Open Access

A Wavelet-Based Encoding Algorithm for High Dynamic Range Images

Frank Y. Shih* and Yuan Yuan

Department of Computer Science, New Jersey Institute of Technology, Newark, NJ 07102, USA

Abstract: High Dynamic Range (HDR) images use a wider range of intensity values than the common Limited Dynamic Range (LDR) images. Because of this, handling HDR images requires a great deal of information to be stored and transferred. To represent and display HDR images efficiently, a trade-off needs to be balanced between size and accuracy. We present a new wavelet-based algorithm for encoding HDR images, so that their data are feasible in the internet-based society for image communication. Experiments are conducted using the library provided by the Munsell Color Science Laboratory of Rochester Institute of Technology and HDR Images provided by Industrial Light and Magic (ILM, a motion picture visual effects company). Experimental results show that the encoded format is able to achieve a good balance between visual quality and image size for internet users.

Keywords: Image representation, Image compression, High dynamic range.

1. INTRODUCTION

Human beings are accustomed to seeing a very limited range of intensity levels in image reproduction [1]. Much of the current technology up-to-date has been unable to capture or display imagery in nearly the depth of intensity level that human vision can perceive. Thus, the high dynamic range imagery is put forward as a solution. Lewotsky stated, "High dynamic range imaging (HDRI) delivers images that are truer to life" [2]. That is, HDRI offers a far better approximation of the real world [3-6]. We probably have seen an under-exposed or an over-exposed region in a picture; the former loses detail in excessively dark regions, and the latter loses detail in excessively bright regions. With HDRI, there will be enough dynamic range to see details, simultaneously in the very dark and in the very bright regions of a given image.

The HDRI technology is used in cinema, especially for special effects, because it aids in the blending of computer rendered images and real life imagery [7]. Also, because it is possible to view HDR images on an everyday consumer computer screen with special software, a subject can be photographed at different levels of exposure, and then the multiple images can be composed into a single HDR image [8, 9]. The viewer will then get to appreciate the details in the regions that are extremely dark or extremely bright. However, because of the limitations of photographic sensors, it is not yet possible for everyday consumers to snap HDR pictures with their digital cameras. Some companies and other research organizations, however, are currently investigating more sophisticated mechanisms that allow HDR image capture.

Computers can render artificial HDR imagery and create HDR imagery from several photographs of varying exposure, but we have not yet seen consumer cameras that can deliver these pictures although this may not be the case for much longer [10, 11]. Because HDRI is a technology whose time is coming, it is important that we are ready for it. As more consumers begin to have access to HDRI, there will undoubtedly be a demand to share images over the Internet (through email or the web). This requires the creation and subsequent standardization of small yet accurate HDR digital representation.

The most commonly used HDR image formats include Radiance RGBE Encoding (HDR) and ILM OpenEXR (EXR) [1]. The Radiance RGBE Encoding was developed in 1985 at the Lawrence Berkeley National Laboratory, located in Berkeley, California. Because the system was designed to compute photometric quantities, a 4-byte is used to represent each pixel where three 8-bit mantissas share a common 8-bit exponent. The OpenEXR image format was developed in 2002 by the Industrial Light and Magic (ILM, a motion picture visual effects company) located in San Francisco, California. This format has been internally used by the company for many years for special effect rendering and composing. This format is a general-purpose wrapper for the 16-bit Half data type, which has been adopted by both companies, NVidia™ and ATI™, in their floating-point frame buffers. The Half data type is a logical contraction of the IEEE-754 floating-point representation to 16 bits. It is also called the S5E10 (Sign plus 5 exponent plus 10 mantissa) format. This format has been floating around the computer graphics hardware developer network for some time. The ILM OpenEXR supports a standard IEEE 32-bit/component format as well as a 24-bit/component format.

However, there has not been a focus on devising an HDR image format appropriate for the Internet up to date. A format is conducive to the web page paradigm of 10 to 20 images loading and formatting instantaneously. Since much of our daily lives are coordinated, planned, or conducted using the Internet, as the HDR technology becomes more readily available to the average consumer, the expectation for an HDR online experience will grow.

