IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received July 19, 2021, accepted July 29, 2021, date of publication August 3, 2021, date of current version August 19, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3102235

Improvement of JPEG XL Lossy Image Coding Using Region Adaptive DCT Block Partitioning Structure

JOONHYUNG CHO, OH-JIN KWON, AND SEUNGCHEOL CHOI
Department of Electrical Engineering, Sejong University, Seoul 05006, Republic of Korea

Corresponding author: Oh-Jin Kwon (ojkwon@sejong.ac.kr)

This work was supported by the Institute for Information and Communications Technology Promotion (IITP) funded by Korea Government, development of JPEG systems standard for snack culture contents, under Grant 2020-0-00347.

ABSTRACT The Joint Picture Expert Group (JPEG) committee has been standardizing next-generation image compression, called JPEG XL, to meet the specific needs for a responsive web, wide color gamut, and high dynamic range. JPEG XL supports lossy and lossless compression. A variable-sized discrete cosine transform (DCT) block is used for lossy compression. A block partitioning method is regarded as a critical function for the performance of JPEG XL. The current DCT block partitioning method used in JPEG XL is highly dependent on the compression rate and tends to assign small-sized DCT blocks to homogeneously textured regions (HTRs) having similar or regular patterns. We propose a region-adaptive DCT block partitioning method that assigns larger blocks to the HTR. The proposed method identifies the HTRs by using a combined metric employing a sum-modified Laplacian, zero-crossing, and colorfulness metric for measuring the region homogeneity. Objective, subjective, and visual comparison evaluations with the ten images recommended by the JPEG working group were provided to show the improvement in coding performance. The proposed method shows its superiority in terms of the compression efficiency evaluated using six objective metrics, subjective tests with 15 participants, visual comparison improvements in the HTR, and gains in the execution time.

INDEX TERMS JPEG XL, DCT block partitioning structure, homogeneous region, sum-modified Laplacian, zero-crossing, colorfulness.

I. INTRODUCTION

With the advent of modern information and communication technology, millions of images are generated by mobile phones and digital cameras, uploaded, and shared through various applications such as cloud services and social networking services. Currently, the most widely used image compression format in the digital imaging industry is the Joint Picture Expert Group (JPEG) standard (ISO/IEC 10918-1) [1]. However, with an increase in the recent demand for supporting massive images, such as 360-degree and high-dynamic range (HDR) images, an image compression format with better coding performance is needed for a band-limited network transmission of images. Therefore, to address this demand, the JPEG committee has begun developing the next-generation image coding standard, JPEG XL (ISO/IEC 18181) in March 2019 [2].

JPEG XL is being developed to provide various features and is specially designed to be optimized for web environments that render images on various devices. It supports lossless transcoding from the JPEG format, which is a useful feature for reducing storage costs because this feature enables servers to store a single JPEG XL file to be serviced on both JPEG and JPEG XL clients. JPEG XL also aims to support wide color gamut coding; HDR coding; lossless/visually lossless coding; alpha channel/transparency coding; progressive coding in terms of quality, spatial resolution, and scanning order; animation image sequences; and significant compression efficiency improvements over commonly used coding standards for high-quality imaging and professional photography [3].

In the JPEG standard, the input image was converted into the YCbCr color coordinate system. Each of the Y, Cb,

The associate editor coordinating the review of this manuscript and approving it for publication was Andrea F. Abate.

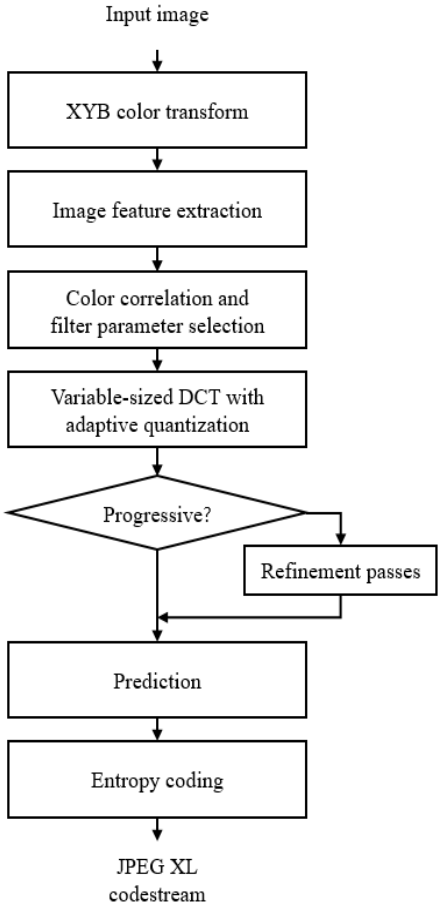

FIGURE 1. Overview of the encoding process of JPEG XL lossy image coding [5].

and Cr color components were divided into non-overlapping 8 × 8 blocks. The 8 × 8 discrete cosine transform (DCT) is applied to each color component, quantized, and losslessly entropy coded independently. A more advanced image coding standard, such as the high-efficiency video coding (HEVC) intra mode standard [4], [5] and versatile video coding (VVC) [6], [7] improve the coding efficiency of JPEG by using variable-sized DCT blocks instead of using a fixed-sized 8 × 8 DCT block. JPEG XL is also designed to adopt variable-sized DCT blocks for its image coding strategy. In this study, a JPEG XL block partitioning method is introduced and a region-adaptive DCT block partitioning method is proposed to improve the coding efficiency of JPEG XL.

The remainder of this paper is organized as follows. In Section II, the JPEG XL image coding standard is reviewed, and the current block partitioning method is introduced. Section III describes the proposed region-adaptive DCT block-partitioning method. Section IV presents the experimental results showing that the proposed method outperforms the current JPEG XL method, and Section V provides some concluding remarks regarding this study.

II. JPEG XL OVERVIEW

A. OVERVIEW OF JPEG XL STANDARD

The international image coding standards, including JPEG, HEVC, and JPEG XL, define the normative decoding process only. They do not define their encoding processes. Therefore, as long as the encoded bitstream is compatible with the standard decoder, the encoding process is free from using its own various encoding functions. Although JPEG XL supports both lossy and lossless compression modes, we overview only the lossy compression mode in this section because we focus solely on the improvement of the block partitioning method applied to the lossy compression mode.

Figure 1 shows the corresponding encoding process recommended by the JPEG committee [8]. As shown in Fig. 1, the encoding process of JPEG XL is defined as a flow of functional blocks, and the decoding process is conducted in a process opposite the encoding process. Each block performs different types of processing as follows:

- The XYB color transform transforms the input color into XYB color, which is a hybrid color model inspired by the human visual system.
- Image feature extraction allows for more precise and economical representations.
- Color correlation allows the decoder to predict the chroma from the luma.
- A filter parameter selection helps suppress artifacts while preserving fine details.
- Variable-sized DCT serves as a fast approximation of the optimal decorrelating transform.
- Adaptive quantization uses per-block quantization step sizes
- Prediction provides an adaptive, pixel-by-pixel decorrelation without side information.
- Entropy coding is performed using an asymmetric numeral system (ANS) or Huffman.

Concerning the flexible block size, JPEG XL supports a variable-sized DCT block that is square or rectangular with a size of 2 × 2 to 256 × 256. Within the variable-sized DCT block, a block size optimized for the coding performance should be provided, which operates differently depending on the encoder implementations. Because DCT blocks of larger than 64 × 64 in size typically occupy an extremely small bitrate in JPEG XL image coding, we only consider DCT blocks with a size of 64 × 64 or smaller.

