Pre-print of: "Visually lossless strategies to decode and transmit JPEG2000 imagery" / L. Jiménez-Rodríguez, at al. in IEEE signal processing letters (IEEE), Jan. 2014, vol. 21, issue 1, p. 35-38. DOI 10.1109/LSP.2013.2290317

Visually Lossless Strategies to Decode and Transmit JPEG2000 Imagery

Leandro Jiménez-Rodríguez, Francesc Aulí-Llinàs, *Member, IEEE*, and Michael W. Marcellin, *Fellow, IEEE*

***Abstract*—Visually lossless coding allows image codecs to achieve high compression ratios while producing images without visually noticeable distortion. In general, visually lossless coding is approached from the point of view of the encoder, so most methods are not applicable to already compressed codestreams. This paper presents two algorithms focused on the visually lossless decoding and transmission of JPEG2000 codestreams. The proposed strategies can be employed by a decoder, or a JPIP server, to reduce the decoding or transmission rate without penalizing the visual quality of the resulting images.**

***Index Terms*—Visually lossless coding, visibility thresholds, human visual system, JPEG2000.**

I. INTRODUCTION

VISUALLY lossless coding refers to the ability of an image coding system to identify and encapsulate the information of an image that is visually relevant to a human observer. Often, this is achieved by determining visibility thresholds (VTs) for the human visual system (HVS) that are introduced into the coding system to preserve the visually relevant information [1]. In the context of transform coding, the VT for a particular transform coefficient is the maximum absolute error between the original and the coded coefficient that results in just imperceptible distortion in the image.

The use of visually lossless coding has several advantages. First, images coded in this regime look to a human observer as if they were compressed losslessly. Second, visually lossless compression achieves higher compression ratios than numerically lossless compression [2]. Third, combined with transmission protocols, visually lossless coding enhances the interactive image transmission by reducing response times [3].

Early attempts toward visually lossless coding employed the Gabor filter and the cortex transform. Currently, the discrete wavelet transform (DWT) is more commonly employed due to its suitability for both perceptual models and image coding schemes. In one of the first applications of the DWT to perceptual coding, Watson et al. measured the VTs for individual wavelet subbands based on the HVS contrast sensitivity function using randomly generated *uniform* noise as a substitute for quantization error [1]. These VTs were then employed to code the coefficients of each subband until the threshold for that subband was reached. The thresholds from [1] were introduced in the framework of JPEG2000 in [4], but the resulting images were not strictly visually lossless. This stems from the fact that JPEG2000 employs a deadzone quantizer, which introduces non-uniform quantization noise. Other approaches to obtain VTs such as [5] achieve more accurate thresholds, though they still assume uniform quantization, rather than deadzone quantization. A more suitable model for the quantization noise caused by the quantizer of JPEG2000 was proposed in [6]. When that model is applied to JPEG2000, the resulting compressed images are indistinguishable from the original ones at superior compression ratios.

Despite numerous studies on visually lossless codecs, the focus of most work has been on the encoder side. To the best of our knowledge, there are no methods to decode, or to parse and transmit, a visually lossless image from an already compressed (very high fidelity, or even numerically lossless) codestream. Since most methods are devised from the point of view of the encoder, an obvious approach would be to perform a full decoding and re-encoding. In situations where it is desirable to maintain the original (super-visually-lossless) quality, the re-encoded codestream could include side information to allow subsequent parsing of a visually lossless version. In a layered system such as JPEG2000, the re-encoded codestream could be constructed so that decoding or transmitting the first n layers would guarantee a visually lossless image. Nevertheless, there may exist large repositories of images encoded using numerically lossless or very high fidelity lossy methods. In such repositories, re-encoding may not be viable due to high computational costs. Thus, visually lossless decoding or parsing is of great interest.

