

# A New Hardware-Efficient Algorithm and Reconfigurable Architecture for Image Contrast Enhancement

Shih-Chia Huang, *Member, IEEE*, and Wen-Chieh Chen

**Abstract**—Contrast enhancement is crucial when generating high quality images for image processing applications, such as digital image or video photography, liquid crystal display processing, and medical image analysis. In order to achieve real-time performance for high-definition video applications, it is necessary to design efficient contrast enhancement hardware architecture to meet the needs of real-time processing. In this paper, we propose a novel hardware-oriented contrast enhancement algorithm which can be implemented effectively for hardware design. In order to be considered for hardware implementation, approximation techniques are proposed to reduce these complex computations during performance of the contrast enhancement algorithm. The proposed hardware-oriented contrast enhancement algorithm achieves good image quality by measuring the results of qualitative and quantitative analyzes. To decrease hardware cost and improve hardware utilization for real-time performance, a reduction in circuit area is proposed through use of parameter-controlled reconfigurable architecture. The experiment results show that the proposed hardware-oriented contrast enhancement algorithm can provide an average frame rate of 48.23 frames/s at high definition resolution  $1920 \times 1080$ .

**Index Terms**—Image contrast enhancement, reconfigurable architecture.

## I. INTRODUCTION

CONTRAST enhancement plays an important role in the improvement of visual quality for processing of computer vision, such as video surveillance [1]–[3], pattern matching [4]–[6], face detection [7], face recognition [8], [9] and face annotation [10]. This is because several conditions may lead to poor contrast in digital video or images, including lack of operator expertise and inadequacy of the image capture device [11]. Essentially, if the overall luminance is insufficient, then the details of the image or video features will be obscured. Contrast enhancement is one of the most effective techniques by which to improve the perceptual quality of images presented on liquid crystal displays (LCDs) under dim backlight [12]–[16]. In previous studies, many enhancement techniques



Fig. 1. Backlight dimming equipment of a 42-inch LCD.

for dimmed images have been developed to eliminate these problems [17]–[21]. We have proposed an efficient contrast enhancement approach based on the gamma correction and probability density of the luminance pixel to improve the brightness of dimmed images in [22]–[24]. This approach improves the brightness of dimmed images and produces enhanced images of higher quality than previous state-of-the-art methods. The proposed contrast enhancement algorithm [22], [23] has also been evaluated and applied to the backlight dimming equipment in liquid crystal displays successfully, as shown in Fig. 1. Due to real-time requirements for high-definition video applications, it is necessary to accelerate the proposed contrast enhancement algorithm [22], [23] by hardware implementations. Therefore, the proposed contrast enhancement algorithm [22], [23] needs to be modified for use in association with a hardware-oriented contrast enhancement algorithm, whereupon efficient contrast enhancement architecture is designed to meet the requirements for real-time processing. Considering the requirements for hardware implementation, there exist three potential problems in the proposed algorithm. They are defined below.

- 1) *High Computational Complexity*: There are many highly complex mathematical operations in the proposed contrast enhancement algorithm [22], [23] which may make it difficult to implement in hardware. This is especially apparent in variable exponentiation with non-integer computation. In response, we propose the use of approximation techniques in the hardware-oriented

Manuscript received February 6, 2014; revised May 26, 2014; accepted July 30, 2014. Date of publication August 18, 2014; date of current version September 5, 2014. This work was supported by the National Science Council of Taiwan under Grant NSC-100-2628-E-027-012-MY3 and Grant NSC-102-2221-E-027-065. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Quanzheng Li.

The authors are with the Department of Electronic Engineering, National Taipei University of Technology, Taipei 106, Taiwan (e-mail: schuang@ntut.edu.tw; jay2070@gmail.com).

Color versions of one or more of the figures in this paper are available online at <http://ieeexplore.ieee.org>.

Digital Object Identifier 10.1109/TIP.2014.2348869

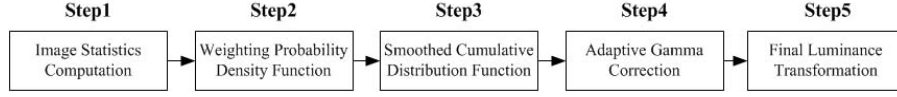


Fig. 2. The flow chart of the proposed contrast enhancement method.

contrast enhancement algorithm by which to solve these computational problems.

- 2) *Real-Time Requirement*: The implementation of the proposed algorithm must meet real-time constraints. We propose efficient hardware architecture to achieve the best throughputs. Each module of the proposed hardware-oriented contrast enhancement algorithm can be processed in parallel.
- 3) *Hardware Utilization*: In order to achieve optimal cost efficiency, it is important to achieve a compromise between efficient hardware utilization and real-time constraint. Therefore, we propose parameter-controlled reconfigurable architecture to decrease the hardware cost and improve hardware utilization.

According to the experimental results, the proposed architecture and chip design can process an average frame rate of up to 48.23 fps at high definition resolution  $1920 \times 1080$ . The proposed hardware-oriented contrast enhancement algorithm can provide an approximate solution for the original algorithm and achieve good image quality by measuring the results of qualitative and quantitative analyses. The rest of this paper is organized as follows: Section II provides a brief discussion of prior work. Section III details our proposed hardware-oriented contrast enhancement algorithm and hardware consideration. Section IV describes the proposed hardware architecture. In Section V, the efficacy of the proposed method is supported through comparison of its experimental results with those obtained through existing methods; the implementation results and comparisons are summarized. Finally, our concluding remarks are presented in Section VI.

## II. PRIOR WORK

The flowchart of the proposed contrast enhancement algorithm [22], [23] is shown in Fig. 2. Our approach achieves a good balance between contrast enhancement and feature preservation, and involves five important proposed modules: an image statistics computation (ISC) module, a weighting probability density function (WPDF) module, a smoothed cumulative distribution function (SCDF) module, an adaptive gamma correction (AGC) module, and a final luminance transformation (FLT) module. The core of the proposed contrast enhancement algorithm is described as follows:

- Step 1) *Image Statistics Computation*: Suppose that  $X = \{X(i, j)\}$  denotes an incoming 2D image composed of  $l$  discrete gray levels in the range  $[l_{min}, l_{max}]$ , where  $l_{max}$  is the maximum luminance of the incoming image, and  $l_{min}$  is the minimum luminance of the incoming image. In other words,  $X(i, j)$  represents the intensity of the incoming image at the location  $(i, j)$  and  $X(i, j) \in \{l_{min}, l_{min}+1, l_{min}+2, \dots, l_{max}\}$ .

For the incoming image  $X$  with  $l$  discrete gray levels, the probability density function (PDF) is defined as follows:

$$PDF(l) = \frac{n_l}{MN} \quad (1)$$

where  $l = l_{min}, l_{min}+1, l_{min}+2, \dots, l_{max}$ ,  $n_l$  denotes the number of pixels for luminance  $l$ , and  $MN$  denotes the total number of pixels in the incoming image.

- Step 2) *Weighting Probability Density Function*: The weighting probability density function can be expressed as follows:

$$PDF_w(l) = \max(PDF) \times \left( \frac{PDF(l) - \min(PDF)}{\max(PDF) - \min(PDF)} \right)^\alpha \quad (2)$$

where  $l = l_{min}, l_{min}+1, l_{min}+2, \dots, l_{max}$ ,  $PDF_w(l)$  represents the weighting probability density function,  $\max(PDF)$  denotes the maximum probability density of  $PDF(l)$ ,  $\min(PDF)$  denotes the minimum probability density of  $PDF(l)$ , and  $\alpha$  represents the adaptive parameter that can be empirically set to 0.5. Here, the range  $[0.1, 0.5]$  is determined experimentally.

