Power-Constrained Contrast Enhancement Algorithm Using Multiscale Retinex for OLED Display

Yeon-Oh Nam, Dong-Yoon Choi, and Byung Cheol Song, *Senior Member, IEEE*

***Abstract*— This paper presents a power-constrained contrast enhancement algorithm for organic light-emitting diode display based on multiscale retinex (MSR). In general, MSR, which is the key component of the proposed algorithm, consists of power controllable log operation and subbandwise gain control. First, we decompose an input image to MSRs of different sub-bands, and compute a proper gain for each MSR. Second, we apply a coarse-to-fine power control mechanism, which recomputes the MSRs and gains. This step iterates until the target power saving is accurately accomplished. With video sequences, the contrast levels of adjacent images are determined consistently using temporal coherence in order to avoid flickering artifacts. Finally, we present several optimization skills for real-time processing. Experimental results show that the proposed algorithm provides better visual quality than previous methods, and a consistent power-saving ratio without flickering artifacts, even for video sequences.**

***Index Terms*— Power consumption, contrast enhancement, OLED, multi-scale retinex.**

I. Introduction

MODERN display panels can be categorized into emissive and non-emissive displays. The cathode-ray tube (CRT), the plasma display panel (PDP) and the organic light-emitting diode (OLED) are representative emissive displays that do not require external light sources, whereas the thin-film transistor liquid crystal display (TFT-LCD) is non-emissive. In general, emissive displays have several advantages over non-emissive ones [1], [2]. First, since an emissive display can turn off individual pixels, it can express complete darkness and achieve a high contrast ratio. Second, emissive displays consume less power than non-emissive ones because each pixel in an emissive display can be independently driven and the power consumption of the pixel is proportional to its intensity level. Note that non-emissive displays should turn on their backlight regardless of pixel intensity.

Thus, the OLED is regarded as the most promising candidate for the next-generation display [3], [4], which will replace the TFT-LCD displays currently dominating the commercial market. Even though the OLED is mainly used for small panels in mobile devices, its mass-production technology is being developed rapidly. So large-size OLED panels may soon be adopted in a wider range of devices such as high-definition television (HDTV) and ultra HDTV [3].

Note that display modules consume most of the power in digital media devices [5]. So techniques to minimize power consumption in the display are inevitably required. Several image processing techniques for power saving in display panels have been proposed, beyond circuit-level power savings. Unfortunately, such techniques focus on reducing backlight intensity for TFT-LCDs while preserving the same level of perceived quality [6]–[10]. So they cannot be applied to power saving in emissive display devices such as OLED.

Lee et al. proposed a power-constrained contrast-enhancement algorithm (PCCE) for emissive displays based on histogram equalization (HE) [11], [12] They developed a power-consumption model for OLED displays and formulated an objective function that consists of the HE term and the power term. By minimizing the objective function based on the convex optimization theory, they tried to simultaneously achieve contrast enhancement and power savings. However, with HE-based contrast enhancement there is an inherent risk of overstretching. Also, since their method depends on certain parameters it cannot automatically, consistently and accurately provide the power-saving level desired by a user.

On the other hand, various contrast enhancement techniques have been developed for dozens of years [13]–[20]. As one of them retinex is a well-known non-linear enhancement method used for contrast enhancement as well as dynamic range compression. The retinex theory was proposed by Land and McCann [13], and Jobson et al. adapted their theory to single-scale retinex (SSR) [14] and multi-scale retinex (MSR) [15]. Retinex theory assumes that the human visual system has three independent ways to perceive short, medium, and long wavelengths in the visible light spectrum. Based on the retinex theory, SSR utilizes Gaussian low pass filter (LPF) and log operation to accentuate a specific wavelength range of the image, and MSR gives an output image as the weighted sum of the retinex output images by using several linear LPFs having different support regions.

This paper proposes a power-constrained contrast enhancement algorithm using a sub-band decomposed MSR (SD-MSR) [16] for OLED display. First, we designed a modified log function for dynamic power saving. Second

Manuscript received August 28, 2013; revised December 17, 2013 and March 24, 2014; accepted May 6, 2014. Date of publication May 16, 2014; date of current version June 24, 2014. This work was supported by the National Research Foundation of Korea through the Korean Government under Grant 2012000446. The associate editor coordinating the review of this manuscript and approving it for publication was Mr. Pierre-Marc Jodoin.

The authors are with the Department of Electronic Engineering, Inha University, Incheon 402-751, Korea (e-mail: namyo@inha.edu; pride0723@inha.edu; bcsong@inha.ac.kr).

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.2014.2324288

we propose a coarse-to-fine power control mechanism based on SD-MSR, which jointly achieves contrast enhancement and dynamic range compression using an adaptive weighting strategy proper for an input image. Finally, we present a power control scheme for a constant power reduction ratio in video sequences by using temporal coherence in video sequences. Experimental results show that the proposed algorithm provides better visual quality than previous methods, and a consistent power-saving ratio without flickering artifacts for video sequences.

The rest of this paper is organized as follows. Section II reviews a typical MSR and the SD-MSR which is a basic framework of the proposed algorithm. Section III defines a power model for OLED display and proposes a power-constrained contrast enhancement algorithm for video sequences as well as still images in detail. Section IV presents several optimization skills for real-time processing. Section V provides intensive experimental results. Finally, Section VI concludes this paper.

II. Sub-Band Decomposed Multi-Scale Retinex

MSR is an extended SSR with multiple kernel windows of different sizes. MSR output is a weighted sum of several different SSR outputs [14]–[16]. The MSR output for a single spectral component can be represented as

$$R^{MSR}(x,y)=\sum_{n=1}^{N} w_n \cdot R_n(x,y) \tag{1}$$

where

$$R_n(x,y)=\log I(x,y)-\log(F_n(x,y)^{*}I(x,y)) \tag{2}$$

Here, $R_n(x,y)$ denotes a retinex output associated with the n-th scale for an input image $I(x,y)$. Note that gain w_n is determined so that it can satisfy the condition of $\sum w_n=1$. The symbol "*" in Eq. (2) denotes the convolution operation and N is the number of scales. $F_n(x,y)$ denotes a surround function and is given by

$$F_n(x,y)=K_n e^{(x^2+y^2)/\sigma_n^2} \tag{3}$$

where K_n is determined so that $F_n(x,y)$ can satisfy $\sum_x\sum_y F_n(x,y)=1$. σ_n^2 denotes the variance of the Gaussian kernel at the n-th sub-band. Under the condition $\sigma_n>\sigma_{n-1}$ for every SSR, we can derive successive frequency sub-bands. Note that a small σ_n is suitable for enhancing fine details, whereas a large σ_n is suitable for improving tonality. Thus, it is important to select an appropriate value of σ_n in the MSR.

Based on this rationale, Jang et al. proposed an SD-MSR that consists of a modified logarithmic function, sub-band decomposition, space varying sub-band gain, and an automatic gain/offset control [16] (see Fig. 1). The modified log ($m$log) is defined as

$$m\log(I(x,y))=\begin{cases} w_L\log(I(x,y)+1) & I(x,y)\le\tau \\ -w_H\log(D-I(x,y))+\log D & I(x,y)>\tau \end{cases} \tag{4}$$

where τ is a user-defined threshold and D denotes an image dynamic range. For example, D is 256 for an 8-bit image.

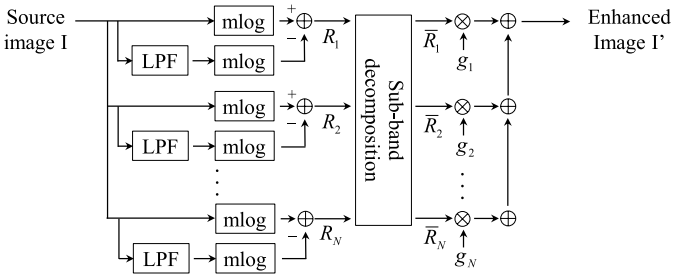


Fig. 1. Block diagram of the conventional SD-MSR [16].

w_L and w_H denote weighting parameters according to τ and are defined as

$$w_L=\frac{\frac{\tau}{D-1}\log D}{\log(\tau+1)},\qquad w_H=\frac{(1-\frac{\tau}{D-1})\log D}{\log(D-\tau)} \tag{5}$$

As a result, the $m$log function of Eq. (4) enhances the contrasts of dark regions as well as bright regions. In this way, we can enhance image details both in highlights and shadows.