B. OVERVIEW OF JPEG XL DCT BLOCK PARTITIONING

The JPEG XL block partitioning process is applied to every 64 × 64 block by calculating the entropy of each possible child-blocks whose size ranges from 64 × 64 to 2 × 2. The size of the child-blocks was mainly determined based on the desired compression ratio. Figure 2 shows the result of the process according to the target bitrate, in bits per pixel (bpp), using the JPEG XL verification model, called JPEG XLM version 0.2 [9]. When the target bpp is set to 1.0, most block sizes are determined as 8 × 8 or smaller, except for flat

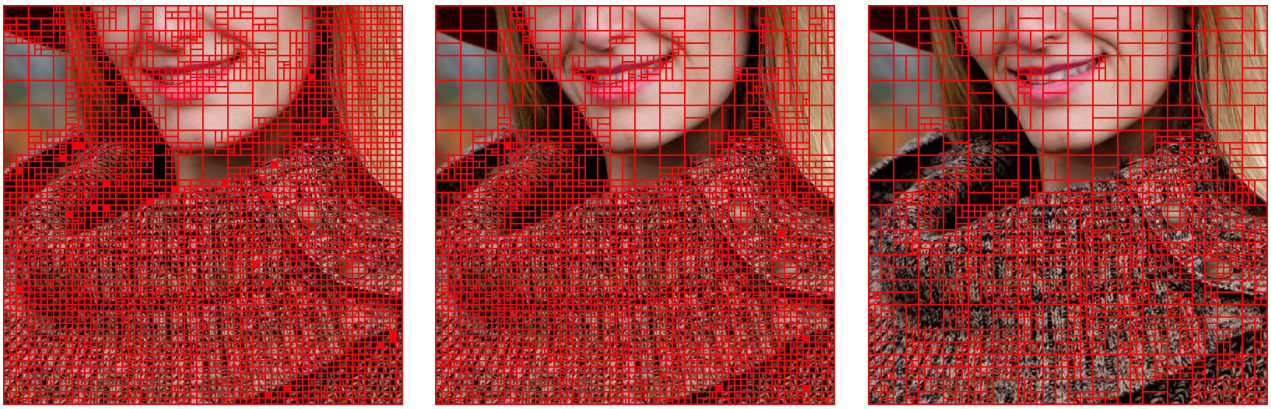

FIGURE 2. Images representing DCT block sizes of JPEG XL depending on target bpp.

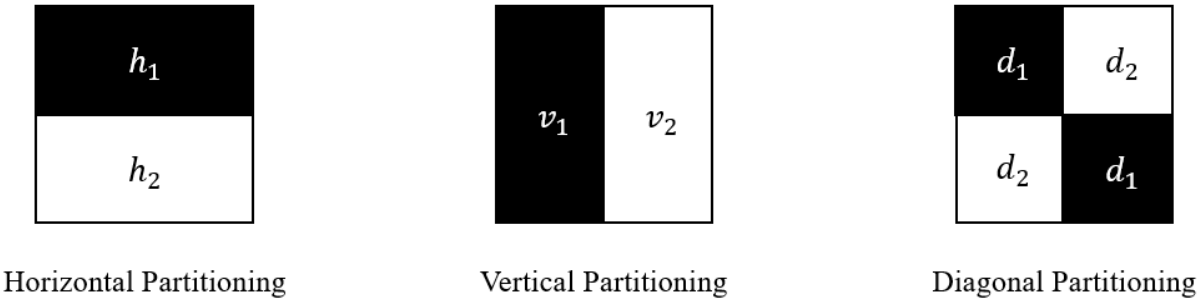

FIGURE 3. Three types of proposed block partitioning.

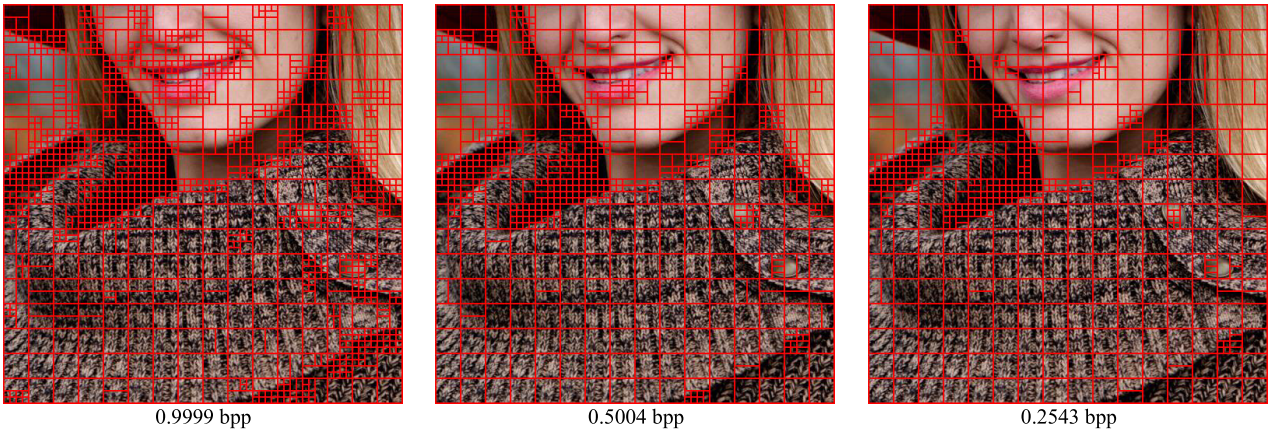

FIGURE 4. Images representing DCT block sizes of proposed partitioning.

regions such as a smooth face and blurred background in the image. As the target bpp decreases, larger blocks tend to be used. Therefore, it was found that the partitioning process of the JPEG XLM is dependent mainly on the target compression ratio, regardless of the visual characteristics of the block regions. In this paper, we propose a more efficient and region-adaptive block partitioning method that improves the encoding performance of the current JPEG XLM based on the features that allow humans to perceive a better image quality.

III. PROPOSED DCT BLOCK PARTITIONING

A. PROPOSED TEXTURE HOMOGENEITY METRIC

A homogeneously textured region (HTR) indicates a region with a similar texture regularity. The characteristics of the HTR can be utilized in various ways, and various studies on HTR have been conducted [10]–[12]. Considering that merging blocks in an HTR improves the image coding efficiency, we propose a texture homogeneity metric that combines three metrics: zero-crossing (ZC) [13], sum-modified Laplacian (SML) [14], and colorfulness $\hat{M}^{(3)}$ [15]. These three metrics

FIGURE 5. Test images used in JPEG XL WG.

were chosen based on their effectiveness and the possibility of fast calculation [16]. The definitions of ZC, SML, and colorfulness used in this study are referenced from [16].

It should be noted that ZC and SML are normally computed using the luminance component. The XYB color space is a hybrid color model inspired by the human visual system, which is used in JPEG XL lossy image coding. In the XYB color space, the Y component behaves as a luminance component [17], which is used for calculating ZC and SML.

B. REGION ADAPTIVE DCT BLOCK PARTITIONING

For implementation of the entire image coding, we modified the reference software provided by the JPEG XL working group (WG) [9].

Since we noted that the 64 × 64 block, which is the maximum size used in the current JPEG XL, causes an excessive blurring effect visually in the reconstructed image, we limit the maximum DCT block size to 32 × 32. The input image is first divided into 32 × 32 sized blocks. A mother-block whose size is initially set to 32 × 32 is partitioned into two child-blocks in three types: horizontal, vertical, and diagonal partitioning. Figure 3 shows the two child-blocks of three types denoted in black and white region, respectively. The average values: a_b of the black region and a_w of the white region, are calculated based on each of the three homogeneity metrics, respectively. For three partitioning types, we calculate the value which is the greater value between (a_b/a_w) and (a_w/a_b) and denote each of them as R_h, R_v, and R_d for horizontal, vertical, and diagonal partitioning type, respectively. If this ratio is larger than a threshold, the mother-block is divided into child-blocks as shown in Fig. 3.

