POWER-CONSTRAINED CONTRAST ENHANCEMENT ALGORITHM USING MULTISCALE RETINEX FOR OLED DISPLAY

GRANDHI GOPINATH¹, G. RAMANA REDDY²

¹PG Scholar, Dept. of Electronics and Communication Engineering, Nalanda Institute of Engineering & Technology, AP, India, E-mail: gopisiri.col@gmail.com
²Head of the Department, Dept. of Electronics and Communication Engineering, Nalanda Institute of Engineering & Technology, AP, India, E-mail: gujjula.ramanareddy@gmail.com

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.

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. This paper proposes a power-constrained contrast enhancement algorithm using a sub-band decomposed MSR (SD-MSR) [3] for OLED display. First, we designed a modified log function for dynamic power saving. Second 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. Experimental results showed that the proposed algorithm provides better visual quality than previous works, and a consistent power-saving ratio without the flickering artifacts 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 [4]). In addition, we proved the possibility of real-time processing by accomplishing an entire execution time of 9 ms per 1080p image.

II. DIGITAL IMAGE PROCESSING (DIP) BACKGROUND:

Digital image processing is an area characterized by the need for extensive experimental work to establish the viability of proposed solutions to a given problem. An important characteristic underlying the design of image processing systems is the significant level of testing & experimentation that normally is required before arriving at an acceptable solution. This characteristic implies that the ability to formulate approaches & quickly prototype candidate solutions generally plays a major role in reducing the cost & time required to arrive at a viable system implementation.

What is DIP?

An image may be defined as a two-dimensional function f(x, y), where x & y are spatial coordinates, & the amplitude of f at any
pair of coordinates (x, y) is called the intensity or gray level of the image at that point. When x, y & the amplitude values of f are all finite discrete quantities, we call the image a digital image. The field of DIP refers to processing digital image by means of digital computer. Digital image is composed of a finite number of elements, each of which has a particular location & value. The elements are called pixels.

Vision is the most advanced of our sensor, so it is not surprising that image play the single most important role in human perception. However, unlike humans, who are limited to the visual band of the EM spectrum imaging machines cover almost the entire EM spectrum, ranging from gamma to radio waves. They can operate also on images generated by sources that humans are not accustomed to associating with image.

There is no general agreement among authors regarding where image processing stops & other related areas such as image analysis & computer vision start. Sometimes a distinction is made by defining image processing as a discipline in which both the input & output at a process are images. This is limiting & somewhat artificial boundary. The area of image analysis (image understanding) is in between image processing & computer vision.

There are no clear-cut boundaries in the continuum from image processing at one end to complete vision at the other. However, one useful paradigm is to consider three types of computerized processes in this continuum: low-, mid-, & high-level processes. Low-level process involves primitive operations such as image processing to reduce noise, contrast enhancement & image sharpening. A low- level process is characterized by the fact that both its inputs & outputs are images. Mid-level process on images involves tasks such as segmentation, description of that object to reduce them to a form suitable for computer processing & classification of individual objects. A mid-level process is characterized by the fact that its inputs generally are images but its outputs are attributes extracted from those images. Finally higher- level processing involves “Making sense” of an ensemble of recognized objects, as in image analysis & at the far end of the continuum performing the cognitive functions normally associated with human vision.

Digital image processing, as already defined is used successfully in a broad range of areas of exceptional social & economic value.

**What is an image?**

An image is represented as a two dimensional function f(x, y) where x and y are spatial coordinates and the amplitude of f at any pair of coordinates (x, y) is called the intensity of the image at that point.

**Gray scale image:**

A grayscale image is a function I(xylem) of the two spatial coordinates of the image plane. I(x, y) is the intensity of the image at the point (x, y) on the image plane.

I (xylem) takes non-negative values assume the image is bounded by a rectangle [0, a] \times [0, b] \rightarrow [0, a] \times [0, b] \rightarrow [0, \text{info}]

**Color image:**

It can be represented by three functions, R (xylem) for red, G (xylem) for green and B (xylem) for blue.

An image may be continuous with respect to the x and y coordinates and also in amplitude. Converting such an image to digital form requires that the coordinates as well as the amplitude to be digitized. Digitizing the coordinate’s values is called sampling. Digitizing the amplitude values is called quantization.