Another feature of SD-MSR is to decompose the modified retinex outputs into nearly non-overlapping spectral bands. The following equation accomplishes this sub-band decomposition:

$$\begin{aligned} \overline{R}_1&=R_1, & n&=1 \\ \overline{R}_n&=R_n-R_{n-1}, & 2\le n&<N \end{aligned} \tag{6}$$

As n increases, $\overline{R}_n$ corresponds to the low frequency region more and more. Here, R_n is computed by replacing the log of Eq. (2) with the $m$log of Eq. (4) Next, the space varying sub-band gain at the n-th sub-band is defined as

$$g_n(x,y)=\left(\frac{1}{NR_n(x,y)+\varepsilon_g}\right)^{1-\frac{\sigma_n}{\sigma_{\max}+\varepsilon_\sigma}} \tag{7}$$

where $\sigma_{\max}=\max_{n\in\{1,2,\ldots,N\}}\sigma_n$. Also, ε_g and ε_σ are two constants to avoid dividing by zero. In this paper, ε_g and ε_σ are set to 0.1 and 0, respectively. NR_n denotes the normalized SSR at the n-th sub-band and is defined as

$$NR_n(x,y)=\frac{\left|\overline{R}_n(x,y)\right|}{\left|\overline{R}_n\right|_{\max}} \tag{8}$$

where $\left|\overline{R}_n\right|_{\max}=\max_{(x,y)}\left|\overline{R}_n\right|$. In a high spectral band of small n, they make the gain difference between pixels larger, especially for the pixels with low $NR_n(x,y)$. This is because this spectral band has large high-frequency components representing image details. Meanwhile, they lower the gain difference between pixels in a high spectral band of large n to maintain the characteristics of a natural scene. Thus, using Eq. (7), the final enhanced image I' is output as follows:

$$I'=\sum_{n=1}^{N} g_n\bar{R}_n \tag{9}$$

III. The Proposed Algorithm

We propose a power controllable contrast enhancement algorithm for OLED display based on SD-MSR. Fig. 2 describes the proposed algorithm, which is composed of three stages. The first stage coarsely reduces the power of an input

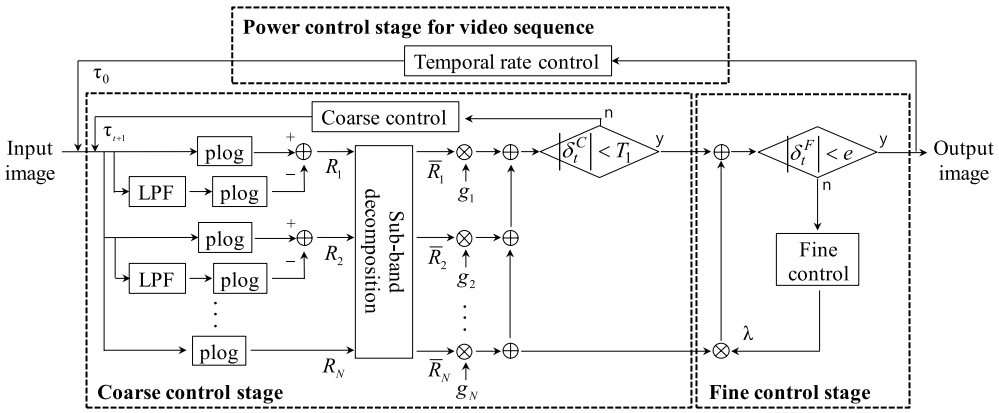


Fig. 2. Block diagram of the proposed algorithm.

image nearer to the target power with contrast enhancement, and the second stage finely controls the image power such that it is very close to the target power. If the input is a video sequence, the final stage adjusts the power of each image so that it is similar to those of its neighbors by considering the temporal coherence of the input video sequence.

The proposed algorithm is differentiated from previous methods in the following three aspects. First, we control the target power level automatically. Second, we avoid the flickering phenomenon by keeping the power levels of adjacent images constant for video sequences. Third, we achieve real-time processing of the proposed algorithm on a general-purpose graphics processing unit (GPU) even for full HD video sequences.

A. Power Modeling for OLED Display

Before presenting a detailed explanation of the proposed algorithm, we need to model power for an OLED display. Dong et al. presented a pixel-based power model that estimates the power consumption of OLED modules based on the red-green-blue (RGB) specification of each pixel [21]. The power consumption of an OLED display with K pixels, i.e., P is

$$P_{OLED} = C + \sum_{i=1}^{K} \{f_R(R_i) + f_G(G_i) + f_B(B_i)\} \quad (10)$$

where $f_R(R_i)$, $f_G(G_i)$, $f_B(B_i)$ indicate the power consumption of red, green, and blue devices of the pixel, respectively and i stands for the pixel index in an image. C is a constant to account for the static power contribution made by the non-pixel part of the display, which is independent of the pixel values. In this paper, the constant C is not considered for convenience. Also, we consider only the Y-component because it dominates the entire overall power. Note that the Y-component indicates the luminance component in YUV color format. So we use the Y-component power consumption (YP) of an OLED display with K pixels [11] as

$$YP = \sum_{i=1}^{K} Y_i^{\gamma} \quad (11)$$

where γ is a parameter for gamma correction for a given display device.

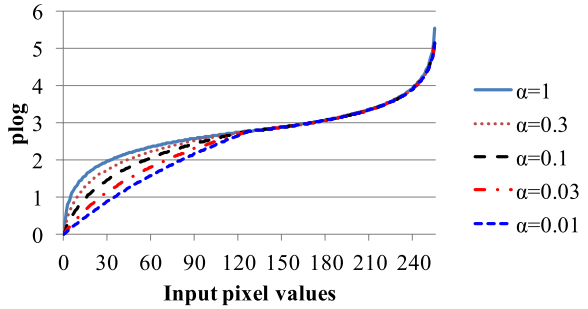


Fig. 3. The $p$log function according to α values. Here, τ is 128.

B. The Proposed Algorithm

This section details the proposed algorithm.

1) Coarse Control Stage: The $m$log of conventional SD-MSR plays a role in enhancing the contrasts of highlight and shadow regions. In other words, contrast in the dark region becomes high by increasing the intensity level of the pixels in the region, and contrast in the bright region also becomes high by decreasing the intensity level of the pixels in the region. However, the increase of the intensity values in the shadow region results in the increase in power consumption for the OLED display. So, for low power consumption as well as contrast enhancement, even in the shadow region, we re-define a so-called power-constrained log ($p$log) from the $m$log of Eq. (4) as follows:

$$p\log(I(x,y)) = \begin{cases} \frac{\tau \log D \log(\alpha I(x,y)+1)}{(D-1)\log(\alpha\tau+1)} & I(x,y) \le \tau \\ m\log(I(x,y)) & I(x,y) > \tau \end{cases} \quad (12)$$

Here, α is a sort of penalty parameter. Fig. 3 describes the $p$log graph according to α values. In Fig. 3, $p$log is equivalent to $m$log when α is 1. We can observe that $m$log tends to increase the contrast of extremely darker range within the dark interval below τ, and decrease the contrast of the middle range around τ. So $m$log works for images having extremely dark region, but its effect on contrast enhancement may be marginal for the other images. On the other hand, we can see that $p$log makes the pixel values in the dark region change less as α decreases. So $p$log provides less contrast enhancement than $m$log for dark pixels, but the former shows better contrast

enhancement than the latter for middle intensity pixels. Note that the human visual system tends to be relatively insensitive to contrast change in the dark region [15]. Therefore, the $p$log of Eq. (12) has the effect of controlling the increase in power consumption while partially lowering the contrast in the dark region. From Eq. (7) and MSRs computed by $p$log, i.e., $\{\overline{R}_n\}$, we can derive the following output image.

$$\widehat{R}_t = \sum_{n=1}^{N} g_n \bar{R}_n \tag{13}$$

where t indicates the iteration number. Initially, t is set to 0.

On the other hand, basing YP on Eq. (11), the power reduction ratio of an input image and its output image is defined as follows:

$$p_t = 1 - \frac{YP(\widehat{R}_t^N)}{YP(I)} \tag{14}$$

where $\widehat{R}_t^N$ is the normalized version of $\widehat{R}_t$ such that it has the same dynamic range as the input image I, i.e., $[l, L]$. In this paper, $\widehat{R}_t^N$ can be computed with Eq. (15) as in [16].

$$f(X) = X^N = \frac{X - m}{M - m}(L - l) + l \tag{15}$$

where $M = \max(\widehat{R}_t)$ and $m = \min(\widehat{R}_t)$. p_t should be controlled to be close to target power reduction ratio P. Let δ_t denote the difference between p_t and P as in Eq. (16).

$$\delta_t = P - p_t \tag{16}$$

δ_t has an identical meaning in the fine control stage (FCS) as well as the coarse control stage (CCS), and it is re-defined as δ_t^C and δ_t^F in the CCS and FCS, respectively. We make use of δ_t^C as a criterion to determine the range of power reduction at every iteration of the CCS. Using δ_t^C, we propose an effective strategy to achieve quick convergence speed. If $|\delta_t^C|$ is within a sufficiently small range, we just terminate the iteration(s) in the CCS. Also, if $|\delta_t^C|$ is too large, we increase the variation of τ in Eq. (12) to achieve fast convergence. Let T_1 and T_2 be the thresholds for the two decisions, respectively. Section V depicts how T_1 and T_2 are determined.

