Brightness Calculation in Digital Image Processing

Sergey Bezryadin, Pavel Bourov*, Dmitry Ilinih*; KWE Int.Inc., San Francisco, CA, USA; *UniqueIC's, Saratov, Russia

Abstract

Brightness is one of the most significant pixel characteristics. It is involved in many image-editing algorithms such as contrast or shadow/highlight.

Currently, there is no conventional formula for brightness calculation, and the same image-processing tool may employ several different brightness measures. However, stimuli, equi-bright according to one measure, may differ more, than ten times according to another.

This paper suggests using length of a color vector for Brightness and demonstrates with major image editing procedures the advantage of this measure.

Suggested definition for Brightness is convenient in terms of software development because it simplifies design of algorithms that perform only intended operations without concurrent unwilling modification some other image parameters. For example, suggested contrast editing algorithm modifies only pixel Brightness and does not change chromatic coordinates. An advantage of the algorithms is especially visible when they are applied to a high dynamic range image.

Introduction

Usually, term Brightness should be used only for non-quantitative references to physiological sensations and perceptions of light. Wyszecki and Stiles [1] define Brightness as *an attribute of a visual sensation according to which a given visual stimulus appears to be more or less intense; or, according to which the area in which the visual stimulus is presented appears to emit more or less light*, and range variation in Brightness from "bright" to "dim". Given definition is useless for digital image processing, because provides no foundation for image editing.

Developers of algorithms for digital image processing are obliged to find a way to describe Brightness quantitatively. However, currently, there is no conventional numerical description for this stimulus characteristic. This paper proposes a review and analysis of the most popular values used for Brightness representation and discusses the effectiveness of those values in image editing algorithms heavily dependent on the choice of Brightness measure.

Brightness Models

Not so long ago, **Luminance** was used as a synonym for Brightness. Thus, a value Photoshop employs for Brightness in Color-to-Grayscale transformation well correlates with Luminance definition.

Another popular brightness substitution is **Luma**. According to ITU-R BT.601 standard, it is Brightness equivalent in MPEG and JPEG algorithms

$$Y' = 0.299\ r + 0.587\ g + 0.114\ b \tag{1}$$

where r, g, and b are stimulus sRGB coordinates.

Luma is widely used in image processing algorithms imitating performance of corresponding Color TV adjusting knobs. Thus, Photoshop uses it in contrast editing algorithms to calculate average Brightness. There is a myth that Luma well approximates Brightness. It is not always true. For example, two stimuli having (0,0,255) and (38,21,45) sRGB coordinates, respectively, characterized by the same Luma value (Y' = 29), while their Luminance differs 6.4 times.

The most popular Brightness editing algorithm is based on **Arithmetic mean** model

$$= (r + g + b) / 3 \tag{2}$$

This Brightness measure has the biggest difference with Luminance. For example, stimuli with (0,255,0) and (69,21,165) sRGB coordinates are characterized by the same value = 85, while their Luminance differs 15.8 times.

Introduced by Alvy Ray Smith, **HSV** (Hue, Saturation, Value) also known as HSB (Hue, Saturation, Brightness) model is prevalent in Saturation and Hue editing algorithms

$$V = \max\ (r, g, b) \tag{3}$$

According to (3), stimuli with sRGB coordinates (255,255,255) and (0,0,255), respectively, are characterized by the same V = 255. Luminance of the stimuli differs 13.9 times.

Presented examples demonstrate that for stimuli corresponding to saturated colors, there is a large diversity in determination which of them have the same Brightness, and there is a question, which value is more appropriate for Brightness calculation. None of considered values works equaly well for all image editing procedures, and developer's preference, as it has been illustrated with Photoshop example, usually depends on an area of application.

Use of stimulus length as a measure of Brightness (4), introduced in **BCH** (Brightness, Chroma, Hue) model [2], provides Brightness definition effective for all image-editing algorithms. Length is calculated according to Cohen metrics [3].

$$B = \sqrt{D^2 + E^2 + F^2} \tag{4}$$

$$\begin{pmatrix} D \\ E \\ F \end{pmatrix} = \begin{pmatrix} 0.2053 & 0.7125 & 0.4670 \\ 1.8537 & -1.2797 & -0.4429 \\ -0.3655 & 1.0120 & -0.6104 \end{pmatrix} \cdot \begin{pmatrix} X \\ Y \\ Z \end{pmatrix}$$

where X, Y, and Z are Tristimulus values.