- Step 3) *Smoothed Cumulative Distribution Function*: The original cumulative distribution function (CDF) is smoothed and can be expressed by using the  $PDF_w(l)$  as follows:

$$CDF_s(l) = \sum_{l=l_{min}}^{l_{max}} \frac{PDF_w(l)}{\sum PDF_w} \quad (3)$$

for  $l = l_{min}, l_{min}+1, l_{min}+2, \dots, l_{max}$ ,  $\sum PDF_w$  represents the sum of the weighting probabilities, and  $CDF_s$  represents the smoothed CDF.

- Step 4) *Adaptive Gamma Correction*: By using the gamma correction, the transform function can be calculated as follows:

$$T(l) = (l_{max} - l_{min}) \times \left( \frac{l}{l_{max} - l_{min}} \right)^\gamma \quad (4)$$

where  $l = l_{min}, l_{min}+1, l_{min}+2, \dots, l_{max}$ , with  $T(l)$  representing the transform function;  $\gamma = 1 - CDF_s(l) \times P$  with  $P$  represents the adaptive parameter that can be empirically set to 1. Here, the range  $[0.5, 1]$  is determined experimentally.

- Step 5) *Final Luminance Transformation*: The output image of the proposed algorithm  $Y = \{Y(i, j)\}$  can be expressed as:

$$Y = \{T(X(i, j)) | \forall X(i, j) \in X\} \quad (5)$$

where  $X(i, j)$  represents the intensity of the incoming image at the location  $(i, j)$  and  $Y(i, j)$  represents the intensity of the output image at the location  $(i, j)$ .

### III. PROPOSED HARDWARE-ORIENTED CONTRAST ENHANCEMENT ALGORITHM AND HARDWARE CONSIDERATION

The afore-mentioned algorithm has an important advantage - it offers a good balance between defense against the creation of noise artifacts and preservation of features. Consequently, this proposed software-oriented contrast enhancement method is very attractive for hardware architecture implementation [22], [23].

In order to be considered for hardware implementation, the proposed algorithm [22], [23] needs to be modified for the development of an efficient procedure which encompasses both contrast enhancement and the hardware-oriented algorithm. Therefore, a novel hardware-efficient algorithm is proposed by which to increase the processing speed while reducing the hardware cost. Implementation of the original variable exponentiation with a non-integer base computation in (2) and (4), along with division computation in (3), requires that the contrast enhancement algorithm have very high computational complexity, as well as high hardware cost. Thus, in order to achieve high image quality and throughput, approximation techniques are proposed to reduce these complex computations during performance of the contrast enhancement algorithm. To reduce the computational complexity of the variable exponentiation with non-integer base operations in (2) and (4), an approximation method is proposed for hardware consideration. The general formula can be expressed as follows:

$$y = m \times \left(\frac{x}{n}\right)^\gamma \quad (6)$$

This can be rewritten as:

$$y = m \times 2^{\log_2 \left(\frac{x}{n}\right)^\gamma} \quad (7)$$

then

$$y = m \times 2^{\gamma \left(\log_2 \frac{x}{n}\right)} \quad (8)$$

Let us rewrite (8) in regard to its exponent:

$$y = m \times 2^{\gamma (\log_2 x - \log_2 n)} \quad (9)$$

As can be observed in (9), the variable exponentiation with a non-integer base computation is simplified to a variable exponentiation with base two and computation of log base two. Both of these computations can be implemented effectively for hardware design. Although the use of different bases leads to the same result, we adopted base two because of hardware considerations. In order to reduce the computational complexity of division operations in (3), we employed an approximation method instead of the original division technique. Both  $m$  and  $\gamma$  are set to 1 in (6); a general formula can be expressed as follow:

$$y = \frac{x}{n} \quad (10)$$

The division operation also makes use of a procedure similar to the variable exponentiation with a non-integer base operation in (6). The formula can be expressed as follows:

$$y = 2^{\log_2 \left(\frac{x}{n}\right)} \quad (11)$$

Let us rewrite (11) in regard to its exponent:

$$y = 2^{(\log_2 x - \log_2 n)} \quad (12)$$

Similarly to the hardware consideration for (9), the division computation is simplified to a variable exponentiation with base two, along with computation of log base two.

1) *Hardware Consideration of  $\log_2 k$* : The general computation of  $\log_2 k$  is limited to a range:

$$k \in [1, 2^{p+1}) \quad (13)$$

$$\log_2 k = p + \log_2 h \quad (14)$$

where  $p$  and  $h$  are real positive numbers. We can thereby find:

$$p = \text{floor}(\log_2 k) \quad (15)$$

and

$$k = h2^p \quad (16)$$

$$h = \frac{k}{2^p}, h \in [1, 2) \quad (17)$$

where  $p$  can be computed by finding the largest nonzero bit in  $k$ , and  $h$  is determined by the right-shifting of  $k$ .

To achieve low computational complexity, we employ the Newton's divided difference interpolation formula that approximates the function over the desired interval.

Let

$$\pi_n(x) = \prod_{k=0}^n (x - x_k) \quad (18)$$

then

$$f(x) = f_0 + \sum_{k=1}^n \pi_{k-1}(x)[x_0, x_1, \dots, x_k] + R_n \quad (19)$$

and the remainder is:

$$R_n(x) = \pi_n(x)[x_0, x_1, \dots, x_n, x] = \pi_n(x) \frac{f^{(n+1)}(\zeta)}{(n+1)!} \quad (20)$$

for  $x_0 < \zeta < x_n$ .

From Newton's divided difference interpolation formula, we can find the approximation function of  $\log_2 h$  as:

$$\log_2 h \simeq [(0.1519h - 1.02123)h + 3]h - 2.13, h \in [1, 2) \quad (21)$$

As the result of (21), we replace the  $\log_2 h$  calculation for an approximation of its range, thereby reducing computational complexity and promoting efficiency.

2) *Hardware Consideration of  $2^k$* : The computation of  $2^k$  is limited to a range as follows:

$$2^k, k \in [p, 0) \quad (22)$$

We separate  $k$  into  $p$  and  $h$

$$k = p + h \quad (23)$$

and

$$h = k - p, h \in [0, 1) \quad (24)$$

where  $p$  is a positive real number.

Then

$$2^k = 2^p \times 2^h \quad (25)$$

and

$$p = \text{floor}(k) \quad (26)$$

From Newton's divided difference interpolation formula, we can find the approximation function of  $2^h$  as:

$$2^h \simeq [(0.079h + 0.2242)h + 0.6967]h + 0.999, h \in [0, 1) \quad (27)$$

Experimental results show that the absolute error of the approximation equation is close to the correct value, and it can be implemented easily by performing simple operations of addition and multiplication.

After modification of the proposed algorithm for hardware consideration, we propose implementation of a novel and effective hardware-oriented contrast enhancement algorithm with which to examine hardware implementation at the algorithm level and reduce computational complexity. The core of the proposed hardware-oriented contrast enhancement algorithm is described as follows:

Step 1) *Image Statistics Computation*: To reduce the complexity of the division computation, we can rewrite equation (1) as:

$$PDF'(l) = n_l \quad (28)$$

where  $PDF'(l) = PDF(l) \times MN$ ,  $l = l_{min}, l_{min} + 1, l_{min} + 2, \dots, l_{max}$ , and  $n_l$  represents the number of times that level  $l$  appears in the incoming image.