According to the above-mentioned strategy, if p_t is sufficiently close to P, that is, $-T_1 < \delta_t^C < T_1$, we go to the FCS without further iterations (see Fig. 2). Otherwise, we have to control τ for approaching P. Note that the decrease of τ suppresses the range of intensity increase by reducing the shadow region, and it concurrently expands the range of intensity decrease by widening the highlight region. So we can achieve P by controlling τ properly. The basic mechanism to control τ is to quickly approach target power reduction ratio P with a big change in τ for a large δ_t^C, and get close to P with a small change in τ for a small δ_t^C. More specifically, if $\delta_t^C < -T_2$, we significantly increases τ as in Eq. (17) because such a condition indicates an excess of power reduction over P.

$$\tau_{t+1} = \tau_t + (D - \tau_t)/2 \tag{17}$$

Simultaneously, $\hat{R}_t$ is reset to $\hat{R}_{t-1}$ because it exceeds the target power. If $-T_2 < \delta_t^C < -T_1$, we increase τ relatively small as in Eq. (18) because δ_t^C weakly overruns P.

$$\tau_{t+1} = \tau_t + (D - \tau_t)/4 \tag{18}$$

In this case, $\hat{R}_t$ is also reset to $\hat{R}_{t-1}$. On the other hand, $T_1 < \delta_t^C < T_2$ indicates weak power reduction below P. So we approach P by decreasing τ relatively small as in Eq. (19).

$$\tau_{t+1} = \tau_t - \tau_t/4 \tag{19}$$

Finally, if $\delta_t^C > T_2$, p_t falls short of P noticeably. So we rapidly approach P by decreasing τ significantly.

$$\tau_{t+1} = \tau_t - \tau_t/2 \tag{20}$$

As a result, τ_{t+1} is adaptively computed with the above-mentioned update rule, and this entire CCS iterates with the updated τ.

Note that the initial τ, i.e., τ_0 affects the number of iterations in the CCS. If τ_0 is considerably larger or smaller than an optimal value for each frame, the convergence speed may get slower due to the increased number of iterations. Since the optimal τ_0 per frame is generally unknown, this paper empirically sets τ_0 to 80 for the first frame of each video sequence. For the following frames τ_0 is updated according to a specific rule, which will be discussed in the next subsections.

For extremely high P values, the CCS does not often achieve the target range of $-T_1 < \delta_t^C < T_1$, and can be caught in an infinite loop trap. In order to mitigate computational damages by such a phenomenon, we limit the maximum number of iterations to 5. However, the CCS works well except such an extreme case.

2) Fine Control Stage: Since p_t still cannot get close to P through the proposed CCS, we require additional power control. Let δ_t^F denote the difference between the power reduction ratio at the t-th iteration of the fine control stage and P. Initially, δ_0^F is set to the final δ^C obtained from the CCS. If $|\delta_t^F| < e$, this FCS terminates as in Fig. 2, and the output image $\overline{I}$ is obtained by normalizing the final $\hat{R}$ from the previous stage. Here, e indicates the allowable tolerance. Fig. 4 (a) shows the number of iterations in the FCS according to e, and Fig. 4 (b) is the graph of the power control error vs. e. For this experiment, the first frames of all the test video sequences and still images (as mentioned in section V) are employed, and their average values are displayed in Fig. 4. We can observe that the number of iterations decreases as e increases. In other words, the increase of e indicates quick convergence speed. On the contrary, the increase of e tends to augment the power control errors. Especially, if e becomes larger than 0.001, the power control error rapidly increases. Also if e is large, the so-called flickering artifact may happen for video sequences due to inconsistent luminance levels between adjacent images. So we set e to 0.001. For instance, if P is 20%, this FCS iterates until p_t is included in [19.9% 20.1%].

If $|\delta_t^F| \geq e$, a regular FCS begins. In general, the high-frequency region of an image is highly involved in details of the image, but its impact on the power of the image is

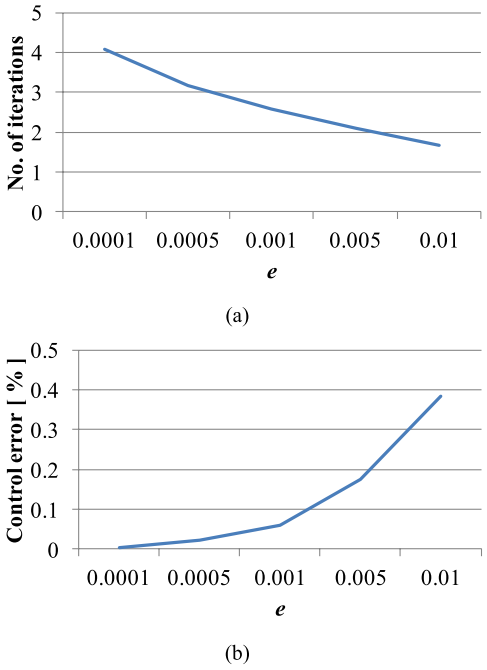


Fig. 4. The effect of e on the performance of the proposed algorithm. (a) The number of iterations according to e, and (b) the control error according to e.

negligible. On the other hand, the low-frequency region is rarely related to image details, but it dominates image power as a whole. So we try to approach P by finely controlling the proportion of the lowest-band MSR which may have most of the image power. In detail, we control the gain of $\overline{R}_N$ as follows:

$$\hat{R} = \sum_{n=1}^{N-1} g_n \overline{R}_n + (g_N + \lambda)\overline{R}_N \tag{21}$$

where λ indicates a control parameter for the lowest-band MSR. λ, which is updated according to Eq. (22) enables the FCS to approach the target power with little change of contrast

$$\lambda_{t+1} = \lambda_t - \delta_t^F \tag{22}$$

where λ_0 is set to 0. This entire FCS iterates until $\left|\delta_t^F\right| < e$. Finally, the output image $\overline{I}$ is obtained by normalizing $\hat{R}$.

3) Power Control Stage for Video Sequence: With image sequences, the τ parameter for the current image can be derived from that of the previous image because of high temporal correlation. So, if we know the power reduction ratio of the previous image in advance, we can determine the increase and decrease in τ for the current image. Based on this rationale, we determine τ_0 of the next image, i.e., $\tau_{0,next}$ from τ_0 of the current image according to δ_0^C. This helps the CCS in rapidly converging with a reduced number of iterations for video sequences. Since a significant change of τ_0 between adjacent images may cause flickering artifacts due to contrast fluctuation, we limit $\tau_{0,next}$ to [τ_0-5, τ_0+5].

In detail, since $\delta_t^C < -T_2$ indicates an excessive power reduction over P, we try to decrease the power consumption ratio of the next image by setting $\tau_{0,next}$ to τ_0+5. Similarly, with $-T_2 < \delta_t^C < -T_1$, we determine $\tau_{0,next}$ to τ_0+3 for weak power consumption. On the other hand, with$T_1 < \delta_t^C < T_2$, we set $\tau_{0,next}$ to τ_0-3. Finally, with$T_2 < \delta_t^C$, we set $\tau_{0,next}$ to τ_0-5 for strong power consumption.

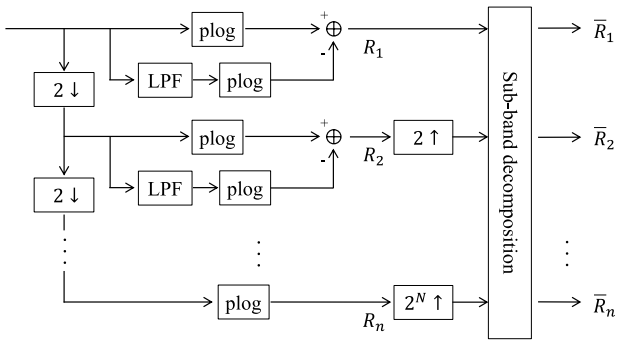


Fig. 5. The proposed algorithm when MRP is employed.

IV. Computational Optimization

In order to reduce the computational complexity of the proposed algorithm, we employed two algorithmic optimization approaches. The first approach is multi-resolution processing (MRP) where all the operations in the low-frequency bands are performed using the low-resolution images instead of the full-resolution counterparts. Another approach is to replace nonlinear log operations with look-up table (LUT). After the algorithmic optimization, we dramatically reduced the entire running time by Compute Unified Device Architecture (CUDA) programming on the GPU.

A. Algorithm-Level Optimization

1) Multi-Resolution Processing: As seen in Eq. (2), the computation of MSRs includes convolution in the spatial domain. Since the kernel size of Eq. (3) is normally very large, the computational load due to convolutional operation is significant. We reduce such a huge computational cost by employing MRP. For instance, MRP has ever been used for fast motion estimation [22]. Since MRP coincides with hierarchical frequency structure of MSRs, we applied MRP for computational optimization of MSRs.

Note that the high-frequency band tends to have a low spatial correlation, but the low-frequency band has a high spatial correlation in general. So we use high-resolution images for computing MSRs at high-frequency bands and employ low-resolution images for calculating MSRs at low-frequency bands. We call this multi-resolution processing, and describe its structure in Fig. 5. In this figure, '2↓' indicates down-sampling with a scale ratio of 2:1, and '2^N ↑' stands for up-sampling with a scale ratio of 2^N:1. In this paper, we adopted Lanczos filter [23] for up-sampling. As a result, if the down-sampling ratio for a specific frequency band is set to β, the computational complexity to compute the corresponding MSR decreases up to $1/\beta^4$.

2) LUT-Based Log Operation:: The fastest solution for implementing a nonlinear function in a computer program is the LUT [24].The *p*log operation in deriving MSRs generally requires significant computation due to its nonlinearity. So

we replace such an operation with LUT, already computed in off-line. Since there are D different τ values and D plog values per τ, the dictionary has D^2 pre-computed $p$log values. Assuming a fixed α, $p$log of 4-byte float-type, and D of 256, the dictionary size is 256 Kbytes.

