - 2\sigma_{w_m}^2\sigma_{w_n}^2 \ln \left( \frac{P(w_n)}{P(w_m)} \right) \frac{\sigma_{w_m}}{\sigma_{w_n}}.
\]

The parametric equation (9) has two roots, i.e.,

\[
x_m^{(1)} = \frac{-b + \sqrt{b^2 - 4ac}}{2a}, \quad x_m^{(2)} = \frac{-b - \sqrt{b^2 - 4ac}}{2a}.
\]

In Fig. 1(b), all intersection points between GMM components are denoted by yellow circles. The numerical values of the intersection points determined using (10) are shown in Table I. Table I is symmetric, i.e., the intersection points between components \( w_1 \) and \( w_2 \) are the same as the intersection points between components \( w_2 \) and \( w_1 \). The intersection points of two components are independent of the order of the components. All possible intersection points that are within the dynamic range of the image are detected. The leftmost intersection point between components \( w_1 \) and \( w_2 \) is at -718.88, which is not within the dynamic range of the input image; thus, it could not be considered. In order to allow the combination of intersection points to cover only the entire dynamic range of the input image, a further process is needed.

The total number of intersection points calculated is \( N(N - 1) \). The significant intersection points \( x_m^{(d)} \), where \( d \in \{1, \ldots, D\}, \quad D \leq N(N - 1) \), are selected among all points. For a given intersection point \( x_m^{(k)} \), where \( k = \{1, 2\} \), between Gaussian components \( w_m \) and \( w_n \), it is selected as a significant intersection point if and only if it is a real number in the dynamic range of the input image, i.e., \( x_m^{(k)} \in [x_{a1}, x_{a2}] \) and the Gaussian components \( w_m \) and \( w_n \) contain the maximum value in the mixture for point \( x_m^{(k)} \), i.e.,

\[
P(w_m)p(x_m^{(k)}|w_m) \geq P(w_n)p(x_m^{(k)}|w_n)
\]

or equivalently

\[
P(w_m)p(x_m^{(k)}|w_m) > P(w_k)p(x_m^{(k)}|w_k)
\]

where \( \forall w_k \neq \{w_m, w_n\} \).

The significant intersection points are sorted in ascending order of their value and are partitioned into gray-level intervals to cover the entire dynamic range of \( X \), i.e., \( x \in x = [x_1^{(1)}, x_2^{(1)}] \cup [x_1^{(2)}, x_2^{(2)}] \cup \cdots \cup [x_1^{(D)}, x_2^{(D)}] \). The leftmost significant intersection point \( x_s^{(1)} \) is selected as the value of \( x \) for which

\[
x_s^{(1)} = x \quad F(x) \geq \frac{T_h}{HW} \quad F(x - \Delta) < \frac{T_h}{HW}
\]

where the minimum distance between two consecutive numbers is \( \Delta \), e.g., \( \Delta = 1 \) in the case of the 8-bit input image \( X \) considered in this paper. \( F(x) \) is the CDF of \( x \), and \( T_h \) is the minimum number of pixels that will be excluded from the tails of the gray-level distribution of \( x \). To consider all pixel gray values of \( X \), we set \( T_h = 1 \). Similarly, the rightmost significant intersection point \( x_r^{(r)} \) is selected by considering the tail of the gray-level distribution of \( x \) for which

\[
x_r^{(r)} = x \quad 1 - F(x) \geq \frac{T_h}{HW} \quad 1 - F(x + \Delta) < \frac{T_h}{HW}
\]

The significant intersection points that fall outside of interval \([x_s^{(1)}, x_s^{(r)}]\) are ignored since they are the intersection points between two Gaussian components that fall outside the dynamic range of \( X \), and \( x_s^{(r)} \) is updated as \( x_s^{(r)} = [x_1^{(1)}, x_2^{(2)}, \ldots, x_s^{(K)}] \) with \( x_s^{(1)} < x_s^{(2)} < \ldots < x_s^{(K)} \), where \( K \) is the maximum number of significant intersection points. In Fig. 1, the six significant intersection points are denoted by black circles, and the range of \( x_s \) covers the entire dynamic range of \( X \).