Motivated by the discussion above, this work introduces strategies to decode or transmit the information necessary to reconstruct a visually lossless image from a codestream previously encoded using a *conventional* JPEG2000 encoder. Clearly, this is not possible unless the original codestream contains sufficient information to produce a visually lossless image in the first place. The goal pursued here is to provide visually lossless quality while decoding or transmitting the smallest subset possible from the original codestream. The proposed strategies employ the perceptual model of [6] to produce techniques that can be employed in a JPEG2000 decoder or in a JPIP server.

Section II of this paper overviews the model of [6] and describes the proposed strategies. Section III assesses the performance of the proposed methods through experimental results, while the last section concludes with some remarks.

Leandro Jiménez-Rodríguez and Francesc Aulí-Llinàs are with the Department of Information and Communications Engineering, Universitat Autònoma de Barcelona, Spain (phone: +34 935813571; fax: +34 935814477; e-mail: {ljimenez, fauli}@deic.uab.cat). Michael W. Marcellin is with the Department of Electrical and Computer Engineering, University of Arizona, USA (e-mail: marcellin@ece.arizona.edu). This work has been partially supported by the Spanish Government (MINECO), by FEDER, and by the Catalan Government, under Grants UAB-472-01-2/09, RYC-2010-05671, TIN2012-38102-C03-03, and 2009-SGR-1224.

II. Proposed strategies

A. Visually lossless encoding

The model of distortion produced by the JPEG2000 deadzone quantizer [6] is employed to determine VTs for wavelet subbands. To do so, a stimulus image is generated by applying the inverse DWT to wavelet data that contain simulated quantization distortions for an assumed coefficient variance σ^2 and quantization step size Δ. The inverse DWT then produces an image with a distortion corresponding to quantization error for that subband, variance, and step size. To determine the VT for the assumed subband and variance, a two-alternative forced choice method is used. In this method, the stimulus and a mid-gray level image are displayed together and a human subject decides which is the stimulus. The experiment is iterated varying Δ to find the largest Δ for which the stimulus is not distinguished from the mid-gray level image, which is then the VT for that subband and variance, denoted as $\text{VT}(\sigma^2)$.

In a JPEG2000 encoder, each subband of the DWT is quantized using an initial step size Δ_i. In this work, the initial step size for a given subband is set equal to the square root of the energy gain factor [4, Ch. 4.3.2] for that subband, although other choices are allowed by the standard. After quantization, the wavelet subbands are divided into small sets of coefficients called codeblocks. Each codeblock is coded employing three coding passes per bitplane called significance propagation (SPP), magnitude refinement (MRP), and cleanup (CP) [4]. A bitplane is defined as the collection of bits from all quantized coefficients corresponding to the same position of their binary representation. In the *en*coder of [6], the above perceptual model is applied in each codeblock as follows. First, $\text{VT}(\sigma^2_{\mathcal{B}})$ is computed employing the variance of the coefficients within codeblock $\mathcal{B}$. At the end of each coding pass, the maximum absolute error produced by the partially transmitted coefficients is computed as $D = \max_{w \in \mathcal{B}}(|w - \hat{w}|)$, with w and $\hat{w}$ denoting the original and the reconstructed coefficient, respectively. When $D \leq \text{VT}(\sigma^2_{\mathcal{B}})$, the encoding procedure is stopped.

B. Application to the decoder

In a JPEG2000 decoder, the bitstream corresponding to a codeblock is decoded from the most significant bitplane of the codeblock to the least significant bitplane, until the last coding pass included in the bitstream for that codeblock is reached. The first difficulty that arises when attempting to apply the perceptual model in the decoder is that the variance for the codeblock is not available since the image is already encoded. So an estimate for $\sigma^2_{\mathcal{B}}$ is needed. One piece of information relevant to the variance of a codeblock is the bitplane number of the most significant bitplane of the codeblock, denoted as M, which is coded in the headers of the codestream. Empirical evidence indicates that variance estimates can be obtained via M. Fig. 1 depicts the average variance of codeblocks found in three different wavelet subbands. Results for other subbands are similar. Each point in the plots corresponds to the average variance of codeblocks in one wavelet subband that have the same value of M. The results indicate that the variance of codeblocks is strongly related to the wavelet subband and to M. Note, for instance, that the average variance of codeblocks with $M < 4$ is almost zero for all subbands, and then increases exponentially as M grows. The proposed strategy employs these average variances as estimates, denoted as $\hat{\sigma}^2_{\mathcal{B}}$.