B. CUDA Programming

In addition, we implemented the proposed algorithm using CUDA toolkit 4.0 on GPU, i.e., nVidia GeForce GTX 460, which consists of 336 core processors. As the main central processing unit (CPU), an Intel Core2 Quad CPU Q9400 at 2.66GHz was used.

All the operations except computing τ and λ were implemented through parallel processing on the GPU. Note that τ and λ are computed on the CPU. For CUDA programming, 128 threads were used, and parallel processing by 32 blocks was performed. The execution time of the proposed algorithm is explained in Section V.

V. Experimental Results

To evaluate the performance of the proposed algorithm, we chose two images from Kodak Lossless True Color Image Suite[1] (*caps* and *beach*) and a high dynamic range (HDR) test image *memorial*. Also, we employed six common intermediate format (CIF) video sequences: *container* (500 frames), *football* (90 frames), *Paris* (300 frames), *foreman* (300 frames), *bus* (500 frames), *Stefan* (90 frames) and five 720p sequences: *big ship* (60 frames), *crew* (60 frames), *jets* (60 frames), *night* (60 frames), *raven* (60 frames), and four 1080p sequences: *crowd run* (500 frames), *park joy* (60 frames), *toys and calendar* (60 frames) *traffic* (60 frames). We processed only the luminance components in the experiments. More specifically, given a color image, we converted it to the YUV color space and then process only the Y-component without modifying the U- and V-components.

A. Evaluation in Terms of Subjective Visual Quality

We compared the proposed algorithm with a typical linear algorithm as the most trivial approach and a power-constrained contrast enhancement (PCCE) proposed in [11] in terms of qualitative visual quality. Here, the linear algorithm indicates a method that transforms I to $I\sqrt{1-P}$ for a given P. In the proposed algorithm, α and τ_0 were set to 0.1 and 80, respectively. Also, T_1 and T_2 were set to 0.05 and 0.1. Since PCCE cannot automatically control its parameters, μ was set to 5 and ρ was manually chosen for a given P.

First, Fig. 6 and Fig. 7 are the results of several algorithms for the *caps* and *beach* images when P is 10%. Both the proposed algorithm and the PCCE achieve significant contrast enhancement compared to the linear algorithm. However, the PCCE loses the details in the yellow cap and shows too-darkened shadows as seen in Fig. 6(c). Similarly, the PCCE loses details in the cloud in Fig. 7(c) and darkens the forest too much. This is a phenomenon that histogram-based approaches such as the PCCE often suffer from. On the other hand, the proposed algorithm can enhance details while effectively preserving the overall intensity level of the input image.

[1] http://r0k.us/graphics/kodak

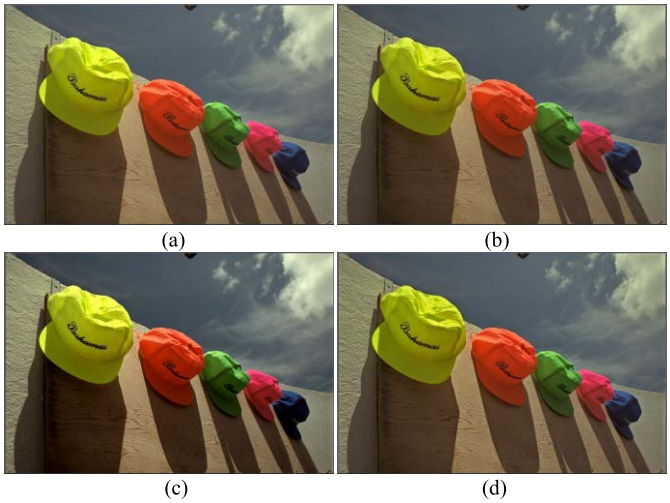


Fig. 6. The resulting images from *caps* when P is set to 10% (a) original, (b) linear algorithm, (c) PCCE [11], (d) the proposed algorithm ($\tau_t = 40$).

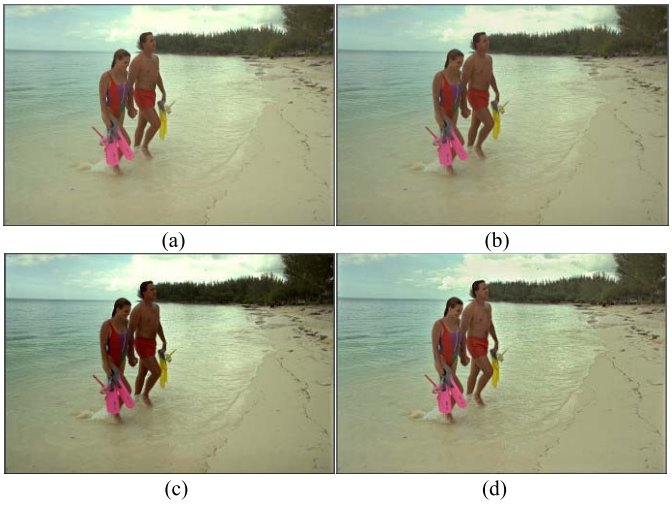
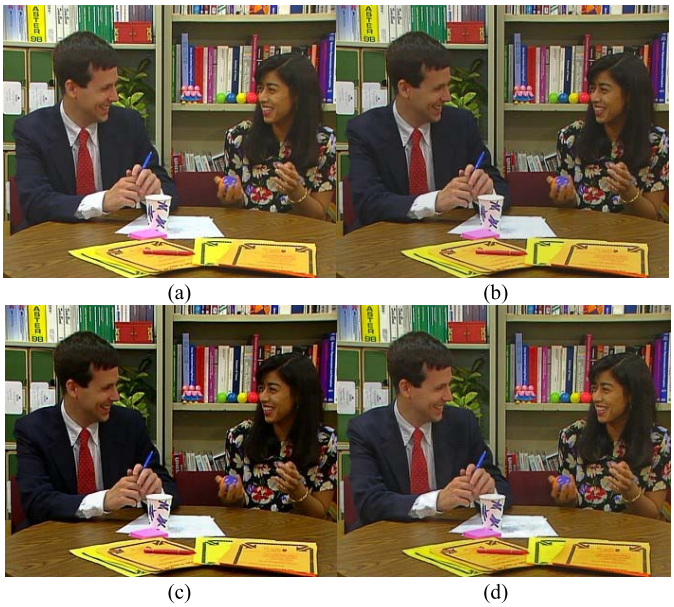


Fig. 7. The resulting images from *beach* when P is set to 10% (a) original, (b) linear algorithm, (c) PCCE [11], (d) the proposed algorithm ($\tau_t = 160$).

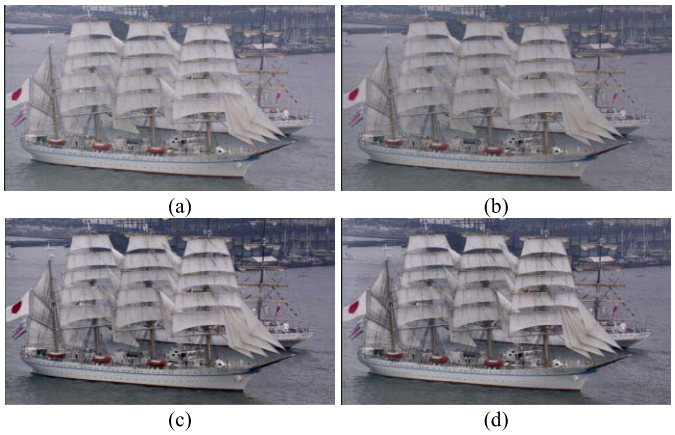
Figs. 8 and 9 compare several algorithms for the first frames of *Paris* and *big ship* sequences when P is 20%. Fig. 8 shows that the proposed algorithm enhances the details such as a man's suit and cup's shade better than the PCCE and the linear method. Similarly, Fig. 9 proves that the proposed algorithm successfully represents the details in a sail on a ship in comparison with the PCCE.

Figs. 10 and 11 show the comparison results of several algorithms when P is 30%. They result from the first frames of *traffic* and *crowd run* sequences. The PCCE still provides high performance in enhancing contrasts, but loses details in the dark region (for example, the shaded buildings and trees) as seen in Fig. 10(c). However, Fig. 10(a) and Fig. 10(d) show that the proposed algorithm preserves the overall luminance level of the input image with enhanced contrasts even for a high P of 30%. Similarly, Fig. 11 proves that the proposed algorithm is superior to the other algorithms.



(a) (b) (c) (d)

Fig. 8. The resulting images from *Paris* when P is set to 20% (a) original, (b) linear algorithm, (c) PCCE [11], (d) the proposed algorithm ($\tau_t = 60$).



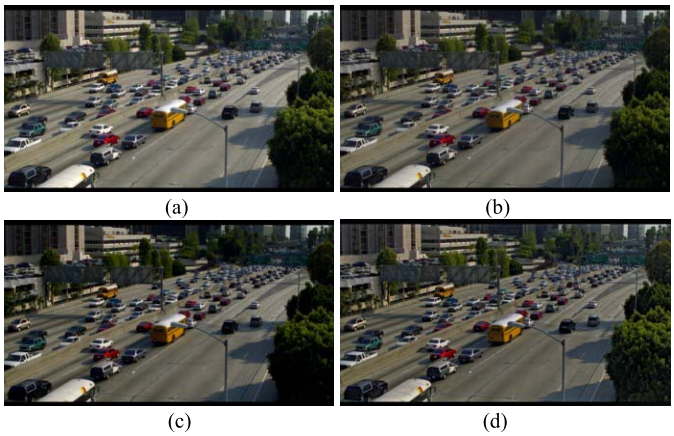
(a) (b) (c) (d)

Fig. 9. The resulting images from *big ship* when P is set to 20% (a) original, (b) linear algorithm, (c) PCCE [11], (d) the proposed algorithm ($\tau_t = 60$).