The CDF of \( x \) is

\[
F(x) = \int_{-\infty}^{x} p(x)dx = \int_{-\infty}^{x} \sum_{n=1}^{N} P(w_n)p(x|w_n)dx
\]

<table>
<thead>
<tr>
<th>TABLE I</th>
<th>NUMERICAL VALUES OF INTERSECTION POINTS DENOTED BY YELLOW CIRCLES IN FIG. 1(b) BETWEEN COMPONENTS OF THE GMM FIT TO THE GRAY-LEVEL IMAGE SHOWN IN FIG. 1(a)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>w_1</td>
<td>-</td>
</tr>
<tr>
<td>w_2</td>
<td>-718.88, 90.05</td>
</tr>
<tr>
<td>w_3</td>
<td>115.18, 225.52</td>
</tr>
<tr>
<td>w_4</td>
<td>129.46, 193.08</td>
</tr>
</tbody>
</table>
\begin{align}
&= \sum_{n=1}^{N} P(w_n) \int_{x_n}^{x} \frac{1}{\sqrt{2\pi\sigma_{w_n}^2}} \exp\left(-\frac{(x - \mu_{w_n})^2}{2\sigma_{w_n}^2}\right) \, dx \\
&= \sum_{n=1}^{N} P(w_n) \int_{-\infty}^{\infty} \frac{1}{\sqrt{\pi}} \exp(-t^2) \, dt.
\end{align}
\tag{15}

It can be calculated using the closed-form expression, i.e.,
\begin{equation}
F(x) = \sum_{n=1}^{N} P(w_n) F_{w_n}(x) \tag{16}
\end{equation}
where \(F_{w_n}(x)\) is the CDF of the Gaussian component \(w_n\), and using the error function \(\text{erf}(x)\) [20], it is computed as \(F_{w_n}(x) = \beta(x - \mu_{w_n} / \sqrt{2\sigma_{w_n}^2})\), where
\begin{equation}
\beta(x) = \begin{cases} 
(1 + \text{erf}(x)) / 2, & \text{if } x \geq 0 \\
(1 - \text{erf}(x)) / 2, & \text{otherwise}
\end{cases} \tag{17}
\end{equation}
and the numerical values of \(\text{erf}(x)\) are tabulated in [20]. Function \(\beta(x)\) is invertible, i.e., for a given \(\beta(x) = a\), \(x = \beta^{-1}(a)\) exists.

The consecutive pairs of significant intersection points are used to partition the dynamic range of \(X\) into subintervals, i.e., \([x_s^{(1)}, x_s^{(2)}] \cup [x_s^{(2)}, x_s^{(3)}] \cup \cdots \cup [x_s^{(K-2)}, x_s^{(K-1)}] \cup [x_s^{(K-1)}, x_s^{(K)}]\). Subinterval \([x_s^{(k)}, x_s^{(k+1)}]\) is represented by a Gaussian component \(u_k\), which is dominant with respect to the other Gaussian components in the subinterval. The dominant Gaussian component is found by considering the \textit{a posteriori} probability of each component in interval \([x_s^{(k)}, x_s^{(k+1)}]\), which is equivalent to the area under the Gaussian component, i.e.,
\begin{equation}
u_k = \text{arg max}_{\forall u_j} \left(F_{u_j}(x_s^{(k+1)}) - F_{u_j}(x_s^{(k)})\right). \tag{18}
\end{equation}

C. Mapping

Interval \([x_s^{(k)}, x_s^{(k+1)}]\), where \(k = 1, 2, \ldots, K - 1\), is mapped onto the dynamic range of the output image \(Y\). In the mapping, each interval covers a certain range, which is proportional to weight \(\alpha_k\), where \(\alpha_k \in [0, 1]\), which is calculated by considering two figure of merits simultaneously, i.e., the rate of the total number of pixels that fall into interval \([x_s^{(k)}, x_s^{(k+1)}]\) and the standard deviation of the dominant Gaussian component \(u_k\), i.e.,
\begin{equation}
\alpha_k = \frac{\sigma_{u_k}}{\sum_{j=1}^{K} \sigma_{u_j}^2} \frac{F\left(x_s^{(k+1)}\right) - F\left(x_s^{(k)}\right)}{\sum_{i=1}^{K-1} \sum_{j=1}^{K} F\left(x_s^{(i+1)}\right) - F\left(x_s^{(i)}\right)}, \tag{19}
\end{equation}

The first term adjusts the brightness of the equalized image, and \(\gamma \in [0, 1]\) is brightness constant (in this paper, \(\gamma = 0.5\) is used). The lower the value of \(\gamma\), the brighter the output image is. The second term in (19) is related to the gray-level distribution and is used to retain the overall content of the data in the interval. Equation (19) maintains a balance between the data distribution and the variance of the data in a certain interval. Since the human eye is more sensitive to sudden changes in widely scattered data and less sensitive to smooth changes in densely scattered data, (19) gives larger weights to widely scattered data (larger variance) and vice versa.