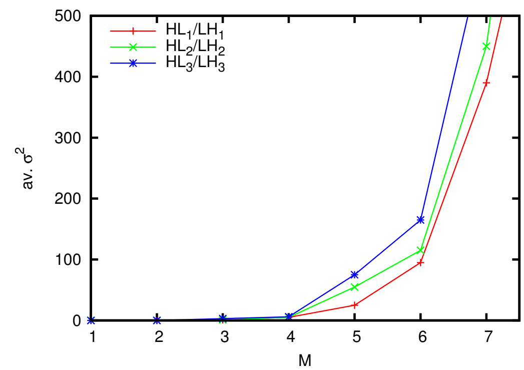

Fig. 1: Average variance of codeblocks having the same M in different subbands. Results are obtained for the images of Section III using the irreversible 9/7 DWT. Similar results are obtained for other subbands.

Another difficulty that arises is that D cannot be computed at the decoder because the original image is not available. The proposed strategy upper bounds the maximum absolute error at the end of a coding pass in bitplane P by noting that the effective (embedded) quantization step size of a coefficient, after bit P of its magnitude representation has been decoded, is $\Delta_i 2^P$. This fact, together with the knowledge of whether any coefficient from the codeblock is in the deadzone of the effective quantizer, can be used to upper bound the maximum absolute error as

$$D' = \left.\begin{array}{ll} \Delta_i 2^P & \text{if pass = CP} \\ \Delta_i 2^{P+1} & \text{otherwise} \\ \Delta_i 2^P & \text{if pass = SPP} \\ \Delta_i 2^{P-1} & \text{otherwise} \end{array} \right\} \begin{array}{l} \text{if } \exists\ \hat{w} = 0 \\ \\ \text{otherwise} \end{array} . \quad (1)$$

Masking effects can also help to reduce the (de)coding rate without sacrificing visual quality. We adopt the strategy described in [6], in which the VT for a codeblock is multiplied by a masking factor α, $\alpha > 1$ when self- and/or texture-masking are present. Since the masking factor is computed from quantized coefficients, its implementation in the decoder presents no problems.

In summary, the proposed strategy for the decoder is as follows. First, the bitplane number of the most significant bitplane M for codeblock $\mathcal{B}$ is extracted from the codestream headers. Second, the variance of the codeblock $\hat{\sigma}^2_{\mathcal{B}}$ is estimated through a lookup table containing the average variances computed for a large corpus of images. Third, the VT for the codeblock is computed using the estimated variance $\hat{\sigma}^2_{\mathcal{B}}$. Fourth, the decoding process begins and, at the end of each coding pass, the maximum error D' and the masking factor α are computed.[1] Decoding for codeblock $\mathcal{B}$ is stopped when $D' \leq \alpha\text{VT}(\hat{\sigma}^2_{\mathcal{B}})$. Evidently, if the codestream does not contain

[1] A slight increment in coding performance can be achieved by re-estimating the codeblock variance at the end of each coding pass using partially reconstructed coefficients.