The main advantage of this model is that it simplifies design of an algorithm that performs only intended operation without unwilling concurrent modification other image parameters. Thus, Brightness and contrast editing algorithms based on BCH model modify only pixel Brightness and preserve chromatic coordinates.

This Brightness definition is also noticeably different from Luminance. Thus, stimuli with sRGB coordinates (0,0,255) and

(196,234,0), respectively, have the same length, while their Luminance differs 9.8 times.

Color to Grayscale Transformation

The most natural way to turn a colored image into a grayscale one is with an algorithm preserving pixel Brightness. This transformation may serve as a test revealing the quality of Brightness measure.

The biggest discrepancy in Brightness values is on the edge of sRGB gamut, and this fact has determined the selection of stimuli (Tab.1) used for model analysis and investigation of their conformity with human sensation.

Tab.1 presents sRGB coordinates of seven stimuli with the same Luminance (accuracy 0.2%).

Table 1. sRGB coordinates for a set of equi-Luminance stimuli

Red			Green			Blue			Gray		
157	0	0	0	89	0	0	0	255			
Cyan			Magenta			Yellow			76	76	76
0	85	85	138	0	138	79	79	0			

A colored image for Tab.1 is not displayed because corresponding colors are beyond allowed for this publication gamut, but anyone may restore the image on a monitor using provided stimuli coordinates, or download it from www.kweii.com/ref/2007LV.png

And when colors corresponding to coordinates provided by the Tab. 1 are displayed, it becomes clear, that they are not equally bright from a human point of view. The brightest stimulus is obviously Blue, but Red and Magenta are also perceived brighter, than Cyan, Green and Yellow.

The image corresponding to Tab.1 processed with color-to-grayscale transformation using Luminance for Brightness turns into equally grey picture. Processing the same image with alternative Brightness representatives according to discussed above models makes it possible to compare the models.

Brightness values for stimuli presented in Tab.1 calculated with formulas (1) – (4) are in Tab. 2. Corresponding grayscale images are displayed in Fig. 1- 4.

Table 2: Brightness calculated according to considered models for the set of equi-Luminance stimuli

r	g	b	Y'	μ	V	B
157	0	0	46.9	52.3	157	10.4
0	89	0	52.2	29.7	89	5.0
0	0	255	29.1	85.0	255	45.3
0	85	85	59.6	56.7	85	6.5
138	0	138	57.0	92.0	138	14.1
79	79	0	70.0	52.7	79	4.7
76	76	76	76.0	76.0	76	5.8

While Luminance underrates Brightness of the Blue stimulus, the value provided for it by Luma may be considered as unacceptably small. Rating of colors in Fig.1 looks inversed, marking Blue and Red less bright than Cyan and Yellow.

Use of improves relation between Blue and Grey stimuli, but underrates Brightness of Green and overrates Magenta, grading its Brightness closely to Blue (Fig.2).