Figs. 12, 13 and 14 compare several algorithms for *night*, *crew*, and *memorial* images, respectively when P is 50%. The PCCE is still good for enhancing contrasts in comparison with the other algorithms. Especially observing lights and electric signs in Fig. 12(c), we can find that the PCCE outperforms the other methods in terms of contrast enhancement. However, the PCCE loses some details in the dark region. On the other hand, note that the proposed algorithm preserves most of the details of the original image while providing reasonably enhanced contrasts in comparison with the other algorithms as shown in Fig. 12(d). Fig. 13 and Fig. 14 also prove that the proposed algorithm produces better visual quality than the other algorithms.

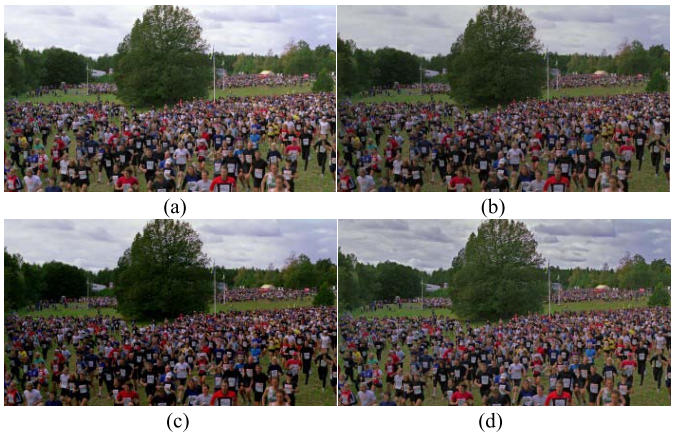
B. Evaluation in Terms of Objective Visual Quality

In order to quantitatively evaluate the performance of power control algorithms, some metrics to measure the degree of edge preservation were employed [9], [10]. Among them, we



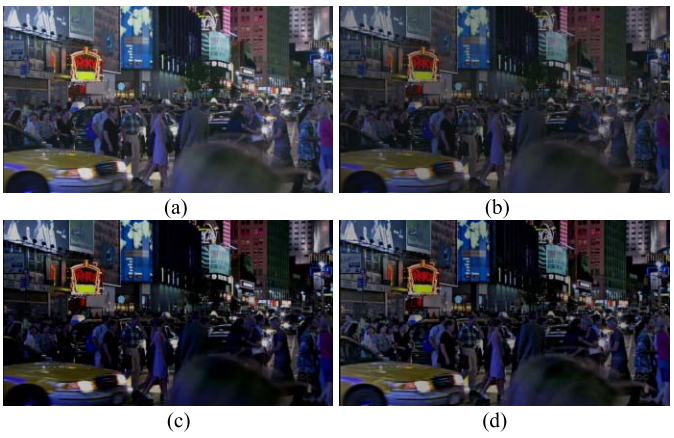
(a) (b) (c) (d)

Fig. 10. The resulting images from *traffic* when P is set to 30% (a) original, (b) linear algorithm, (c) PCCE [11], (d) the proposed algorithm ($\tau_t = 80$).



(a) (b) (c) (d)

Fig. 11. The resulting images from *crowd run* when P is set to 30% (a) original, (b) linear algorithm, (c) PCCE [11], (d) the proposed algorithm ($\tau_t = 40$).

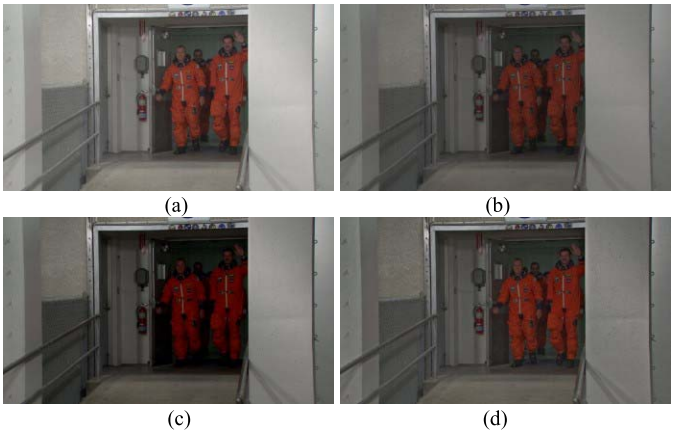


(a) (b) (c) (d)

Fig. 12. The resulting images from *night* when P is set to 50% (a) original, (b) linear algorithm, (c) PCCE [11], (d) the proposed algorithm ($\tau_t = 40$).

adopted a metric using the Lapacian operator. First, we obtain a Laplacian image using the following equation.

$$\psi(x, y) = \begin{cases} 1 & Lap(x, y) > T \\ 0, & otherwise \end{cases} \tag{23}$$



(a) (b) (c) (d)

Fig. 13. The resulting images from *crew* when P is set to 50% (a) original, (b) linear algorithm, (c) PCCE [11], (d) the proposed algorithm ($\tau_t = 80$).

(a) (b) (c) (d)

Fig. 14. The resulting images from *memorial* when P is set to 50% (a) original, (b) linear algorithm, (c) PCCE [11], (d) the proposed algorithm ($\tau_t = 80$).

where Lap indicates an image obtained by the conventional 3×3 Lapacian operator, and T is a threshold to determine edge pixels. Let ψ_T denote the operator for counting the number of pixels whose $\psi(.,.)$ is 1. Now, we calculate the edge-preservation ratio ψ_E, defined as the ratio between the number of preserved edge pixels $\psi_T(I')$ and the number of original edge pixels $\psi_T(I)$

$$\psi_E = \frac{\psi_T(I')}{\psi_T(I)} \tag{24}$$

where I and $I\prime$ stand for the input and output images. Here, a pixel is declared to be a preserved edge pixel if it is an edge image in the original image but not in the processed image. So if ψ_E is less than 1, we presume that the edges were lost by processing. On the other hand, if ψ_E is larger than 1, we can say that the edges were enhanced after processing. In this subsection, we quantitatively compared the proposed algorithm with the other algorithms in terms of ψ_E from Eq. (24).

Fig. 15 is the ψ_E graph for different T values in Eq. (24). In this graph, the average ψ_E values for all the 1080p images are plotted when P is 10%. The linear algorithm shows noticeably lower ψ_E values than the PCCE and the proposed algorithm regardless of T. Note that the proposed algorithm provides much better ψ_E values than the PCCE when T is less than 120. Even though the PCCE is better than the proposed algorithm if T is larger than 120, such a T range

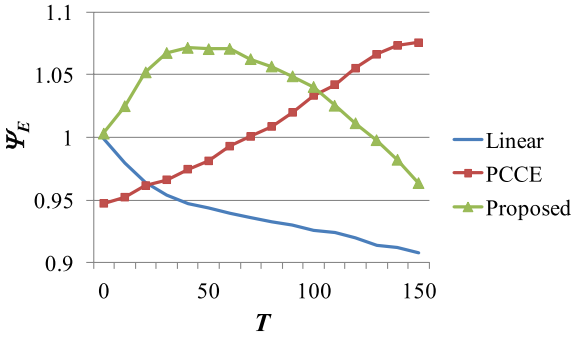


Fig. 15. Comparison of edge-preservation ratios of several algorithms according to T values.

TABLE I
COMPARISON OF THE EDGE PRESERVATION RATIOS ACCORDING TO T_1

Name	T_1 values					
	0.03	0.05	0.08	0.1	0.12	0.15
bus	1.0116	1.0147	1.0140	1.0103	1.0103	1.0103
football	1.2703	1.2515	1.2154	1.2154	1.1688	1.1688
Paris	1.0144	1.0180	1.0180	1.0180	1.0180	0.9941
Stefan	1.0167	1.0171	1.0171	1.0171	0.9909	0.9909
beach	1.6328	1.6328	1.6328	1.6328	1.6328	1.6328
caps	1.2750	1.2750	1.2750	1.2750	1.2750	1.2750
big ship	1.6544	1.6542	1.6542	1.6542	1.6542	1.6542
crew	1.2759	1.2573	1.2573	1.2573	1.2573	1.2573
jets	1.5962	1.5962	1.5962	1.5962	1.5962	1.5962
raven	1.2076	1.2076	1.2076	1.2076	1.2076	1.2076
park joy	1.0710	1.0693	1.0693	1.0644	1.0644	1.0644
toys & cal.	0.9397	0.9397	0.9725	0.9725	0.9117	0.9117
traffic	1.0125	1.0125	1.0125	1.0125	1.0125	1.0125
crowd run	1.1443	1.1443	1.1443	1.1443	1.1443	1.1443
Average	1.2225	1.2202	1.2200	1.2193	1.2098	1.2081

is meaningless because effective edges rarely exist in the T range in practice. As a result, we find that the proposed algorithm is superior to the previous algorithms in the effective T range.