For each $N \times N$ mother-block, the diagonal partitioning is checked first. If R_d is larger than a threshold, $N \times N$ block is partitioned to four $\frac{N}{2} \times \frac{N}{2}$ child-blocks and we finish the partitioning for this mother-block. If not, we decide which one of horizontal or vertical partitioning is checked next. If R_h is larger than R_v, the horizontal partitioning is checked. Otherwise, the vertical partitioning is checked. If one of horizontal or vertical partitioning is performed, we check if we have to proceed further dividing $N \times \frac{N}{2}$ or $\frac{N}{2} \times N$ sized block into $\frac{N}{2} \times \frac{N}{2}$ sized blocks or not and perform the partitioning in the same fashion. When the above procedures for a $N \times N$ sized mother-block is complete, the mother-block results into $N \times N$ (no partitioning case), $N \times \frac{N}{2}$, $\frac{N}{2} \times N$, and $\frac{N}{2} \times \frac{N}{2}$ sized blocks. An example of pseudo code can be introduced as follows:

Procedure Partitioning ($L \times L\_Block$, *Butteraugli_Target*)

Input: $L \times L\_block$, *Butteraugli_target*
Output: *Partitioned_M* × *N _blocks*
// *Butteraugli_target* is the compression ratio in JPEG XLM

// Set the threshold value r for determining whether to partition
$r \leftarrow 1.35$
if *Butteraugli_target* > 8.0
$r \leftarrow 1.50$
else if *Butteraugli_target* < 2.0
$r \leftarrow 1.20$

$h_M \leftarrow \max(h_1, h_2)$
$h_m \leftarrow \min(h_1, h_2)$
$v_M \leftarrow \max(v_1, v_2)$
$v_m \leftarrow \min(v_1, v_2)$
$d_M \leftarrow \max(d_1, d_2)$
$d_m \leftarrow \min(d_1, d_2)$

// Calculate ratio to each of three modes
$R_h \leftarrow h_M/h_m$
$R_v \leftarrow v_M/v_m$
$R_d \leftarrow d_M/d_m$
// Choose one of three modes

15: if $R_d > r$
16: Partition $L \times L$ block to $\frac{L}{2} \times \frac{L}{2}$ blocks
17: else
18: if $R_h > R_v$
19: Call procedure HorizontalPartition ($L \times L\_block$, R_h, r)
20: else
21: Call procedure VerticalPartition ($L \times L\_block$, R_v, r)

where

Procedure HorizontalPartition ($L \times L\_block$, R_h, r)

1: $t_1 \leftarrow$ value calculated by metrics in top-left block
2: $t_2 \leftarrow$ value calculated by metrics in top-right block
3: $b_1 \leftarrow$ value calculated by metrics in bottom-left block
4: $b_2 \leftarrow$ value calculated by metrics in bottom-right block

5: $t_M \leftarrow \max(t_1, t_2)$
6: $t_m \leftarrow \min(t_1, t_2)$
7: $b_M \leftarrow \max(b_1, b_2)$
8: $b_m \leftarrow \min(b_1, b_2)$

9: $R_t \leftarrow t_M/t_m$
10: $R_b \leftarrow b_M/b_m$

11: if $R_h > r$
12: Partition $L \times L$ block to $\frac{L}{2} \times L$ blocks
12: if $R_t > r$
13: Partition top $\frac{L}{2} \times L$ block to $\frac{L}{2} \times \frac{L}{2}$ blocks
14: if $R_b > r$
15: Partition bottom $\frac{L}{2} \times L$ block to $\frac{L}{2} \times \frac{L}{2}$ blocks

Procedure VerticalPartition ($L \times L\_block$, R_v, r)

1: $l_1 \leftarrow$ value calculated by metrics in top-left block
2: $l_2 \leftarrow$ value calculated by metrics in bottom-left block
3: $r_1 \leftarrow$ value calculated by metrics in top-right block
4: $r_2 \leftarrow$ value calculated by metrics in bottom-right block

5: $l_M \leftarrow \max(l_1, l_2)$
6: $l_m \leftarrow \min(l_1, l_2)$
7: $r_M \leftarrow \max(r_1, r_2)$
8: $r_m \leftarrow \min(r_1, r_2)$

9: $R_l \leftarrow l_M/l_m$
10: $R_r \leftarrow r_M/r_m$

11: if $R_v > r$
12: Partition $L \times L$ block to $L \times \frac{L}{2}$ blocks
12: if $R_l > r$
13: Partition left $L \times \frac{L}{2}$ block to $\frac{L}{2} \times \frac{L}{2}$ blocks
14: if $R_r > r$
15: Partition right $L \times \frac{L}{2}$ block to $\frac{L}{2} \times \frac{L}{2}$ blocks

Our proposed partitioning method is performed by the following order:

1) Perform partitioning for every 32×32 blocks using SML metric.
2) Perform partitioning for every 16×16 blocks resulting from 1) using SML metric.
3) Repeat 1) and 2) using ZC and then colorfulness metric.

Figure 4 shows the results of the partitioned DCT blocks using the proposed method. Compared to the result of JPEG XL shown in Fig. 2, the size of the DCT block is obviously larger in HTRs at comparable bit rates. In particular, in the HTR of women's clothes, the proposed method mainly uses a DCT with a block size of 32×32 because it determines that there is no need to partition the blocks.

IV. EXPERIMENTAL RESULTS

We evaluated the performance of the proposed block partitioning method compared with state-of-the-art JPEG XL block partitioning. Ten sample images recommended by the JPEG WG were chosen for our test, as shown in Fig. 5 [18].

First, we conduct a comparison using objective image quality metrics to measure the distortion between the original and coded images. We selected seven metrics, i.e., those mostly adopted in the field of image coding research and additional metrics recently recommended by the JPEG WG [19], [20]:

- the PSNR as a quality metric in terms of the ratio between the maximum image value and the mean square error;
- Butteraugli [21], which estimates the psychovisual similarity;
- CIEDE2000 [22], which calculates the difference between two colors in the CIELab color space;
- feature similarity index metric (FSIM) [23], which uses phase congruency and gradient magnitude to assess the local image quality;
- structural similarity index metric (SSIM) [24] focusing on structural image differences;
- multi-scale SSIM (MS-SSIM) [25], which computes the SSIM over different resolutions to represent the quality at different resolutions and viewing conditions;
- visual information fidelity (VIF) [26], which uses natural image statistics to exploit the human visual system.

The objective test results are shown in Figs. 6 and 7. Table 1 provides the average values over the sample images for each metric. Table 2 lists all values drawn in Figs. 6 and 7. Note that the SSIM, MS-SSIM, FSIM, and VIF metrics use only the luminance component for their measurements. We obtained our test results for specified bit rates of 0.25, 0.50, 0.75, and 1.00 bpp by adjusting the quality factor of JPEG XLM through control of the targeted bit rate. Our observations of the objective results can be summarized as follows:

- The proposed method is more effective on average at lower bit rates than at higher bit rates.
- The proposed method is more effective in natural scenes, such as images of houses or waterfalls, than artificial scenes such as building images.