Step 2) *Weighting Probability Density Function*: Let us rewrite equation (2) in terms of equation (28):

$$PDF'_w(l) = \max(PDF') \times \left( \frac{PDF'(l) - \min(PDF')}{\max(PDF') - \min(PDF')} \right)^\alpha \quad (29)$$

where  $l = l_{min}, l_{min} + 1, l_{min} + 2, \dots, l_{max}$ , and  $\alpha$  represents the adaptive parameter that can be empirically set to 0.5. Here, the range [0.1, 0.5] is determined experimentally. Using equation (9), we can rewrite equation (29) as the hardware computation form:

$$PDF'_w(l) = \max(PDF') \times 2^\beta \quad (30)$$

where  $\beta = \alpha[\log_2(PDF'(l) - \min(PDF')) - \log_2(\max(PDF') - \min(PDF'))]$ ;  $l = l_{min}, l_{min} + 1,$

$l_{min} + 2, \dots, l_{max}$ , and  $\alpha$  represents the adaptive parameter that can be empirically set to 0.5. Here, the range [0.1, 0.5] is determined experimentally.

Step 3) *Smoothed Cumulative Distribution Function*: Let us rewrite equation (3) in terms of equation (30):

$$CDF'_s(l) = \frac{\sum_{l=l_{min}}^{l_{max}} PDF'_w(l)}{\sum PDF'_w} \quad (31)$$

where  $l = l_{min}, l_{min} + 1, l_{min} + 2, \dots, l_{max}$ .  $\sum PDF'_w$  represents the sum of the weighting probabilities, and  $CDF'_s(l)$  represents the smoothed  $CDF'$ . Using equation (12), we can rewrite equation (31) as the hardware computation form:

$$CDF'_s(l) = 2^{(\log_2(\sum_{l=l_{min}}^{l_{max}} PDF'_w(l)) - \log_2(\sum PDF'_w))} \quad (32)$$

where  $l = l_{min}, l_{min} + 1, l_{min} + 2, \dots, l_{max}$ ,  $\sum PDF'_w$  represents the sum of the weighting probabilities, and  $CDF'_s(l)$  represents the smoothed  $CDF'$ .

Step 4) *Adaptive Gamma Correction*: Let us rewrite equation (4) from equation (9) in terms of hardware consideration:

$$T(l) = (l_{max} - l_{min}) \times 2^{\gamma(\log_2 l - \log_2(l_{max} - l_{min}))} \quad (33)$$

where  $l = l_{min}, l_{min} + 1, l_{min} + 2, \dots, l_{max}$ , with  $T(l)$  represents the transform function;  $\gamma = 1 - CDF'_s(l) \times P$  with  $P$  represents the adaptive parameter that can be empirically set to 1. Here, the range [0.5, 1] is determined experimentally.

Step 5) *Final Luminance Transformation*: After the simplification of steps 1 through 4, we can get the final enhanced image via equation (5). The output image of the proposed hardware-oriented algorithm  $Y = \{Y(i, j)\}$  can be expressed as:

$$Y = \{T(X(i, j)) | \forall X(i, j) \in X\} \quad (34)$$

where  $X(i, j)$  represents the intensity of the incoming image at the location  $(i, j)$  and  $Y(i, j)$  represents the intensity of the output image at the location  $(i, j)$ .

#### IV. HARDWARE ARCHITECTURE DESIGN

In the following paper, we present a new hardware-oriented contrast enhancement algorithm which is highly amenable to efficient hardware acceleration. We propose new hardware architecture based on our hardware-oriented contrast enhancement algorithm. Our architecture is more efficient for low cost hardware implementation and real-time performance for high-definition video applications. In order to improve the throughput of the proposed hardware for real-time applications, the proposed hardware-oriented algorithm is dedicatedly pipelined in several stages. In order for the proposed hardware-oriented contrast enhancement algorithm to achieve the best throughputs, the architecture of our hardware-oriented algorithm can be divided into five stages, as shown in Fig. 3(a). Conversely, these five independent modules of the proposed hardware-oriented contrast enhancement algorithm can be processed in parallel. However, the original pipeline design will cause several problems. These problems are stated as follows:

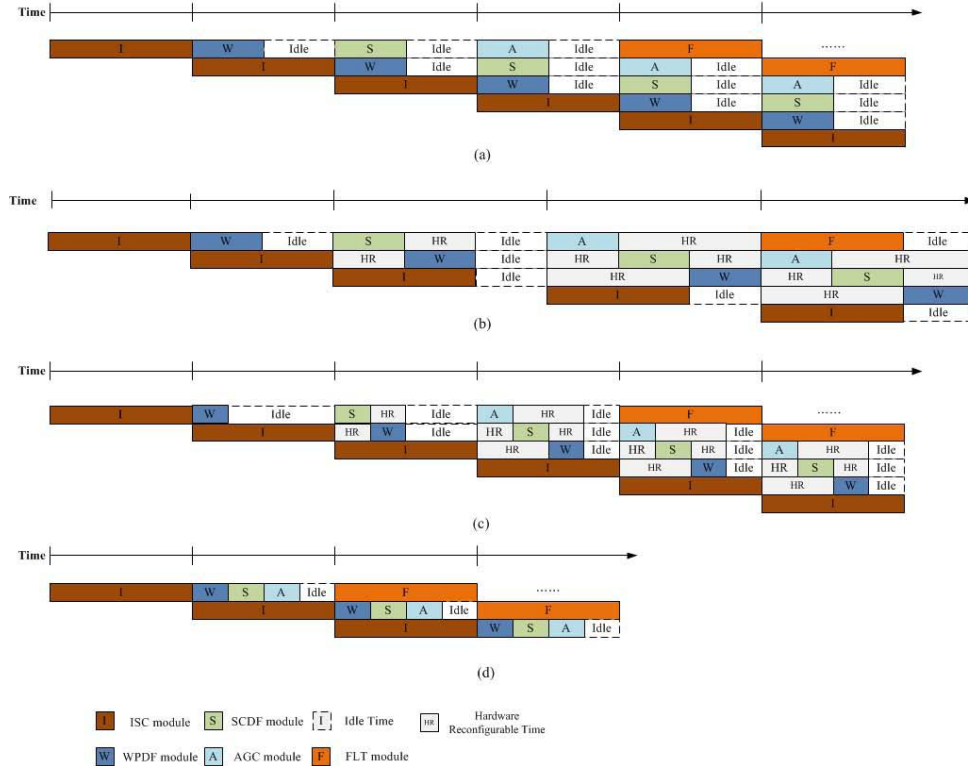


Fig. 3. Pipeline stages of the proposed method in: (a) Original five-stage pipeline architecture design. (b) After the proposed reconfigurable hardware design. (c) After the proposed half histogram design. (d) After the proposed three-stage pipeline architecture design.

**Problem 1)** The processing time of the ISC module is the throughput bottleneck of our system and makes the WPDF, SCDF, and AGC modules idle during most of the processing cycles. However, by adopting the proposed parameter-controlled reconfigurable architecture, the WPDF, SCDF, and AGC modules can be united into a new module, thus improving hardware utilization. This can be seen in Fig. 3(b).

**Problem 2)** Due to the parameter-controlled reconfigurable design of the proposed system, the combination of the WPDF, SCDF, and AGC modules during the reconfigurable pipeline stage has a tendency to cause a major throughput bottleneck. Hence, we propose the half-histogram (HH) method. A subsample of the histogram not only has the capacity to reduce processing time and stage registers, but also maintains good image quality. This can be seen in Fig. 3(c). Once these problems have been solved through hardware consideration, real-time performance can be achieved with efficient hardware utilization. The initial five stages are then consolidated into three stages, as shown in Fig. 3(d).

The functionality of each module and the corresponding hardware architecture will be introduced in the following subsections.