Table I shows the ψ_E values according to T_1 when P and T were set to 20% and 60. As T_1 is smaller, we obtain better ψ_E values. On the other hand, Table II shows that as T_1 is smaller, the number of iterations increases. Based on this observation, we decided to set T_1 to 0.05.

Table III shows the δ_t^C values according to T_2 when T_1 is fixed at 0.05. In this experiment, the δ_t^C values are the averages for all of the test video sequences. We find that as T_2 is above 0.12, the error increases. Thus, we set T_2 to 0.1.

Finally, Table IV compares the ψ_E values of several algorithms for two different P values when T_1, T_2, and T were set to 0.05, 0.1, and 60, respectively. The proposed algorithm outperformed the other algorithms in both cases. For example, when P is 10%, the proposed algorithm provides at maximum 36% and on average 15% higher ψ_E values than the PCCE. Also, when P is 30%, we can see that the proposed algorithm shows at maximum 30% and on average 6% higher ψ_E values than the PCCE. For this experiment, we used noise-free video sequences, and we confirmed that the proposed algorithm shows outstanding detail enhancement performance. Note that noise components in video sequences normally boost

TABLE II
COMPARISON OF THE NUMBER OF ITERATIONS ACCORDING TO T_1 IN THE CCS

Name	T_1 values					
	0.03	0.05	0.08	0.1	0.12	0.15
bus	3	3	3	2	2	2
football	3	2	2	2	1	1
Paris	3	2	2	2	2	1
Stefan	4	2	2	2	1	1
beach	1	1	1	1	1	1
caps	2	2	2	2	2	2
big ship	2	1	1	1	1	1
crew	2	1	1	1	1	1
jets	1	1	1	1	1	1
raven	1	1	1	1	1	1
park joy	4	3	3	2	2	2
toys & cal.	3	3	2	2	1	1
traffic	1	1	1	1	1	1
crowd run	1	1	1	1	1	1
Average	2.21	1.71	1.64	1.5	1.29	1.21

TABLE III
COMPARISON OF THE CONTROL ERROR ACCORDING TO T_2 IN THE CCS

# of iterations	T_2 values				
	0.08	0.1	0.12	0.14	0.16
0	0.0993	0.0993	0.0993	0.0993	0.0993
1	0.0363	0.0363	0.0414	0.0436	0.0464
2	0.0100	0.0100	0.0170	0.0165	0.0165

with detail enhancement. However, if the noise level of an input video sequence is known in advance, we can suppress noise boosting by imposing restrictions on highfrequency gain, i.e., g_1.

In addition, we evaluated the proposed algorithm in terms of several quantitative quality metrics. Agaian et al. presented a contrast enhancement metric called *EME* in [25].

$$EME = \frac{1}{H \times V} \sum_{h=1}^{H} \sum_{v=1}^{V} 20 \ln \frac{I_{\max;h,v}}{I_{\min;h,v} + \xi} \tag{25}$$

where the image is broken up into $H \times V$ blocks, $I_{\max;h,v}$ and $I_{\min;h,v}$ are the maximum and minimum in a given block, and ξ is a small constant equal to 0.0001 to avoid dividing by 0. Table V compares the *EME* values for several algorithms when P is set to 10% and 30%, respectively. Note that the proposed algorithm provides significantly higher *EME* values than those of the input images in spite of the reduced power. We can observe that the proposed algorithm noticeably outperforms the linear method, but shows slightly lower *EME* values on average than the PCCE. Even though the PCCE provides the best *EME* values, it tends to further darken dark areas and is often hard to discriminate some details in the dark areas as shown in Fig. 10(c).

We also evaluated the proposed algorithm in terms of a sharpness enhancement metric proposed by Fan and Au in [26]. The sharpness enhancement metric, i.e., S is defined as

$$S = \left\| \sqrt{G_H^2 + G_V^2} \right\| \tag{26}$$

TABLE IV
COMPARISON IN TERMS OF THE EDGE PRESERVING RATIO

Name	P=10%			P=30%		
	linear	PCCE	Proposed	linear	PCCE	Proposed
bus	0.93	1.02	1.14	0.78	0.88	0.89
football	0.94	1.15	1.29	0.84	0.99	0.86
Paris	0.96	1.07	1.17	0.88	1.01	1.04
Stefan	0.93	1.05	1.27	0.80	0.95	1.06
beach	0.91	1.26	1.62	0.74	1.12	1.42
caps	0.92	1.29	1.57	0.75	1.19	1.35
big ship	0.90	1.41	1.62	0.7	1.19	1.32
crew	0.94	1.1	1.23	0.83	1.05	0.96
jets	0.91	1.44	1.65	0.75	1.12	1.24
raven	0.97	1.16	1.21	0.93	1.18	1.14
park joy	0.97	1.01	1.12	0.91	0.96	0.96
toys & cal.	0.89	1.12	0.97	0.68	0.99	0.84
traffic	0.94	0.95	1.06	0.79	0.87	0.89
crowd run	0.95	0.95	1.09	0.81	0.87	1.10
Average	0.93	1.14	1.29	0.80	1.02	1.08

TABLE V
COMPARISON IN TERMS OF THE *EME* VALUE

Name	Org	P = 10%			P = 30%		
		Linear	PCCE	Proposed	Linear	PCCE	Proposed
bus	79.08	74.99	79.13	82.71	66.16	72.02	68.49
football	50.88	48.29	54.55	53.67	42.58	50.51	47.28
Paris	77.67	73.68	87.39	84.42	65.02	80.73	73.35
Stefan	78.72	74.70	82.99	86.94	65.81	76.28	72.25
beach	25.83	24.5	32.64	34.89	21.61	30.77	27.18
caps	23.05	21.87	28.41	27.84	19.29	26.7	27.1
big ship	35.92	34.07	48.00	42.07	30.05	44.21	40.34
crew	21.08	19.81	24.63	27.14	17.64	22.41	25.04
jets	19.02	17.97	26.91	24.14	15.92	24.91	32.22
raven	19.58	18.54	27.74	27.01	16.39	26.24	25.6
park joy	43.9	41.49	48.33	44.41	36.74	44.52	38.74
toys & cal.	31.2	30.27	34.05	32.83	26.11	31.2	27.1
traffic	43.73	40.1	45.02	44.91	36.59	41.21	37.57
crowd run	60.01	56.41	63.33	59.63	50.21	56.31	55.27
Average	43.55	41.19	48.79	48.04	36.43	44.85	42.68

TABLE VI
COMPARISON IN TERMS OF SHARPNESS ENHANCEMENT METRIC

Name	Org	P=10%			P=30%		
		Linear	PCCE	Proposed	Linear	PCCE	Proposed
bus	4.98	4.72	5.03	5.86	4.16	4.54	4.86
football	3.08	2.91	3.29	3.91	2.56	3.02	3.23
Paris	5.72	5.43	6.38	6.82	4.78	5.88	6.11
Stefan	5.27	4.99	5.56	6.23	4.39	5.06	5.82
beach	3.48	3.31	4.43	5.07	2.91	4.19	4.33
caps	3.1	2.95	3.89	3.97	2.59	3.68	3.81
big ship	4.55	4.33	6.12	7.33	3.81	5.71	6.27
crew	2.61	2.48	3.07	3.49	2.18	2.78	3.18
jets	2.49	2.37	3.58	4.43	2.08	3.32	4.15
raven	2.06	1.98	2.94	3.53	1.72	2.76	3.43
park joy	5.58	5.30	6.15	5.83	4.67	5.36	5.19
toys & cal.	3.61	3.44	3.91	4.08	3.02	3.46	3.54
traffic	4.84	4.60	4.91	5.09	4.05	4.51	4.92
crowd run	4.84	7.00	7.80	8.34	6.17	6.89	7.73
Average	4.02	3.98	4.79	5.28	3.51	4.37	4.76

where G_H and G_V indicate horizontal and vertical gradient values, respectively, and S is the average for all the pixels