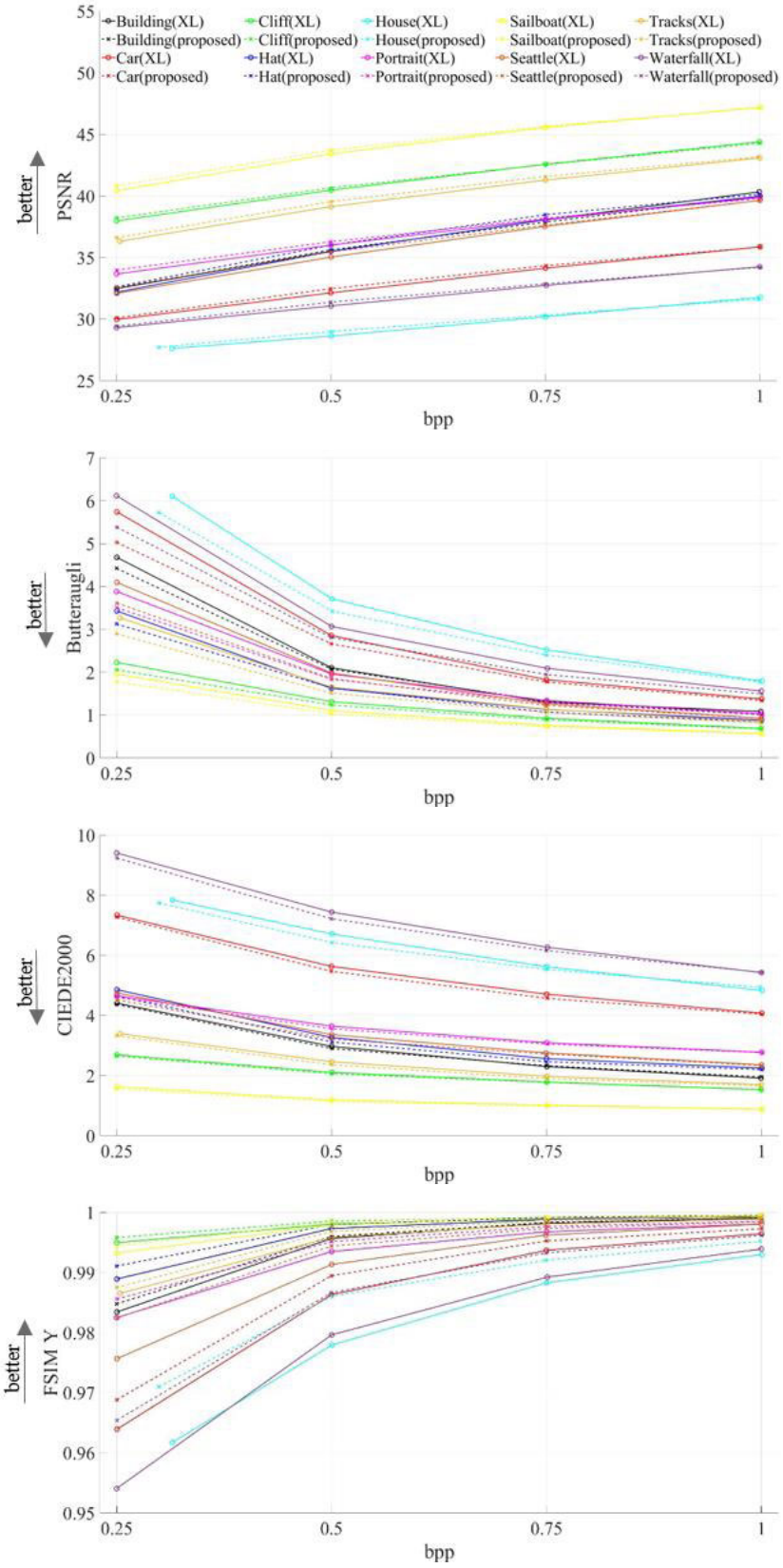

FIGURE 6. Comparison of objective coding performance of JPEG XL and proposed block partitioning using PSNR, Butteraugli, CIEDE2000, and FSIM metrics.

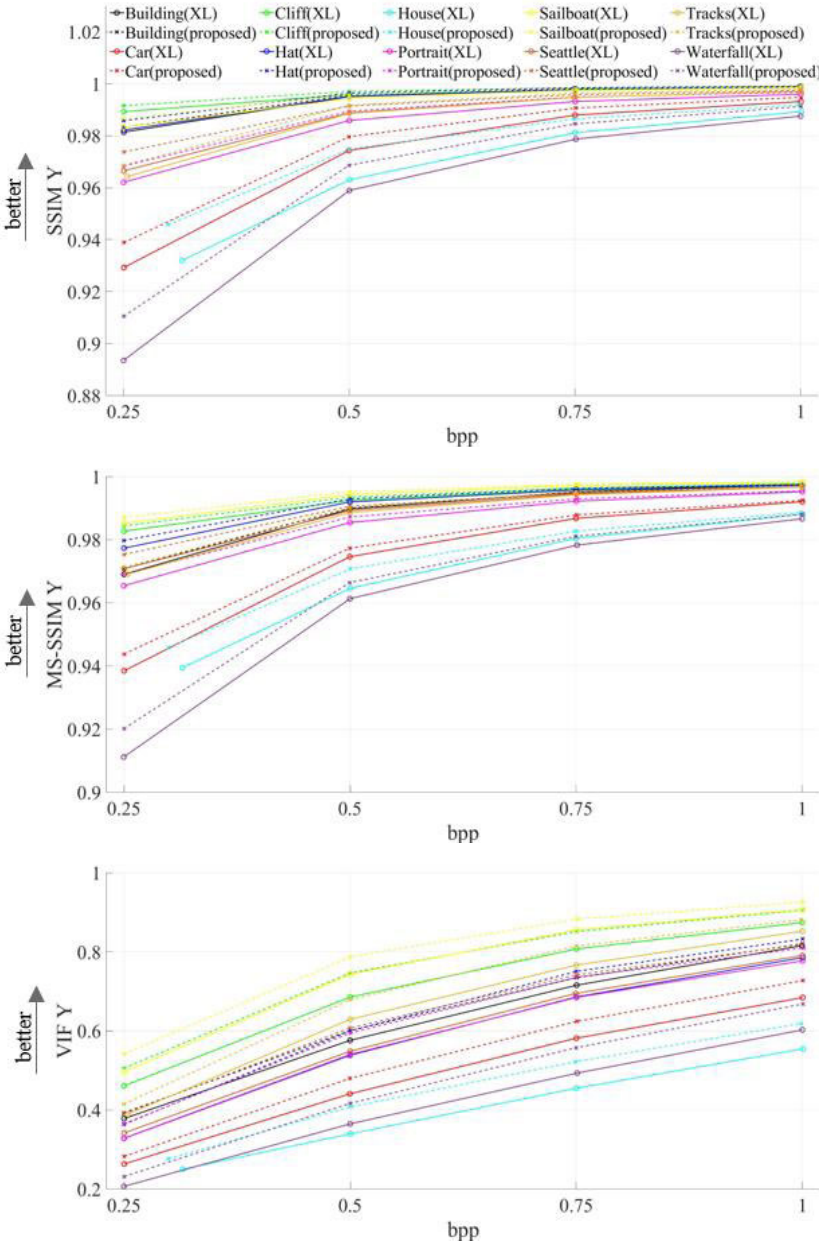

FIGURE 7. Comparison of objective coding performance of JPEG XL and the proposed block partitioning using SSIM, MS-SSIM, and VIF metrics.

TABLE 1. Average objective metric results of JPEG XL and the proposed block partitioning.

Image	XL				Proposed			
	bpp = 0.25	0.50	0.75	1.00	bpp = 0.25	0.50	0.75	1.00
PSNR	33.2057	35.6902	37.8218	39.6403	33.4596	36.0139	37.9487	39.5855
Butteraugli	4.1494	2.1378	1.4307	1.0771	3.7532	2.0074	1.3715	1.0453
CIEDE2000	5.0967	3.8786	3.2085	2.7751	4.9927	3.7474	3.1489	2.7732
FSIM	0.9785	0.9913	0.9957	0.9976	0.9827	0.9940	0.9970	0.9982
SSIM	0.9584	0.9840	0.9921	0.9955	0.9653	0.9881	0.9942	0.9966
MS-SSIM	0.9609	0.9832	0.9911	0.9946	0.9648	0.9854	0.9920	0.9950
VIF	0.3443	0.5406	0.6739	0.7644	0.3765	0.5920	0.7215	0.8008

- The proposed method achieves a better performance than that of the JPEG XL method for most of the test images, employed metrics, and bit rates.