#### A. Implementation of the ISC Module

The functionality of the ISC module is to generate the histogram of the current frame and calculate the  $\max(PDF')$  and  $\min(PDF')$  simultaneously. The histogram data corresponding to the luminance value of each input pixel is cumulated in the cumulating unit. Meanwhile, the minimum

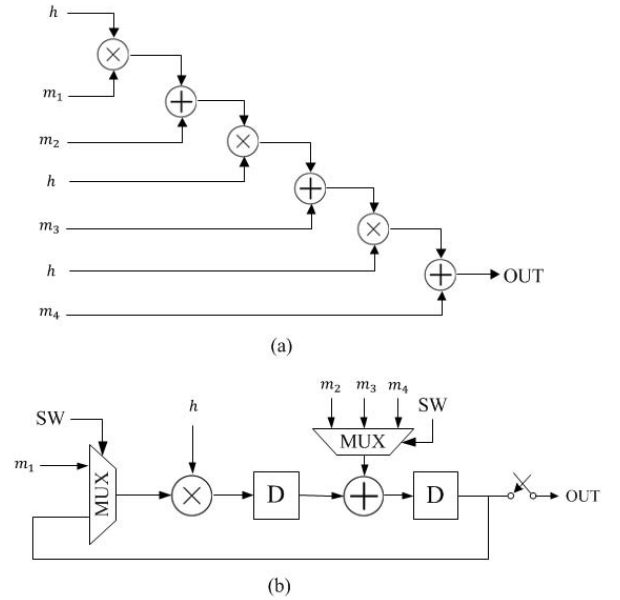


Fig. 4. The approximation function of the  $\log_2 h$  and  $2^h$  modules of (a) the traditional architecture design, and (b) the folding architecture design.

and maximum  $PDF'$  values of the input frame are calculated and held in the registers. Subsequently, the data in the register is converted to floating-point number format on a continuous basis for the next stage of processing.

#### B. Implementation of the WPDF, SCDF, and AGC Modules

1) *Implementation of the  $\log_2 k$  Module and  $2^k$  Module:* According to the results in (21) and (27), we can observe

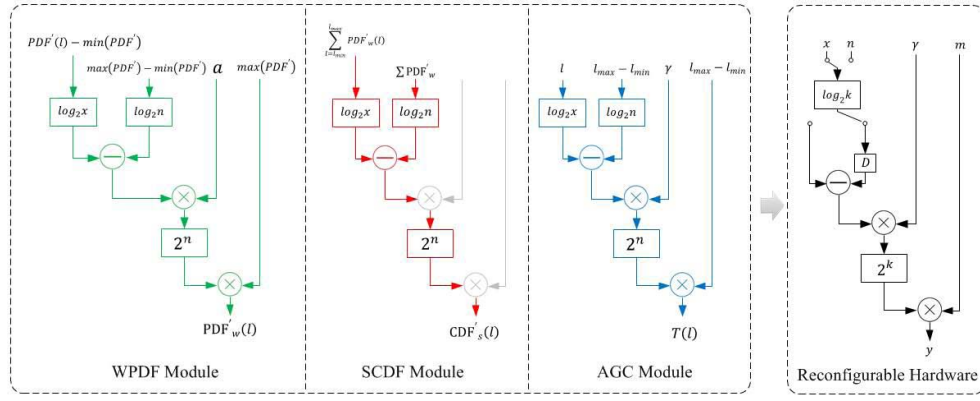


Fig. 5. Hardware reduction and reuse by the proposed parameter-controlled reconfigurable architecture.

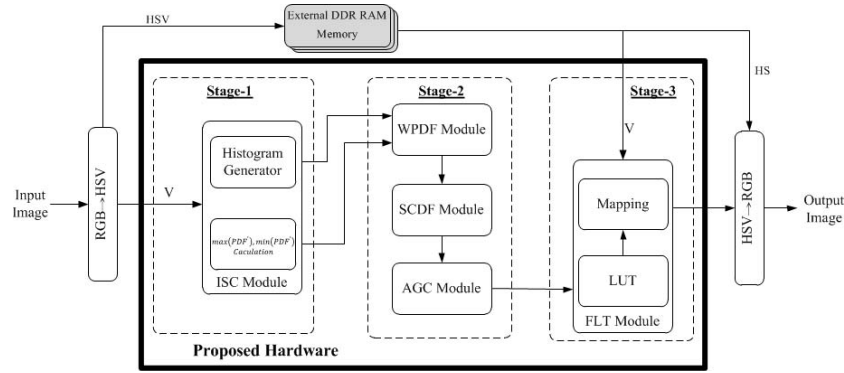


Fig. 6. Architecture of the proposed hardware implementation.

that the proposed approximation function of  $\log_2 h$  and  $2^h$  can be calculated by using the same hardware architecture but with different input parameters. Hardware implementation of the proposed approximation function of the  $\log_2 h$  and  $2^h$  modules is determined in the usual manner as shown in Fig. 4(a). To decrease hardware cost and improve hardware utilization, a reduction in circuit area is proposed through use of folding architecture. The proposed folding architecture of the approximation function of the  $\log_2 h$  and  $2^h$  modules is shown in Fig. 4(b). With use of the parameter-controlled folding architecture, the hardware implementation of the proposed approximation function of the  $\log_2 h$  and  $2^h$  modules features a reduction in the circuit size yet results in the same throughput as that of traditional design.

2) *Implementation of Reconfigurable Structure*: The respective architectures of the WPDF, SCDF, and AGC modules are shown in the left part of Fig. 5. According to the pipeline stages diagram in Fig. 3(a), we can see that the WPDF, SCDF, and AGC modules idle in most of the processing cycles. In regard to hardware implementation, the WPDF, SCDF, and AGC modules of proposed hardware-oriented algorithm exhibit similarity. To further decrease hardware cost and improve hardware utilization, we propose a parameter-controlled reconfigurable scheme. With this scheme, the WPDF, SCDF, and AGC modules can be united into a new module which will reduce the idle time of these modules and thus improve hardware utilization. From the

parameter-controlled reconfigurable structural design, the individual parameters of the WPDF, SCDF, and AGC modules can be fed as input vectors to the same reconfigurable hardware architecture. The right part of Fig. 5 illustrates the parameter-controlled reconfigurable architecture which supports the three modules of the proposed hardware-oriented algorithm. With the reconfigurable architecture, the implementation of the proposed hardware-oriented algorithm has the capacity to result not only in dramatically reduced hardware cost but also in high throughput for real-time application.

### C. Implementation of Half Histogram Design

However, if the WPDF, SCDF, and AGC modules are combined in a single pipeline stage, this reconfigurable stage may become a throughput bottleneck for the system. Hence, we propose the half-histogram scheme. Using this subsample of the histogram not only can reduce processing time and stage registers, but also maintain high image quality. Division by powers of two can be expressed as a right shift in the hardware implementation. For the hardware implementation of the proposed half-histogram scheme, each input luminance value needs to be right-shifted one bit, which means subsampling half of the gray level distribution of the histogram. A decline in the gray level distribution of the histogram results in a subsequent fifty percent reduction in the processing time of the WPDF, SCDF, and AGC modules.



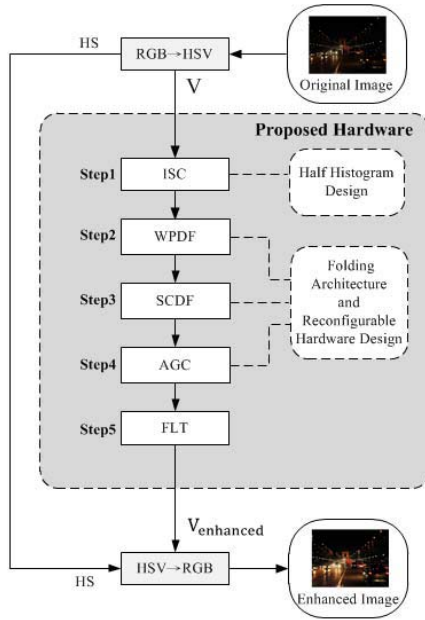


Fig. 7. The overview of the proposed contrast enhancement hardware.

