Implementation of Adaptive Shading Correction System Supporting Multi-Resolution for Camera

Jooyoung Ha*, Jingeun Song*, Jeonguk Im*, Kyoungjoong Min**, Bongsoon Kang*
Department of Electronics Engineering, Dong-A University*
SAMSUNG Electro-Mechanics Co. Ltd.**

Abstract

In this paper, we say the shading correction system supporting multi-resolution for camera. The shading effect is caused by non-uniform illumination, non-uniform camera sensitivity, or even dirt and dust on glass (lens) surfaces. In general this shading effect is undesirable [1]. Eliminating it is frequently necessary for subsequent processing and especially when quantitative microscopy is the fine goal. The proposed system is available on thirty nine kinds of image resolutions scanned by interlaced and progressive type. Moreover, the system is using various continuous quadratic equations instead of using the piece-wise linear curve which is composed of multiple line segments. Finally, the system could correct the correct effect without discontinuity in any image resolution.

The proposed system is also experimentally demonstrated with Xilinx Virtex FPGA XCV2000E- 6BG560 and the TV set.

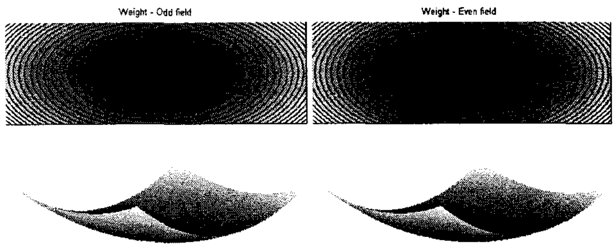
I. Introduction

Virtually all imaging systems produce shading [2]. By this we mean that if the physical input image *a(x,y) = constant*, then the digital version of the image will not be constant. The source of the shading might be outside the camera such as in the scene illumination or the result of the camera itself where a gain and offset might vary from pixel to pixel. The model for shading is given by:

$$c[m,n] = gain[m,n] \cdot a[m,n] + offset[m,n] \qquad (1)$$

where *a[m,n]* is the digital image that would have been recorded if there were no shading in the image, that is, *a[m,n] = constant.*

In some cases the image might be bright in the center and decrease in brightness as one goes to the edge of the field-of-view. In other cases the image might be darker on the left side and lighter on the right side [3-4]. In the problems, the proposed system is concentrating on the former case shown in Fig. 1 because the center shaded image is happened most frequently. Moreover, the proposed shading correction method is using retrospective type which can be applied to any image, because the method only uses the information already present in an image [5].



(a) Shaded image - Odd (b) Shaded image - Even
Fig.1 Center shaded image

Figure 1 explains the shaded image displayed by interlaced method - Fig.1-(a) odd field and Fig.1-(b) even field.

The rest of the paper is organized as follows. In Section II, we show the proposed algorithm for shading correction. Finally, the conclusion is given in Section III.

II. Proposed algorithm

1. Generating Weight to Correct Shading Effect

Figure 2 shows the fundamental theory of shading correction to eliminate the shading effect. As you can see, the system is using the two major terms such as d and d_{max} which are the distance from image center to any input pixel and maximum distance, respectively.

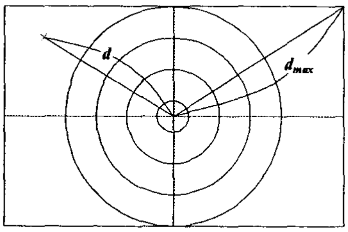


Fig. 2 Fundamental theory of shading correction

The correction weight is decided by using the rate between any input pixel's distance and maximum distance and then, the relationship is shown in Eq. (1).

$$rate = \frac{d}{d_{\max}} \quad (1)$$

Equation (2) explains the process of generating rate shown in Eq. (1). The terms, (x_{center}, y_{center}), are the center of input image and (x, y) are the corresponding pixel position.

$$rate = \frac{\sqrt{(x_{center} - x)^2 + (y_{center} - y)^2}}{d_{\max}} \quad (2)$$

As stated above, the proposed system is available on thirty nine kinds of image resolution shown in Table 1. and image center is different each other along image size.