*Address correspondence to this author at the Department of Computer Science, New Jersey Institute of Technology, Newark, NJ 07102, USA; Tel: 973-596-5654; Fax: 973-596-5777; E-mail: shih@njit.edu

1876-8253/10

In order to represent HDR images, a trade-off occurs between size and accuracy. This paper proposes a new method for encoding HDR images that are acceptable in our internet-based society, both in terms of size and accuracy. The rest of the paper is organized as follows. We present the considerations in choosing a format in Section 2. The proposed encoding algorithm is presented in Section 3. Experimental results are provided in Section 4. Finally, we draw conclusions in Section 5.

2. CONSIDERATIONS IN CHOOSING A FORMAT

There are several issues to be considered in seeking a small and accurate image representation. Finding an appropriate format that can represent HDR images in a small and accurate footprint will be a multifarious endeavor [12]. It includes finding the appropriate color representation, finding an optimal bit depth, and applying appropriate compression technologies. In choosing a suitable format, we consider the following guidelines.

First, it is important that the format is able to represent the entire visible gamut. Even though many devices are not yet able to display the entire gamut, it is important that this information is there. In the future, the format will be able to represent the full gamut, allowing any devices that can display it to do so. We need to consider how to encode a pixel's color information efficiently. Scientists have studied our perception of color for many years, dating back to the 1600's, with Isaac Newton [13]. Typically, in TVs, computer screens, and other display devices, the primaries red, green, and blue are used as the static colors that are combined to create the color in question. However, this situation has a drawback that unless the red, green and blue lights are 100% pure, containing exactly the single correct frequency, they will be unable to produce every color in the visible gamut. Alternatively, color can be represented using luminance and chromaticity. Luminance is the intensity of a color, and chromaticity is coordinates on a color plane that covers all visible colors. Therefore, we have a two-dimensional grid of all colors and a third variable that represents directly the intensity of the color. Typically, using the three-primary (red, green, and blue) approach to store color representations, equal resolution is given to all three of the primaries. This is significant because the value of each primary is equally important. However, with luminance and chrominance, the resolution of the two chrominance channels can be much less than that of the luminance channel — only 8 bits are needed to achieve a 0.0017 unit tolerance in chrominance space. We can allocate our storage specifically to the channel that needs it the most, which is luminance channel. With luminance/chrominance, we assign the majority of our pixel's data to the luminance channel.

Second, the number of actual bits allocated for each of the pixel attributes has a great impact on size. If we give 48 bits rather than 32 bits for each of the pixel attributes, the size of each attribute is increased by 16 bits and the total size of a raw 320 × 200 image is increased by 384 KB, which is a significant change for a relatively small image. According to [14], human eyes can sense a total dynamic range of five orders of magnitude in a single view. It is important that HDR image format should cover at least five orders of magnitude of dynamic range. As this format is intended for a rich multimedia experience, granting slightly beyond the range visible in a single view is far more than sufficient, and will provide more than sufficient dynamic range.

Finally, we must consider the manner with which we compress the pixel data. The standard JPEG and the JPEG HDR [15] formats use Discrete Cosine Transform (DCT) to analyze image features and find redundancies; the compression method in these formats is always lossy [13]. Other methods use Discrete Wavelet Transform (DWT) and can be lossy or lossless to yield good compression ratios [16].

3. THE PROPOSED ENCODING ALGORITHM

The proposed encoding algorithm consists of five major steps: color space transformation, floating-point number to integer transformation, discrete wavelet transform, quantization, and entropy coding [17].

3.1. Color Space Transformation

To make a compromise between compression efficiency and visual quality, we use the following criteria for choosing the color space:

(1) The color space must encode the full color gamut and the full range of luminance that are visible to human eyes.

(2) A unit distance in the color space correlates with the Just Noticeable Difference (JND), which simplifies the control over distortions for lossy compression. The JND is defined as the smallest difference in a specified modality of sensory input that can be detected by a human perception. According to Weber's Law, the size of the JND is a constant proportion of the original stimulus value.

(3) Since most compression algorithms use positive integer values, we choose positive integers to encode luminance and color.

(4) The correlation between different color channels should be minimal. If different color channels are correlated, it means that the same information is encoded twice; thus this will worsen the compression performance.

