Improved Video Compression using Perceptual Modeling

***Mark Q. Shaw* †*, Albert Parra* ‡ *and Jan P. Allebach* ‡**
† ***LaserJet and Enterprise Systems, Hewlett-Packard Company, Boise, ID.***
‡ ***Department of Electrical and Computer Engineering, Purdue University, West Lafayette, IN***

Abstract

In this paper we investigate a method for selectively modifying a video stream using a color contrast sensitivity model based on the human visual system. The model identifies regions of high variance with frame to frame differences that are visually imperceptible to a human observer with normal color vision. The model is based on the CIELAB and the CIE ΔE_{94} color difference formula, and takes advantage of the nature of frame-based progressive video coding.

The method was found to achieve up to 35% improvement in data compression without perceptible degradation of the video quality. As expected, the amount of compression improvement obtained is dependent on the type of video content being compressed.

Introduction

Compression of video data is becoming increasingly essential in today's information age. Video content is growing at an alarming rate, and the bandwidth requirements to support such video content are staggering.

This paper investigates a method to selectively discard inter-frame differences based on underlying assumptions about the human visual system (HVS). By taking into account the variation of the sensitivity of the HVS as a function of hue, chroma and lightness [1], the method proposed herein selectively attenuates inter-frame differences based on a variance-weighted chromatic activity map. In the developed model, the attenuated differences are then transformed and quantized in a manor similar to the standard MPEG 2 workflow, and finally encoded using Huffman Coding.

Standardized video coding frameworks have been developed by the International Telecommunication Union (ITU) and the Motion Picture Group (MPEG) [2]-[8]. Within these frameworks, the structure of the decoder is clearly defined, but the method by which the encoder creates the video stream is left to the discretion of the author of the encoder [9]. A variety of methods have been published in the literature to incorporate aspects of the HVS into the encoder. Zheng, *et al* [10] developed a method that focused on spatial frequency sensitivity. Watson *et al* [11] developed a video quality metric (DVQ) to compare two video sequences using a model that incorporates aspects of visual processing. Most notably, Yang *et al* [12] published a paper on residue pre-processing based on a just noticeable distortion profile.

The approach of using pre-processing of the video sequence is not new [13]-[15]. Leung, *et al*, optimize the compression based on visual masking. In [16] and [17], the authors evaluate the performance of various color difference models for moving images. When evaluating this work, it is important to consider that the original color difference models were developed using uniform, static solid patches of color. Applying these models to moving images introduces a new dimension that is not typically accounted for by existing color difference models.

In this paper, we outline an approach to pre-processing the video stream that incorporates principles of color science for still images and a variance based weighting to account for the impact of motion. The performance of this method is evaluated by comparing the file size of the Huffman compressed bit streams with and without our method, and by subjective evaluation of the compressed video sequences. The method was found to yield between 1% and 35% improvement in compression without visibly degrading the video quality. As expected, the amount of compression improvement obtained is dependent on the type of video content being compressed.

Theoretical Framework

Inter-frame video coding takes advantage of the fact that not every pixel within a video sequence may change significantly from one frame to the next. By removing the redundancy of unchanging pixels, the video stream will only code those pixels that are changing from frame to frame. This results in a significant improvement in the bit rate. One of the underlying assumptions is that the pixel differences to be encoded from frame to frame are perceptually significant. Research in the area of color science has shown that not all color differences are equally likely to be perceptible, the HVS sensitivity to changes in color varies as a function of lightness, chroma, and hue [18]. This can be seen in Fig. 1 wherein

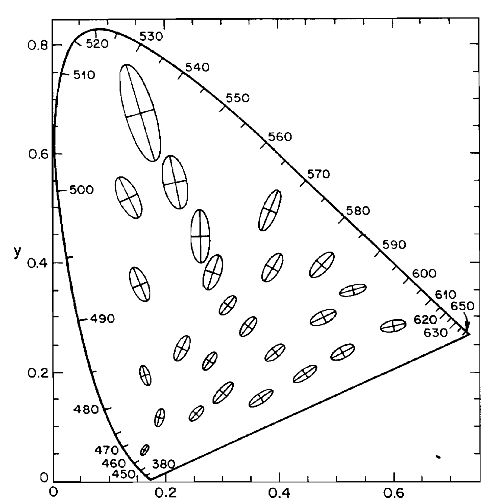


Figure 1. CIE 1931 xy chromaticity diagram showing MacAdam's Ellipses (ten times enlarged) [1]