Table 1 Available image resolution specification

No.	Image size	No.	Image size	No.	Image size
1	352×288	14	2304×1728	27	2460×1836
2	320×240	15	2272×1704	28	2560×1920
3	640×480	16	2048×1536	29	2592×1944
4	800×600	17	1800×1200	30	2576×1932
5	1024×768	18	1900×1600	31	3008×1960
6	1152×864	19	2048×1536	32	3008×2008
7	1280×1024	20	2967×2232	33	2976×2232
8	1600×1200	21	2016×1512	34	3072×2304
9	1632×1224	22	2282×1712	35	3264×2448
10	1728×1152	23	2272×1704	36	3264×2176
11	1728×1168	24	2304×1728	37	3560×2336
12	2160×1440	25	2400×1600	38	720×488 (NTSC 27MHz)
13	1792×1200	26	2240×1680	39	720×576 (PAL 27MHz)

The weight generation is shown in Eq. (3) and the value has nine kinds of $weight_{max}$ to cope with various shading effects.

$$weight = (weitht_{\max} - 1) \times rate^2 + 1 \quad (3)$$

$weight_{max}$ = {1, 1+1/8, 1+2/8, 1+3/8, 1+4/8, 1+5/8, 1+6/8, 1+7/8, 2}

The nine kinds of $weight_{max}$ calculated in Eq. (3) are shown in Fig. 3.

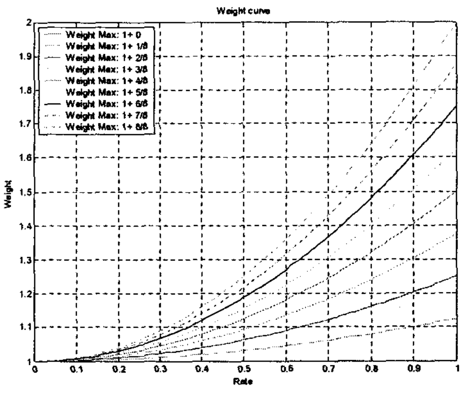


Fig. 3 Relation between rate and weight

As you can see, the input image is not treated in case that the $weight_{max}$ is '1' because every input color is multiplied by '1.'

The Eqs (2) and (3) are the rate generation method in case of progressive scan. However, the process in case of interlaced scan should be different with the Eqs (2) and (3) because interlaced scan displays one frame by using two fields. The Eq. (4) is used when the input image is an odd field.

$$rate = \frac{\sqrt{(x_{center} - x)^2 + \{2 \times (y_{center} - y)\}^2}}{d_{\max}} \quad (4)$$

Figure 4 shows the interlaced scan which consists of two kinds of field such as odd and even. Here, image center is on the odd field scan line and the even filed lines are shown between odd field scan lines.

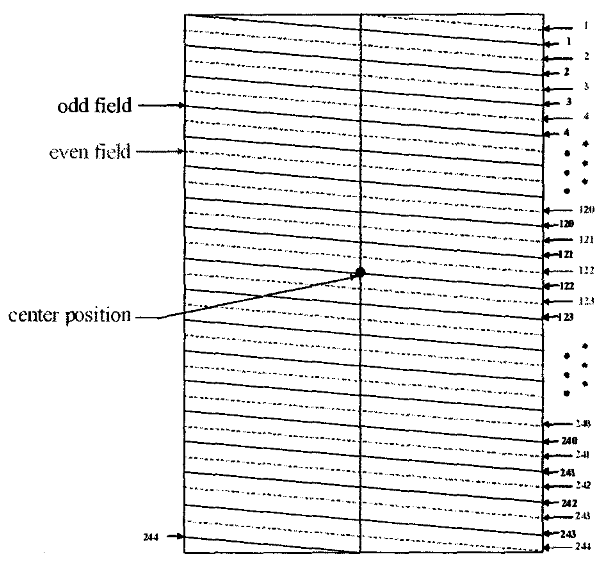


Fig. 4 Interlaced scan

The generation process of even field rate is shown in Equations (5) and (6) which are corresponding to upper lines (1st line~122nd line) and lower lines(123rd line~244th line) than center position, respectively.

$$rate = \frac{\sqrt{(x_{center} - x)^2 + \{2 \times (y_{center} - y) + 1\}^2}}{d_{\max}} \quad (5)$$

$$rate = \frac{\sqrt{(x_{center} - x)^2 + \{2 \times (y_{center} - y) - 1\}^2}}{d_{\max}} \quad (6)$$

The terms, '+1' and '-1', in Eqs.(5) and (6) imply that the relationship between center and even filed lines in Fig. 3.