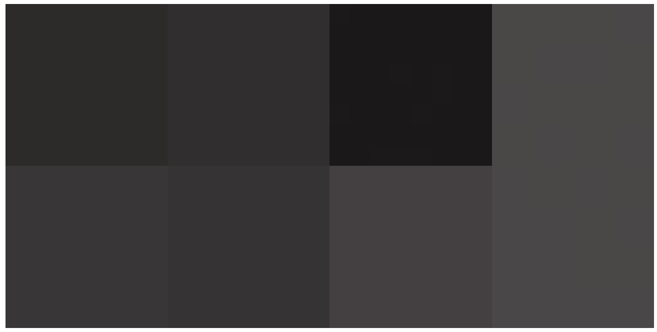

Figure 1. *Color- to-grayscale transformation. Luma model.*

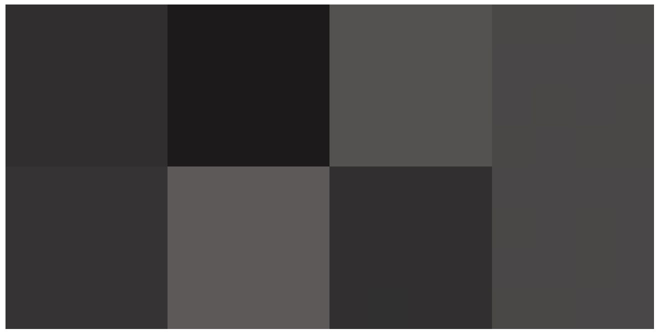

Figure 2 *Color- to-grayscale transformation. Arithmetic mean model.*

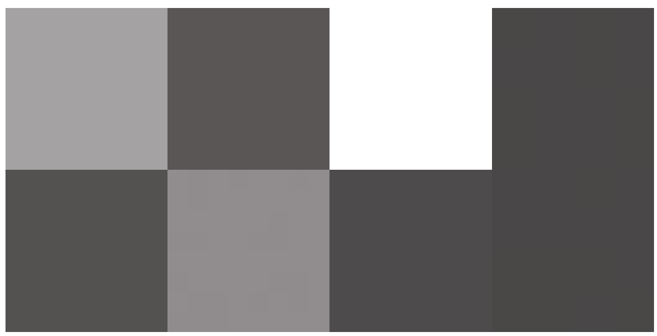

Figure 3. *Color- to-grayscale transformation. HSV model*

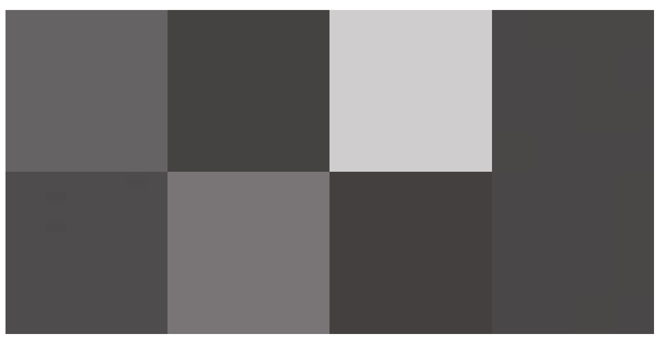

Figure 4 *Color-to-grayscale transformation. BCH model*

Brightness rating provided by HSV model better corresponds to human perseption, than Luma or , and it makes this model relatively popular among photographers. However, Brightness of the Blue stimulus is graded as high as Brightness of White stimulus (Fig.3), and this defect reduces the model value.

In BCH model evaluation of Blue is improved comparing to HSV model and, in general, its Brightness rating corresponds to human perception (Fig.4).

It is a moot point, whether Luminance or BCH model provides better measure for Brightness, both of them are not optimal, but they definitely have advantage against other considered values.

Brightness Editing

Natural Choise

An algorithm that is equivalent to expocorrection and which may be described with the following formula

$$B' = 2^{EV} \cdot B \qquad (5)$$

looks like the most natural choice for Brightness editing. Fig. 5 illustrates a performance of the algorithm, while Tab. 3 presents corresponding sRGB coordinates.

Table 3: Brightness editing. Natural choice

Color	Original color			EV = +2			EV = +4		
Grey	56	56	56	111	111	111	208	208	208
Red	56	5	3	111	18	12	208	43	32
Green	2	37	15	8	77	38	25	148	78
Blue	5	4	29	18	15	63	43	38	123
Cyan	4	36	53	15	75	105	38	145	199
Magenta	54	2	28	107	8	61	202	25	119
Yellow	60	58	10	118	114	29	221	215	62
Dark	1	1	1	4	4	4	15	15	15

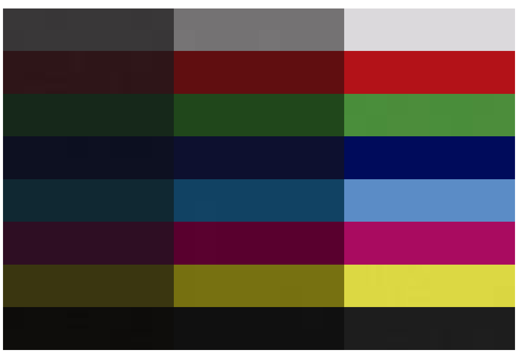

***Figure 5**. Brightness editing. Natural choice*

The algorithm is designed for BCH and may easily be adapted for any other Color Coordinate Systems (CCSs). Although this method of Brightness editing provides better result, than those described below, it is less common in present-day digital image processing.

TV based algorithm

Modern image processing tools, such as, Corel, Photoshop etc., make Brightness modification according to formula (6)

$$(r', g', b') = (r + M_0, g + M_0, b + M_0) \qquad (6)$$

where M_0 is parameter determining Brightness modification.

This algorithm imitates Brightness control embodied in TV. Brightness modification should transform any set of equi-bright stimuli into equi-bright stimuli. For equation (6), this requirement is fulfilled only by the Arithmetic mean model. Other Brightness measures, including Luma, do not support this property, although Luma is used for Brightness in other TV-imitating image-processing algorithms. Fig. 6 illustrates a performance of the algorithm, while Tab. 4 presents corresponding sRGB coordinates.

