ONE SCAN SHADOW COMPENSATION AND VISUAL ENHANCEMENT OF COLOR IMAGES

Felix Albu, Constantin Vertan, Corneliu Florea, Alexandru Drimbarean

Tessera Romania SRL

ABSTRACT

The paper presents a new technique of efficient dynamic range compression and shadow compensation for still color images. The proposed method enhances low light areas while preserving the colors and details, without generating visual artifacts. The approach is based on recursive filtering and contrast stretching techniques, driven by statistical measures of the image and implemented under a logarithmic image processing model. The implementation can be used for any image represented in the RGB or YCbCr color spaces.

Index Terms—dynamic range compression, shadow compensation, recursive filtering, logarithmic model, color image processing

1. INTRODUCTION

Most real-world images seen by the human eye have a very wide dynamic range. Typically, digital camera sensors can capture a lower value dynamic range (in the order of 256), while the human eye, according to the experiments lead by Blackwell [1] can distinguish up to 10000. Therefore, the imaging devices need to use some processing algorithms, called dynamic range compression (DRC) techniques in order to obtain a suitable display range of intensities (see [2] – [8] and the references therein). Although they compress the dynamic range, they also introduce some undesired artifacts such as de-saturated colors, halos, etc. The techniques can be used as a post-processing step for photos with exposure problems or significant darker areas hiding important details; other possible applications include medical imaging (such as Computed Tomography or Nuclear Magnetic Resonance), surveillance systems and consumer appliances.

The majority of these techniques use global contrast and brightness adjustment. These include point transforms such as automatic gain/offset, non-linear gamma correction, logarithmic transform, and global transforms such as histogram equalization etc [5]. The gain/offset correction is a linear method and can produce artifacts due to saturation and clipping. The non-linear gamma correction or logarithmic correction highlights some details from darker areas, but destroy information stored by brighter areas as

well [4]. The histogram equalization is a global technique that works well for a wide variety of images, but it fails on images that contain very dark and very bright regions.

Other popular techniques for image enhancement are based on Retinex methods (e.g. [2], [5] – [7] and the references therein). The primary aim is to decompose a given image into a reflectance image and an illumination image. These are process differently and the output image is a combination of them. The Retinex-based methods can produce sometimes images with de-saturated colors, or halos [2], [5].

Recently, a DRC method of combining gamma corrected images according to weight functions based on local variances has been proposed in [8]. The local variances have to be computed for both gamma corrected pixels and therefore, the complexity of the method is rather high.

Our proposed technique enhances the shadow regions using adaptive, pixel-specific color component amplification factors, which are computed according to the local contrast and pre-computed setup parameters. The original global contrast is increased in the case of low-contrast, shadowed images and kept at near constant values in the remainder of the cases. The amplification is performed at pixel-level, using a non-linear, perceptual-resembling logarithmic multiplication. Unlike most of the aforementioned techniques, the proposed approach can be implemented as a one-pass image processing operation, using a specially-designed recursive filter. The color processing approach uses the common paradigm of marginal (independent) color component processing, being suited for any intensity-based color representation spaces (such as RGB or YCbCr). Thus, the processing algorithm is based on three steps: an optional white-balance compensation method (used for the RGB version), the core shadow correction method and the final pixel amplification.

The remainder of the paper is organized as follows: Section 2 presents the details of the proposed method. The experimental results are shown in Section 3. Section 4 concludes the paper.

2. THE PROPOSED METHOD

In the following, we will describe the steps of the proposed algorithm within the RGB color representation. Let us denote by **I** the original (possibly under-exposed, with shadow areas) color image. The color components of a

given pixel located at coordinates (i, j) are red, $\mathbf{I}(i, j, 1)$ green, $\mathbf{I}(i, j, 2)$ and blue, $\mathbf{I}(i, j, 3)$. The three mentioned steps of the algorithm, namely the optional white balance compensation for the RGB version, the core processing and the pixel amplification are detailed in below.