Using \(\alpha_k\), the input interval \([x_s^{(k)}, x_s^{(k+1)}]\) is mapped onto the output interval \([y^{(k)}, y^{(k+1)}]\) according to
\begin{equation}
y^{(k)} = y_{ld} + (y_{ud} - y_{ld}) \sum_{k=1}^{K-1} \alpha_i \tag{20}
\end{equation}

The aforementioned mapping guarantees that the output dynamic range is covered by the mapping, i.e., \([y_{ld}, y_{ud}] = [y^{(1)}, y^{(K)}]\). In the final mapping of pixel values from the input interval onto the output interval, the CDF of the distribution in the output interval is preserved. Let the Gaussian distribution \(u_{k'}\) with parameters \(\mu_{u_{k'}}^1\) and \(\sigma_{u_{k'}}^2\) represent the Gaussian component \(u_k\) in range \([y^{(k)}, y^{(k+1)}]\). Parameters \(\mu_{u_{k'}}^1\) and \(\sigma_{u_{k'}}^2\) are found by solving the following equations simultaneously:
\begin{align}
F_{u_{k'}}\left(x_s^{(k)}\right) &= F_{u_{k'}}\left(y^{(k)}\right) \tag{21} \\
F_{u_{k'}}\left(x_s^{(k+1)}\right) &= F_{u_{k'}}\left(y^{(k+1)}\right) \tag{22}
\end{align}

so that the area under the Gaussian distribution \(u_{k'}\) between \([x_s^{(k)}, x_s^{(k+1)}]\) is equal to the area under the Gaussian distribution \(u_{k'}\) in interval \([y^{(k)}, y^{(k+1)}]\). Using (17), together with (21) and (22), one can write
\begin{align}
\beta\left(\frac{x_s^{(k)} - \mu_{u_{k'}}}{\sqrt{2\sigma_{u_{k'}}}}\right) &= \beta\left(\frac{y^{(k)} - \mu_{u_{k'}}}{\sqrt{2\sigma_{u_{k'}}}}\right) \tag{23} \\
\beta\left(\frac{x_s^{(k+1)} - \mu_{u_{k'}}}{\sqrt{2\sigma_{u_{k'}}}}\right) &= \beta\left(\frac{y^{(k+1)} - \mu_{u_{k'}}}{\sqrt{2\sigma_{u_{k'}}}}\right) \tag{24}
\end{align}

which is equivalent to
\begin{align}
x_s^{(k)} - \mu_{u_{k'}} &= \frac{y^{(k)} - \mu_{u_{k'}}}{\sqrt{2\sigma_{u_{k'}}}} \sqrt{2\sigma_{u_{k'}}} \tag{25} \\
x_s^{(k+1)} - \mu_{u_{k'}} &= \frac{y^{(k+1)} - \mu_{u_{k'}}}{\sqrt{2\sigma_{u_{k'}}}} \sqrt{2\sigma_{u_{k'}}}. \tag{26}
\end{align}

Using (25) and (26), the parameters of the Gaussian distribution \(u_{k'}\) are computed as follows:
\begin{align}
\mu_{u_{k'}} &= \frac{\left(y^{(k)} - \mu_{u_{k'}}\right) x_s^{(k+1)} - \mu_{u_{k'}} y^{(k+1)} - \mu_{u_{k'}} y^{(k)}}{\left(y^{(k)} - \mu_{u_{k'}}\right) x_s^{(k+1)} - \mu_{u_{k'}} y^{(k+1)} - \mu_{u_{k'}} y^{(k)} - 1} \tag{27} \\
\sigma_{u_{k'}} &= \left(y^{(k)} - \mu_{u_{k'}}\right) \sigma_{u_{k'}}. \tag{28}
\end{align}