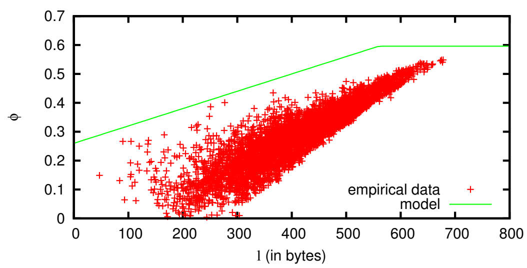

Fig. 2: Percentage of codeblock bitstream needed to reach the VT. Results are for the images of Section III when using the irreversible 9/7 DWT and 32×32 codeblocks.

enough coding passes to achieve $D' \leq \alpha \mathrm{VT}(\hat{\sigma}^2_{\mathcal{B}})$, the decoder stops the procedure after decoding the last available coding pass and then visually lossless quality cannot be guaranteed.

C. Application to JPIP servers

The application of the visually lossless decoding procedure discussed above to a JPIP server is complicated by the fact that partial decoding of the file is required. It is preferable that the server not be required to decode any bitplane data, so that neither D' nor α can be computed. The only useful information about the codeblock that is then available is M, the number of coding passes, and the length of the bitstream generated for the codeblock, denoted as l.

Experiments indicate that M and l are good indicators of the amount of data that have to be transmitted to produce a visually lossless image. This can be seen as follows. Fig. 2 depicts the percentage of a codeblock bitstream required to reach its VT. The horizontal axis of the figure is l, whereas the vertical axis is the percentage of l, denoted as ϕ, that is required to reach the VT. Each point in the scatter plot corresponds to one codeblock in the HH_1 subband having $M = 4$. When l is small, ϕ is also small. As l increases, ϕ increases, until reaching a point at which ϕ does not grow more. Similar behavior holds for other subbands and Ms.

Results corresponding to Fig. 2 are upper bounded for each wavelet subband (and each value of M) by the function

$$\phi' = \begin{cases} s \cdot l + \phi_{\text{min}} & \text{if } l < l_{\text{max}} \\ \phi_{\text{max}} & \text{otherwise} \end{cases}. \tag{2}$$

The parameters s, ϕ_{min}, and ϕ_{max} employed in the upper bound (as functions of M) are reported in Table I. The solid line in Fig. 2 depicts the upper bound of (2) for the corresponding subband and value of M. This upper bound to the actual value of ϕ was computed over a wide corpus of images, being (2) an overly conservative estimate to assure visually lossless.

The results of Fig. 2 were generated using initial step sizes as discussed in Section II-A and by including all coding passes of each codeblock bitstream. Since all images are assumed to have been previously encoded by "non-aware" JPEG2000 encoders, different initial step sizes may have been employed, and codeblocks may have some missing coding passes (due to rate allocation procedures, etc.). In the case of missing coding passes (only), the resulting difference in l is approximated by noting that missing passes correspond to the least significant bitplanes, which are nearly incompressible. Thus, the length of such coding passes is well approximated by one bit per coefficient per bitplane. In the case of different initial step sizes, the resulting difference in l can be approximated by $\log_2$ of the ratio between the true and the assumed step size, in units of bits per coefficient. The true step size can be read from the codestream headers.

In summary, the proposed strategy for the JPIP server is as follows. First, M and l are extracted (or in the case of l, estimated as needed) from the codestream headers. Second, the percentage of each codeblock bitstream that needs to be transmitted to achieve a visually lossless image is computed via (2). Third, the server transmits the corresponding portions of the codeblock bitstreams to the client. Fourth, the client decodes data until reaching the end of each codeblock bitstream segment. The decoder must be aware that the end of a bitstream segment may not coincide with the end of a coding pass, so it must stop when all bytes are consumed (see [7]).

TABLE I: Parameters for the upper bound to ϕ as a function of M.