Table 4: Brightness editing. TV based algorithm

Color	Original color			M_0 = +55			M_0 = +152		
Grey	56	56	56	111	111	111	208	208	208
Red	56	5	3	111	60	58	208	157	155
Green	2	37	15	57	92	70	154	189	167
Blue	5	4	29	60	59	84	157	156	181
Cyan	4	36	53	59	91	108	156	188	205
Magenta	54	2	28	109	57	83	206	154	180
Yellow	60	58	10	115	113	65	212	210	162
Dark	1	1	1	56	56	56	153	153	153

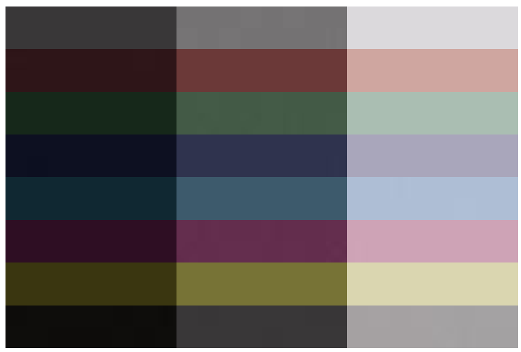

***Figure 6**. Brightness editing. TV based algorithm*

Comparison of Fig 5 and Fig.6 reveals the main defects of the method based on Arithmetic mean model. The Brightness transformation changes stimuli chromatic coordinates and increasing Brightness entails contrast and saturation decrease. Roughly speaking, this method may be reduced to addition (or subtraction) a White stimulus. However, as it may be seen from a more accurate analisys, it is not exactly true, because coordinate addition in a non-linear space has no sense.

Lightness editing (Lab)

Some image editors provide an option to choose Lab CCS as Workflow. And there is a common believe, that Brightness editing may be well done by lightness modification according to the following algorithm

$$(L', a', b') = (L + L_0, a, b) \qquad (7)$$

where L_0 is parameter determining Brightness modification.

Fig. 7 illustrates a performance of the algorithm, while Tab.5 presents corresponding sRGB coordinates. As it may be seen from the pictures, lightness editing result is very similar to TV based algorithm result and significantly worse than expocorrection.

Table 5: Lightness editing.

Color	Original color			L_0 = +23.3			L_0 = +60		
Grey	56	56	56	111	111	111	208	208	208
Red	56	5	3	115	59	57	217	150	146
Green	2	37	15	55	90	65	146	185	156
Blue	5	4	29	57	58	80	147	147	173
Cyan	4	36	53	62	88	108	155	182	205
Magenta	54	2	28	112	56	79	213	148	172
Yellow	60	58	10	118	113	63	218	210	155
Dark	1	1	1	56	56	56	145	145	145

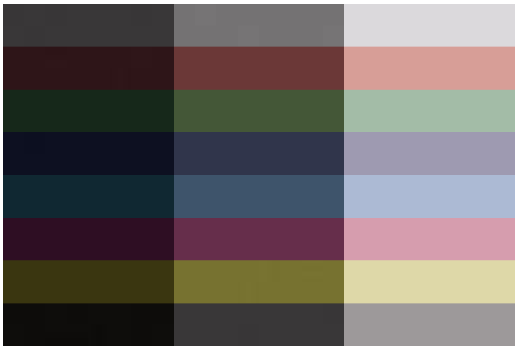

***Figure 7**. Lightness editing*

Curves editing

To make Brightness editing with curves, method widely accepted by professionals, one needs expensive equipment that guarantees accurate visual control of the processed image. However, even a profesional needs to make a lot of manual work in order to preserve stimuli chromatic coordinates, while his efforts not always benefited, especially in case of complicated HDR scenes.

Use of BCH and Bef CCSs as Workflow simplifies design of algorithms that, while preserving chromatic coordinates, do not require advanced training in order to achieve an acurate color image editing.

Contrast and Dynamic Range Editing

According to Federal Standard 1037C, contrast in display systems is the brightness ratio. Therefore, it is reasonable to expect from a correct contrast and dynamic range editing algorithms to act according to the rule: if some two pairs of pixels had the same brightness ratio prior contrast modification, after contrast modification their brightness ratios stay equal as well

$$B_1 : B_2 = B_3 : B_4 \;\Rightarrow\; B_1 \;: B_2 \;= B_3 \;: B_4 \qquad (8)$$

However, most of popular dynamic range editing algorithms do not embody this feature and don't preserve pixel chromatic coordinates.