**Coordinate convention:**

The result of sampling and quantization is a matrix of real numbers. We use two principal ways to represent digital images. Assume that an image f(x, y) is sampled so that the resulting image has M rows and N columns. We say that the image is of size M X N. The values of the coordinates (xylem) are discrete quantities. For notational clarity and convenience, we use integer values for these discrete coordinates. In many image processing books, the image origin is defined to be at (xylem)=(0,0). The next coordinate values along the first row of the image are (xylem)=(0,1). It is important to keep in mind that the notation (0,1) is used to signify the second sample along the first row. It does not mean that these are the actual values of physical coordinates when the image was sampled. Following figure shows the coordinate convention. Note that x ranges from 0 to M-1 and y from 0 to N-1 in integer increments.

The coordinate convention used in the toolbox to denote arrays is different from the preceding paragraph in two minor ways. First, instead of using (xylem) the toolbox uses the notation (race) to indicate rows and columns. Note, however, that the order of coordinates is the same as the order discussed in the previous paragraph, in the sense that the first element of a coordinate tuples, (alb), refers to a row and the second to a column. The other difference is that the origin of the coordinate system is at (r, c) = (1, 1); thus, r ranges from 1 to M and c from 1 to N in integer increments. IPT documentation refers to the coordinates. Less frequently the toolbox also employs another coordinate convention called spatial coordinates which uses x to refer to columns and y to refers to rows. This is the opposite of our use of variables x and y.

**III. PROPOSED ALGORITHM**
We propose a power controllable contrast enhancement algorithm for OLED display based on SD-MSR. The first stage coarsely reduces the power of an input 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.

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.

IV. PCCE FOR VIDEO SEQUENCES
We extend the proposed PCCE algorithm to enhance video sequences. The proposed algorithm provides a power-reduced output image using the power-control parameter \( \beta \). We can apply the proposed algorithm with fixed \( \beta \) to each frame in a video sequence. However, a typical video sequence is composed of frames with fluctuating brightness levels. Experiments in Section V-B will show that a bright frame can be enhanced with large \( \beta \) to save power aggressively, whereas a dark frame can be severely degraded if its overall brightness is reduced further with the same \( \beta \). Therefore, we develop a scheme that determines \( \beta \) adaptively according to the brightness level of each frame.

For each frame, we first set the target power consumption

\[
TDP_{\text{out}} = \sum_{k=0}^{L-1} h_k \cdot \beta^k
\]

TDP based on the input power consumption TDP and then control parameter \( TDP_{\text{out}} \) to achieve TDP. Specifically, we set

\[
TDP_{\text{out}} = \kappa \cdot TDP_{\text{in}}
\]

Where \( \kappa \) is the power-reduction ratio. When \( \kappa = 1 \), the proposed algorithm saves no power during the contrast enhancement.

On the other hand, when \( \kappa \) is smaller, the proposed algorithm darkens the output frame and decreases the power consumption. The power model in Section III-A indicates that a bright frame consumes more power than a dark frame. Therefore, more power saving can be achieved for a brighter frame, and the power-reduction ratio \( \kappa \) in (32) can be set to a smaller value. On the other hand, the ratio for a dark frame should be close to 1 since even a small power reduction may yield poor image quality by reducing the contrast further and erasing details. Based on these observations, we set the power-reduction ratio \( \kappa \) by

\[
\kappa = \left( 1 - \frac{\sum Y}{L - 1} \right)^{\rho}
\]

Where \( \bar{Y} \) denotes the average gray level of an input frame and \( \rho \) is a user-controllable parameter. For a bright input frame with high \( \bar{Y} \), is set to a small value to achieve aggressive power saving. On the contrary, for a dark input frame with low \( \bar{Y} \), \( \kappa \) is set to be close to 1 to avoid the brightness reduction. To summarize, given an input frame, we determine the
target power consumption $TDP_{\text{min}}$ using (32) and (33). Then, we find parameter $\beta$ to achieve $TDP_{\text{min}}$. Since $TDP_{\text{min}}$ is inversely proportional to $\beta$, we can easily obtain the desired $\beta$ using the bisection method [27], which iteratively halves a candidate range of the solution into two subdivisions and selects the subdivision containing the solution. Thus, in the video enhancement $\beta$, is automatically determined, and the only power-control parameter is in $\rho$ (33). Note that, for the same $\overline{\rho}$, larger $\rho$ yields smaller $K$ and saves more power.