Based on the analysis above, given a HDR image with photometrically calibrated *XYZ* color values. The physical luminance *Y* is transformed to luma values according to [5] as

$$luma(Y) = \begin{cases} a \times Y & \text{if} \quad Y < YL \\ b \times Y^{c} + d & \text{if} \quad YL \le Y \le YH \\ e \times \log(Y) + f & \text{if} \quad Y > YH \end{cases} \tag{1}$$

where the parameters are chosen to be a = 17.554, b = 826.81, c = 0.10013, d = –884.17, e = 209.16, f = –731.28, YL = 5.6046, and YH = 10469.

The inverse conversion from luma to luminance is given by

$$Y(luma) = \begin{cases} a' \times L & \text{if} \quad L < LL \\ b' \times \left(L + d'\right)^{c'} & \text{if} \quad LL \le L < LH \\ e' \times \exp(f' \times L) & \text{if} \quad L \ge LH \end{cases} \tag{2}$$

where $a' = 0.056968$, $b' = 7.3014\text{e-}30$, $c' = 9.9872$, $d' = 884.17$, $e' = 32.994$, $f' = 0.0047811$, $LL = 98.381$, $LH = 1204.7$.

The transformation function is plotted in Fig. (**1**), which shows for the luminance range from 1e-04 to 1e+10 cd/m^2 (candela per square meter), and the output luma range from 0 to 4000, which can be encoded as 16-bit integer after rounding to the nearest integer.

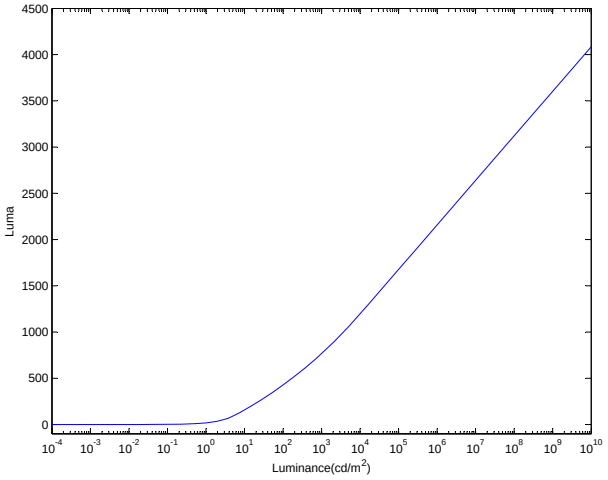

Fig. (1). Function mapping physical luminance to luma.

We deal with encoding chrominance as two 8-bit chroma channels because human eyes are not as keen to chroma as to chrominance. We use the CIE 1976 Uniform Chromacity Scales (u, v). The values for chrominance channels are calculated by

$$\begin{cases} u = \dfrac{4X}{X+15Y+3Z} \\ v = \dfrac{9Y}{X+15Y+3Z} \end{cases} \quad (3)$$

3.2. Floating-Point Number to Integer Transformation

The transformation function in Fig. (**1**) shows that the output luma ranges from 0 to 4000, which can be encoded as a 16-bit integer after rounding to the nearest integer. Since the gamut of perceivable u and v values is between 0 and 0.62, we perform a multiplication by a scale factor of 410 to change the ranges into between 0 and 255, which can be encoded into 8-bit integers by

$$\begin{cases} u_{8bit} = 410u \\ v_{8bit} = 410v \end{cases} \quad (4)$$

3.3. Discrete Wavelet Transform

Industrial standards for compressing still images (e.g. JPEG) and motion pictures (e.g. MPEG) have been based on the DCT. Both standards have produced good results, but have limitations at high compression ratios. At low data rates, the DCT-based transforms suffer from a "blocking effect" due to the unnatural block partition that is required in the computation. Other drawbacks include mosquito noise and aliasing distortions. Furthermore, the DCT does not improve the performance as well as the complexities of motion compensation and estimation in video coding.

Due to the shortcomings of DCT, discrete wavelet transform (DWT) has become increasingly important. The main advantage of DWT is that it provides space-frequency decomposition of images, overcoming the DCT and Fourier transform that only provide frequency decomposition. By providing space-frequency decomposition, the DWT allows energy compaction at the low-frequency subbands and the space localization of edges at the high-frequency subbands. Furthermore, the DWT does not present a blocking effect at the low data rates.