Contrast editing

A transformation that satisfies stated above condition (8) might be written as follows:

$$B\ (m,n) = B_{Avr}(m,n)\left(\frac{B(m,n)}{B_{Avr}(m,n)}\right)^{k} \qquad (9)$$

where k is a variable parameter,
$B(m,n)$ is the Brightness of a pixel with an order number (m,n),
$B_{Avr}(m,n)$ is averaged over an area surrounding the pixel (m,n).

Use of the BCH model or Luminance for brightness in this formula guarantees preservation of pixel chromatic coordinates. Fig. 8 illustrates the difference between the algorithm preserving chromatic coordinates (center) and a typical algorithm that do not (right). A change in chromatic coordinates that accompanies a contrast adjustment on the right image is particularly noticeable when it is compared to the center picture edited with the algorithm employed the BCH model (9).

***Figure 8** Comparison of contrast editing algorithms.*

Dynamic Range Editing Preserving Local Contrast

The dynamic range editing preserving chromatic coordinates and not affecting local contrast is very important for HDR (High Dynamic Range) image processing. The most popular tone mapping algorithms [5], [6] do not satisfy these conditions. But an algorithm providing listed qualities may easily be created with the BCH model:

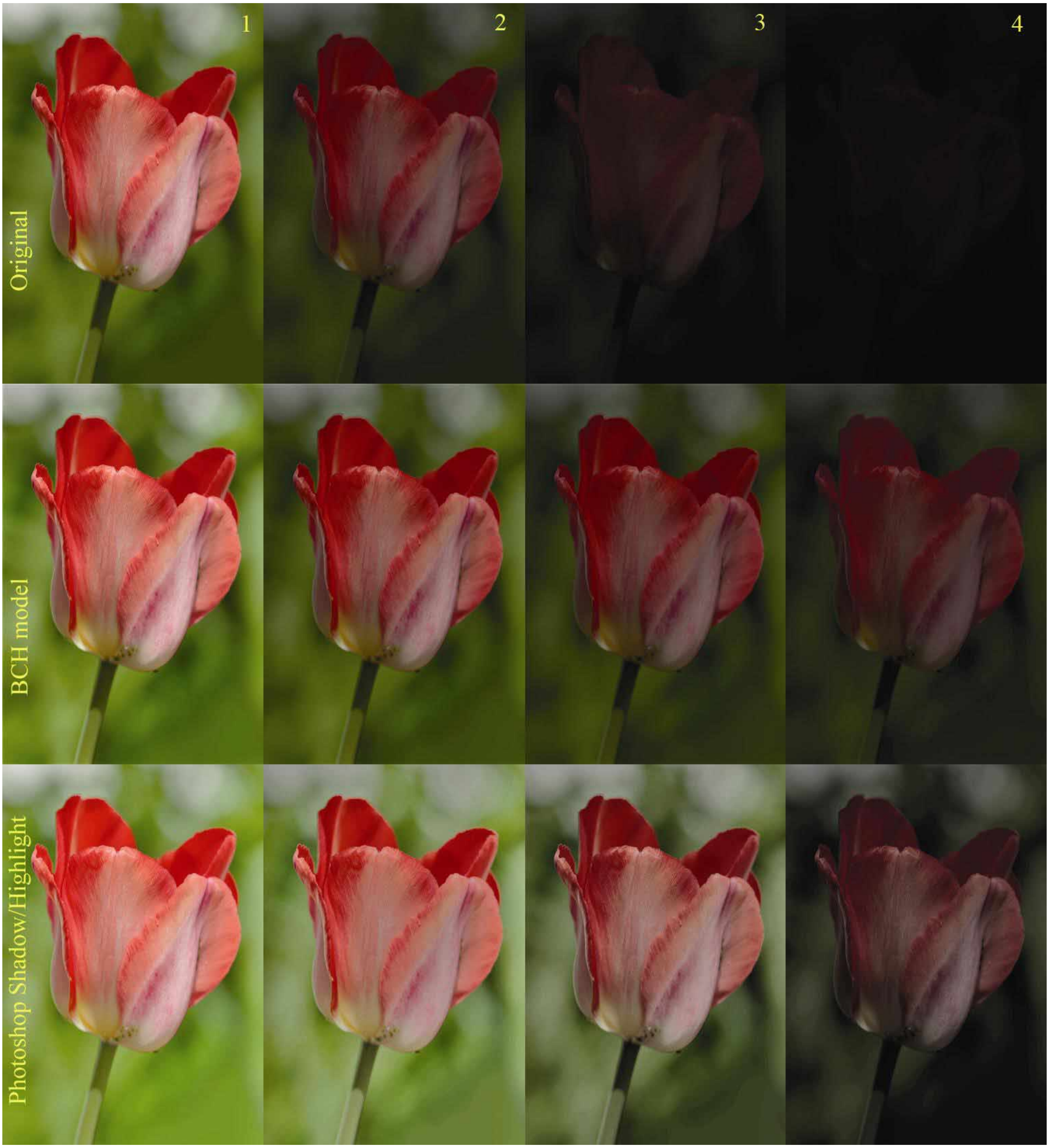

Figure 9 *Dynamic Range editing.*

$$B(m,n) = B(m,n)\left(\frac{B_0}{B_{Avr}(m,n)}\right)^k \qquad (10)$$

where B_0 is a variable parameter.

Presented algorithm preserves the relation (8) for B_{Avr}

$$B_{Avr,1} : B_{Avr,2} = B_{Avr,3} : B_{Avr,4} \qquad (11)$$

and this feature helps maintain an impression of large dynamic range. Moreover, it provides an opportunity for an accurate reverse transformation.

The performance of dynamic range editing algorithms may be illustrated with Fig. 9. A synthetic HDR image (top) was constructed from a single photograph. It has four pictures in a row, and the second, third and forth elements were created from the first one by successive increase of expocorrection, -2 EV, -4 EV and -6 EV, respectively, so brightness ratio of corresponding pixels in the first and forth quarters is 64, while their chromatic coordinates are the same (within sRGB allowable accuracy). This artificial image (it may be downloaded from www.kweii.com/ref/HDR.AT1.png) helps easily visualise changes in local contrast and chromatic cordinates. In Fig.9 the central image has been processed with the algorithm (10), and the result of Photoshop shadow/highlight processing is in the bottom.

Conclusion