2. Considering and Applying Weight

Equation 7 shows the procedure which is appling five kinds of curvature to weight.

$$weight = (weight_{max} - 1) \times \frac{rate^2 + curve\_control \times rate}{curve\_control + 1} + 1 \quad (7)$$

$$curve\_control = \{0, 1/8, 1/4, 1/2, 1\}$$

Figure 5 shows the applied five kinds of curvatures in $weight_{max}$, '1+8/8.' As you know, the proposed system is using eight kinds of $weight_{max}$ except $weight_{max}$ is '1.' Therefore, the system is using forty kind of correction curves with $weight_{max}$ and *curve_control.*

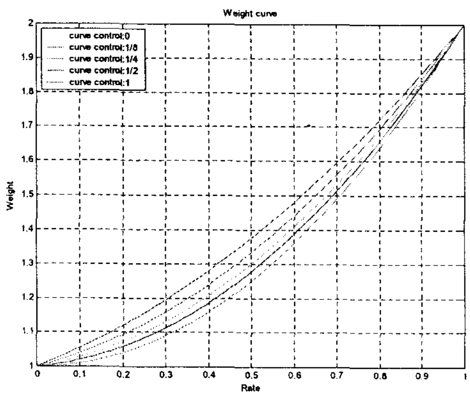


Fig. 5 Modified weight curve

Equations (8)-(10) shows the processes which convert red, green and blue into Y, Cb and Cr.

$$R = 1.164 \times (Y - 16) + 1.596 \times (Cr - 128) \quad (8)$$

$$G = 1.164 \times (Y - 16) - 0.813 \times (Cr - 128) - 0.392 \times (Cb - 128) \quad (9)$$

$$B = 1.164 \times (Y - 16) + 2.017 \times (Cb - 128) \quad (10)$$

By Eqs. (8)-(10), we could know that we should use *(Y-16)*, *(Cb-128)* and *(Cr-128)* to convert red, green and blue on keeping color mixture ratio.

$$Y'_{in} - 16 = \frac{1}{weight} \times (Y_{in} - 16) \rightarrow$$

$$Y'_{in} = \frac{1}{weight} \times (Y_{in} - 16) + 16 \quad (11)$$

$$Cr'_{in} - 128 = \frac{1}{weight} \times (Cr_{in} - 128) \rightarrow$$

$$Cr'_{in} = \frac{1}{weight} \times (Cr_{in} - 128) + 128 \quad (12)$$

$$Cb'_{in} - 128 = \frac{1}{weight} \times (Cb_{in} - 128) \rightarrow$$

$$Cb'_{in} = \frac{1}{weight} \times (Cb_{in} - 128) + 128 \quad (13)$$

Equations (11)-(13) show the process that the weight is applied to *(Y-16)*, *(Cb-128)* and *(Cr-128)*.

3. Simulation Result

Figure 6 shows the resulting images obtained by using the system proposed in this paper.

(a) Input image with shading effect

(b) Output image with shading correction processing

Fig. 6 Input image and simulation result image

As you can see, the background dark bluish color in Fig. 6-(a) becomes more light bluish in Fig. 6-(b) and the dark reddish-color flowers of the garden in Fig. 6-(a) becomes more light reddish in Fig. 6-(b). Thereby, the corrected colors can be achieved by using the proposed system without shading effect.

Figure 7 shows the block diagram of the proposed system. The proposed system is comprised of four major building blocks and designed by using Verilog-HDL models [6].