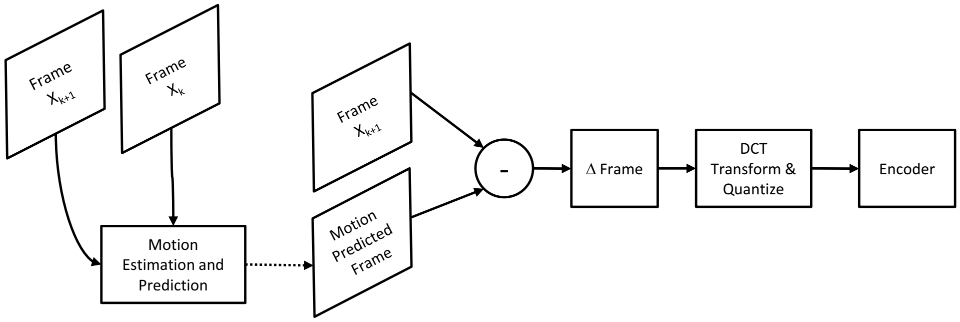


Figure 2. *Baseline video CODEC workflow for interframe video coding*

the enlarged ellipses represent loci of colors that were perceived as having a just noticeable difference (JND) to an observer. It is evident that the ellipses in the top-center (green region) are much larger than those in the bottom left (blue region). This indicates that an observer is much more sensitive to color changes in the blues, than in the greens.

More recent developments in color science have led to the standardization of color difference equations [19], known as the CIE ΔE_{ab}, ΔE_{94}, and most recently ΔE_{2000} equations. In order to use such a color difference model, first the colors to be compared must be converted from the source color space into a perceptual color space, such as CIELAB. CIELAB models the perception of a human observer with normal, two degree color vision according to the simplified form of the CIELAB equations in Eq. 1, where X, Y, Z are the tristimulus values of the color under observation, and X_n, Y_n, Z_n are the tristimulus values of the reference white.

$$
\begin{aligned}
L^* &= 116\left(\frac{Y}{Y_n}\right)^{\frac{1}{3}} - 16 \\
a* &= 500\left[\ \left(\frac{X}{X_n}\right)^{\frac{1}{3}} - \left(\frac{Y}{Y_n}\right)^{\frac{1}{3}}\ \right] \qquad (1) \\
b* &= 200\left[\ \left(\frac{Y}{Y_n}\right)^{\frac{1}{3}} - \left(\frac{Z}{Z_n}\right)^{\frac{1}{3}}\ \right]
\end{aligned}
$$

For a solid color patch, once the CIELAB coordinates have been computed for two different samples, the color difference can then be computed using the color difference equations. ΔE_{94} and ΔE_{2000} are more commonly used because they attempt to account for the non-linear dependency on hue and chroma of the samples. The ΔE_{94} color difference Eq. is calculated using

$$
\Delta E_{94} = \sqrt{\ \left(\frac{\Delta L^*}{k_l}\right)^2 + \left(\frac{\Delta C^*_{ab}}{1+K_1 C^*_1}\right)^2 + \left(\frac{\Delta H^*_{ab}}{1+K_2 C^*_2}\right)^2\ }, \qquad (2)
$$

where the constants are defined in [19]. The ellipses of equivalent just noticeable difference, when converted to the CIELAB color space calculated using the ΔE_{94} equation, are shown in Fig. 3. It can be seen that relative to the CIE ΔE_{94} equations, the CIELAB color space is still quite nonuniform in terms of color difference.