The mapping of \(x\) to \(y\), where \(x \in [x_s^{(k)}, x_s^{(k+1)}]\) and \(y \in [y^{(k)}, y^{(k+1)}]\), is achieved by keeping the CDFs of the Gaussian distribution \(u_k\) and the Gaussian distribution \(u_{k'}\) equal, i.e.,
\begin{equation}
\beta\left(\frac{x - \mu_{u_k}}{\sqrt{2\sigma_{u_k}}}\right) = \beta\left(\frac{x_s^{(k)} - \mu_{u_{k'}}}{\sqrt{2\sigma_{u_{k'}}}}\right) \tag{29}
\end{equation}
and is the probability of the pixel intensity according to (31). Fig. 2(c) shows to , i.e., and an output image color space [1], and measures its content, where a higher low- , and , re- is thus defined as

\[
\begin{align*}
\beta \left( \frac{y - \mu_{w_{k,l}}}{\sqrt{2\sigma_{w_{k,l}}}} \right) - \beta \left( \frac{y^{(k)} - \mu_{w_{j,l}}}{\sqrt{2\sigma_{w_{j,l}}}} \right)
\end{align*}
\] (29)

where using the equality in (23), i.e.,

\[
\beta \left( \frac{x - \mu_{w_{k,l}}}{\sqrt{2\sigma_{w_{k,l}}}} \right) = \beta \left( \frac{y - \mu_{w_{j,l}}}{\sqrt{2\sigma_{w_{j,l}}}} \right) \Rightarrow x - \mu_{w_{k,l}} = \frac{y - \mu_{w_{j,l}}}{\sqrt{2\sigma_{w_{j,l}}}}
\]

results in the following mapping of given \(x\) to the corresponding \(y\) according to the Gaussian distributions \(w_{k,l}\) and \(w_{j,l}\), i.e.,

\[
y = \left( \frac{x - \mu_{w_{k,l}}}{\sigma_{w_{k,l}}} \right) \sigma_{w_{j,l}} + \mu_{w_{j,l}},
\] (30)

The final mapping from \(x\) to \(y\) is achieved by considering all Gaussian components in the GMM to retain the pixel distributions in input and output intervals equal by using the superposition of distributions, together with (30), i.e.,

\[
y = \sum_{i=1}^{N} \left( \left( \frac{x - \mu_{w_{i,l}}}{\sigma_{w_{i,l}}} \right) \sigma_{w_{i,l}} + \mu_{w_{i,l}} \right) P_{w_{i,l}}.
\] (31)

Fig. 2(a)–(c) shows the input images and the equalized images using the proposed algorithm, respectively, where the dynamic range of the output image is \([y_{\text{min}}, y_{\text{max}}] = [0, 255]\), and the mappings between input-image data points \(x\) and equalized output-image data points \(y\) are according to (31). Fig. 2(c) shows that a different mapping is applied to a different input gray-level interval. Fig. 2(b) shows that the proposed algorithm increases the brightness of the input image while keeping the high contrast between object boundaries. The input image in the second row of Fig. 2(a) only has 15 different gray levels; thus, it is difficult to observe the image features. The proposed algorithm linearly transforms the gray levels, as shown in Fig. 2(c), so that the image features are easily discernible in Fig. 2(b).

D. Extending the Proposed Method to Color Images

One approach to extend the grayscale contrast enhancement to color images is to apply the method to their luminance component only and preserve the chrominance components. Another is to multiply the chrominance values with the ratio of their input and output luminance values to preserve the hue. The former approach is employed in this paper, where an input RGB image is transformed to the CIE \(L^*a^*b^*\) color space [1], and the luminance component \(L^*\) is processed for contrast enhancement. The inverse transformation is then applied to obtain the contrast-enhanced RGB image.

III. EXPERIMENTAL RESULTS

A data set comprising of standard test images from [21]–[24] is used to evaluate and compare the proposed algorithm with our implementations of GHE [1], BPHEME [13], FHSABP [14], CEBGA [16], and the weighted histogram approximation of HMF [15]. GHE, BPHEME, FHSABP, and CEBGA are free of parameter selection, but HMF requires parameter tuning, which is manually set according to the input test images. It is worth to note that exact HS is used in FHSABP [14] to achieve a high degree of brightness preservation between input and output images. The test images show wide variations in terms of average image intensity and contrast. Thus, they are suitable for measuring the strength of a contrast enhancement algorithm under different circumstances.

An output image is said to have been enhanced over the input image if it enables the image details to be better perceived. An assessment of image enhancement is not an easy task as an improved perception is difficult to quantify. Thus, it is desirable to have both quantitative and subjective assessments. It is also necessary to establish measures for defining good enhancement. We use absolute MB error (AMBE) [10], discrete entropy (DE) [25], and edge-based contrast measure (EBCM) [26] as quantitative measures. For color images, the contrast enhancement is quantified by computing these measures on their luminance channel \(L^*\) only.

