

## Research Article

# Bit-Error Aware Lossless Image Compression with 2D-Layer-Block Coding

Jungan Chen <sup>1</sup>, Jean Jiang <sup>2</sup>, Xinnian Guo <sup>3</sup>, and Lizhe Tan <sup>4</sup>

<sup>1</sup>Department of Electronic and Computer Science, Zhejiang Wanli University, Ningbo, China

<sup>2</sup>College of Technology, Purdue University Northwest, Indiana, USA

<sup>3</sup>Department of Electronic Information Engineering, Huaiyin Institute of Technology, Huaian, China

<sup>4</sup>Department of Electrical and Computer Engineering, Purdue University Northwest, Indiana, USA

Correspondence should be addressed to Jungan Chen; friendcen21@hotmail.com

Received 23 June 2021; Revised 10 October 2021; Accepted 12 October 2021; Published 3 November 2021

Academic Editor: Sang-Hoon Hong

Copyright © 2021 Jungan Chen et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

With IoT development, it becomes more popular that image data is transmitted via wireless communication systems. If bit errors occur during transmission, the recovered image will become useless. To solve this problem, a bit-error aware lossless image compression based on bi-level coding is proposed for gray image compression. But bi-level coding has not considered the inherent statistical correlation in 2D context region. To resolve this shortage, a novel variable-size 2D-block extraction and encoding method with built-in bi-level coding for color image is developed to decrease the entropy of information and improve the compression ratio. A lossless color transformation from RGB to the YCrCb color space is used for the decorrelation of color components. Particularly, the layer-extraction method is proposed to keep the Laplacian distribution of the data in 2D blocks which is suitable for bi-level coding. In addition, optimization of 2D-block start bits is used to improve the performance. To evaluate the performance of our proposed method, many experiments including the comparison with state-of-the-art methods, the effects with different color space, etc. are conducted. The comparison experiments under a bit-error environment show that the average compression rate of our method is better than bi-level, Jpeg2000, WebP, FLIF, and L3C (deep learning method) with hamming code. Also, our method achieves the same image quality with the bi-level method. Other experiments illustrate the positive effect of built-in bi-level encoding and encoding with zero-mean values, which can maintain high image quality. At last, the results of the decrease of entropy and the procedure of our method are given and discussed.

## 1. Introduction

With cloud computing and Internet of Things (IoT) development, the requirement for data transmission and storage is increasing. Fast and efficient compression of data plays a very important role in many applications. For instance, image data compression has been used in many areas such as medical, satellite remote sensing, and multimedia.

There are many methods to compress image data including prediction-based, transformation-based, and other methods such as fractal image compression and deep learning with Auto Encoder (AE) [1, 2], Recurrent Neural Network (RNN), Convolutional Neural Network (CNN) [3], and Residual Neural Network (ResNet) [4]. The

transformation-based method includes Discrete Cosine Transform (DCT), Karhunen-Loeve Transform (KLT), Hadamard transform, Slant transform, Haar transform, and singular value decomposition [5]. Usually, transformation-based or deep learning methods are used in lossy compression while prediction-based methods are used for lossless compression.

In some cases, lossless compression must be applied when data acquisition is expensive. For example, lossless image compression must be applied in aerial, medical, and space images [6, 7]. In industry, many engineered lossless compression methods including Portable Network Graphics (PNG), WebP [8], and Free Lossless Image Format (FLIF) [9] are used. Also, some deep learning-based lossless

compression methods [10–12] are early researched. As one classical method for lossless compression, the prediction-based method takes into account the difference between pixel values and their predicted values, which are generally smaller numbers than the pixel values themselves. Thus, each difference value needs a smaller number of bits to encode [13]. It mainly has three kinds of methods including context-based, least-square (LS)-based, and spatial structure-based. Among these methods, the method based on spatial structure with 2D context region is an effective solution to improve the compression ratio (CR) because of considering the inherent statistical correlation using block-based methods such as quadtree-based block [14], reference block [15], template-matching[16], and hierarchical decomposition [17]. Quadtree-based block and hierarchical decomposition methods split image to many subimages. And the reference block method considers the phenomenon that a physical object is constructed from numbers of structure components. Inspired from these methods, splitting image into many blocks where each block has similar color is taken as an effective method used by this work.