It can be seen from Fig. 3 (and Eq. 2) that the perception of color differences changes as a function of the distance from the neutral axis (origin) of the $a^* - b^*$ diagram. By taking into account this feature of the HVS, a larger deviation from the actual pixel color in inter-frame coding can be allowed, as defined by Eq. 2, without a perceptible loss in image quality.

Video Encoding Workflow

In order to evaluate the potential of such a technique in a video coder, a baseline video codec workflow was constructed in Matlab, as shown in Fig. 2. This workflow uses an 8×8 block matching Adaptive Motion Estimation (AME) algorithm to determine the motion vectors from frame to frame, the MPEG 2 Discrete Cosine Transform (DCT) and quantization scheme, with lossless Huffman Coding of the inter-frame pixel differences.

The modified CODEC workflow is shown in Fig. 4. It can be seen that in addition to the standard steps, a weighting is used to perturb the inter-frame differences based on the colorimetric properties of the frames under analysis. The following steps are performed :

1. The motion predicted and current frame are converted from YUV to CIELAB using the REC 601 primaries and respective white point. The YUV reference [1,0,0] is assumed to the white point of the scene.

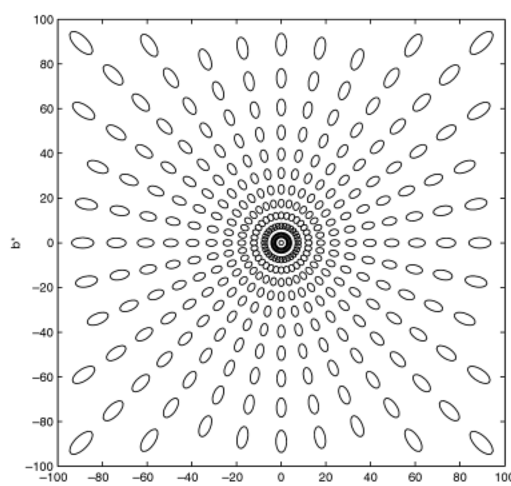


Figure 3. *JND Ellipses corresponding to a color difference of $\Delta E_{94} = 1.0$ in the a^*-b^* plane [19]*

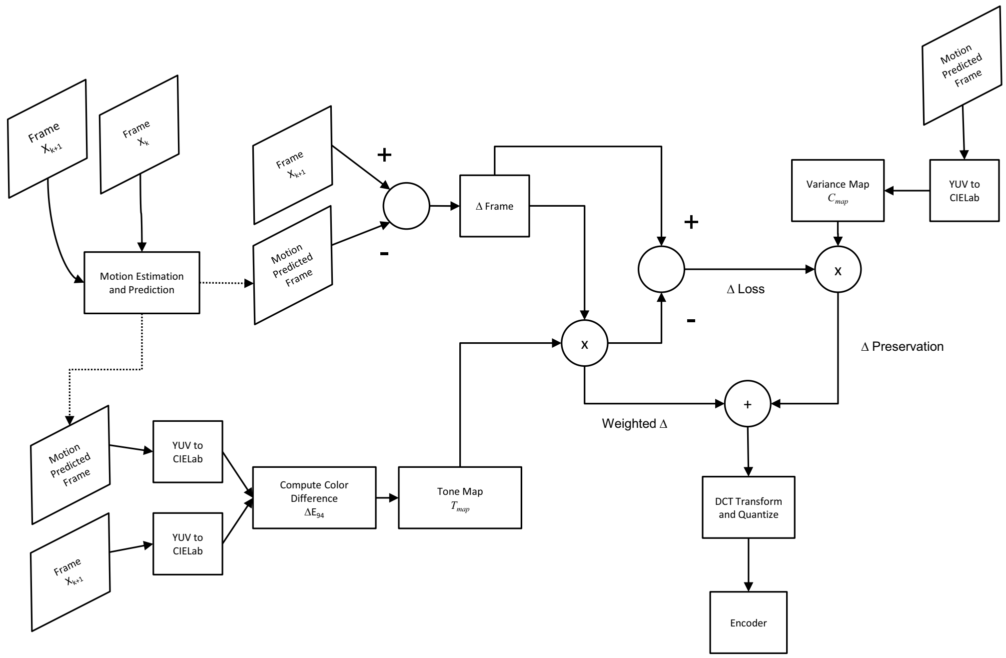


Figure 4. *Modified video CODEC workflow for inter frame video coding with color difference compression*