We also performed a human subjective evaluation. The subjective test was conducted in accordance with the International Telecommunication Union (ITU-T P.910)

TABLE 2. Objective results of JPEG XL and proposed block partitioning.

Metric	Image	XL				Proposed			
		bpp = 0.25	0.50	0.75	1.00	bpp = 0.25	0.50	0.75	1.00
PSNR	Building	32.4878	35.5097	38.0612	40.3348	32.4989	35.6263	37.8768	40.0040
	Car	29.9696	32.1335	34.1363	35.8637	30.1053	32.4709	34.3441	35.8582
	Cliff	37.9720	40.4694	42.5699	44.3935	38.2180	40.6575	42.5446	44.2740
	Hat	32.1880	35.4958	38.0572	39.9783	32.6153	36.0102	38.4750	40.1495
	House	27.6272	28.6280	30.1919	31.7737	27.7269	28.9970	30.3147	31.6219
	Portrait	33.6617	36.0084	38.1023	39.8799	33.9846	36.2829	38.2008	39.8003
	Sailboat	40.4227	43.4163	45.5479	47.2036	40.8082	43.7132	45.6558	47.1400
	Seattle	32.0931	35.0371	37.5186	39.6408	32.5944	35.4583	37.6263	39.6351
	Tracks	36.3237	39.1294	41.2892	43.0860	36.6138	39.5450	41.5809	43.1628
	Waterfall	29.3109	31.0738	32.7436	34.2487	29.4309	31.3773	32.8678	34.2092
Butteraugli	Building	4.67952	2.10459	1.30340	1.08421	4.42270	2.07098	1.32627	1.01384
	Car	5.73851	2.85713	1.82577	1.37546	5.02757	2.66170	1.77126	1.34055
	Cliff	2.22791	1.31371	0.92650	0.69060	2.05707	1.23501	0.89183	0.67727
	Hat	3.42897	1.62606	1.13302	0.88344	3.12630	1.63631	1.07004	0.84343
	House	6.10056	3.70997	2.52327	1.79268	5.72297	3.42267	2.40330	1.77482
	Portrait	3.88276	1.95961	1.34104	1.03280	3.50352	1.83459	1.28877	1.00808
	Sailboat	1.95748	1.10646	0.75912	0.57180	1.77691	1.03278	0.72702	0.56000
	Seattle	4.09480	1.97955	1.26545	0.92824	3.61401	1.85524	1.22713	0.92531
	Tracks	3.26696	1.65299	1.13904	0.84888	2.89730	1.51280	1.06193	0.81663
	Waterfall	6.11629	3.06838	2.09013	1.56288	5.38362	2.81175	1.94790	1.49320
CIEDE2000	Building	4.40903	2.97187	2.30010	1.91755	4.37009	2.90040	2.32673	1.95563
	Car	7.33429	5.63186	4.70885	4.07928	7.27251	5.46746	4.57031	4.04172
	Cliff	2.69299	2.10595	1.78557	1.53306	2.65163	2.06082	1.77692	1.52603
	Hat	4.85809	3.26929	2.56338	2.24584	4.64213	3.11868	2.46453	2.19845
	House	7.83714	6.70799	5.61954	4.83219	7.74007	6.42434	5.53357	4.93670
	Portrait	4.65720	3.64515	3.09343	2.77820	4.60601	3.55710	3.04920	2.76669
	Sailboat	1.62971	1.19062	1.01402	0.88155	1.56173	1.14590	0.99045	0.86944
	Seattle	4.75011	3.36398	2.75631	2.35646	4.52232	3.21942	2.71726	2.33707
	Tracks	3.39478	2.46048	1.97723	1.70100	3.32622	2.36056	1.90245	1.66467
	Waterfall	9.40330	7.43838	6.26695	5.42626	9.23417	7.21980	6.15801	5.43572
FSIM	Building	0.98344	0.99566	0.99818	0.99906	0.98478	0.99601	0.99832	0.99909
	Car	0.96393	0.98630	0.99369	0.99646	0.96884	0.98946	0.99526	0.99728
	Cliff	0.99497	0.99804	0.99887	0.99934	0.99582	0.99855	0.99915	0.99950
	Hat	0.98893	0.99733	0.99884	0.99937	0.99109	0.99798	0.99916	0.99951
	House	0.96171	0.97795	0.98828	0.99299	0.97098	0.98607	0.99205	0.99518
	Portrait	0.98249	0.99349	0.99679	0.99808	0.98558	0.99515	0.99767	0.99852
	Sailboat	0.99326	0.99789	0.99901	0.99942	0.99441	0.99839	0.99924	0.99955
	Seattle	0.97567	0.99135	0.99622	0.99805	0.98244	0.99440	0.99733	0.99843
	Tracks	0.98654	0.99583	0.99808	0.99891	0.98751	0.99685	0.99861	0.99915
	Waterfall	0.95406	0.97963	0.98924	0.99391	0.96542	0.98665	0.99325	0.99616
SSIM	Building	0.98138	0.99504	0.99784	0.99889	0.98322	0.99564	0.99808	0.99892
	Car	0.92927	0.97430	0.98793	0.99316	0.93886	0.97962	0.99062	0.99479
	Cliff	0.98924	0.99580	0.99775	0.99866	0.99155	0.99709	0.99841	0.99906
	Hat	0.98216	0.99514	0.99789	0.99883	0.98577	0.99640	0.99847	0.99910
	House	0.93199	0.96305	0.98113	0.98924	0.94572	0.97488	0.98644	0.99186
	Portrait	0.96208	0.98585	0.99315	0.99598	0.96835	0.98944	0.99489	0.99677
	Sailboat	0.98325	0.99430	0.99738	0.99848	0.98659	0.99579	0.99804	0.99884
	Seattle	0.96647	0.98878	0.99475	0.99709	0.97378	0.99139	0.99579	0.99740
	Tracks	0.96427	0.98856	0.99473	0.99713	0.96857	0.99170	0.99637	0.99787
	Waterfall	0.89354	0.95894	0.97869	0.98756	0.91052	0.96862	0.98455	0.99124
MS-SSIM	Building	0.96905	0.98974	0.99521	0.99740	0.97084	0.99012	0.99492	0.99717
	Car	0.93848	0.97463	0.98677	0.99204	0.94374	0.97735	0.98787	0.99248
	Cliff	0.98272	0.99249	0.99597	0.99757	0.98437	0.99355	0.99645	0.99783
	Hat	0.97735	0.99198	0.99579	0.99732	0.97977	0.99294	0.99633	0.99757
	House	0.93944	0.96463	0.98044	0.98809	0.94587	0.97079	0.98270	0.98867
	Portrait	0.96543	0.98545	0.99223	0.99522	0.96909	0.98724	0.99297	0.99551
	Sailboat	0.98517	0.99435	0.99718	0.99828	0.98708	0.99522	0.99754	0.99845
	Seattle	0.97083	0.98924	0.99455	0.99674	0.97536	0.99062	0.99482	0.99681
	Tracks	0.96912	0.98860	0.99427	0.99669	0.97132	0.99022	0.99508	0.99705
	Waterfall	0.91118	0.96132	0.97830	0.98666	0.92016	0.96642	0.98110	0.98815
VIF	Building	0.37832	0.57600	0.71556	0.81557	0.39264	0.60097	0.73472	0.82046
	Car	0.26348	0.44079	0.58148	0.68418	0.28252	0.47966	0.62395	0.72753
	Cliff	0.46096	0.68546	0.80730	0.87407	0.50530	0.74662	0.85114	0.90557
	Hat	0.32824	0.53836	0.68571	0.78550	0.36342	0.60008	0.75052	0.83237
	House	0.25017	0.33973	0.45494	0.55453	0.27649	0.40800	0.52199	0.61868
	Portrait	0.32840	0.54027	0.68466	0.77724	0.36530	0.59463	0.73579	0.80968
	Sailboat	0.49621	0.74314	0.85511	0.90714	0.54174	0.78756	0.88276	0.92572
	Seattle	0.34199	0.54850	0.69460	0.79093	0.39061	0.60719	0.74179	0.81894
	Tracks	0.38907	0.62917	0.76638	0.85209	0.41487	0.67878	0.81509	0.88105
	Waterfall	0.20664	0.36481	0.49336	0.60258	0.23164	0.41632	0.55757	0.66812

publication [27]. For evaluating the subjective visual quality of coded image, the Degradation Category Rating (DCR) method was used. We have shown a pair of original and compared image. The compared image was randomly chosen among the images code by JPEG XL and proposed method at 4 different bit rates: 0.25, 0.50, 0.75 and 1.00 bpp.