On the one hand, Brightness, by definition, is a psychophysical non-measurable characteristic. On the other hand, de-facto, it is a quantitative parameter essential for digital image processing, and algorithm developers have to use some formula for Brightness calculation. There is no conventional measure for Brightness, and it is a regular situation, when an image-editing tool uses several different formulas for its calculation. However, stimuli equally bright by one measure may differ more than 10 times by another. Moreover, many formulas are designed for sRGB gamut and their use for extended gamut (for example, WideGamutRGB) results in even bigger difference.

The BCH model as Brightness measure has a clear physical meaning and convenient for software development. All considered in the paper Brightness measures do not fully correspond to human perception, but while each of the first four has its advantageous and disadvantageous area of application, the BCH model works well in all image editing procedures. Considered algorithms which performance significantly depends on employed Brightness formula, such as color-to-grayscale transformation, or Brightness, Contrast and HDR image editing, illustrate the statement.

References

[1] G. Wyszecki, W. S. Stiles, “Color Science. Concepts and Methods, Quantitative Data and Formulae”, Second Edition, John Wiley & Sons, (2000)

[2] S. Bezryadin, P. Bourov, “Color Coordinate System for Accurate Color Image Editing Software”, The International Conference Printing Technology SPb’06 proceedings, p. 145–148, St. Peterburg State University of Technology and Design, (2006)

[3] Cohen J. B., Visual Color and Color Mixture: The Fundamental Color Space, Univ. of Illinois Pr, (2000).

[4] S. Bezryadin, “New generation of image editing algorithms”, Electronic Imaging 2007, Digital Photography III, (2007).

[5] E. Reinhard, M. Stark, P. Shirley, and J. Ferwerda, "Photographic Tone Reproduction for Digital Images." ACM Transactions on Graphics, 21(3), 267-276 (2002). http://www.cs.ucf.edu/~reinhard/cdrom/

[6] R. Fattal, D. Lischinski, and M. Werman, "Gradient Domain High Dynamic Range Compression." ACM Transactions on Graphics, 21(3), 257-266 (2002). http://www.cs.huji.ac.il/~danix/hdr/

Author Biography

Sergey Bezryadin received MS in physics from the Moscow State University (1976) and PhD in physics and mathematics from the Moscow Institute of Electronic Technique (1982). Since 2001 he has worked in the KWE Int. Inc., San Francisco, USA. He is a member of IS&T.