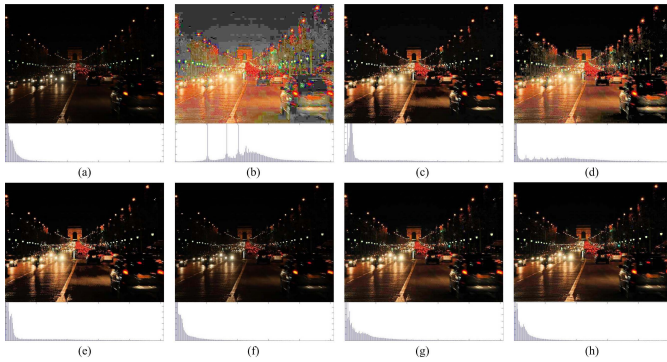


Fig. 8. The "Road" image: (a) is the original image with the statistical histogram; the remaining seven images are the enhancement results with modified histograms by the (b) THE, (c) BBHE, (d) DSIHE, (e) RSIHE, (f) RSWHE, (g) proposed software, and (h) proposed hardware.

#### D. Implementation of FLT Module

Because the final mapping function operations tend to be computationally intensive, a Look-Up-Table (LUT) can be utilized after computation of the AGC module. The original luminance values then can be fed into the mapping unit. The output is the corresponding final enhanced pixel, which is dependent on the LUT. Each input luminance value needs to be right-shifted one bit to the mapping unit in order to suit the half-histogram design. The application of the final luminance transformation at every pixel of the frame requires that each output luminance value be left-shifted one bit, along with the addition of one bit 0 at LSB. An overview of the architecture of the proposed contrast enhancement hardware design is illustrated in Fig. 6. An overview of the architecture of the proposed contrast enhancement hardware is shown in Fig. 7.

### V. EXPERIMENTAL RESULTS

This section presents the results of our five-part analysis. The first part consists of an image quality comparison of

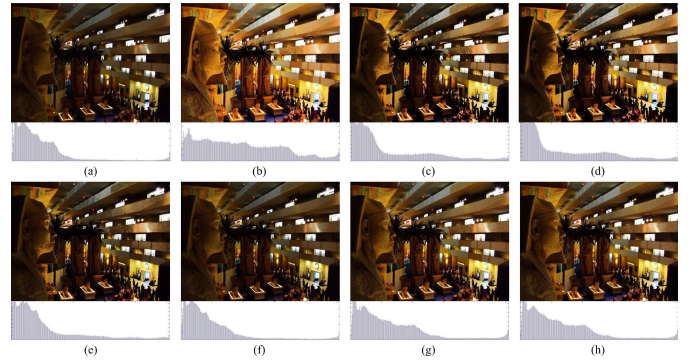


Fig. 9. The "Indoor" image: (a) is the original image with the statistical histogram; the remaining seven images are the enhancement results with modified histograms by the (b) THE, (c) BBHE, (d) DSIHE, (e) RSIHE, (f) RSWHE, (g) proposed software, and (h) proposed hardware.

the contrast enhancement algorithms of the seven different methods, including the proposed method. Various image patterns were used during the testing process. The second part details the ASIC implementation, the specification results of which are summarized in Table III. The third part presents the analysis of hardware utilization. In this portion, the critical path and costs of the proposed hardware are examined. In the fourth part, the analysis of the reduction in processing time is presented. The processing time of the proposed method is examined in regard to software and hardware implementation. The proposed contrast enhancement algorithm also has been evaluated and applied to the backlight dimming equipment in the final part.

#### A. Image Quality Comparison

1) *Visual Assessment*: In our previous study [22], [23], an efficient contrast enhancement algorithm was proposed by which to produce enhanced images of higher quality than those produced using previous state-of-the-art methods. Initially, we performed visual assessments of the results of each of the seven methods. The seven methods include Traditional Histogram Equalization (THE) [17], Brightness Preserving Bi-Histogram Equalization (BBHE) [18], Dualistic Sub-Image Histogram Equalization (DSIHE) [19], Recursive Sub-Image Histogram Equalization (RSIHE) [20], Recursively Separated and Weighted Histogram Equalization (RSWHE) [21], our proposed contrast enhancement algorithm [22], [23], and our proposed hardware-oriented contrast enhancement algorithm. All methods were applied to enhance various color images. We demonstrate the experimental results using seven image contrast enhancement algorithms in dark (test images: Road, Indoor), middle (test images: Sea, Dock), and bright (test images: River, Park) grayscale images. The summarized results are shown in Fig. 8, Fig. 9, Fig. 10, Fig. 11, Fig. 12, and Fig. 13, respectively. The hardware computation results of the proposed hardware-oriented contrast enhancement algorithm were obtained via FPGA testing and Verilog HDL simulation.

Fig. 8 shows the original Road dark image and seven enhancement results produced by THE, BBHE, DSIHE,

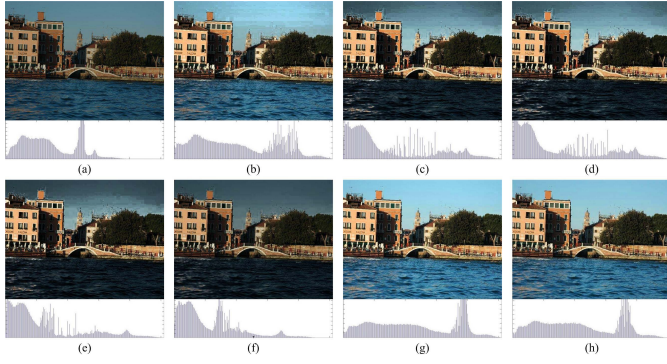


Fig. 10. The “Sea” image: (a) is the original image with the statistical histogram; the remaining seven images are the enhancement results with modified histograms by the (b) THE, (c) BBHE, (d) DSIHE, (e) RSIHE, (f) RSWHE, (g) proposed software, and (h) proposed hardware.

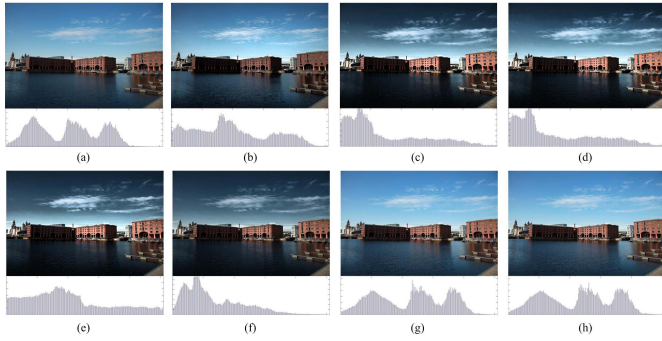


Fig. 11. The “Dock” image: (a) is the original image with the statistical histogram; the remaining seven images are the enhancement results with modified histograms by the (b) THE, (c) BBHE, (d) DSIHE, (e) RSIHE, (f) RSWHE, (g) proposed software, and (h) proposed hardware.

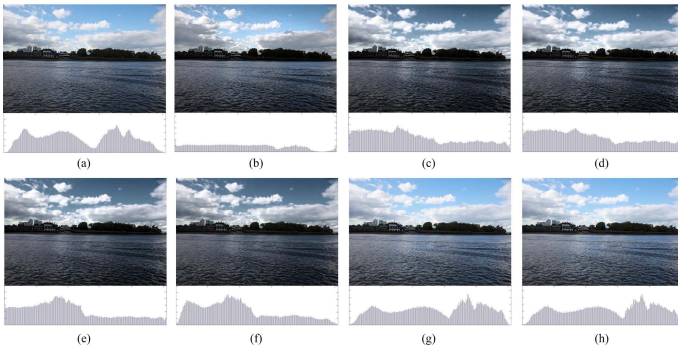


Fig. 12. The “River” image: (a) is the original image with the statistical histogram; the remaining seven images are the enhancement results with modified histograms by the (b) THE, (c) BBHE, (d) DSIHE, (e) RSIHE, (f) RSWHE, (g) proposed software, and (h) proposed hardware.