AMBE is the absolute difference between the mean values of an input image \(X\) and an output image \(Y\), i.e.,

\[
\text{AMBE}(X, Y) = |\text{MB}(X) - \text{MB}(Y)|
\] (32)

where \(\text{MB}(X)\) and \(\text{MB}(Y)\) are the MB values of \(X\) and \(Y\), respectively. The lower the value of AMBE, the better the brightness preservation is.

The DE of image \(X\) measures its content, where a higher value indicates an image with richer details. It is defined as

\[
\text{DE}(X) = -\sum_{x_i} p(x_i) \log(p(x_i))
\] (33)

where \(p(x_i)\) is the probability of the pixel intensity \(x_i\), which is estimated from the normalized histogram.

The EBCM is based on the observation that the human perception mechanisms are very sensitive to contours (or edges) [26]. The gray level corresponding to object frontiers is obtained by computing the average value of the pixel gray levels weighted by their edge values. Contrast \(c(i,j)\) for a pixel of image \(X\) located at \((i,j)\) is thus defined as

\[
c(i,j) = \frac{|x(i,j) - c(i,j)|}{|x(i,j) + c(i,j)|}
\] (34)
where the mean edge gray level is

\[ e(i,j) = \sum_{(k,l) \in N(i,j)} g(k,l) \frac{x(k,l)}{\sum_{(k,l) \in N(i,j)} g(k,l)}, \] (35)

\[ N(i,j) \] is the set of all neighboring pixels of pixel \((i,j)\), and \(g(k,l)\) is the edge value at pixel \((k,l)\). Without loss of generality, we employ \(3 \times 3\) neighborhood, and \(g(k,l)\) is the magnitude of the image gradient estimated using the Sobel operators [1]. The EBCM for image \(X\) is thus computed as the average contrast value, i.e.,

\[ \text{EBCM}(X) = \frac{1}{H \times W} \sum_{i=1}^{H} \sum_{j=1}^{W} e(i,j). \] (36)

It is expected that, for an output image \(Y\) of an input image \(X\), the contrast is improved when \(\text{EBCM}(Y) \geq \text{EBCM}(X)\).

A. Qualitative Assessment

1) Grayscale Images: Some contrast enhancement results on grayscale images are shown in Figs. 3–6. The corresponding mapping functions used are shown in Fig. 7.

The dark input image in Fig. 3 shows a firework display [21]. GHE has increased the overall brightness of the image, but the increase in contrast is not significant, and the washout effect is apparent. Both the darker and brighter regions become even brighter. This is verified by the mapping function in Fig. 7(a), which maps input gray level 0 to output gray level 105. BPHEME and FHSABP preserve the input-image average brightness value of 18, resulting in output images with very low brightness, and thus, the contrast enhancement is not noticeable. The mapping functions verify this observation, where the low output brightness and the nonlinear mapping from the input to the output are apparent. BPHEME achieves an almost one-to-one mapping between the input and the output to obtain the maximum entropy. The result of HMF is visually pleasing, providing high contrast and high dynamic range (HDR). However, there are two different spark clusters due to the fireworks and the smoke between sparks. HMF over enhances the brighter pixels of the sparks and the surrounding smoke, so that the smoke pixels are also identified as spark pixels. This over enhancement is represented as a sharp change in the mapping function. Due to the not-sharp image details caused by the smoke from fireworks, CEGBA can only improve the overall brightness of the image. This is verified by the mapping
Fig. 7. Mapping functions of enhanced images: (a) Fig. 3; (b) Fig. 4; (c) Fig. 5; and (d) Fig. 6. Key: (Green solid line) no-change mapping; (black dash-dotted line) GHE; (red solid line) BPHEME; (red dash-dotted line) FHSABP; (blue solid line) HMF; (blue dash-dotted line) CEBGA; and (black solid line) the proposed algorithm.

Fig. 8. Histograms of original and enhanced images shown in Fig. 6: (a) original image; (b) GHE; (c) BPHEME; (d) FHSABP; (e) HMF; (f) CEBGA; and (g) proposed.

function, which is almost parallel to the no-change mapping. However, in the proposed algorithm, the dynamic range of the input image is modeled with the GMM, which makes it possible to model the intensity values of sparks and smoke differently. Input gray-level values are assigned to output gray-level values according to their representative Gaussian components. The nonlinear mapping is designed to utilize the full dynamic range of the output image. Thus, the proposed algorithm improves the overall contrast while preserving image details.