We apply the Daubechies 9-tap/7-tap wavelet in the discrete wavelet transform, which is the same as in the JPEG 2000 compression standard [16]. Each component (luma, u, and v) is converted to a wavelet-domain representation using two-dimensional multi-level discrete wavelet transform. As for the coefficients of the centered scaling and wavelet sequences, we use the numerical values in the implementation as listed in Table **1**.

3.4. Quantization

For each of the luma, u, and v channels, after wavelet transform has been computed, the total number of transform

Table 1. The Coefficients of the Centered Scaling and Wavelet Sequences

Analysis lowpass filter	Analysis highpass filter	Synthesis lowpass filter	Synthesis highpass filter
0.026748757411	0	0	0.026748757411
-0.016864118443	0.091271763114	-0.091271763114	0.016864118443
-0.078223266529	-0.057543526229	-0.057543526229	-0.078223266529
0.266864118443	-0.591271763114	0.591271763114	-0.266864118443
0.602949018236	1.11508705	1.11508705	0.602949018236
0.266864118443	-0.591271763114	0.591271763114	-0.266864118443
-0.078223266529	-0.057543526229	-0.057543526229	-0.078223266529
-0.016864118443	0.091271763114	-0.091271763114	0.016864118443
0.026748757411	0	0	0.026748757411

coefficients is equal to the number of samples in the original image, but the important visual information is concentrated in a few coefficients. To reduce the number of bits needed to represent them, we conduct quantization using the wavelet coefficients a_b of subband b into the value q_b by

$$q_b = \text{sign}[a_b] \times \text{floor}\left[\frac{|a_b|}{2^{R_b-\varepsilon_b}\left(1+\frac{\mu_b}{2^{11}}\right)}\right] \quad (5)$$

where R_b is the nominal dynamic range of subband b, and ε_b and μ_b are respectively the number of bits allotted to the exponent and the mantissa of subband's coefficients. The nominal dynamic range of subband b is the sum of the number of bits used to represent the original image and the analysis gain bits for subband b. We use different quantization parameters for luma, u, and v channels. For luma channel, we set $\varepsilon_b = 16$ $\mu_b = 16$, and for u and v channels, we set $\varepsilon_b = 8$ and $\mu_b = 9$.

3.5. Entropy Coding

After quantization, we perform arithmetic coding to compress the quantized coefficients. Arithmetic coding was used by JPEG 2000, and it generally achieves better compression efficiency than Huffman coding. Like other variable-length entropy encoding techniques, arithmetic coding converts a string of data into another representation that uses fewer bits to represent more frequently occurred characters and uses more bits to represent less frequently occurred characters. However, different from other encoding methods, arithmetic coding possesses several advantages as described below.

When arithmetic coding is applied to independent and identically distributed (i.i.d.) sources, the compression of each symbol is better. It is effective in a wide range of situations and compression ratios. The same arithmetic coding implementation can effectively code all the diverse data created by the different processes. It simplifies automatic modeling of complex sources, yielding near-optimal or significantly improved compression for sources that are not i.i.d. Its main process is arithmetic, which is supported with ever-increasing efficiency by all general-purpose or digital signal processors [18, 19].

The flowchart of the proposed encoding algorithm is shown in Fig. (**2**). The decoding procedure is the inverse of the encoding algorithm which is shown in Fig. (**3**).

4. EXPERIMENTAL RESULTS

We conduct experiments using the library of High Dynamic Range Images provided by the Munsell Color Science Laboratory of Rochester Institute of Technology [20] and the High Dynamic Range Images provided by Industrial Light and Magic (ILM) Company [21]. To measure the quality of the HDR image compression algorithm, we calculate the root-mean-square error as

$$e_{rms} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\left[\left(X_1^i - X_2^i\right)^2 + \left(Y_1^i - Y_2^i\right)^2 + \left(Z_1^i - Z_2^i\right)^2\right]} \quad (6)$$

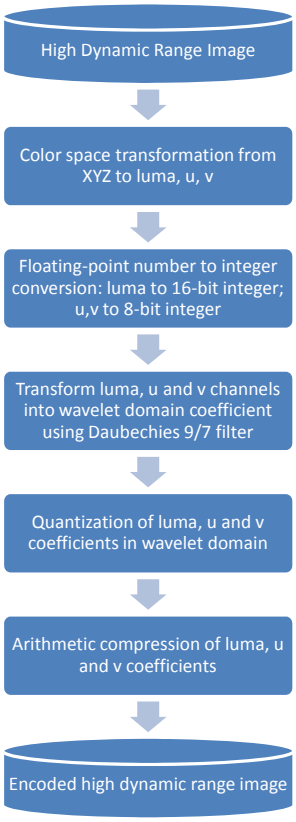

Fig. (2). The flowchart of the proposed encoding algorithm.