V. EXPERIMENTAL RESULTS

We evaluate the performance of the proposed algorithm on ten test images, i.e., “Door,” “Moon,” “Pagoda,” “Beach,” “Sunset,” “Ivy,” “Baboon,” “Lena,” “F-16,” and “Eiffel Tower.” These test images are shown in Figs. 1, 4, and 10. “Beach” and “Door” are from Kodak Lossless True Color Image Suite.1 “Baboon,” “Lena,” and “F-16” are from the USC-SIPI database.2 and the others are taken with a commercial digital camera and resized. The resolution of “Eiffel Tower” is $480 \times 720$, those of the USC-SIPI images are $512 \times 512$, and those of the others are $720 \times 480$. We process only the luminance components in the experiments. More specifically, given a color image, we convert it to the YUV color space and then process only the Y-component without modifying the U- and V-components. Therefore, the TDP is also measured for the Y-component only using (14). In all experiments, $\overline{\rho}$ is set to 2.2.

A. Contrast Enhancement without Power Constraint:

First, we compare the proposed PCCE algorithm without the power constraint ($\beta = 0$) with the conventional HE and HM techniques. Fig. 4 shows the processed images obtained by the conventional HE algorithm, the weighted approximated HE (WAHE) algorithm [17], and the proposed PCCE algorithm ($\beta = 0$). The proposed algorithm is tested in two ways. In Fig. 4(d), the user-controllable parameter $\mu$ for LHM in (10) is set to 2, 6.5, 5.5, 6.5, 5, 5.5, 5, and 5 for the eight test images, respectively, to achieve the best subjective qualities. On the Other hand, in Fig. 4(e), $\mu$ is fixed to 5. For the WAHE results in Fig. 4(c), parameter $g$ is adapted for each image to achieve the best subjective quality. Fig. 5 shows the transformation Functions, which are used to obtain the images in Fig. 4. 1http://r0k.us/graphics/Kodak/2http://sipi.usc.edu/database/

We observe from Fig. 4(b) that the conventional HE algorithm causes excessive contrast stretching. In the “Moon” image, hidden noises become visible, degrading the image quality severely. This noise amplification is due to the steep slope of the transformation function near intensity 0, as shown in Fig. 5. The contrast overstretching suppresses the overall brightness of the “Beach” image. The transformation function reduces the input-pixel range [0, 150] to the output-pixel range [0, 50] by extending the contrast around the input-pixel intensity 170, which corresponds to the background area. Also, contour artifacts are observed in “Sunset.” In general, the conventional including amplified noises, contour artifacts, detail losses, and mood alteration.

Compared with the conventional HE, both WAHE and the proposed algorithm reduce artifacts by alleviating abrupt changes in the transformation functions, as shown in Fig. 4(c) and (d). WAHE exploits spatial variation information to reduce large histogram values, based on the observation that peaks in histograms usually come from background regions.

Specifically, WAHE skips repeated pixel intensities during the construction of an input histogram to focus on the contrast enhancement of textured regions. Thus, it can enhance object details, whereas it may degrade background details. For example, on the “Pagoda” image, WAHE improves the contrast of the tower but loses the details in the clouds. Similarly, since the wall in the “Ivy” image has small intensity variations, its contrast is not enhanced by WAHE significantly.

The proposed PCCE algorithm provides comparable or better results than WAHE on all test images, as shown in Fig. 4(d). On the “Moon,” “Beach,” “Sunset,” “Baboon,” “Lena,” and “F-16” images, the proposed algorithm and WAHE produce similar results. However, on the “Pagoda” and “Ivy” images, the proposed algorithm yields better perceptual quality than WAHE. Notice that the proposed algorithm enhances the clouds in “Pagoda” and the patterns on the wall in “Ivy” more clearly. In Fig. 4(e), we fix the LHM parameter to 5. Except for slight differences in the “Pagoda” image, the output images with the fixed $\mu$ are almost indiscernible from those with the adapted values in Fig. 4(d). Experiments on various other images also confirm that $\mu = 5$ is a reliable choice. Therefore, in the following experiments, is set to 5 unless otherwise specified.
B. Contrast Enhancement with Power Constraint:

Next, we evaluate the performance of the proposed PCCE algorithm with the power constraint (\( \beta > 0 \)). Fig. 6 shows the output images obtained by the proposed algorithm at different values. The images in Fig. 6(a) are exactly the same as those in Fig. 4(e) \( \beta \). As \( \beta \) gets larger, the overall brightness of the output images decreases, but the image contrast is relatively well preserved. Note that the perceptual quality and the subjective contrast of the output images at \( \beta = 0.5 \) are almost the same as those at \( \beta = 0 \). In particular, when these images are displayed on OLED panels, it is hard to distinguish the case without the power constraint \( \beta = 0 \) from the case with the power constraint \( \beta > 0 \) unless \( \beta \) is set to be very high. Fig. 6(e) shows the output images when \( \beta \) has a very high value of 15. Even in this case, the originally bright images “Ivy” and “F-16” retain visual details partly, but the other relatively dark images are severely degraded. In general, \( \beta \), can be set to a higher number for a brighter image to save power more aggressively. On the other hand, for a dark input image, \( \beta \) should be less than 2 for the proposed algorithm to yield good image quality. Fig. 7 shows how the transformation functions vary according to \( \beta \). As \( \beta \) gets larger, the proposed algorithm lowers the transformation functions to save more power, but it preserves the slopes of the functions (or, equivalently, the contrast) for input pixel values with large histogram values. However, as \( \beta \) gets larger, the proposed algorithm inevitably reduces the contrast for infrequent input-pixel values. For example, “Pagoda” has low histogram values for input-pixel values around 90. Thus, at \( \beta = 3 \), the transformation function becomes flat near that pixel.

Values. Fig. 8 compares the TDP measurements for the images in Figs. 4 and 6. For the dark “Moon” image, all three contrast-enhancement methods HE, WAHE, and the proposed algorithm \( \left( \beta = 0 \right) \) increase pixel values to stretch the image contrast, require higher TDPs than the original input images. However, the proposed algorithm can reduce TDPs by increasing parameter \( \beta \). Moreover, for brighter images, such as “Beach” and “Ivy,” the proposed algorithm \( \beta = 1.5 \) can reduce the power consumption more significantly while improving the overall contrast. For instance, on the “Ivy” image, the proposed algorithm at \( \beta = 1.5 \) reduces the TDP by more than 70%, as compared with the input image, but it still improves the contrast.

Fig. 9 compares the outputs of the proposed algorithm at \( \kappa = \frac{\text{TDP}_{\text{out}}}{\text{TDP}_{\text{in}}} \) with those of the linear mapping method. Let us recall that the power-reduction ratio is defined as TDP in (32). The linear mapping method uses a linear transformation function \( \mathcal{T}_k = C \cdot k \cdot k \), where constant is set for each image in such a way that the method achieves the same as the proposed algorithm. Whereas the linear mapping method provides dull output images due to the reduced dynamic ranges, the proposed algorithm provides significantly better image contrast and perceptual quality. An exception is the “Sunset” image, on which the proposed algorithm sacrifices the details in the mountain region to improve the contrast in the sky region. In this test \( \left( \beta = 1.5 \right) \), the mean and the variance of the power-reduction ratios for the eight test images are 0.36 and 0.009, respectively. At \( \beta = 3 \), the mean becomes.
0.26, and the variance becomes 0.015. At $\beta = 15$, the mean is 0.14, and the variance is 0.010.

As discussed in the last section, is directly related to the power consumption. However, the LHM parameter also affects the power consumption since it influences the transformation function, as illustrated in Fig. 1. In Figs. 10 and 11, we show the output images and the power-reduction ratios for various combinations of and . In both Figs. 10 and 11, it can be observed that, for fixed , the TDP consistently decreases as gets larger. On the contrary, the effects of on are inconsistent, depending on the characteristics of the input images. Larger modifies the input histograms less strongly and overstretches the contrast. Because of the contrast overstretching, larger increases the TDP on the dark “Eiffel Tower” image but decreases the TDP on the bright “F-16” image. These inconsistent effects make less suitable for the power control. The LHM parameter controls the level of contrast enhancement, but larger does not always provide better subjective quality. In the extreme case $\mu = \infty$, the histogram is not modified at all, and LHM becomes the conventional HE algorithm, which has several drawbacks. In Section V-A, we showed that, when is fixed to 5, the proposed algorithm without the power constraint suppresses the drawbacks of the conventional HE and provides good image quality reliably. Similarly, Figs. 10 and 11 show that case $\mu = 5$, enclosed by the solid rectangle, yields satisfactory image quality for various values. In other words, each image within the rectangle provides comparable or better quality than the images outside the rectangle with similar power reduction ratios. An improper value of may yield undesirable artifacts in the output image. Therefore, we suggest fixing $\mu$ to 5 and varying only $\beta$ to control the power consumption.