The input image of an island in Fig. 4 [22] has an average brightness value of 125. The results obtained by the different algorithms are similar as verified by the similar mapping functions in Fig. 7(b). Since the average brightness value of the input image is very close to 127.5, BPHEME and FHSABP obtain similar target histograms. The slight difference of FHSABP from BPHEME is due to the exact HS used in FHSABP. The results of HMF and CEBGA are also a match because both algorithms employ similar edge information. Where the sky and sea converge, GHE, BPHEME, and FHSABP provide a higher contrast than HMF and CEBGA. The proposed algorithm provides a contrast that is neither too high nor too low.

The bright input image in Fig. 5 shows an aerial view of a junction in a city [23]. GHE increases the overall contrast of the image significantly, but the image looks darker as verified by its mapping function in Fig. 7(c). The contrast improvements obtained by using BPHEME, FHSABP, HMF, and CEBGA are very slight. HMF fails to provide an improvement due to weak edge information. The proposed algorithm, on the other hand, does not darken the image and produces sufficient contrast for the different objects to be recognized.

The input image Girl in Fig. 6 consists of challenging conditions for an enhancement algorithm, i.e., very bright and dark objects, and an average brightness value of 139 [see its histogram in Fig. 8(a)]. Since the average brightness value of the input image is near to 127.5, GHE, BPHEME, and FHSABP very similarly perform. This can be verified by the visual results shown in Fig. 6(b)–(d), the mapping functions shown in Fig. 7(d), and the histograms in Fig. 8(b)–(d). The output histograms fail to achieve smooth distribution between high and low gray levels. Thus, the enhancement results of GHE, BPHEME, and FHSABP are visually unpleasing. Since the output histogram of HMF achieves a smoother distribution in between low and high gray values, as shown in Fig. 8(e), its result is more natural looking. CEBGA also produces a natural-looking output image; however, the overall enhancement is not significant. The proposed algorithm preserves the overall shape of the gray-level distribution and redistributes the gray levels of the input image within the dynamic range [see Fig. 8(g)], thus retaining the natural look of the enhanced image [see Fig. 6(g)]. Although it slightly darkens the girl’s hair, the perceived contrast has significantly improved.

2) Color Images: In Fig. 9, the over enhancement provided by GHE, BPHEME, and FHSABP whitens some areas of the concrete ground. HMF and CEBGA provide similar results, whereas the proposed algorithm enhances the contrast and the average brightness to improve the overall image quality. In Fig. 10, GHE, BPHEME, FHSABP, and HMF cause a part of the sky to be too bright. CEBGA and the proposed algorithm improve the overall contrast considerably and maintain high visual quality. In Fig. 11, GHE, BPHEME, and FHSABP result in the loss of details in the clouds and on the top of the yellow hat, whereas HMF and CEBGA retain the details while increasing the contrast. However, the contrast between the right side of the wall and the sky is not sufficiently high. The proposed algorithm keeps the details while improving the overall contrast. Finally, in Fig. 12, GHE makes the stones around the window and the pink flower very bright; hence, the enhanced image has an unnatural look. Although BPHEME and FHSABP perform better than GHE, they do not remove this effect completely. This effect is reduced by HMF, CEBGA, and the proposed
algorithm. Moreover, the colors of the window and the wall are better differentiated in the result of the proposed algorithm.

**B. Visual Assessment Score**

In order to assign a visual assessment score to each algorithm for each enhanced image, subjective perceived quality tests are performed by a group of 30 subjects on the results of the six algorithms for the eight test images. For each test on a test image, a subject is shown seven images at the same time, i.e., the original test image (placed in the center of view) and the output images processed by the six algorithms (randomly placed around the original test image). The subject is then asked to score the quality of the processed image by assigning one of the six numeric scores (0, 1, 2, 3, 4, and 5), where score “0” is for very bad and annoying enhancement (the image quality is totally distorted), score “3” is for no noticeable enhancement (natural and similar to the original image), score “5” is for significant enhancement without annoying distortion (looks natural across the overall image), and other values are selected according to the perceived image quality.

The mean opinion scores (MOSs) and the corresponding standard deviations of the visual assessment are shown in Table II. The MOSs support the qualitative assessments in Section III-A. It shows that the proposed method performs best for six out of the eight test images. For the remaining two images, the proposed method is ranked second and fourth. For each test image, the standard deviations of the MOSs of different algorithms are similar, which indicate that the uncertainty of each subject in scoring is similar. Table II also shows that only CEBGA and the proposed method always improve the image quality since all of their MOSs are greater than 3 (where score “3” indicates natural and similar to the original image), with the proposed method achieving higher MOSs than CEBGA.