2.1 White Balance Compensation

This version of white balance adapts the local image content and is based on the global intensities of the color components and uses the green component as reference. A running average on each color channel is computed using a recursive filter. Thus, for each pixel, located at coordinates (i, j) we compute:

$$
\begin{aligned}
\overline{R}(i, j) &= \beta \overline{R}(i, j-1) + (1-\beta) \mathbf{I}(i, j, 1) \\
\overline{G}(i, j) &= \beta \overline{G}(i, j-1) + (1-\beta) \mathbf{I}(i, j, 2) \\
\overline{B}(i, j) &= \beta \overline{B}(i, j-1) + (1-\beta) \mathbf{I}(i, j, 3)
\end{aligned}
\tag{1}
$$

In the equations above β is a tuning coefficient with possible values in the range $[0, 1]$, but values around 1/8 are a good compromise between averaging and signal variations tracking abilities. Based on the local accumulated color components averages, two correction terms, γ_R and γ_B, are computed for each pixel, considering the G color component as reference.

$$
\begin{aligned}
\gamma_R(i, j) &= \frac{\overline{G}(i, j)}{\overline{R}(i, j)} \left(\frac{(1-\alpha)\overline{R}(i, j) + 255\alpha}{(1-\alpha)\overline{G}(i, j) + 255\alpha} \right) \\
\gamma_B(i, j) &= \frac{\overline{G}(i, j)}{\overline{B}(i, j)} \left(\frac{(1-\alpha)\overline{B}(i, j) + 255\alpha}{(1-\alpha)\overline{G}(i, j) + 255\alpha} \right)
\end{aligned}
\tag{2}
$$

The computed values of the correction terms, γ_R and γ_B, are limited to $[0.95 ... 1.05]$ interval, in order to reduce the over-amplification of the red and blue channels if the following shadow correction is applied on the RGB image. The final, modified pixel values $\mathbf{F}(i, j)$ are given by:

$$
\begin{aligned}
\mathbf{F}(i, j, 1) &= \gamma_R \mathbf{I}(i, j, 1) \\
\mathbf{F}(i, j, 2) &= \mathbf{I}(i, j, 2) \\
\mathbf{F}(i, j, 3) &= \gamma_B \mathbf{I}(i, j, 3)
\end{aligned}
\tag{3}
$$

2.2. Core processing: Shadow Correction

Let image $\mathbf{F}$ be the result of the optional white balance compensation procedure described above (obviously, if no compensation is applied, $\mathbf{F} \equiv \mathbf{I}$). The core processing of the proposed method determines adaptively an amplification factor for each image pixel. The amplification is related to

the relative average intensity (or luminance) in the neighborhood of each pixel. Darker regions should be amplified more than the medium-illuminated areas; the very bright areas should be left unchanged. The need for a single image pass together with the use of a one-pass image scan, the use of some average pixel intensity as basis for the value of the amplification factor and the inverse dependence of the amplification factor with the intensity motivates the use of a recursive filter as mean of practical computation of the desired amplification factor. The amplification factor $H(i, j)$ is computed as follows:

$$
H(i, j) = p \cdot H(i, j-1) + (1-p) \left((1-\alpha) + \frac{255\alpha}{Y(i, j)} \right)
\tag{4}
$$

In the equation (4) p is the pole of the recursive filter and α is a positive tuning factor, with $\alpha \in (0, 1)$; the luminance Y of the processed color image $\mathbf{F}$ can be either the YCbCr luminance or the average of the RGB components in case of the RGB version.

The role of the recursive filter is to smooth fast transitions of the $H(i, j)$ coefficients due to its low-pass filtering properties. It is known that the recursive filter is more efficient than the non-recursive filter. A more powerful filtering is applied if the pole value is increased. However, if the pole value exceeds the value of 0.5 some visible artifacts can appear in pictures with high contrast areas.