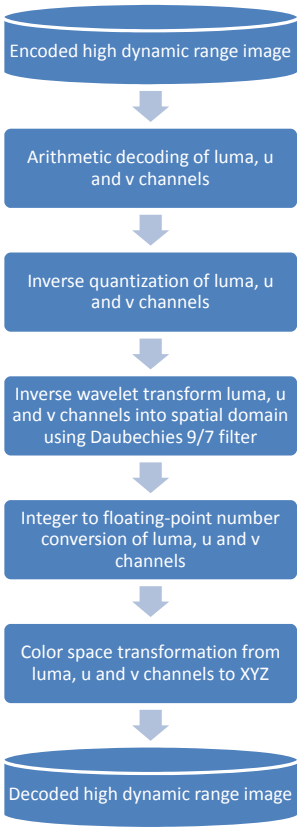

Fig. (3). The flowchart of the proposed decoding algorithm.

where X_1, Y_1, and Z_1 are the *XYZ* channels of the original image, and X_2, Y_2, and Z_2 are the *XYZ* channels of the

Table 2. Experimental Results of HDR Images

Image Name	Original Size (byte)	Encoded Size (byte)	Visible Difference Metrics		Root Mean Square	Encoding Time (second)
			>75%	>95%		
CrissyField.exr	1,304,619	750,382	0%	0%	5.7327	5.0798
Flowers.exr	758,083	200,512	0%	0%	6.8101	4.7732
MtTamNorth.exr	1,422,492	487,324	0%	0%	9.8594	4.9972
cis_front.hdr	7,893,724	640,179	0.01%	0%	0.3659	17.6852
atrium.hdr	8,110,240	360,418	0.36%	0.11%	0.0515	17.3147
Parkinglot.hdr	7,767,193	666,102	0.01%	0%	0.3025	17.2710
rooftops.hdr	7,336,883	500,505	0%	0%	0.1733	15.3573
albers.hdr	6,444,172	463,477	0%	0%	0.4152	15.7201

compressed image. Note that they are in the range of $[10^{-4}, 10^{10}]$.

In addition to the root-mean-square error, we also use the high dynamic range image visible difference metrics, which can predict whether differences between two images are visible to the human observer or not. Such metrics are used for testing either visibility of information (for seeing important visual information) or visibility of noise (for not seeing any distortions in images, e.g. due to lossy compression).

The high dynamic range image visible difference predictor, developed by Mantiuk [22, 23], enables the prediction of visible differences in HDR images. The HDR VDP receives a pair of images (original and distorted) as input and generates a map of probability values, which indicates how likely the differences between the two images are perceived. Both images should be scaled in the unit of luminance. In case of low-dynamic range images, pixel values should be inverse gamma corrected and calibrated according to the maximum luminance of the display device. The original image is transformed by the Optical Transfer Function (OTF), which simulates light scattering in the cornea, lens, and retina. To account for the nonlinear response of photoreceptors to light, the amplitude of the signal is nonlinearly compressed and expressed in the unit of the JND. The probabilities of visible differences are finally calculated from all channels and a map of detection probabilities is generated.

Table **2** shows the experimental results of some HDR images. The first column is image name, the second column is the original HDR image size before encoding, the third column is the HDR image size after encoding, the fourth column is the Visible Difference Metrics (the percentage of pixels with probability of detection) between the original HDR image and the encoded HDR image, the fifth column is the root-mean-square error between the original HDR image and the encoded HDR image and the last column shows the encoding time. Note that we implement the proposed algorithm in Matlab™. By changing to C or assembly language, the encoding time could be significantly reduced.

The input images whose names end with an extension ".exr" are provided by Industrial Light and Magic (ILM) Company. These images are compressed by wavelet-base techniques [21]. The input images whose names end with an extension ".hdr" are provided by Munsell Color Science Laboratory of Rochester Institute of Technology. These images are in run-length encoded RGBE format [20].