2. The ΔE_{94} color difference of every pixel in the image is computed between the current frame and the motion predicted frame.
3. It has been well established that the theoretical visual tolerance for a just noticeable difference is considered to be equivalent to one ΔE_{94} for large area solid color patches [1][18]. In this case, we are looking at small (pixel sized) colors that are changing as a function of time. For that reason, we propose that a variable margin of error be allowed, and define a tone mapping function that maps input color difference values to an output between $0 and 1$. The tone mapping function is shown in Fig. 5 for two levels of color thresholding. This is a tunable parameter that can influence the bit rate and quality of the resulting video sequence.
4. The tone mapped color difference image T_{map} and the Δ-frame are then point-wise multiplied. The difference between the Δ-frame and the tone mapped response, called the Δ-loss, is then multiplied by the spatial variance map C_{map} of the motion predicted frame. The resulting image called the Δ-preservation frame shows the variance of the pixels that will be removed in the encoding process.
5. Preservation of the color differences in the smooth regions of the frame is attained by applying a point-wise summation of the weighted Δ-frame and the scaled Δ-preservation frame. This is possible because the detail and texture of the video sequence are masking the visibility of the change in the color differences.

Therefore, this framework enables the selective compression of video content based on regions that have small inter-frame color differences and medium to high variance.

The modified Δ-frame pixels are then transformed and quantized using the MPEG-2 DCT transformation and quantization scheme, with lossless Huffman Coding, as in the baseline MPEG-2 CODEC workflow.

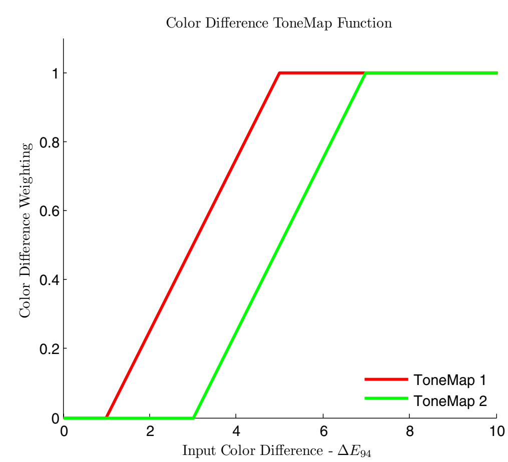


Figure 5. *Color difference tonemap function*

Test Framework

In order to compare the compression performance of the proposed method to the baseline video CODEC, the two video CODEC workflows were tested for a number of sequences listed in Table 1. To determine the bit rate savings, the DCT transformed, quantized coefficients for each frame are Huffman Coded and appended to a binary file to simulate the video bitstream. Once all of the frames have been coded, the size (number of bytes) of the two bit-streams (with and without the proposed method) are compared and the percent compression improvement is calculated using Eq. 3.

$$\%\text{ Compression} = 100 \times \left[1 - \frac{\text{size after}}{\text{size before}}\right] \quad (3)$$

Table 1. Video sequences used in testing

Sequence	Format	# of Frames
Akiyo	cif	300
BigBuckBunny	cif	14315
Bridge-Close	cif	2000
Bus	cif	150
Carphone	qcif	382
Caire	qcif	494
Coastguard	cif	300
Container	cif	300
ElephantsDream	hd	1127
Flower	cif	250
Football	cif	260
Foreman	cif	300
Hall	cif	300
Highway	cif	2000
Miss-America	qcif	150
Mobile	cif	300
Mother-Daughter	cif	300
Paris	cif	1065
Stefan	cif	90
Tempete	cif	260
Waterfall	cif	260

Given that the improvement is highly correlated to the video content, a number of video sequences were tested. The video sequences used were a range of standard test scenes acquired in the QCIF, CIF, and HD formats.