Darker areas are amplified more than the illuminated areas due to the inverse values recursive averaging (the factor $\frac{255\alpha}{Y(i, j)}$ from the second term of equation (4)). Therefore, an automatic correction of uneven luminance in the foreground/background is obtained. If higher values for α are used, the $H(i, j)$ coefficients will have increased values in the darker areas. Therefore, different DRC strength levels can be obtained by varying the parameters of (4). However, if $\alpha > 1/4$ the noise can become visible in low quality pictures and some haziness could affect very bright pictures.

2.3. Pixel Amplification

The final, modified pixel values, $\mathbf{F_{out}}(i, j)$, are given by:

$$
\begin{aligned}
\mathbf{F_{out}}(i, j, 1) &= H(i, j) \otimes \mathbf{F}(i, j, 1) \\
\mathbf{F_{out}}(i, j, 2) &= H(i, j) \otimes \mathbf{F}(i, j, 2) \\
\mathbf{F_{out}}(i, j, 3) &= H(i, j) \otimes \mathbf{F}(i, j, 3)
\end{aligned}
\tag{5}
$$

The multiplications $\otimes$ used in equations (5) above can be either the usual real-number multiplications or can be replaced by logarithmic scalar multiplications, in order to

avoid possible saturation problems (and the clipping of the color components).

The use of a logarithmic image processing (LIP) model [9] ensures range preservation of the image components due to its closing property. LIP models were successfully used in different image processing application such as image enhancement [9], edge detection, image stabilization [10], etc. We will further use the classical LIP scalar multiplication proposed in [9].

Hence the output pixel value is given by:

$$\mathbf{F_{out}}(i, j, k) = M - M\left(1 - \frac{\mathbf{F}(i, j, k)}{M}\right)^{H(i,j)} \quad (6)$$

In the equation above, M is the maximal value allowed for the luminance and color component ($M = 255$ for the RGB color model and Y plane, $M = 128$ for the Cb and Cr color components) and $k = 1,2,3$ is the index of the color components. The pixel amplification part can be efficiently implemented using LUT's. Figure 1a shows the logarithmic curves for the luminance or RGB channels, while figure 1b shows the logarithmic curves for the chrominance channels. It can be seen that the maximum absolute values in equation 6 are limited by $M = 255$ and respectively by $M = 128$. The color and brightness boost can be obtained by varying the pole value p; therefore, for each pixel, depending on computed $H(i, j)$ coefficients, an individual curve is chosen for luminance and respectively for chrominance channels.
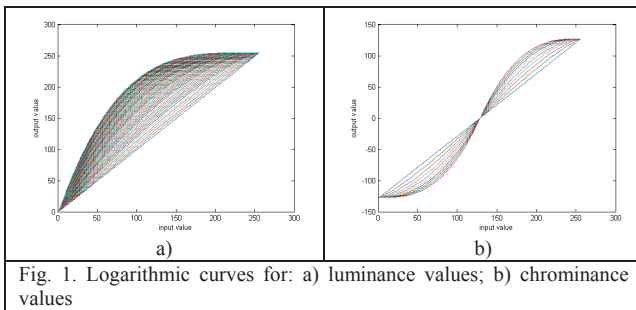

Fig. 1. Logarithmic curves for: a) luminance values; b) chrominance values

3. EXPERIMENTAL RESULTS

In this section we shall present some simulation results in order to demonstrate the performance of the proposed algorithm. The following parameters are used as standard $\alpha = p = \beta = 1/8$. We shall illustrate the method on the original image shown in Fig. 2. Figure 3 shows the scaled computed coefficients for various combinations of the p and α parameters of the recursive filter in (4). It can be noticed that they are adapted to each pixel depending on the local image content. They are close to one in bright areas and tend to increase to higher values in darker areas. The recursive filter smoothes the transition from one pixel to

another. Therefore, the artifacts are greatly reduced and no halo effects appear in the enhanced image.