Two sample images from the experiment are shown in Figs. (**4** and **5**). Note that in order to display the high dynamic images on monitor, tone mapping technique [24, 25] is used to map the high dynamic images into low dynamic images. It is hard to notice the difference of these figures in Figs. (**4** and **5**) at first sight although those images are shrunk to 26.4% and 4.4% of its original size. However, after careful comparison, we find out the color of the flowers on the upper left corner of the compressed image is a little fainted

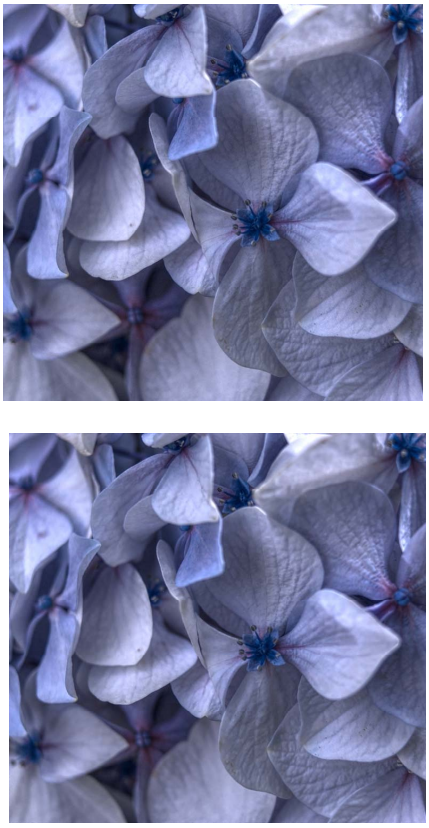

Fig. (4). (**a**) The original Flowers.exr and (**b**) its compressed version by the proposed algorithm. The displayed images are the tone mapped results of the original high dynamic range images.

in Fig. (**4**). And we also find out that the compressed image is not as crisp as the original uncompressed image in Fig. (**5**). These effects should be the combined effects of compression and tone mapping. Anyway, these tiny distortions are not easy to notice, which is consistent with the "Visible Difference Metrics" measurement in Table **2**.

Fig. (5). (**a**) The original atrium.hdr and (**b**) its compressed version by the propose algorithm. The displayed images are the tone mapped results of the original high dynamic range images.

5. CONCLUSIONS

In this paper, we present a new wavelet-based algorithm for encoding HDR images in a way that it is suitable for the internet-based multimedia communication in terms of size and accuracy. The new format reduces HDR image files to a fraction of their original size while still preserving the original high dynamic range. Experiments are conducted using the library of High Dynamic Range Images provided by the Munsell Color Science Laboratory of Rochester Institute of Technology and High Dynamic Range Images provided by the Industrial Light and Magic Company. The results show that the encoded format is able to reduce image size significantly with very low visible difference metrics and root-mean-square errors.

Several model parameters affect the performance of the proposed compression model. There is always a balance between image quality, image size and compression speed. Under the proposed framework, if we use increase the number of bit to represent luminance and chrominance channel, the quality of the image would increase but at the same time the image size would increase. Different wavelet transform would also affect the image compression, but probability the adopted Daubechies 9-tap/7-tap wavelet transform is optimal since it is also adopted in JPEG 2000 [16]. Except arithmetic coding, there are many other encoding technique like Huffman coding, which runs faster but produce larger output. People can adjust these parameters to suit their own needs, to achieve a customized balance on speed, "image size" and quality.

REFERENCES

[1] G. Ward, "High dynamic range image encodings," 2008 [Online] Available from: URL: http://www.anyhere.com/gward/hdrenc/hdr_encodings.html.

[2] K. Lewotsky, "Riding the range," *SPIE Professional*, Newsroom, Jan. 27, 2007.

[3] F. Durand and J. Dorsey, "Fast bilateral filtering for the display of high-dynamic-range images," *ACM Trans. Graph.*, vol. 21, no. 3, pp. 257-266, 2002.

[4] R. Fattal, D. Lischinski, and M. Werman, "Gradient domain high dynamic range compression," *ACM Trans. Graph.*, vol. 21, no. 3, pp. 249-256, 2002.