Quantitative Experimental Results

The results for compression of the different video sequences compressed using the two different tone maps shown in Fig. 5, are given in Table 2. The quantization of the DCT transformation is controlled by a stepsize parameter γ, as shown in Eq. 4. The DCT stepsize used in the quantization stage of the compressor was $\gamma = 0.25$,

$$QDCT(i,j) = \frac{DCT(i,j)}{\gamma\, Q_{stepsize}} \tag{4}$$

It can be seen from Table 2 that the compression gain can vary greatly depending on the video sequence. This is not surprising, since the algorithm is specifically looking for small color differences within the scene on a frame to frame basis. Interestingly,

(a)

(b)

Figure 6. Sequential frames from the Akiyo video sequence

Table 2. Percent compression improvement over the baseline encoder when $\gamma = 0.25$

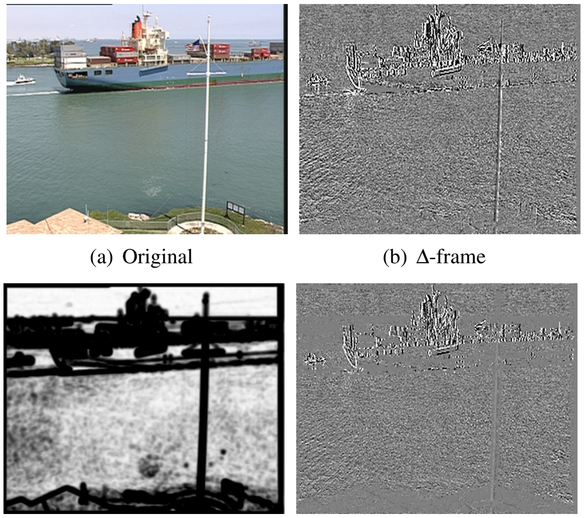
Sequence	Tone Map 1 % Compression	Tone Map 2 % Compression
Akiyo	26.2	30.9
BigBuckBunny	6.1	7.1
Bridge-Close	11.5	31.4
Bus	0.8	1.3
Carphone	9.1	14.6
Claire	20.4	24.0
Coastguard	2.7	5.8
Container	33.1	38.5
ElephantsDream	3.5	5.2
Flower	0.6	1.0
Football	2.8	5.3
Foreman	3.8	5.9
Hall	16.5	37.9
Highway	4.8	11.3
Miss-America	15.0	19.1
Mobile	0.9	1.2
Mother-Daughter	18.6	20.8
Paris	34.5	40.3
Stefan	1.9	2.3
Tempete	1.9	2.1
Waterfall	8.6	10.3
Average	10.6	15.0

the compression gains on video sequences such as *Akiyo*, *Container* and *Paris* were very significant. When watching the video sequences, it was evident that a large majority of the gain was coming from the background regions of the image that were not changing dramatically from frame to frame, as shown in Fig. 6. Although much of the differences from frame to frame were relatively small, it is important to capture the progressive nature of the image differences. Early attempts at compressing the video sequences resulted in noticeable visual artifacts appearing on the left lapel of Akiyo's jacket. This was addressed by feeding back the Δ-preservation frame, which attempts to preserve the Δ-frame changes in smooth regions. This can also be seen in Fig. 7, where the original frame and the Δ-frame are shown in (a) and (b) respectively. The Δ-loss is then attenuated using the variance image shown in (c), resulting in the corrected Δ-frame (d). By comparing (b) and (d), one can see that the intra-frame differences in the grass in the foreground at the bottom of the image and the trees just above and to the left of the stern of the ship have been compressed more than the differences in the sky and water.

As was expected, other video sequences such as *Flower*, *Mobile*, and *Bus* did not demonstrate as significant a benefit from the quantization scheme. On average the compression improvement is around 10% for ToneMap 1 and 15% for ToneMap 2, which confirms our expectation that the more aggressive ToneMap will yield greater compression.