The RGB and YCbCr enhanced images are shown in Figs. 4a and 4b, respectively. The Figs. 5a and 5b show their corresponding histograms. It can be noticed that both highlighted and darker areas are enhanced, and especially the darker area details become noticeable. As comparison, the combined gamma corrected enhanced picture [8] and its corresponding histogram are shown in Figs. 5c and 6c respectively. Also, the Retinex based result [7] and its corresponding histogram are shown in Figs. 5d and 6d respectively.
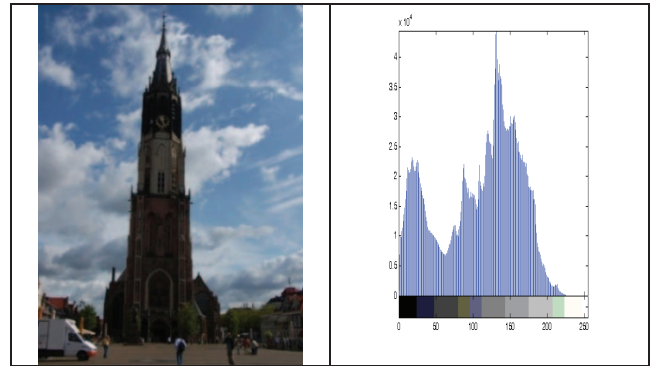

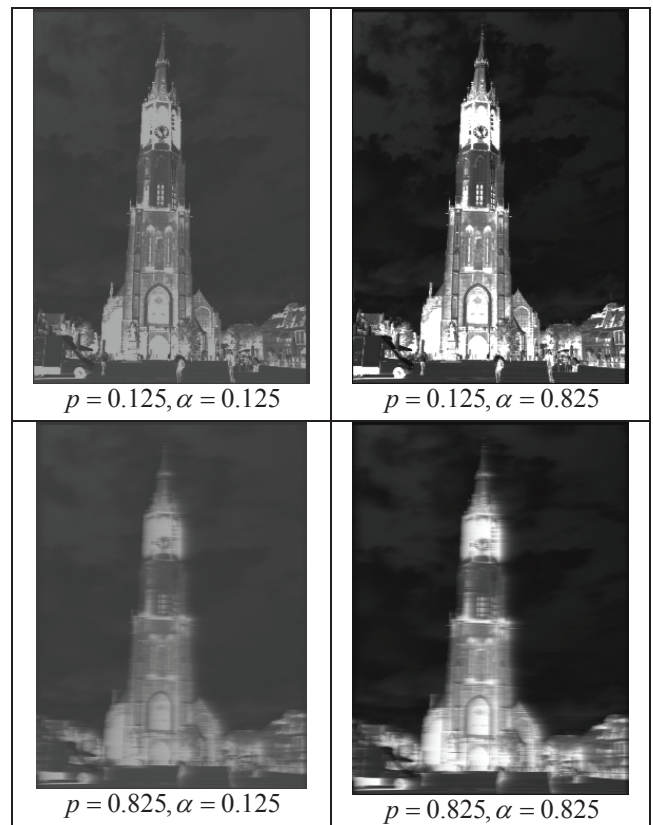**Fig. 2.** Original color image and its corresponding histogram.

Fig. 3. H image map amplification variation with respect to the filter pole p and α