RSIHE, RSWHE, our proposed contrast enhancement algorithm [22], [23], and our proposed hardware-oriented contrast enhancement algorithm methods, respectively. The various contrast enhancement methods [17]–[21] all improve the dark region with the exception of the RSWHE method, which preserves the low brightness as a weak enhancement. Overall, the two proposed methods produce the most optimal balance between image intensity preservation and contrast enhancement.

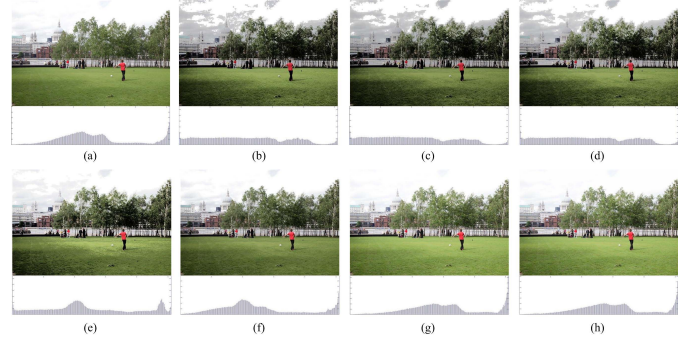


Fig. 13. The “Park” image: (a) is the original image with the statistical histogram; the remaining seven images are the enhancement results with modified histograms by the (b) THE, (c) BBHE, (d) DSIHE, (e) RSIHE, (f) RSWHE, (g) proposed software, and (h) proposed hardware.

TABLE I

THE PSNR MEASUREMENTS OF THE SIMILARITIES BETWEEN THE PROPOSED SOFTWARE AND PROPOSED HARDWARE

Test images	PSNR
Road	44.86
Indoor	47.01
Sea	41.04
Dock	43.37
River	45.83
Park	44.25
Average	44.39

Fig. 9 presents the Indoor dark image and enhancement results with statistical histograms produced by THE, BBHE, DSIHE, RSIHE, RSWHE, our proposed contrast enhancement algorithm, and our proposed hardware-oriented contrast enhancement algorithm methods, respectively. As shown in Fig. 9(b), the THE method produces an image with unnatural contrast due to its non-uniform histogram. Since most of the gray levels are less than the median, neither the BBHE method nor the DSIHE method produces a good contrast image result as indicated in Fig. 9(c) and Fig. 9(d), respectively. The reason why the two methods fail is that the low-level of their histograms has large probability density equalized similarly to those of the THE method. The RSIHE method improves the DSIHE method via multi-equalizations.

Figs. 10 and 11 display the enhanced images for the Sea, and Dock middle grayscale images, which are captured during overcast conditions, along with the seven enhanced versions generated by each method. The THE, BBHE, DSIHE, RSIHE, and RSWHE methods cannot uniformly equalize the statistical histogram, and thus produce serious artifacts which appear in the sky regions of the enhanced versions. Conversely, both the proposed methods improve the luminance which is distributed from lowest to highest intensity.

Figs. 12–13 show the enhanced images using different methods for the River, and Park bright images, respectively. Some adverse effects were generated by the THE, BBHE, DSIHE, RSIHE, and RSWHE methods due to non-uniform equalizations. For example, the color of the sky region is distorted after performing these methods. Compared with the other methods, the proposed method produces an acceptable image without unnatural or insufficient contrast.



TABLE II  
EDGE LOSS RATE (%)

Test images	Average intensity	THE	BBHE	DSIHE	RSIHE	RSWHE	Proposed software	Proposed hardware
Road	13	59.38	50.15	55.98	47.36	45.98	34.26	36.31
Indoor	45	48.24	48.95	41.36	38.73	39.35	22.12	23.68
Sea	75	55.56	36.59	45.561	46.81	36.75	25.56	28.51
Dock	101	52.12	38.98	43.14	44.58	41.12	29.13	31.57
River	122	53.21	43.98	44.56	40.13	39.84	28.34	30.35
Park	152	51.22	50.34	49.83	48.35	43.54	33.24	35.81
Average		53.28	44.93	46.73	44.32	41.09	28.77	31.03

When comparing the products of the contrast enhancement algorithms using the human visual system, the proposed contrast enhancement algorithm [22], [23] and proposed hardware-oriented contrast enhancement algorithm produce superior results.

2) *Comparison of the Results Between the Software and Hardware Implementations:* Contrary to software implementation, hardware implementation has limited hardware resources. Therefore, some precision reductions were performed in the hardware implementation. Hence, it is essential to examine the similarities between the software algorithm image and the hardware-oriented algorithm image. Table I lists the peak signal-to-noise ratio (PSNR) calculated between the enhanced results of proposed contrast enhancement and proposed hardware-oriented contrast enhancement algorithms. From Table I, we can see that the image quality remains almost the same before and after the hardware design; not only is the register reduced, but the quality of the enhanced image is retained as well. The proposed hardware-oriented contrast enhancement algorithm can provide an approximate solution to hardware implementation and efficiently reduce its computational complexity.

3) *Quantitative Evaluation:* Contrast enhancement is not easily measured by quantitative criteria. To judge the preservation of image details quantitatively, a measure of the edge loss rates was adopted [13] for the test cases for the seven algorithms. The edge loss rate  $\varepsilon E$  is defined as the ratio between the number of missed edge pixels  $\psi E_m$  and the number of original edge pixels  $\psi E_i$ .

$$\varepsilon E = \frac{\psi E_m}{\psi E_i}, \quad (35)$$

The lower edge loss rate values indicate greater preservation of image details. The qualitative results for the test images with the edge loss rates measurements are shown in Table II. These were generated by THE, BBHE, DSIHE, RSIHE, RSWHE, proposed contrast enhancement algorithm [22], [23], and proposed hardware-oriented contrast enhancement algorithm methods, respectively.

For the test image patterns, the edge loss rates produced by each method were 53.28%, 44.93%, 46.73%, 44.32%, 41.09%, 28.77%, and 31.03%, respectively. Generally, the proposed method attains the lowest edge loss rates when compared with other state-of-the-art methods. According to the edge loss rate results of the quantitative measurements, the proposed contrast enhancement algorithm achieves an average improvement of

TABLE III  
SPECIFICATION AND IMPLEMENTATION OF PROPOSED HARDWARE

	Specification	Design on FPGA
Technology	TSMC 0.18um	Xilinx Virtex5
Clock Rate	100 MHz	66 MHz
Gate Count	150.1 K Gate	11.5 K Slices
Memory	9216 Kb	35.2 K Slices
Image Resolution	1920 × 1080	1920 × 1080
Frame Rate	48.22 fps	31.82 fps

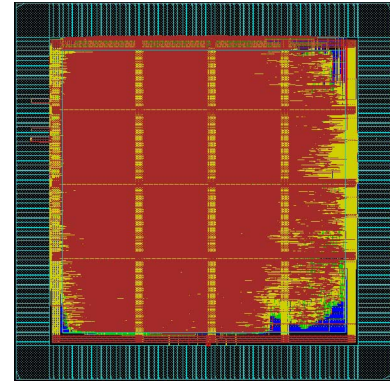


Fig. 14. Chip layout.

more than 20% over the other six previous algorithms. According to the edge loss rate results of the quantitative measurements, the proposed hardware-oriented contrast enhancement algorithm achieves an average improvement of more than 20% over the other five previous algorithms. The proposed hardware-oriented contrast enhancement algorithm adopts both approximation function and half histogram techniques, and produces good results as measured through qualitative and quantitative analysis.