Next, we enhance video sequences using the algorithm in Section IV. Two video clips from the movies “Avatar” and “The Shaw shank Redemption” are employed as test sequences, and each clip consists of 700 frames. In the video enhancement, the power consumption is affected by the LHM parameter and the power-control parameter $\rho$ in (33). However, as mentioned in the last section, is not suitable for the power control. Therefore, we fix $\mu$ to 5 and vary only $\rho$ to control the power consumption. Figs. 12 and 13 compare the TDPs of input and output frames. They also show selected frames. “Adaptive” denotes the proposed algorithm, and “Static” means the static method that maintains a constant output TDP regardless of an input TDP. Let us first compare the proposed algorithm at $\rho = 0.5$ with the static method. The constant output TDP of the static method is set to be equal to the average TDP of the proposed algorithm at $\rho = 0.5$ over all frames. The proposed algorithm reduces more power for brighter input frames adaptively, whereas the static method fixes the output power and thus even increases power for some dark input frames. We see that the proposed algorithm provides better perceptual image quality. For bright input frames, e.g., the 200th frame in Fig. 12 and the 693rd frame in Fig. 13, the proposed algorithm
reduces the power consumption by 26.8% and 11.8%, respectively, without decreasing the image quality. On the contrary, the static method darkens those frames too much and hides the details. For dark input frames, the proposed algorithm decreases the power consumption slightly, whereas the static method increases the power consumption. For instance, on the 135th frame in Fig. 12, the static method consumes TDP about twice higher than the input frame but improves the image contrast only marginally.

In Figs. 12 and 13, we also see that the proposed algorithm saves more power, as parameter gets larger. On average, when \( \rho \) is set to 0.5, 1.0, and 1.5, the proposed algorithm reduces the power consumption by 19.3%, 34.7%, and 46.9% for “Avatar” and by 21.2%, 36.3%, and 47.4% for “Shaw shank Redemption,” respectively. The proposed algorithm saves more power for a brighter input frame, whereas it attempts to avoid the brightness reduction for a darker frame. Thus, although the proposed algorithm reduces the average power consumption significantly, it provides good subjective image quality by exploiting the characteristics of input frames.

### Table I

<table>
<thead>
<tr>
<th></th>
<th># of bisection iterations</th>
<th># of variable changes</th>
<th># of secant iterations</th>
<th>Processing time</th>
</tr>
</thead>
<tbody>
<tr>
<td>Still image</td>
<td>-</td>
<td>16.9</td>
<td>3.84</td>
<td>6.23 ms</td>
</tr>
<tr>
<td>Video frame</td>
<td>9.34</td>
<td>3.37</td>
<td>2.85</td>
<td>13.12 ms</td>
</tr>
</tbody>
</table>

Table I summarizes the computational complexity, which is required for the proposed
PCCE algorithm to process a still image or a video frame. It lists the average performance over all test images in Figs. 1, 4, and 10, as well as the average performance over all frames in the two video sequences in Figs. 12 and 13. We use a personal computer with a 3.3-GHz central processing unit for this test. The proposed algorithm is implemented in C but not optimized. In the still image processing, the secant formula in (27) is iteratively applied to find a solution to . The average number of secant iterations is about 3.84. As mentioned in the Appendix, if solution is less than or equal to , we change from to . The average number of variable changes is 16.9. The proposed algorithm takes only 6.23 ms to enhance a still image on the average. In the video enhancement, for each frame, to find that produces a target , the proposed algorithm uses the bisection processing time for a video frame is longer than that for a still image. However, both secant and bisection iterations are performed with vector , the dimension of which is just 256. Therefore, even our software implementation takes only 15.12ms to process a video frame on the average. Moreover, the PCCE algorithm can be efficiently implemented on hardware such as field-programmable gate arrays.

CONCLUSION

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.

BIBLIOGRAPHY