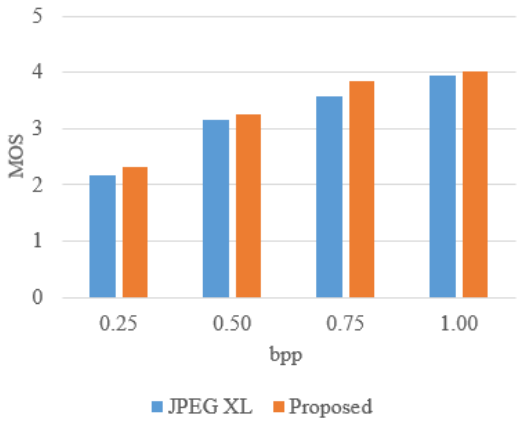

FIGURE 8. Overall MOS result of DCR test.

15 participants have participated. The participants were asked to grade the impairment using a five-level scale as follows:

5 – imperceptible
4 – perceptible, but not annoying
3 – slightly annoying
2 – annoying
1 – very annoying.

We collect 10 images × 15 participants × 2 compared images coded by JPEG XL and proposed method × 4 bit rates = 1,200 values in total from participants in our DCR test. Figure 8 represents the mean opinion score (MOS) of DCR for each bpp of our subjective test. The subjective test shown in Fig. 8 represents that the proposed method achieves the better performance than that of JPEG XL method for all tested bit rates.

For a visual comparison, we selected several parts of interest in the coded images (Fig. 9). These parts were selected such that they included various features (e.g., regions with texture and homogeneous regions). Figs. 10–14 show the resulting images coded at 0.5, 0.75, and 1.0 bpp for two DCT block partitioning methods. The visual comparison shows that the proposed block partitioning method shows a significant difference from the JPEG XL DCT block partitioning method. The observations shown in Figs. 10–14 can be summarized as follows:

- The proposed method is less affected by the compression ratio.
- The visual difference between the two methods becomes greater at a lower bit rate.
- The proposed method is more resistant to blocking and blurring artifacts.

Most importantly, the proposed method objectively and visually exhibits a better performance than the current JPEG XL by using a relatively larger-sized DCT block at the HTR. The proposed method suppresses the blocking and blurring artifacts and helps preserve the HTR information.

We also compared the execution times of JPEG XL and the proposed method. Because the encoding process, except for the block partitioning part, is the same, the times are measured only for each partitioning procedure. The execution times were measured using a platform powered by a quad-core 3.6 GHz CPU and 20 GB of RAM. Table 3 shows the average execution times for 10 attempts using 10 test images with 2 partitioning methods. For all images, the proposed approach takes less time than the JPEG XL method.

TABLE 3. Execution times for partitioning DCT blocks using JPEG XL and proposed method (units are given in seconds).

Image	Average execution time	
	JPEG XL	Proposed
Building	0.205	0.111
Car	0.106	0.037
Cliff	0.185	0.123
Hat	0.239	0.055
House	0.114	0.061
Portrait	0.124	0.049
Sailboat	0.146	0.057
Seattle	0.115	0.050
Tracks	0.174	0.062
Waterfall	0.093	0.032

(a) (b)

FIGURE 9. Selected parts of test images for visual comparison: (a) House and (b) Seattle.

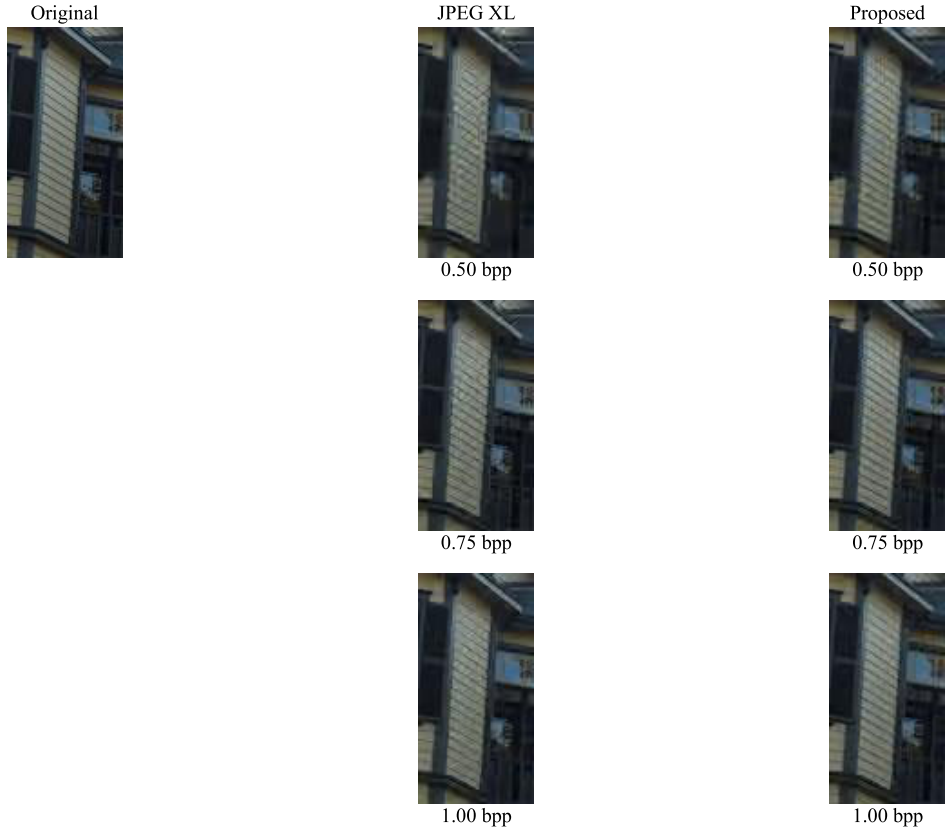

FIGURE 10. Coded image of the compared structures with two different DCT block partitioning structures at approximately 1.0 bpp.