	$s = s_1 \cdot M + s_2$		$\phi_{min} = n_1 \cdot M + n_2$		$\phi_{max} = m_1 \cdot M + m_2$	
	s_1	s_2	n_1	n_2	m_1	m_2
HH_1	-0.000105	0.00102	90	260	0.045	0.08
HL_1/LH_1	-0.00014	0.0012	80	260	0.055	0
HH_2	-0.000172	0.00155	90	260	0.072	0
HL_2/LH_2	-0.000191	0.00172	80	260	0.067	0
HH_3	-0.000155	0.00155	90	260	0.048	0
HL_3/LH_3	-0.00012	0.0012	80	260	0.06	0
HH_4	-0.000165	0.0018	90	260	0.048	0
HL_4/LH_4	-0.00013	0.00135	80	260	0.06	0
HH_5	-0.000165	0.0018	90	260	0.048	0
HL_5/LH_5	-0.00013	0.0014	80	260	0.06	0

III. Experimental results

Experimental results are reported in Table II (all images are 8 bit, grayscale). The JPEG2000 coding parameters employed are: 5 levels of DWT, and codeblocks of size 32×32. The reversible 5/3 DWT is employed for numerically lossless results, otherwise the irreversible 9/7 DWT is used. A three-alternative forced-choice (3AFC) procedure is used to validate the results, using the same procedures and viewing conditions as those in [6]. The 3AFC test is performed with a HP ZR2440w monitor that has an IPS panel, contrast ratio of 1:1000, brightness of 350 cd/m^2, and a dot pitch of 0.27mm. A total of 12 subjects participated in the validation test. When the images are visually lossless, the probability of correct response for the 3AFC test should be $1/3$. The 95% confidence intervals for the mean frequency at which observers selected the correct image in this test are reported in the first row of the table. When the appropriate confidence interval contains $1/3$, the images are visually lossless for these viewing conditions.

Table II includes compression results (in bps) for the strategy of Section II-B (labeled "decoder") and for the strategy of Section II-C (labeled "server"). Also included for comparison are results for numerically lossless encoding, and for the encoder based procedure of [6] (labeled "encoder"). The 3AFC results achieved by the "encoder," "decoder," and "server" strategies suggest that each produces visually lossless images.