TABLE VII
THE COMPARISON OF FLICKERING ARTIFACTS IN TERMS OF THE F VALUE. HERE, THE UNIT IS 10^{-2}

Name	*Linear*			*PCCE*			*Proposed*		
	S.N.	S.F.	Q	S.N.	S.F.	Q	S.N.	S.F.	Q
bus	0.07	0.99	0.83	0.02	0.99	0.81	0.10	0.99	0.84
football	0.57	0.99	0.93	0.41	0.99	0.91	0.52	0.99	0.93
Paris	0.30	0.99	0.89	0.07	0.99	0.83	0.29	0.99	0.88
Stefan	0.40	0.99	0.90	0.26	0.99	0.88	0.63	0.98	0.94
beach	0.34	0.95	0.88	0.62	0.98	0.94	0.57	0.92	0.92
caps	0.41	0.94	0.89	0.52	0.96	0.92	0.61	0.95	0.93
big ship	0.77	0.99	0.87	1.00	0.99	0.99	0.99	0.98	0.99
crew	0.13	0.91	0.82	0.14	0.96	0.84	0.21	0.94	0.85
jets	0.23	0.83	0.83	0.45	0.91	0.89	0.47	0.89	0.89
raven	0.23	0.99	0.87	0.47	0.99	0.91	0.46	0.98	0.91
park joy	0.04	0.91	0.80	0.02	0.81	0.76	0.04	0.96	0.81
toys & cal.	0.12	0.98	0.84	0.09	0.95	0.82	0.15	0.97	0.85
traffic	0.22	0.99	0.87	0.15	0.97	0.84	0.21	0.98	0.86
crowd run	0.72	0.99	0.96	0.49	0.99	0.92	0.68	0.99	0.95
Average	0.32	0.96	0.88	0.34	0.96	0.88	0.42	0.96	0.90

in the image. In this paper, typical 1D gradient operators are employed for computing gradients. Table VI compares the S values for several algorithms when P is set to 10% and 30%, respectively. We can observe that the proposed algorithm shows higher S values of 1.21 at maximum and 0.49 on average than the PCCE when P is 10%. Even when P is 30%, the proposed algorithm still outperforms the PCCE as well as the linear method. This is because the proposed algorithm based on the MSR can easily control the highest frequency gain for detail enhancement in comparison with the other algorithms. Also, in order to evaluate the dynamic range compression performance of the proposed algorithm, we adopted a tone-mapped image quality index (TMQI) proposed by Yehaneh et al. [27]. First, the structural fidelity (SF) score and the statistical naturalness (SN) score are computed from an input low dynamic range (LDR) image and the corresponding high dynamic range (HDR) image. Second, the TMQI value for the LDR image is derived from the two scores. Table VII shows the TMQI values for several algorithms when P is set to 20%. We can find that all the algorithms have similar SF scores, but the SN value of the proposed algorithm is noticeably higher than those of the other algorithms. This indicates that the proposed algorithm produces more natural images than the other algorithms. As a result, the proposed algorithm provides higher TMQI value of 0.02 on average than the other algorithms.

Table VIII shows the output power reduction ratios for various P values. For fair experiment, we limited the maximum iteration number of the CCS to 5. We can observe that the proposed algorithm keeps the maximum error below 0.1%, and also provides very stable standard deviation less than 0.1. As a result, the proposed power control mechanism is very accurate and robust.

C. Performance Evaluation for Video Sequences

This section deals with the performance evaluation for video sequences. Fig. 16 compares the power reduction ratios

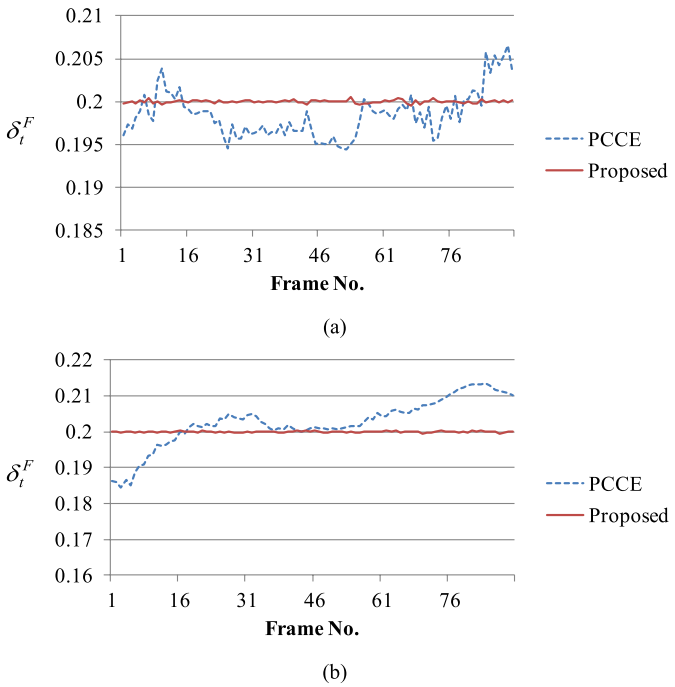


Fig. 16. Variation of the power reduction ratios for two video sequences (a) *foreman* (b) *bus*.

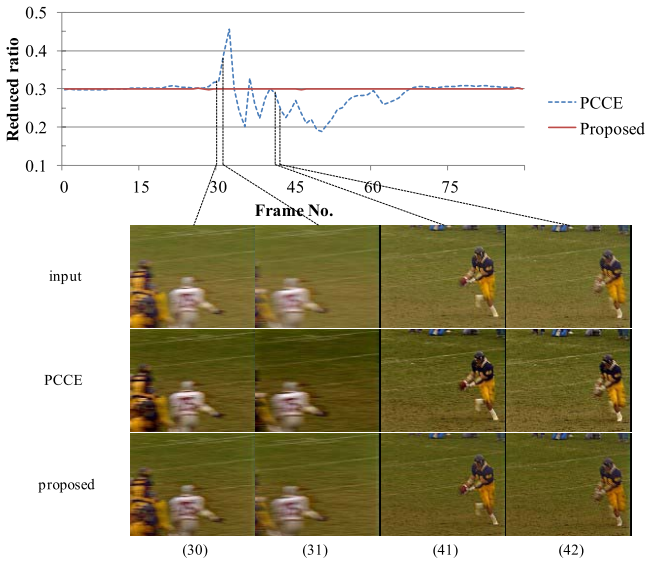


Fig. 17. The power reduction ratio comparison for the *football* video sequence. Here, the lower part provides some image samples for examining flickering artifacts.

(δ_t^F) of the proposed algorithm and the PCCE for two video sequences when P is 20%. For this experiment, ρ for the PCCE is properly tuned to 0.22. We can observe that the proposed algorithm shows consistent δ_t^F values, irrespectively of frame numbers. On the contrary, the PCCE causes significant fluctuation of δ_t^F values. In order to evaluate the flickering artifacts quantitatively, we computed the standard deviation of δ_t^F values (F value) for each video sequence, and compared the F values of the algorithms. In general, if the F value of a video sequence is high, the video sequence may

TABLE VIII
THE OUTPUT POWER REDUCTION RATIOS FOR VARIOUS P VALUES

Name	Target power P								
	10%	20%	30%	40%	50%	60%	70%	80%	90%
bus	10.03	20.02	29.96	39.96	49.96	60.04	69.98	80.03	89.96
football	9.97	19.97	30.01	40.01	49.99	60.01	69.98	80.01	90.02
Paris	10.01	19.98	30.04	39.95	49.96	59.95	69.95	79.95	90.01
Stefan	10.00	20.00	29.99	39.97	50.03	59.98	69.98	79.96	89.97
big ship	9.99	20.04	29.97	39.98	49.96	59.96	69.97	79.97	89.95
crew	10.01	19.99	30.02	39.96	49.96	59.97	69.96	79.97	90.03
jets	10.03	19.99	30.00	39.99	49.97	59.99	69.97	79.92	89.99
raven	9.97	19.97	30.01	40.00	50.02	59.97	69.97	79.98	89.96
park joy	10.03	19.98	29.99	39.98	49.96	59.96	69.92	80.04	90.08
toys & cal.	9.96	19.99	30.00	40.01	50.05	60.03	69.96	79.97	90.08
traffic	9.99	20.02	30.00	40.01	50.02	59.95	69.97	79.95	89.99
crowd run	10.00	20.00	29.99	39.98	49.98	59.96	69.96	79.96	89.97
average	10.00	20.00	30.00	39.98	49.99	59.98	69.96	79.98	90.00
Std. dev	0.025	0.022	0.022	0.022	0.034	0.031	0.017	0.036	0.045

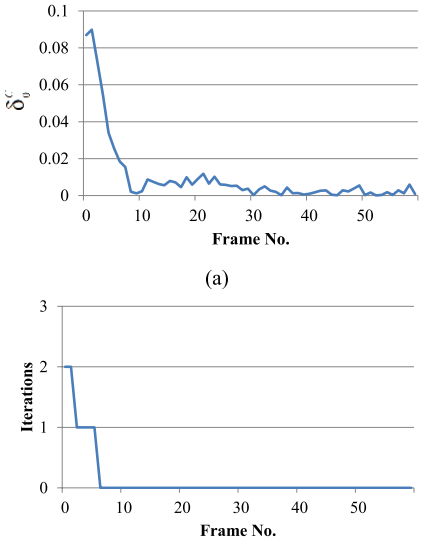


Fig. 18. Fast convergence of the proposed algorithm. (a) The δ_0^C graph according to frame numbers and (b) the number of iterations in the CCS according to frame numbers.

TABLE IX
THE COMPARISON OF FLICKERING ARTIFACTS IN TERMS OF THE F VALUE. HERE, THE UNIT IS 10^{-2}

Name	PCCE	Proposed
bus	1.15	0.03
football	17.7	0.06
Paris	2.08	0.03
Stefan	2.23	0.04
big ship	0.73	0.03
crew	1.15	0.06
jets	2.94	0.05
raven	3.32	0.08
park joy	0.58	0.06
toys and cal.	0.77	0.04
traffic	0.26	0.03
crowd run	0.07	0.05
Average	2.75	0.05

TABLE X
EXECUTION TIME COMPARISON ACCORDING TO OPTIMIZATION LEVELS [MS/FRAME]

Name	No optimization	Used optimization skills		
		MRP	MRP, LUT	All
crowd run	280	187	172	9
park joy	307	196	179	9
toys & cal.	301	199	177	9
traffic	276	184	168	9
Average	291	192	174	9

cause flickering artifacts. Table IX shows that the proposed algorithm provides noticeably lower F values than the PCCE. Especially, the PCCE has an outstanding F value for *football* sequence. This indicates that the PCCE is likely to undergo flickering artifact for the sequence. Actually, Fig. 17 shows the power reduction ratios for the *football* video sequence when P is set to 30%. For this experiment, ρ for the PCCE is manually tuned to 0.3 according to the given P. Since the PCCE cannot keep the power reduction ratio consistently, it may face the so-called flickering artifact. The 41st and 42nd images from the PCCE have different contrast levels, but those of the proposed algorithm show consistent contrast levels. Therefore, the proposed algorithm does not suffer from the visually annoying flickering artifact.