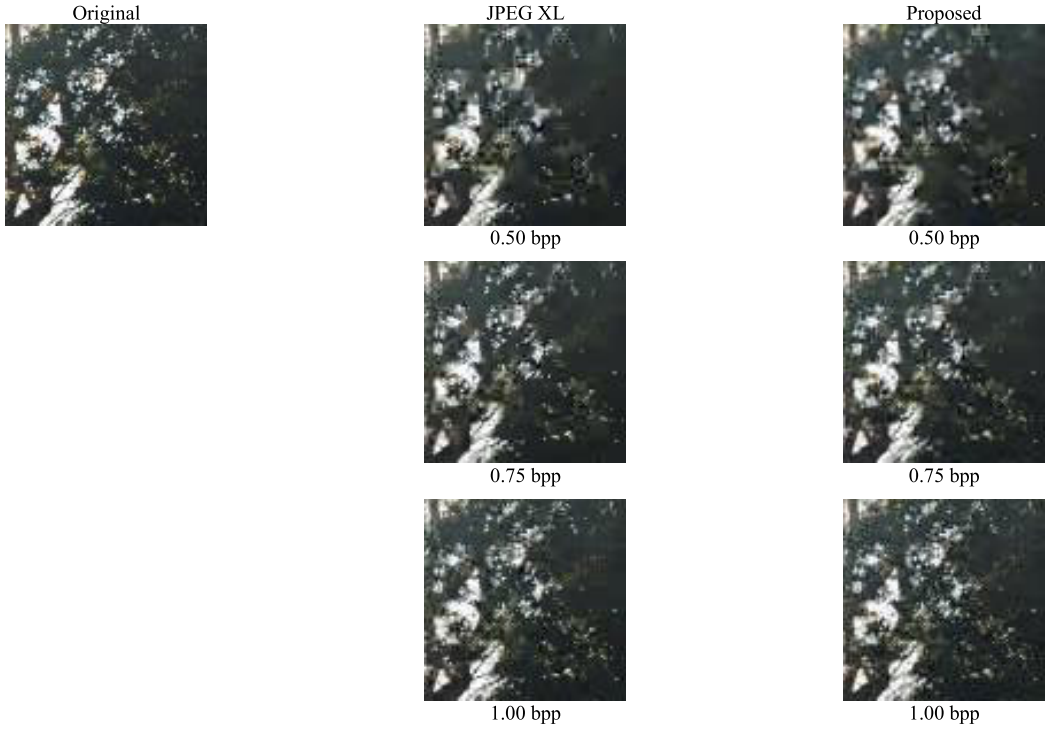

FIGURE 11. Coded image of compared structures with two different DCT block partitioning structures at approximately 1.0 bpp.

FIGURE 12. Coded image of the compared structures with two different DCT block partitioning structure at approximately 1.0 bpp.

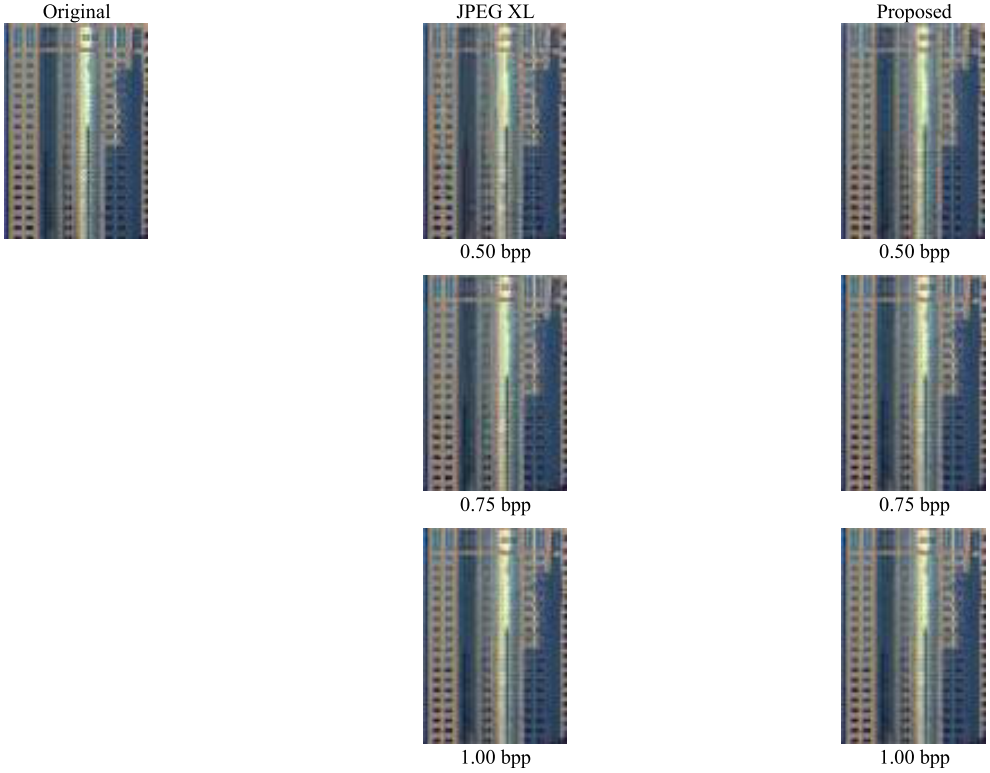

FIGURE 13. Coded image of the compared structures with two different DCT block partitioning structures at approximately 1.0 bpp.

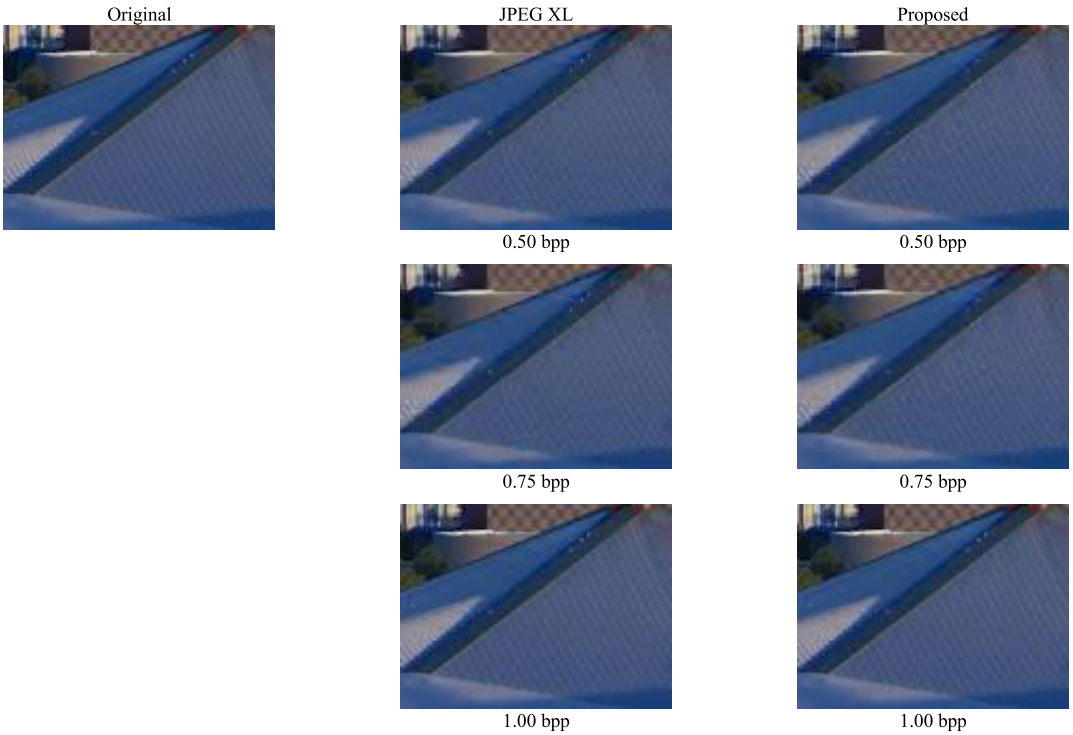

FIGURE 14. Coded image of the compared structures with two different DCT block partitioning structures at approximately 1.0 bpp.

V. CONCLUSION

Herein, we propose a region-adaptive DCT block partitioning method for improving the JPEG XL coding, which is based on three metrics: ZC, SML, and colorfulness. The JPEG XL block partitioning method is mostly dependent on the compression ratio. Therefore, JPEG XL does not use different block sizes, which can improve the performance. The proposed method determines the size of the DCT block based on various metrics, which can effectively affect the performance of JPEG XL for HTRs.

We compared the performance of the region-adaptive DCT block partitioning method against the existing JPEG XL partitioning method. We conducted an objective evaluation, subjective evaluation, and visual comparison to evaluate the quality of the coded images and calculate the execution times. For the objective tests, we compared the results using seven objective image quality metrics: the PSNR, Butteraugli, CIEDE2000, FSIM, SSIM, MS-SSIM, and VIF. The experiments showed significant coding gains for most of the bit rates. We demonstrated the improvement of the proposed method using ten sample images used in the JPEG WG. In the subjective evaluation involving 15 participants, we found that the proposed method shows better performance visually. We confirmed better representations for the HTRs through a visual comparison and less complexity by measuring the execution times. Block partitioning is a major technique of lossy compression image coding, which uses blocks of various sizes. We also prove that a larger DCT block size is effective for HTRs in JPEG XL lossy image coding. Therefore, the proposed method may be a notable function for improving the performance of JPEG XL.