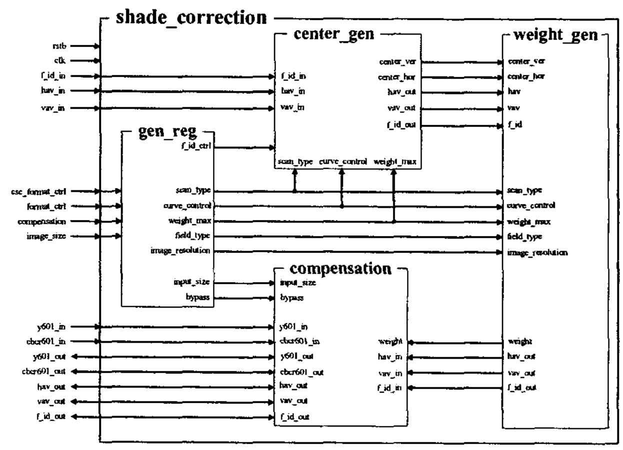


Fig. 7 Block diagram of shading correction system

Table 2 summarizes the required hardware for the proposed method. The gate counts are calculated where a 2-input NAND is counted as one gate.

Table 2 Synthesis and implementation results

Module Name	Xilinx FPGA XCB2000E-BG560		Synopsys Design Analyzer	
	Slice (19,200)	Max Timing [ns]	Gate counts	Max Timing [ns]
gen_reg	13 (0.06%)	-	145	14.5
center_gen	73 (0.4%)	7.852	1110	19.05
weight_gen	1644 (8%)	18.249	25696	19.05
sqrt	514 (2%)	11.521	8444	15.43
weight_gen_core	1131 (5%)	18.249	17244	19.30
compensation	664 (3%)	14.592	9300	17.26

The Xilinx Virtex FPGA XCV2000E-6BG560 device in Fig. 8 is used to demonstrate the performance of the proposed method.

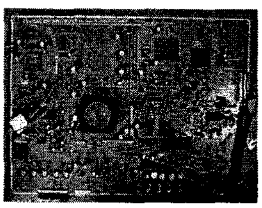


Fig. 8 Demonstration board of proposed system

III. Conclusion

In this paper, we proposed the shading correction system supporting multi-resolution for camera. The proposed system is available on thirty nine kinds of image resolutions scanned by interlaced and progressive type. Moreover, the system is using some continuous quadratic equations instead of using the piece-wise linear curve which is composed of multiple line segments. And then the curves have eight kinds of $weight_{max}$ in Eq.(3) and five kinds of *curvature* in Eq.(7). In other words, the proposed system corrects the shading effect by using forty kinds of compensation curve without discontinuity.

Therefore, the proposed system could be applied to the any resolution image and cope with various kinds of shading environment.

Acknowledgement

This work has been supported in part by the 2nd BK21 funded by the Korean Government and Samsung Electro-Mechanics, Co. LTD. The authors also wish to thank the IDEC and IT-SoC Practice Project for their software assistances.

References

[1] I. T. Young, "Shading Correction: Compensation for Illumination and Sensor Inhomogeneities," On-line: http://www.ph.tn.tndelft.nl, Faculty of Applied Physics. July. 2000.

[2] Tomazevic D, Likar B and Pernus F., *Comparative evaluation of retrospective shading correction methods*, University of Ljubljana, Department of Electrical Engineering, Trzaska 25, 1000 Ljubljana, Slovenia.

[3] I. T. Young, J. J. Gerbrands and L. J. v. Vliet, *Image Processing Fundamentals in The Digital signal processing Handbook*, V. K. Madisetti and D. B. Williams, Eds. Boca Raton, Florida: CRC Press in cooperation with IEEE Press, 1998.

[4] C. R. Giardina and E. R. Dougherty, *Morphological Methods in Image and Signal Processing*, Englewood Cliffs, New Jersey: Prentice-Hall, 1988.

[5] H. J. A. M. Jeijmans, Morphological Image Operators. Boston: Academic Press, 1994.

[6] D. J. Smith. *HDL Chip Design: A practical guide for designing, synthesizing and simulating ASICs and FPGAs using VHDL or Verilog.* Doone Publications, 1996.