On the other hand, Fig. 18 proves that the proposed algorithm converges quickly in terms of the number of iterations as well as the control error. These graphs result from the *park joy* video sequence when P is set to 20%. In Fig. 18(a), we can see that δ_0^C dramatically decreases owing to the proposed video sequence rate control as time goes on. Also,

Fig. 18(b) shows that the proposed algorithm requires no more iterations.

D. Computational Complexity

Table X shows the execution times of the proposed algorithm for different 1080p images. With only MRP, we can reduce the entire running time up to about 30%. If we additionally employ LUT-based optimization, we can further save execution time of 10%. Finally, if we utilize all the optimization skills including GPU-based parallel programming, we can accomplish the entire execution time of 9 ms per image.

VI. Concluding Remarks

This paper proposes an SD-MSR-based image processing algorithm for fine power control in OLED displays. We designed a power-constrained log function for effective power saving in dark regions. Using the power-constrained log function for SD-MSR and an adaptive weighting strategy proper for an input image, we proposed a coarse-to-fine power control mechanism for still images. Finally, we presented a power control scheme for a constant power reduction ratio in video sequences by using temporal coherence in video sequences. Experimental results showed that the proposed algorithm provides better visual quality than previous works, and a consistent power-saving ratio without the flickering artifact even for video sequences. Specifically, the proposed algorithm provides at maximum 36% and on average 13% higher edge-preserving ratios than the state-of-the-art algorithm (i.e., PCCE [11]). In addition, we proved the possibility of real-time processing by accomplishing an entire execution time of 9 ms per 1080p image.

References

[1] J. Jang, S. Lee, and M. Oh, "Technology development and production of flat panel displays in Korea," *IEEE Proc. J., Mag.*, vol. 90, no. 4, pp. 501–513, Apr. 2002.

[2] K. Suzuki, "Past and future technologies of information displays," in *Proc. IEEE IEDM*, Dec. 2005, pp. 16–21.

[3] B. Young, "OLEDs—Promises, myths, and TVs," *Inform. Display*, vol. 25, no. 9, pp. 14–17, Sep. 2009.

[4] H. D. Kim, H. J. Chung, B. H. Berkeley, and S. S. Kim, "Emerging technologies for the commercialization of AMOLED TVs," *Inf. Display*, vol. 25, no. 9, pp. 18–22, Sep. 2009.

[5] W.-C. Cheng and M. Pedram, "Power minimization in a backlit TFT-LCD display by concurrent brightness and contrast scaling," *IEEE Trans. Consum. Electron.*, vol. 50, no. 1, pp. 25–32, Feb. 2004.

[6] P. Greef and H. G. Hulze, "Adaptive dimming and boosting backlight for LCD-TV systems," in *SID Symp. Dig. Tech. Papers*, May 2007, vol. 38, no. 1, pp. 1332–1335.

[7] L. Kerofsky and S. Daly, "Distinguished paper: Brightness preservation for LCD backlight reduction," in *SID Symp. Dig. Tech. Papers*, Jun. 2006, vol. 37, no. 1, pp. 1242–124.

[8] C.-C. Lai and C.-C. Tsai, "Backlight power reduction and image contrast enhancement using adaptive dimming for global backlight applications," *IEEE Trans. Consum. Electron.*, vol. 54, no. 2, pp. 669–674, May 2008.

[9] S. I. Cho, S.-J. Kang, and Y. H. Kim, "Image quality-aware backlight dimming with color and detail enhancement techniques," *IEEE J. Display Technol.*, vol. 9, no. 2, pp. 112–121, Feb. 2013.

[10] 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. 9, pp. 574–583, Apr. 2009.

[11] C. Lee, C. Lee, Y.-Y. Lee, and C.-S. Kim, "Power-constrained contrast enhancement for emissive displays based on histogram equalization," *IEEE Trans. Image Process.*, vol. 21, no. 1, pp. 80–93, Jan. 2012.

[12] C. Lee, C. Lee, and C.-S. Kim, "Power-constrained contrast enhancement for OLED displays based on histogram equalization," in *Proc. IEEE ICIP*, Sep. 2010, pp. 1689–1692.

[13] E. H. Land and J. McCann, "Lightness and retinex theory," *J. Opt. Soci. Amer.*, vol. 61, no. 1, pp. 1–11, Jan. 1971.

[14] D. J. Jobson, Z.-U. Rahman, and G. A. Woodell, "Properties and performance of a center/surround retinex," *IEEE Trans. Image Process.*, vol. 6, no. 3, pp. 451–462, Mar. 1997.

[15] D. J. Jobson, Z.-U. Rahman, and G. A. Woodell, "A multiscale retinex for bridging the gap between color images and the human observation of scenes," *IEEE Trans. Image Process.*, vol. 6, no. 7, pp. 965–976, Jul. 1997.

[16] J. H. Jang, B. Choi, S. D. Kim, and J. B. Ra, "Sub-band decomposed multiscale retinex with space varying gain," in *Proc. IEEE ICIP*, Oct. 2008, pp. 3168–3171.

[17] T.-C. Jen and S.-J. Wang, "Bayesian structure-preserving image contrast enhancement and its simplification," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 22, no. 6, pp. 831–843, Jun. 2012.

[18] C.-H. Lee, L.-H. Chen, and W.-K. Wang, "Image contrast enhancement using classified virtual exposure image fusion," *IEEE Trans. Consum. Electron.*, vol. 58, no. 4, pp. 1253–1261, Nov. 2012.

[19] S. C. Nercessian, K. A. Panetta, and S. S. Agaian, "Non-linear direct multi-scale image enhancement based on the luminance and contrast masking characteristics of the human visual system," *IEEE Trans. Image Process.*, vol. 22, no. 9, pp. 3549–3561, Sep. 2013.

[20] R. Chouhan, R. K. Jha, and P. K. Biswas, "Enhancement of dark and low-contrast images using dynamic stochastic resonance," *IET Image Process.*, vol. 7, no. 2, pp. 174–184, Sep. 2013.

[21] M. Dong, Y.-S. K. Choi, and L. Zhong, "Power modeling of graphical user interfaces on OLED displays," in *Proc. ACM/IEEE DAC*, Jul. 2009, pp. 652–657.

[22] J. Chalidabhongse and C.-C. J. Kuo, "Fast motion vector estimation using multiresolution-spatio-temporal correlations," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 7, no. 3, pp. 477–488, Jun. 1997.

[23] C. E. Duchon, "Lanczos filtering in one and two dimensions," *J. Appl. Meteorol.*, vol. 18, no. 8, pp. 1016–1022, Aug. 1979.

[24] C. Christine and T. Fleiss, "A study of efficiency and accuracy in the transformation from RGB to CIELAB color space," *IEEE Trans. Image Process.*, vol. 6, no. 7, pp. 1046–1048, Jul. 1997.

[25] S. S. Agaian, B. Silver, and K. A. Panetta, "Transform coefficient histogram-based image enhancement algorithms using contrast entropy," *IEEE Trans. Image Process.*, vol. 16, no. 3, pp. 741–758, Mar. 2007.

[26] L. Fang and O. C. Au, "Subpixel-based image down-sampling with min-max directional error for stripe display," *IEEE J. Sel. Topics Signal Process.*, vol. 5, no. 2, pp. 240–251, Apr. 2011.

[27] H. Yehaneh and Z. Wang, "Objective quality assessment of tone-mapped images," *IEEE Trans. Image Process.*, vol. 22, no. 2, pp. 657–667, Feb. 2013.

Yeon-Oh Nam received the B.S. and M.S. degrees in electronic engineering from Inha University, Incheon, Korea, in 2012 and 2014, respectively. His research interests include image processing and video coding.

Dong-Yoon Choi received the B.S. degree in electronic engineering from Inha University, Incheon, Korea, in 2014, where he is currently pursuing the M.S. degree in electronic engineering. His research interests include image processing and superresolution.

Byung Cheol Song received the B.S., M.S., and Ph.D. degrees in electrical engineering from the Korea Advanced Institute of Science and Technology (KAIST), Daejeon, Korea, in 1994, 1996, and 2001, respectively. From 2001 to 2008, he was a Senior Engineer at Digital Media Research and Development Center, Samsung Electronics Company, Ltd., Suwon, Korea. In 2008, he joined the Department of Electronic Engineering at Inha University, Incheon, Korea, where he is currently an Associate Professor. His research interests are in the general areas of video/image coding, superresolution, deblur, image stabilization, deweathering, computer vision, multimedia system design, and data mining.