REFERENCES

[1] W. B. Pennebaker and J. L. Mitchell, *JPEG Still Image Data Compression Standard*. Norwell, MA, USA: Springer, 1992.
[2] J. Wassenberg and J. Sneyers, *JPEG XL Use Cases and Requirement*, document JPEG (ISO/IEC JTC 1/SC 29/WG 1) 83rd Meeting, N83043, Geneva, Switzerland, Mar. 2019, pp. 18–22.
[3] (2021). *JPEG Official Site*. [Online]. Available: https://jpeg.org/jpegxl
[4] I.-K. Kim, J. Min, T. Lee, W.-J. Han, and J. Park, "Block partitioning structure in the HEVC standard," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 22, no. 12, pp. 1697–1706, Dec. 2012.
[5] L. Shen, Z. Zhang, and Z. Liu, "Effective CU size decision for HEVC intracoding," *IEEE Trans. Image Process.*, vol. 23, no. 10, pp. 4232–4241, Oct. 2014.
[6] Y.-W. Huang, J. An, H. Huang, X. Li, S.-T. Hsiang, K. Zhang, H. Gao, J. Ma, and O. Chubach, "Block partitioning structure in the VVC standard," *IEEE Trans. Circuits Syst. Video Technol.*, early access, Jun. 11, 2021, doi: 10.1109/TCSVT.2021.3088134.
[7] J. Pfaff, A. Filippov, S. Liu, X. Zhao, J. Chen, S. De-Luxán-Hernández, T. Wiegand, V. Rufitskiy, A. K. Ramasubramonian, and G. Van der Auwera, "Intra prediction and mode coding in VVC," *IEEE Trans. Circuits Syst. Video Technol.*, early access, Sep. 12, 2021, doi: 10.1109/TCSVT.2021.3072430.
[8] J. Wassenberg and J. Sneyers, *JPEG XL White Paper*, document JPEG (ISO/IEC JTC 1/SC 29/WG 1) 87th Meeting, N87021, Erlangen, Germany, Apr. 2020, pp. 27–30.
[9] (2021). *JPEG Official Site*. [Online]. Available: https://jpeg.org/jpegxl/software.html
[10] H. Feng, B. Hou, and M. Gong, "SAR image despeckling based on local homogeneous-region segmentation by using pixel-relativity measurement," *IEEE Trans. Geosci. Remote Sens.*, vol. 49, no. 7, pp. 2724–2737, Jul. 2011.
[11] H. Ali, M. Sharif, M. Yasmin, and M. H. Rehmani, "Computer-based classification of chromoendoscopy images using homogeneous texture descriptors," *Comput. Biol. Med.*, vol. 88, pp. 84–92, Sep. 2017.

[12] A. Goltsev, V. Gritsenko, E. Kussul, and T. Baidyk, "Finding the texture features characterizing the most homogeneous texture segment in the image," in *Advances in Computational Intelligence* (Lecture Notes in Computer Science), vol. 9094, I. Rojas, G. Joya, and A. Catala, Eds. Cham, Switzerland: Springer, 2015, doi: 10.1007/978-3-319-19258-1_25.
[13] J. Zhang and T. Le, "A new no-reference quality metric for JPEG2000 images," *IEEE Trans. Consum. Electron.*, vol. 56, no. 2, pp. 743–750, May 2010.
[14] S. K. Nayar and Y. Nakagawa, "Shape from focus," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 16, no. 8, pp. 824–831, Aug. 1994.
[15] D. Hasler and S. E. Suesstrunk, "Measuring colorfulness in natural images," *Proc. SPIE* vol. 5007, pp. 87–95, Jun. 2003.
[16] S. Choi, O.-J. Kwon, and J. Lee, "A method for fast multi-exposure image fusion," *IEEE Access*, vol. 5, pp. 7371–7380, 2017.
[17] J. Wassenberg and J. Sneyers, *DIS Text of ISO IEC 18181-1 (JPEG XL)*, document JPEG (ISO/IEC JTC 1/SC 29/WG 1) 86th Meeting, N86049 Sydney, NSW, Australia, Jan. 2020.
[18] M. V. Bernardo, A. Pinheiro, and M. Pereira, *JPEG XL Objective Evaluation Results*, document JPEG (ISO/IEC JTC 1/SC 29/WG 1) 86th Meeting, M86070, Sydney, NSW, Australia, Jan. 2020.
[19] J. Wassenberg and J. Sneyers, *Core Experiments 5 for ISO/IEC 18181*, document JPEG (ISO/IEC JTC 1/SC 29/WG 1) 85th Meeting, N85057, San Jose, CA, USA, Nov. 2019, pp. 2–8.
[20] J. Ascenso, P. Akayzi, M. Testolina, A. Boev, and E. Alshina, *Performance Evaluation of Learning Based Image Coding Solutions and Quality Metrics*, document JPEG (ISO/IEC JTC 1/SC 29/WG 1) 85th Meeting, N85013, San Jose, CA, USA, Nov. 2019.
[21] (2021). *Butteraugli*. [Online]. Available: https://github.com/google/butteraugli
[22] G. Sharma, W. Wu, and E. N. Dalal, "The CIEDE2000 color-difference formula: Implementation notes, supplementary test data, and mathematical observations," *Color Res. Appl.*, vol. 30, no. 1, pp. 21–30, Feb. 2005.
[23] L. Zhang, L. Zhang, X. Mou, and D. Zhang, "FSIM: A feature similarity index for image quality assessment," *IEEE Trans. Image Process.*, vol. 20, no. 8, pp. 2378–2386, Aug. 2011.
[24] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: From error visibility to structural similarity," *IEEE Trans. Image Process.*, vol. 13, no. 4, pp. 600–612, Apr. 2004.
[25] Z. Wang, E. P. Simoncelli, and A. C. Bovik, "Multiscale structural similarity for image quality assessment," in *Proc. 37th Asilomar Conf. Signals, Syst. Comput.*, Nov. 2003, pp. 1398–1402.
[26] H. R. Sheikh and A. C. Bovik, "Image information and visual quality," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, Montreal, QC, Canada, Aug. 2004, pp. 430–444.
[27] *Subjective Video Quality Assessment Methods for Multimedia Applications*, document ITU-T P.910, ITU-T, 2008.

JOONHYUNG CHO received the B.S. and M.S. degrees in electrical engineering from Sejong University, Seoul, South Korea, in 1999 and 2021, respectively. His research interests include image processing and JPEG image coding.

OH-JIN KWON received the M.S. degree in electrical engineering from the University of Southern California, Los Angeles, CA, USA, in 1991, and the Ph.D. degree in electrical engineering from the University of Maryland, College Park, MD, USA, in 1994. He was a Researcher with the Agency for Defense Development, South Korea, from 1984 to 1989, and the Head of the Media Laboratory, Samsung SDS Company Ltd., Seoul, South Korea, from 1995 to 1999. Since 1999, he has been a Faculty Member with Sejong University, Seoul, where he is currently a Professor. His research interests include image and video fusion, coding, watermarking, analysis, and processing.

SEUNGCHEOL CHOI received the B.S. and M.S. degrees in computer science and the Ph.D. degree in electronics engineering from Sejong University, Seoul, South Korea, in 1998, 2001, and 2017, respectively. He was a Researcher with Galaxia Communications, from 2001 to 2013. He has been a Postdoctoral Researcher with Sejong University, since 2017. His research interests include image and video coding, high-dynamic range imaging, image processing, image fusion, and JPEG.