With IoT development, it becomes more usual that image data is transmitted through wireless communication systems and lossless image compression is used to improve transmission throughput. However, if bit errors occur in a wireless noisy channel during transmission, the recovered image will be damaged or become useless. So, lossless image compression must resolve the problem and keep the recovered image be useful. Most methods including engineered lossless compression methods and deep learning based methods are not suitable to transmission in noise channel; to the best of our knowledge, fewer researches have worked on this case except our previous work [7]. By protecting the key information bits with error control coding, our work proposed a bit-aware lossless image compression based on bi-level coding for gray image as a one-dimensional signal. In the coding method, only the linear predictive bi-level block coding parameters are encoded using (7,4) Hamming codes and residue sequences are left as they are to improve the performance of compression rate (CR). One reason for the efficiency of bi-level coding is that it uses the sparsity property of data which required fewer encoding bits.

In this work, we will use bi-level coding [7] for natural images with red (R), green (G), and blue (B) components. As R, G, and B are highly correlated, a linear transformation is applied to map RGB to other color space and achieve better CR [17, 18]. As discussed above, the spatial structure-based method with 2D context region is taken as an effective solution to improve CR. Therefore, image is split into many 2D blocks which has sparsity property and be suitable to be encoded with bi-level coding. Finally, a novel variable-size 2D-block extraction and encoding method with built-in bi-level coding is proposed to improve CR for color image and robust to bit-error environment. An important 2D-layer-block extraction method is used to split the image to many 2D blocks with similar color and keep the Laplacian or Gaussian distribution of data in one 2D block, which has sparsity property.

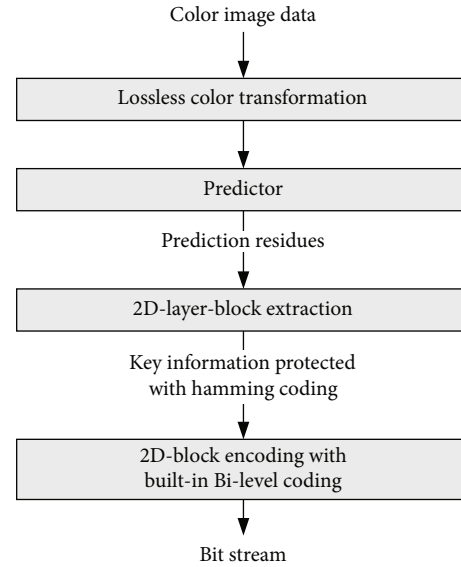


FIGURE 1: The proposed method.

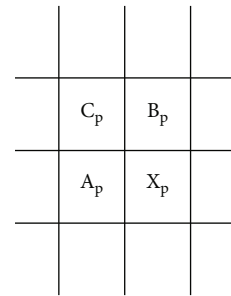


FIGURE 2: Neighboring pixels for the predictor.

The contributions of this paper are summarized as follows:

- (1) For color image compression, a lossless color transformation from RGB to the YCrCb color space is used for the decorrelation of color components. The prediction-based method is used to remove data correlation and produce residue sequence
- (2) To keep the data distribution with the sparsity property and be suitable for bi-level coding, a novel 2D-layer-block extraction method is proposed to keep the Laplacian or Gaussian distribution of data in 2D blocks. Furthermore, by rearranging the order of data encoded, the extraction method can decrease the entropy of data and improve CR
- (3) A novel variable-size 2D-block encoding method with built-in bi-level is proposed to improve CR and robust to bit-error environment just as the bi-level coding method. The mean or min value in one 2D block and key information bits in built-in bi-level coding are protected with hamming code. So, the image can be recovered and useful

The rest of this paper is organized as follows. In Section 2, related works on lossless compression are discussed. In