[5] G. W. Larson, H. Rushmeier, and C. Piatko, "A visibility matching tone reproduction operator for high dynamic range scenes," *IEEE Trans. Vis. Comput. Graph.*, vol. 3, no. 4, pp. 291-306, 1997.

[6] P. Ledda, A. Chalmers, T. Troscianko, and H. Seetzen, "Evaluation of tone mapping operators using a high dynamic range display," *ACM Trans. Graph.*, vol. 24, no. 3, pp. 640-648, 2005.

[7] P. Debevec, "High dynamic range imaging and image-based lighting," in *Intl. Conf. Computer Graphics and Interactive Techniques*, 2004.

[8] C. Schlick, "Quantization technique for visualization of high dynamic range pictures," in *Eurographics Workshop on Rendering*, 1994.

[9] H. Seetzen, W. Stuerzlinger, G. Ward, L. Whitehead, M. Trentacoste, A. Ghosh, and A. Vorozcovs,, "High dynamic range display systems," in *ACM SIGGRAPH Conferences*, 2004.

[10] E. Reinhard and K. Devlin, "Dynamic range reduction inspired by photoreceptor physiology," *IEEE Trans. Vis. Comput. Graph.*, vol. 11, no. 1, pp. 13-24, 2005.

[11] E. Reinhard, G. Ward, S. Pattanaik, and P. Debevec, *High Dynamic Range Imaging: Acquisition, Display and Image-Based Lighting*, Morgan Kaufmann, 2005.

[12] F. Y. Shih, *Image Processing and Pattern Recognition: Fundamentals and Techniques*, Wiley-IEEE Press, 2010.

[13] F. Y. Shih, *Image Processing and Mathematical Morphology: Fundamentals and Applications*, Taylor & Francis Group, CRC Press, Boca Raton, FL, 2009.

[14] H. Seetzen, W. Stuerzlinger, G. Ward, L. Whitehead, M. Trentacoste, A. Ghosh, and A. Vorozcovs, "High dynamic range display systems," *ACM Trans. Graph.*, vol. 23, no. 3, pp. 757-765, 2004.

[15] G. Ward and M. Simmons. "JPEG-HDR: a backwards-compatible, high dynamic range extension to JPEG," in *Intl. Conf. Computer Graphics and Interactive Techniques*, 2005.

[16] D. Taubman and M. Marcellin, *JPEG2000: Image Compression Fundamentals, Standards and Practice*, Springer, 2001.

[17] R. Mantiuk, G. Krawczyk, K. Myszkowski, and H.-P. Seidel, "High dynamic range image and video compression - fidelity matching human visual performance," in *IEEE Intl. Conf. Image Processing*, 2007.

[18] A. Said, "Arithmetic coding," in *Lossless Compression Handbook*, K. Sayood, Editor, Academic Press: San Diego, CA, 2003.

[19] A. Said, "Introduction to arithmetic coding theory and practice," *Hewlett-Packard Laboratories*: Palo Alto, CA, 2004.

[20] Munsell Color Science Lab., "RIT MCSL high dynamic range image database," 2008, Available from: URL: http://www.cis.rit.edu/mcsl/icam/hdr/rit_hdr/

[21] Lucas Digital Ltd. LLC., "OpenEXR sample images," 2008, Available from: URL: http://www.openexr.com/downloads.html

[22] R. Mantiuk, K. Myszkowski, and H.-P. Seidel, "Predicting visible differences in high dynamic range images - model and its

calibration," in *Human Vision and Electronic Imaging X, IS&T/SPIE's 17th Annual Symp Electronic Imaging*, 2005.

[23] R. Mantiuk, R. Myszkowski, and H.-P. Seidel, "*Visible difference predicator for high dynamic range images*," in *IEEE International Conference on Systems, Man and Cybernetics*, 2004.

[24] R. Mantiuk, K. Myszkowski, and H.-P. Seidel, "A perceptual framework for contrast processing of high dynamic range images," in *Second Symposium on Applied Perception in Graphics and Visualization*, 2005.

[25] R. Mantiuk, K. Myszkowski, and H.-P. Seidel, "A perceptual framework for contrast processing of high dynamic range images," *ACM Trans. Appl. Percept.*, vol. 3, no. 3, pp. 286-308, 2006.

Received: February 25, 2010 Revised: April 20, 2010 Accepted: April 23, 2010