	validation test				.346 ± .05			.350 ± .05			.488 ± .15			.431 ± .09		
	lossless	**encoder** [6]			**decoder**			**server**			**server -40%**			**server -2BP**		
image (size)	**bps**	**bps**	**dB**	SSIM	**bps**	**dB**	SSIM	**bps**	**dB**	SSIM	**bps**	**dB**	SSIM	**bps**	**dB**	SSIM
barbara (512×512)	4.79	1.69	39.68	.9988	1.76	40.25	.9990	3.09	48.05	.9998	1.91	41.34	.9992	2.08	43.61	.9952
boats (512×512)	4.42	1.48	41.11	.9991	1.52	41.40	.9991	2.76	48.09	.9998	1.71	42.09	.9993	1.74	43.68	.9960
frog* (621×498)	6.28	3.17	38.42	.9965	3.53	40.38	.9978	4.70	47.86	.9996	2.87	37.52	.9957	3.96	44.51	.9787
goldhill* (512×512)	4.85	1.92	40.66	.9988	2.01	41.21	.9990	3.33	48.38	.9998	2.05	41.11	.9990	2.25	43.30	.9951
horse* (512×512)	5.26	2.24	39.73	.9992	2.32	40.16	.9993	3.67	48.19	.9999	2.26	39.58	.9992	2.75	43.98	.9953
lena* (512×512)	4.33	1.42	41.62	.9990	1.46	41.83	.9991	2.66	47.92	.9998	1.65	42.48	.9992	1.56	43.15	.9967
baboon* (512×512)	6.12	2.78	37.51	.9968	2.93	37.99	.9971	4.58	48.29	.9997	2.80	37.65	.9969	3.71	44.04	.9845
mountain* (640×480)	6.71	2.77	33.99	.9980	2.92	34.41	.9982	4.93	46.30	.9999	3.02	34.90	.9984	4.41	44.68	.9921
onthepad (512×512)	6.52	3.03	36.51	.9986	3.12	36.65	.9986	4.95	48.27	.9999	3.03	37.20	.9988	4.18	44.52	.9939
peppers* (512×512)	4.63	1.62	40.34	.9989	1.66	40.60	.9991	3.04	48.05	.9998	1.88	41.95	.9994	1.94	42.98	.9975
thecook* (512×512)	5.49	2.59	39.65	.9991	2.63	39.78	.9992	4.01	48.60	.9999	2.46	39.52	.9991	3.07	43.78	.9958
zelda* (512×512)	4.01	1.16	42.45	.9994	1.18	42.53	.9989	2.32	48.00	.9997	1.44	43.25	.9991	1.20	43.14	.9968
man (1024×1024)	4.84	1.85	40.46	.9992	1.94	40.90	.9992	3.26	48.21	.9999	2.02	41.28	.9993	2.18	43.13	.9968
woman* (600×800)	3.12	0.86	44.89	.9993	0.90	45.21	.9995	1.32	48.46	.9998	0.84	43.82	.9993	0.88	46.21	.9964
portrait (2048×2560)	4.41	1.57	41.25	.9975	1.64	41.67	.9993	2.73	47.97	.9998	1.68	41.23	.9993	1.78	43.71	.9963
flowers* (600×800)	3.36	1.04	44.00	.9993	1.10	44.56	.9994	1.61	49.02	.9998	1.01	43.27	.9992	1.16	46.20	.9953
cafeteria* (600×750)	6.11	2.42	35.14	.9993	2.54	35.45	.9977	4.31	46.63	.9998	2.65	36.50	.9982	3.73	44.64	.9903
fishing* (600×800)	4.73	1.83	40.67	.9991	1.90	41.04	.9994	2.92	47.15	.9998	1.81	40.32	.9993	2.18	44.11	.9962
fruit* (600×750)	4.49	1.68	41.31	.9991	1.74	41.66	.9994	2.76	48.12	.9997	1.71	41.15	.9993	1.95	44.38	.9964
japanese* (600×800)	5.07	2.18	40.03	.9952	2.26	40.37	.9991	3.35	47.00	.9998	2.07	39.23	.9989	2.68	44.36	.9946
tableware (600×750)	4.51	1.33	39.32	.9993	1.38	39.49	.9992	2.69	47.39	.9999	1.66	41.30	.9994	1.81	44.02	.9960
fieldfire* (600×800)	4.55	1.75	41.89	.9986	1.86	42.23	.9956	2.87	46.98	.9985	1.77	40.87	.9940	1.90	43.21	.9757
bicycle (2048×2560)	4.40	1.48	40.41	.9996	1.54	40.87	.9994	2.70	48.00	.9999	1.67	41.77	.9995	1.80	43.91	.9969
pier* (600×800)	4.80	1.88	39.29	.9993	1.97	39.88	.9988	3.09	47.56	.9998	1.91	38.81	.9985	2.53	45.19	.9922
orchid* (600×750)	3.55	0.82	43.38	.9997	0.86	43.65	.9997	1.71	48.13	.9999	1.07	44.31	.9997	0.91	45.01	.9980
threads* (600×800)	4.14	1.48	41.69	.9993	1.54	42.13	.9993	2.34	47.88	.9998	1.46	41.05	.9992	1.74	45.13	.9956
musicians (600×750)	5.53	2.19	37.52	.9982	2.24	37.68	.9982	3.78	46.92	.9998	2.33	39.05	.9987	2.99	43.64	.9937
silver (600×800)	3.67	1.19	42.75	.9993	1.25	43.40	.9994	1.80	48.24	.9998	1.13	41.86	.9991	1.38	46.06	.9947
candle* (600×750)	6.16	2.52	35.66	.9973	2.59	35.79	.9973	4.38	46.71	.9998	2.69	37.08	.9980	3.78	44.39	.9897
average	*4.86*	*1.86*	*40.05*	*.9986*	*1.94*	*40.45*	*.9988*	*3.16*	*47.81*	*.9998*	*1.95*	*40.40*	*.9987*	*2.35*	*44.23*	*.9935*

TABLE II: Results achieved by the proposed strategies. Images with $*$ are those used in the validation test.