Qualitative Analysis

In order to assess the visual quality of the video sequences when using the proposed method, a tool was developed to display



(a) Original (b) Δ-frame

(c) Variance Image (d) Corrected Δ-frame

Figure 7. *Intra-frame prediction differences for Container sequence*

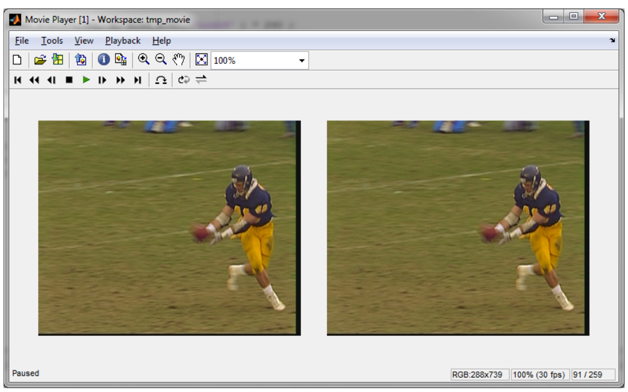


Figure 8. *Visual Comparison GUI*

two video sequences side by side for visual comparison [20]. When evaluated in this way, the video sequences quantized using ToneMap 1 and the video sequences compressed with the benchmark reference encoder were indistinguishable to the author (MQS). When comparing the video sequences quantized using ToneMap 2, some slight differences were visible, but only after many iterations of studying the same video sequence.

Conclusions

We have developed a method to improve the compression rate of the MPEG 2 framework (using inter-frame progressive coding) using a color contrast sensitivity model based on the HVS. The model is based on the principles of color discrimination, and a variance map to model the spatial activity of the motion predicted frames. Given that a human observer has particularly low sensitivity to small color differences in regions of high chroma, a different weight can be given to areas satisfying these properties in order to control the compression rate.

The quantitative analysis, using 21 different video sequences, shows that the compression savings greatly depend on the video content, obtaining gains to up to 35% with averages of 10% for ToneMap 1, and 15% for ToneMap 2. The visual quality of the compressed video sequences was assessed by one of the authors (MQS) and found to be indistinguishable for the sequences compressed with ToneMap 1.

In order to better understand the implications of this approach in a video coding workflow, the authors are investigating the possibility of implementing the algorithm directly into the JM Reference Coder. This would enable a comparison against the current industry standard encoding scheme H.264.

Acknowledgments

The authors wish to express their thanks to the Hewlett-Packard Company and the U.S. Department of Homeland Security's VACCINE Center under Award Number 2009-ST-061-CI000, for their support and partial funding of this project.

References

[1] G. Wyszecki and W. Stiles, *Color Science: Concepts and Methods, Quantitative Data and Formulae*, John Wiley and sons, New York, Third Edition, 1982.

[2] ITU-T Recommendation H.261, "H.261, video codec for audio visual services at p64 kbits/s," March 1993.

[3] ITU-T Recommendation H.263, "H.263, video coding for low bitrate communications", 1995.

[4] ITU-T Recommendation H.263, "H.263+, video coding for low bitrate communications", 1998.

[5] ITU-T Recommendation H.264, "H.264, video coding for low bitrate communications", (05/2003).

[6] ISO/IEC JTC1/SC29 International Standard 11172-2, "Mpeg-1, information technology - generic coding of moving pictures and associated audio for digital storage media at up to about 1.5 mbps - part2: Video," 1993.

[7] ISO/IEC JTC1/SC29 International Standard 13818-2, "Mpeg-2, information technology - generic coding of moving pictures and associated audio information: Video," 1995.

[8] ISO/IEC JTC1/SC29 International Standard 14496-2, "Mpeg-4, information technology - generic coding of audio-visual objects: Visual," 1999.

[9] P. V. Karunaratne, A. K. Katsaggelos and T. N. Pappas, *Pre-Processing of Compressed Digital Video Based on Perceptual Quality Metrics* Proceedings of SPIE - IS&T Electronic Imaging, SPIE Vol. 5007 (2003). pp 137-148.