**C. Quantitative Assessment**

The quantitative measures AMBE, DE, and EBCM are not always good indicators of enhancement that agree with the perceived image quality, e.g., for the Girl image. The AMBE for GHE, BPHEME, FHSABP, HMF, CEBGA, and the proposed method on the Girl image are 5.60, 5.50, 5.50, 10.30, 1.60, and 12.90, respectively, indicating that CEBGA performs the best in terms of brightness preservation, while the proposed method performs the worst. The DE of the Girl image is 3.87, whereas
those of the output images by GHE, BPHEME, FHSABP, HMF, CEBGA, and the proposed method are 3.65, 3.65, 3.65, 3.70, 3.45, and 3.81, respectively, indicating that the proposed method performs the best in terms of entropy. The EBCM of the Girl image is 0.08, whereas those of the output images of GHE, BPHEME, FHSABP, HMF, CEBGA, and the proposed method are, respectively, 0.23, 0.22, 0.22, 0.17, 0.11, and 0.12, indicating that all methods result in higher contrast. Although GHE, BPHEME, and FHSABP result in the highest values of the EBCM, the processed images are not natural.

In order to evaluate the performance of the six algorithms for a wide range of images, they are applied to 300 test images of the Berkeley image data set [24]. Measurement values of the MB, the DE, and the EBCM are computed from the original and output images. The values from the original images are sorted in ascending order, and the images are accordingly indexed (see Fig. 13).

Fig. 13(a) shows that, except for GHE, the low MB in the original image results in the low MB in the output images. GHE consistently maps the MB of the output image close to 127.5, which is the midvalue of the 8-bit gray-level dynamic range. The average of the AMBE for GHE, BPHEME, FHSABP, HMF, CEBGA, and the proposed method are 21.09, 1.30, 1.28, 10.07, 12.23, and 8.80, respectively. Thus, in terms of brightness preservation, BPHEME and FHSABP produce very similar results and perform the best. The proposed method performs better than HMF and CEBGA. In order to support this hypothesis statistically, a statistical significance test is performed. For the purpose of brightness preservation, BPHEME and FHSABP produce very similar results and perform the best. The proposed method performs third best.

The null hypothesis $H_0$ proposes that the contrast enhancement algorithm achieves the MB preservation between the input and output images, whereas the alternative hypothesis $H_1$ proposes otherwise. The probability that a value at least as extreme as the test statistic would be observed under the null hypothesis $H_0$ is the $p$-value [27]. Thus, the higher the $p$-value, the stronger the null hypothesis $H_0$ is. The $p$-values of different algorithms are shown in Table III for hypothesis (37). When the 99% confidence level is considered, BPHEME, FHSABP, and the proposed method do not reject $H_0$, whereas GHE, HMF, and CEBGA reject it in favor of $H_1$. Thus, statistically, BPHEME, FHSABP, and the proposed method achieve brightness preservation between the input and output images on the Berkeley data set when 99% confidence level is considered. With the third largest $p$-value, this test of significance verifies the result from the average of the AMBE over the 300 images that the proposed method performs third best.

Fig. 13(b) shows that the performance of GHE, BPHEME, and FHSABP in terms of the DE are very similar. The average absolute DE difference between the input and output images for GHE, BPHEME, FHSABP, HMF, CEBGA, and the proposed method are 0.12, 0.12, 0.11, 0.05, 0.38, and 0.04, respectively. Since the entropy is related to the overall image content, the proposed method performs best by preserving the overall content of the image while improving its contrast. Similar to (37), in order to statistically show that the proposed method performs
well in terms of entropy preservation, the equivalence of two sets \(\{\text{DE}(X_i)\}_{i=1}^{n}\) and \(\{\text{DE}(Y_i)\}_{i=1}^{n}\) is evaluated. The following hypotheses are evaluated using the KS test:

\[
H_0: \text{the DE is preserved;}
\]
\[
H_1: \text{the DE is not preserved.}
\] (38)

The null hypothesis \(H_0\) is used to propose that the contrast enhancement algorithm achieves the DE preservation, whereas the alternative hypothesis \(H_1\) proposes otherwise. The \(p\)-values of different algorithms are shown in Table III. The results indicate that, when 99% confidence level is considered, only HMF and the proposed algorithm do not reject \(H_0\). The other algorithms reject it in favor of \(H_1\). With the highest \(p\)-value, this test of significance also verifies the result from the average absolute DE difference between the input and output images for the 300 images that the proposed method performs best.