### B. Specification and FPGA Implementation

Our design is implemented on Xilinx's Virtex-5 XC5V110T FPGA and ASIC using the design synthesis tool, Synopsys DesignVision, which is based on Verilog HDL with TSMC 0.18-um process technology for circuit synthesis. The specification and implementation details of the proposed efficient reconfigurable architecture of the hardware-oriented contrast enhancement algorithm and chip layout view are shown in Table III and Fig. 14, respectively. The clock frequency of proposed design is 100 MHz and processes 48.22 fps with

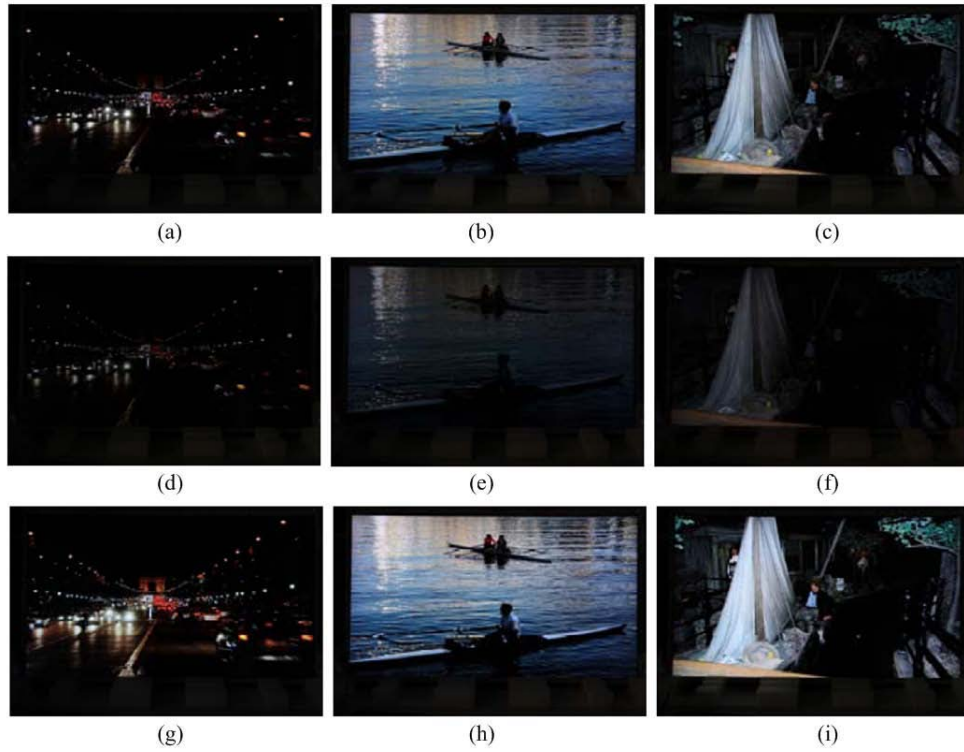


Fig. 15. Experimental results of contrast enhancement with backlight reduction in a 42-inch LCD. (a), (b), and (c) show three original images with full backlight; (d), (e), and (f) show three original images with half backlight modified by the aforementioned equipment, and (g), (h), and (i) show the enhancement of three original images with half backlight modified by the proposed method.

TABLE IV  
HARDWARE UTILIZATION (%)

Image Resolution	1280 × 720	1920 × 1080
without reconfigurable design	46.51	42.89
with reconfigurable design	77.52	71.49
Ratio	1.66	1.66

TABLE V  
PROPOSED TIME PER FRAME OF SOFTWARE AND HARDWARE

Image Resolution	1280 × 720	1920 × 1080
Software	51ms	88ms
Hardware	9.21ms	20.73ms
Ratio	5.53	4.24

high definition video resolution  $1920 \times 1080$  to provide real-time performance.

### C. Analysis of Hardware Utilization

The reconfigurable architecture is proposed for increasing the efficiency of hardware utilization and thus decreasing hardware cost. To examine the design quality, the hardware utilization rate (HUR) is used as a performance index. The HUR can be expressed as follows:

$$HUR = \frac{\sum_{i=1}^n \tau_i}{n \times \tau_{st}}, \quad (36)$$

where  $n$  is the number of stages of the pipeline hardware architecture,  $\tau_i$  is the processing time of each stage, and  $\tau_{st}$  is the slowest stage in the pipeline architecture. The idle time can be represented as  $n \times \tau_{st} - \sum_{i=1}^n \tau_i$ .

The HURs of the proposed hardware architecture with reconfigurable design and non-reconfigurable design are reported in Table IV. In regard to the video sequence with image size  $1280 \times 720$ , the hardware utilization rates of the

proposed hardware architecture with reconfigurable design and non-reconfigurable design are 46.51% and 77.52%, respectively. For the video sequence with image size  $1920 \times 1080$ , the hardware utilization rates of the proposed hardware architecture with reconfigurable design and non-reconfigurable design are 42.89% and 71.49%, respectively. It shows that the HUR of the proposed hardware architecture with reconfigurable design is 1.66 times that of non-reconfigurable design. The circuit size of the proposed parameter-controlled reconfigurable architecture can be reduced from 149920 gate counts to 53024 gate counts. Thus, the proposed module can save 64.6% in hardware implementation cost when compared to the cost of implementing hardware architecture with non-reconfigurable design. To the best of our knowledge, we are the first group to propose the use of highly efficient reconfigurable hardware in image contrast enhancement.

### D. Analysis of Processing Time Reduction

The proposed image contrast enhancement algorithm was implemented using C programming language on

an Intel Core i3 3.07 GHz processor with 4GB of RAM, running the Windows 7 operating system. Table V shows the performances and results of the implementations of the software and proposed hardware. The software implementation achieves an average frame rate of only 11.3 fps at high definition resolution  $1920 \times 1080$ . We have proven that the proposed hardware-oriented contrast enhancement algorithm can achieve an average frame rate of 48.23 fps at high definition resolution  $1920 \times 1080$ . In other words, the proposed hardware architecture provides real-time performance in high definition multimedia applications. The average ratio obtained for the processing time of the proposed hardware is 5.53 and 4.24 times faster than that of the software-oriented contrast enhancement algorithm for resolutions  $1280 \times 720$  and  $1920 \times 1080$ , respectively.

### E. LCD Monitor Demonstration

The proposed contrast enhancement algorithm also has been evaluated and applied to the backlight dimming equipment in a 42-inch LCD successfully, as shown in Fig. 1. This equipment can control the backlight source of the screen to reduce power consumption, yet it degrades the visual perceptibility presented to viewers. After applying the proposed method, a balance between power reduction and preservation of luminance is easily achieved. Fig. 15 shows the experimental results of contrast enhancement with LCD backlight reduction. Fig. 15(a), Fig. 15(b), and Fig. 15(c) show three original images with full backlight; Fig. 15(d), Fig. 15(e), and Fig. 15(f) show three original images with half backlight modified by the aforementioned equipment; Fig. 15(g), Fig. 15(h), and Fig. 15(i) show the enhancement of three original images with half backlight modified by the proposed method. From the results, we can fairly claim that the perceptual quality of images displayed on LCD under dim backlight can be improved by using the proposed method.