[10] H. Zheng, T. Kun and C. Huijuan, *Content Adaptive Quantization Scheme Based on Visual Perception for Video Coding*, Signal Processing, 2008. ICSP 2008. 9th International Conference on, pp 1267-1270.

[11] A.B. Watson, J. Hu and J.F. McGowan III, *DVQ: A Digital Video Quality Metric Based on Human Vision*, Journal of Electronic Imaging, 10(1), pp 20-29.

[12] X. Yang, W. Lin, Z. Lu, E. Ong and S. Yao, *Motion-Compensated Residue Preprocessing in Video Coding Based on Just-Noticeable-Distortion Profile*, IEEE Transactions on Circuits and Systems for Video Technology, Vol. 15, No. 6, June 2005.

[13] R. Leung and D.Taubman, *Perceptual Optimization for Scalable Video Compression Based on Visual Masking Principles* IEEE Transactions on Circuits and Systems for Video Technology, Vol. 19, No. 3, March 2009.

[14] G. Marquant, *Perceptual PreProcessing Techniques Applied to*

Video Compression: Some Result Elements and Analysis Proceedings of the Data Compression Conference (DCC02), 2002.

[15] A. Bhat, I. Richardson and S. Kannangara, *A Novel Perceptual Quality Metric for Video Compression*, Picture Coding Symposium (PCS) 2009, California, Illinois, USA, May 2009.

[16] J.S. Kim, M.S Cho, B. K. Koo, M. R. Luo and S. Westland, *Colour Difference Modeling for Moving Images*, 4th European Conference on Color in Graphics, Imaging, Vision and 10th International Conference on Multispectral Color Science, CGIV 2008/MCS'08.

[17] R. N. Fonseca and M. A. Ramirez, *Using SCIELAB for Image and Video Quality Evaluation*, 2008 IEEE International Symposium on Consumer Electronics (2008).

[18] R.W.G Hunt, *The Reproduction of Color in Photography, Printing and Television*, Fifth Edition, Fountain Press, 1995.

[19] G. Sharma, *Digital Color Imaging Handbook*, CRC Press; 1 Edition (December 23, 2002), pp 31-40.

[20] ITU-R BT.500-11, *Methodology for the Subjective Assessment of the Quality of Television Pictures*, Tech. Rep., International Telecommunication Union/ITU, Radio Communication Sector, 2002.

Author Biography

Mark Q. Shaw received his B.Sc in Graphic Media Studies from the University of Hertfordshire, UK (1997), M.S. degree in Color Science from the Munsell Color Science Laboratory, RIT (1999), and is currently working towards a Ph.D. in Electrical and Computer Engineering at Purdue University. Mark works for the Hewlett-Packard Company as a Color and Imaging Architect. Mark has over 10 years experience in the Color and Imaging Industry, having previously worked for other major corporations including 3M and Xerox.

Albert Parra received his B.S. in Superior Telecommunications Engineering from the Technical University of Catalonia (UPC) in 2010, and his M.S. in Electrical and Computer Engineering from Purdue University in 2011. He is currently pursuing a Ph.D. in Electrical and Computer Engineering at Purdue University, and working as a Research Assistant in the Video and Image Processing Laboratory (VIPER).

Jan P. Allebach is Hewlett-Packard Distinguished Professor of Electrical and Computer Engineering at Purdue University. His current research interests include image rendering, image quality, color imaging and color measurement, document aesthetics, and printer forensics. Allebach is a Fellow of the IEEE, the Society for Imaging Science and Technology (IS&T), and SPIE. He has served as Distinguished or Visiting Lecturer for both IS&T and the IEEE Signal Processing Society. Allebach received the Senior (best paper) Award from the IEEE Signal Processing Society, the Bowman Award and the Itek Best Paper Award from IS&T, was named Electronic Imaging Scientist of the Year by IS&T and SPIE, and was named Honorary Member of IS&T, the highest award that IS&T bestows. From Purdue University, he has received six teaching awards, and separate awards for team leadership, mentoring, and research.