The EBCM values are shown in Fig. 13(c). Although a high EBCM does not always mean good and natural image enhancement, it is at least expected that the EBCM of an output image is higher than that of its input image. Out of 300 images, GHE, BPHEME, FHSABP, HMF, CEBGA, and the proposed method produce 294, 300, 296, 293, 286, and 300 output images, respectively, which is higher than or equal to the EBCM of the input images. The average absolute EBCM difference between the input and output images for GHE, BPHEME, FHSABP, HMF, CEBGA, and the proposed method are 0.0652, 0.0603, 0.0573, 0.0361, 0.0278, and 0.0366, respectively. As expected, GHE provides the highest contrast improvement in terms of average absolute EBCM difference. The proposed method and HMF similarly perform, and CEBGA provides the worst performance. However, it is worth to note that only two algorithms, i.e., BPHEME and the proposed method, increase the EBCM for all 300 images. The following hypotheses are evaluated using the KS test to determine if the output EBCM values are higher than the input EBCM values:

\[
H_0: \text{the contrast is improved;}
\]
\[
H_1: \text{the contrast is not improved.}
\] (39)

The null hypothesis \(H_0\) proposes that the output image resulted from the contrast enhancement algorithm has higher contrast than that of the input image, i.e., \(\{\text{EBCM}(Y_i)\} \geq \text{EBCM}(X_i)\). The \(p\)-values for (39) of different algorithms are shown in Table III. According to the 99% confidence level, all contrast enhancement algorithms do not reject \(H_0\). Thus, statistically, all algorithms produce higher contrast output images. It is worth to note that BPHEME and the proposed method do not reject \(H_0\) with 100% confidence. Thus, for the Berkeley data set, they produce higher contrast output images.

D. Application to HDR Compression

The proposed algorithm can be applied for rendering HDR images on conventional displays. Thus, we compare some of our results with those of the state-of-the-art method proposed in [28]. In the Fattal et al. method, the gradient field of the luminance image is manipulated by attenuating the magnitudes of large gradients. A low dynamic range image is then obtained by solving a Poisson equation on the modified gradient field.

The results in [28], a few of which are in Fig. 14, show that the method is capable of drastic dynamic range compression while preserving fine details and avoiding common artifacts such as halos, gradient reversals, or loss of local contrast. Fig. 14 also shows that the proposed algorithm produces comparable results. It is worth noting that our results are obtained without any parameter tuning.

IV. CONCLUSION

In this paper, we have proposed an automatic image enhancement algorithm that employs Gaussian mixture modeling of an input image to perform nonlinear data mapping for generating visually pleasing enhancement on different types of images. Performance comparisons with state-of-the-art techniques show that the proposed algorithm can achieve image equalization that is good enough even under diverse illumination conditions. The proposed algorithm can be applied to both gray-level and color images without any parameter tuning. It can be also used to render HDR images. It does not distort the overall content of an input image with contrast that is high enough. It further improves the color content, brightness, and contrast of an image automatically. Using the tests of significance on the Berkeley data set, it has been shown that the proposed method achieves brightness preservation, DE preservation, and contrast improvement under the 99% confidence level.

REFERENCES


Turgay Celik received the Ph.D. degree in engineering from the University of Warwick, Coventry, U.K., in 2011. He is currently a Research Fellow with the Bioinformatics Institute, A*STAR, Singapore. He has produced extensive publications in various international journals and conferences. He has been acting as a reviewer for various international journals and conferences. His research interests are in the areas of bioinformatics, digital signal, image and video processing, pattern recognition and artificial intelligence, unusual event detection, remote sensing, and global optimization techniques.

Tardi Tjahjadi (SM’02) received the B.Sc. (Hons.) degree in mechanical engineering from the University College London, London, U.K., in 1980 and the M.Sc. degree in management sciences (operational management) and Ph.D. degree in total technology from the University of Manchester Institute of Science and Technology, Manchester, U.K., in 1981 and 1984, respectively.

He joined the School of Engineering, University of Warwick, Coventry, U.K., and U.K. Daresbury Synchrotron Radiation Source Laboratory as a Joint Teaching Fellow in 1984. He was appointed a Lecturer in computer systems engineering at the same university in 1986, and has been an Associate Professor since 2000. His research interests include multiresolution image processing, image sequence processing, and 3-D computer vision.