## VI. CONCLUSIONS

In this paper, we propose novel hardware architecture based on our hardware-oriented contrast enhancement algorithm. To the best of our knowledge, we are the first group to propose such highly efficient reconfigurable hardware for image contrast enhancement. The proposed hardware-oriented contrast enhancement algorithm can provide an approximate solution for the proposed software-oriented contrast enhancement algorithm, and achieves good image quality by measuring the results of qualitative and quantitative analyze. Experimental image enhancement results show that the proposed method performs well compared with other state-of-the-art methods. We employ parameter-controlled reconfigurable architecture which significantly improves hardware utilization. To demonstrate its performance, the proposed architecture and chip design can process an average frame rate of up to 48.23 fps at high definition resolution  $1920 \times 1080$ . According our analysis of time consumption, the proposed method can be achieved in a real-time video system.

## REFERENCES

- [1] S.-C. Huang, "An advanced motion detection algorithm with video quality analysis for video surveillance systems," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 21, no. 1, pp. 1–14, Jan. 2011.
- [2] S.-C. Huang and B.-H. Do, "Radial basis function based neural network for motion detection in dynamic scenes," *IEEE Trans. Cybern.*, vol. 44, no. 1, pp. 114–125, Jan. 2014.
- [3] S.-C. Huang and B.-H. Chen, "Highly accurate moving object detection in variable bit rate video-based traffic monitoring systems," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 24, no. 12, pp. 1920–1931, Dec. 2013.
- [4] J. Lindblad and N. Sladoje, "Linear time distances between fuzzy sets with applications to pattern matching and classification," *IEEE Trans. Image Process.*, vol. 23, no. 1, pp. 126–136, Jan. 2014.
- [5] W. Ouyang, R. Zhang, and W.-K. Cham, "Segmented gray-code kernels for fast pattern matching," *IEEE Trans. Image Process.*, vol. 22, no. 4, pp. 1512–1525, Apr. 2013.
- [6] Z.-M. Lu and Y. Shi, "Fast video shot boundary detection based on SVD and pattern matching," *IEEE Trans. Image Process.*, vol. 22, no. 12, pp. 5136–5145, Dec. 2013.
- [7] Y. Dupuis, X. Savatier, J.-Y. Ertaud, and P. Vasseur, "Robust radial face detection for omnidirectional vision," *IEEE Trans. Image Process.*, vol. 22, no. 5, pp. 1808–1821, May 2013.
- [8] X.-X. Li, D.-Q. Dai, X.-F. Zhang, and C.-X. Ren, "Structured sparse error coding for face recognition with occlusion," *IEEE Trans. Image Process.*, vol. 22, no. 5, pp. 1889–1900, May 2013.
- [9] P.-H. Lee, S.-W. Wu, and Y.-P. Hung, "Illumination compensation using oriented local histogram equalization and its application to face recognition," *IEEE Trans. Image Process.*, vol. 21, no. 9, pp. 4280–4289, Sep. 2012.
- [10] C.-A. Hsu, M.-K. Jiau, and S.-C. Huang, "An automatic face annotation system featuring high accuracy for online social networks," in *Proc. IEEE Int. Conf. Ubiquitous Intell. Comput.*, Dec. 2013, pp. 163–169.
- [11] T. Arici, S. Dikbas, and Y. Altunbasak, "A histogram modification framework and its application for image contrast enhancement," *IEEE Trans. Image Process.*, vol. 18, no. 9, pp. 1921–1935, Sep. 2009.
- [12] D. Cho, W.-S. Oh, and G.-W. Moon, "A novel adaptive dimming LED backlight system with current compensated X-Y channel drivers for LCD TVs," *J. Display Technol.*, vol. 7, no. 1, pp. 29–35, Jan. 2011.
- [13] P.-S. Tsai, C.-K. Liang, T.-H. Huang, and H. H. Chen, "Image enhancement for backlight-scaled TFT-LCD displays," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 19, no. 4, pp. 574–583, Apr. 2009.
- [14] F.-C. Lin *et al.*, "Dynamic backlight gamma on high dynamic range LCD TVs," *J. Display Technol.*, vol. 4, no. 2, pp. 139–146, Jun. 2008.
- [15] A. Iranli, W. Lee, and M. Pedram, "HVS-aware dynamic backlight scaling in TFT-LCDs," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 14, no. 10, pp. 1103–1116, Oct. 2006.
- [16] Y.-K. Lai, Y.-F. Lai, and P.-Y. Chen, "Content-based LCD backlight power reduction with image contrast enhancement using histogram analysis," *J. Display Technol.*, vol. 7, no. 10, pp. 550–555, Oct. 2011.
- [17] S. L. Dockstader and A. M. Tekalp, "Multiple camera tracking of interacting and occluded human motion," *Proc. IEEE*, vol. 89, no. 10, pp. 1441–1455, Oct. 2001.
- [18] Y.-T. Kim, "Contrast enhancement using brightness preserving bi-histogram equalization," *IEEE Trans. Consum. Electron.*, vol. 43, no. 1, pp. 1–8, Feb. 1997.
- [19] Y. Wan, Q. Chen, and B. Zhang, "Image enhancement based on equal area dualistic sub-image histogram equalization method," *IEEE Trans. Consum. Electron.*, vol. 45, no. 1, pp. 68–75, Feb. 1999.
- [20] K. S. Sim, C. P. Tso, and Y. Y. Tan, "Recursive sub-image histogram equalization applied to gray scale images," *Pattern Recognit. Lett.*, vol. 28, no. 10, pp. 1209–1221, 2007.
- [21] M. Kim and M. Chung, "Recursively separated and weighted histogram equalization for brightness preservation and contrast enhancement," *IEEE Trans. Consum. Electron.*, vol. 54, no. 3, pp. 1389–1397, Aug. 2008.
- [22] Y.-S. Chiu, F.-C. Cheng, and S.-C. Huang, "Efficient contrast enhancement using adaptive gamma correction and cumulative intensity distribution," in *Proc. IEEE Int. Conf. Syst., Man, Cybern.*, Oct. 2011, pp. 2946–2950.
- [23] S.-C. Huang, F.-C. Cheng, and Y.-S. Chiu, "Efficient contrast enhancement using adaptive gamma correction with weighting distribution," *IEEE Trans. Image Process.*, vol. 22, no. 3, pp. 1032–1041, Mar. 2013.
- [24] W.-C. Chen, S.-C. Huang, and T.-Y. Lee, "An efficient reconfigurable architecture design and implementation of image contrast enhancement algorithm," in *Proc. IEEE 9th Int. Conf. High Perform. Comput. Commun.*, Jun. 2012, pp. 1741–1747.



**Shih-Chia Huang** is currently an Associate Professor with the Department of Electronic Engineering, National Taipei University of Technology, Taipei, Taiwan, and an International Adjunct Professor with the Department of Business and Information Technology, University of Ontario Institute of Technology, Oshawa, ON, Canada. He has authored over 40 journal and conference papers, and holds over 30 patents in the U.S., Europe, Taiwan, and China. He received the Ph.D. degree in electrical engineering from National Taiwan University, Taipei, in 2009.

He was a recipient of the Kwoh-Ting Li Young Researcher Award by the Taipei Chapter of the Association for Computing Machinery, in 2011, and the Dr. Shechtman Young Researcher Award by the National Taipei University of Technology in 2012. He is an Associate Editor of the *Journal of Artificial Intelligence*. His research interests include image and video coding, wireless video transmission, video surveillance, error resilience and concealment techniques, digital signal processing, cloud computing, mobile applications and systems, embedded processor design, and embedded software and hardware codesign.



**Wen-Chieh Chen** was born in Taipei, Taiwan, in 1987. He received the B.S. and M.S. degrees from the Department of Electronic Engineering, National Taipei University of Technology, Taipei, in 2010 and 2012, respectively. His research interests include contrast enhancement, embedded software and hardware codesign, and digital image and video processing.