3135

Fig. 4. The enhanced images by: a) proposed RGB version; b) proposed YCC version; c) method from images [8]; d) Retinex based method [7]
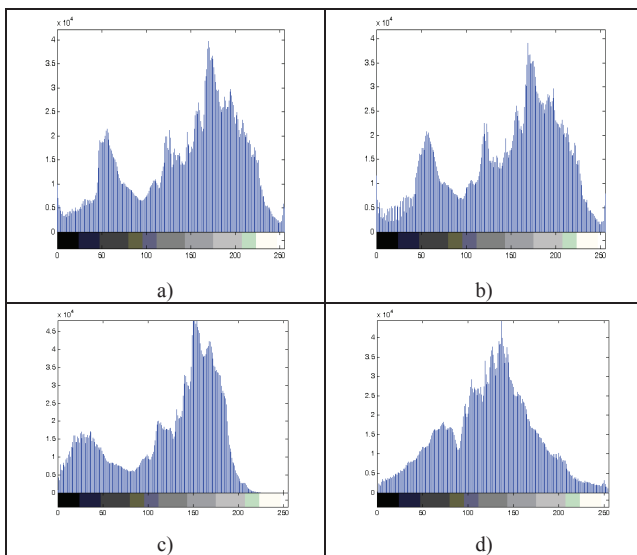

Fig. 5. The histograms of the enhanced images by: a) proposed RGB version; b) proposed YCC version; c) method from [8]; d) Retinex based method [7]

The YCbCr and RGB versions histograms are better spread on all the range, without significant difference towards the higher bins. It can be seen that the enhanced images using the proposed method are looking more pleasant than the combined gamma corrected and Retinex based images. Also, the proposed method can have an efficient implementation by using LUTs.

4. CONCLUSION

This paper has presented an efficient dynamic range compression technique that uses local image information and process the image in a single scan. Therefore it is very suitable for implementation on low power, low memory embedded devices. The core of the method is based on the computation of a pixel amplification factor via a recursive image filtering. Further work will be focused on the adaptation of the recursive filtering (mainly in terms of the choice of the pole p) to image content and to the limitation of the amplification in image areas were the image sensor noise is important.

5. REFERENCES

[1] Blackwell H.R., "Contrast thresholds of the human eye," *Journal of the Optical Society of America*, vol. 36, no. 11,pp. 624–643, November, 1946.

[2] Wanabe, T., Kuwahara, Y., Kojima, A., Kurosawa, T.,"An adaptive multi-scale retinex algorithm realizing high color quality and high-speed processing", *Journal of imaging science and technology*, vol. 49, 2005

[3] Tumblin, J., Rushmeier, H.E., "Tone reproduction for realistic images", *IEEE Computer Graphics and Applications",* vol. 13, no. 6, pp. 42-48, 1993

[4] Capra, A., Castorina, A., Corchs, S., Gasparini, F., and Schettini, R., "Dynamic range optimization by local contrast correction and histogram image analysis", *International Conference on Consumer Electronics,* pp. 309-310, 2006

[5] Rahman, Z., Woodell, G.A., Jobson, D.J., "A comparison of the multiscale Retinex with other image enhancement techniques", *Proceedings of the IS&T 50th anniversary conference*, May 1997

[6] Funt, B., Ciurea, F., and McCann, J., "Retinex in Matlab," *Journal of Electronic Imaging*, vol. 13, no. 1, pp. 48-57, an. 2004

[7] Kimel, R., Elad, M., Shaked, D., Keshet, R., Sobel, I., "A variational framework for Retinex," *International Journal of Computer Vision,* vol. 52, no. 1, pp. 7-23, April 2003

[8] Han, Y.S., and Kang, G.M., "Image Dynamic Compression by Combining Gamma-Corrected Images," *International Technical Conference on Circuits/Systems, Computers and Communications Technical Program (ITC-CSCC 2008)*, Shimonoseki, Japan, July 6-9, 2008

[9] Jourlin, M., and Pinoli, J.-C., "Logarithmic image processing: the mathematical and physical framework for the representation and processing of transmitted images," *Advances in Imaging and Electron Physics*, vol. 115, pp. 129–196, 2001.

[10] Florea, C., Albu, F., Vertan, C., and Drimbarean, A., "Logarithmic tools for in-camera image processing", *Proc. Of 16th IET Irish Signals and Systems Conference ISSCS2008*, pp. 394-399, Galway, Ireland, June 2008