The rates achieved by the decoder are always only slightly larger than those of the encoder. These small differences are due to the use of estimates for the variance and maximum absolute distortion in each codeblock. On the other hand, due to the conservative upper bounds employed for ϕ in the server strategy, its rates are larger than those of the encoder strategy, though still substantially lower than for numerically lossless. Thus, it is of interest to consider less conservative strategies.

As mentioned previously, the upper bounds employed above were computed from a very large corpus of imagery containing images of different types. As the upper bounds apply to every image in this corpus, the proposed system is very robust to images with different statistical properties. A less conservative strategy that might lead to lower encoding rates for certain image types would be to compute different upper bounds for different classes of imagery. We do not pursue this strategy here due to space constraints, as well as our preference for a universal scheme that does not rely on prior knowledge of image types. Rather, the final two columns in the table represent alternate strategies, which decrease the file size significantly, but do not guarantee visually lossless quality. In particular, the strategy labeled "server -40%" is the same strategy as "server" but reduces ϕ' by 40%, resulting in an average rate similar to that achieved by the "decoder" strategy. Observers found that most images are visually lossless (and all have very high quality) for this strategy, so it may be good enough when strictly visually lossless is not required. The strategy labeled "server -2BP" omits the coding passes from the two least significant bitplanes of codestreams produced by the "server" strategy, which also produces slightly visible distortion in some images. For completeness, the PSNR and SSIM achieved for each image is also reported in Table II. Visually lossless images are achieved from 30 to 45 dB, all with SSIM values higher than 0.99.

IV. Conclusions

In general, visually lossless coding methods are done from the perspective of the encoder, and assume that the original image is available. If the image is already coded, most methods cannot identify the visually relevant information within the codestream without fully re-encoding the image. We propose strategies for the decoding and transmission of JPEG2000 codestreams that produce visually lossless images. The proposed strategies can be employed in a decoder, transcoder, or JPIP server to reduce the decoding or transmission rate without penalizing the visual quality of the images.

References

[1] A. Watson, G. Yang, J. Solomon, and J. Villasenor, "Visibility of wavelet quantization noise," *IEEE Transactions on Image Processing*, vol. 6, no. 8, pp. 1164–1175, Aug. 1997.

[2] Z. Liu, L. Karam, and A. Watson, "JPEG2000 encoding with perceptual distortion control," *IEEE Transactions on Image Processing*, vol. 15, no. 7, pp. 1763–1778, Jul. 2006.

[3] H. Oh, A. Bilgin, and M. W. Marcellin, "Visually lossless JPEG2000 at fractional resolutions," in *IEEE International Conference on Image Processing*, Sep. 2011, pp. 309–312.

[4] D. S. Taubman and M. W. Marcellin, *JPEG2000 Image compression fundamentals, standards and practice*. Norwell, Massachusetts 02061 USA: Kluwer Academic Publishers, 2002.

[5] D. Chandler and S. Hemami, "Dynamic contrast-based quantization for lossy wavelet image compression," *IEEE Transactions on Image Processing*, vol. 14, no. 4, pp. 397–410, Apr. 2005.

[6] H. Oh, A. Bilgin, and M. W. Marcellin, "Visually lossless encoding for JPEG2000," *IEEE Transactions on Image Processing*, vol. 22, no. 3, pp. 189–201, Jan. 2013.

[7] F. Auli-Llinas, J. Bartrina-Rapesta, and J. Serra-Sagrista, "Enhanced JPEG2000 quality scalability through block-wise layer truncation," *EURASIP Journal on Advances in Signal Processing*, vol. 2010, pp. 1–11, May 2010, article ID 803542.