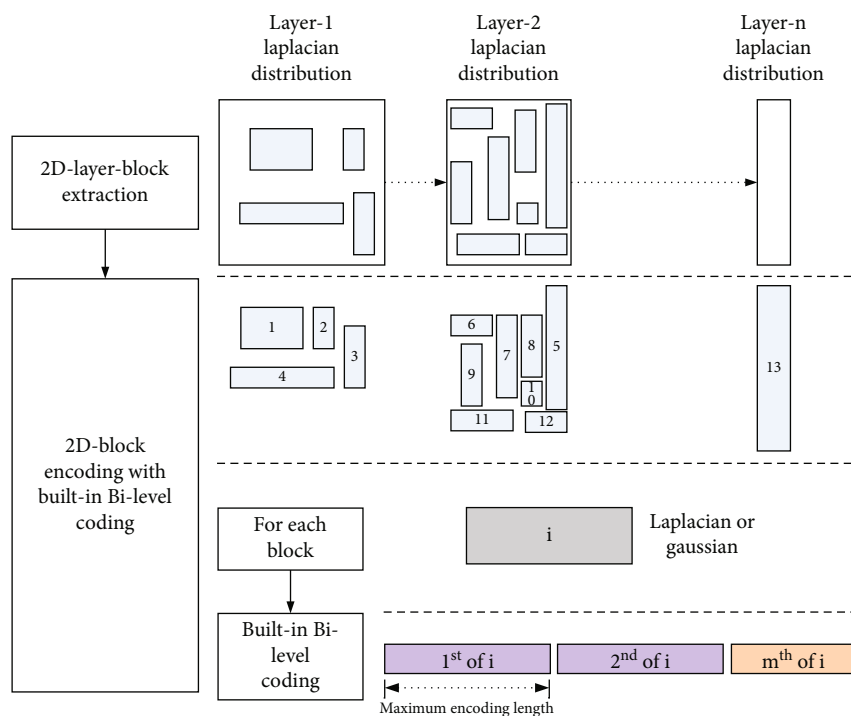


FIGURE 3: The description of 2D-block extraction and encoding.

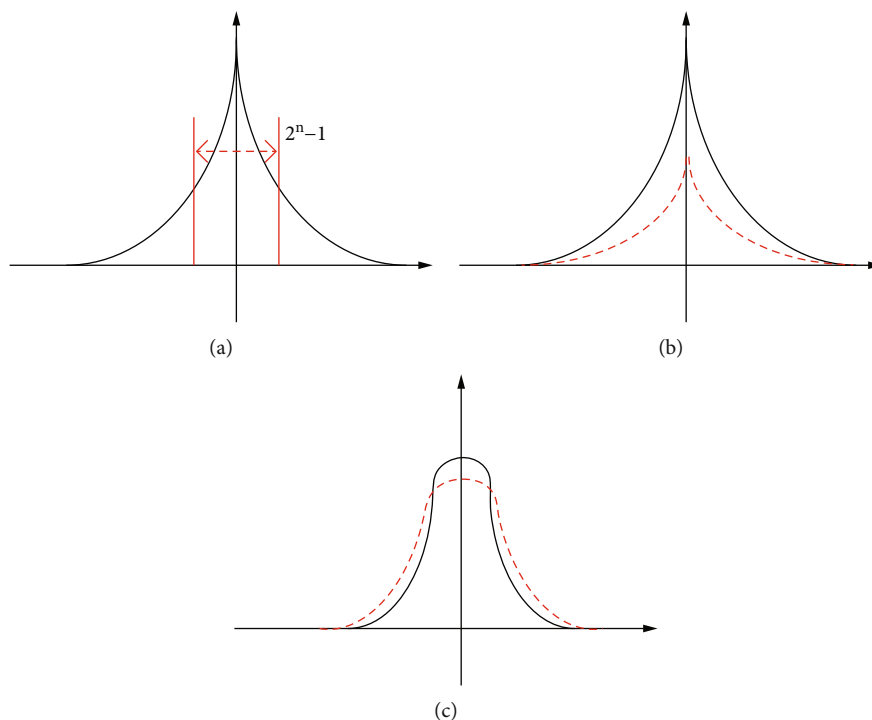


FIGURE 4: Data distribution. (a) data distribution in blocks; (b) data distribution after 2D-block extraction; (c) data in one block.

Section 3, the details of the proposed method are briefly introduced. In Section 4, several experiments including comparison and analysis of basic principles are conducted. Finally, the conclusion and future researches are drawn in Section 5.

## 2. Related Work on Lossless Compression

**2.1. Prediction-Based Methods.** The context-based adaptive prediction method is based on a static predictor which is usually a switching predictor able to adapt to several types

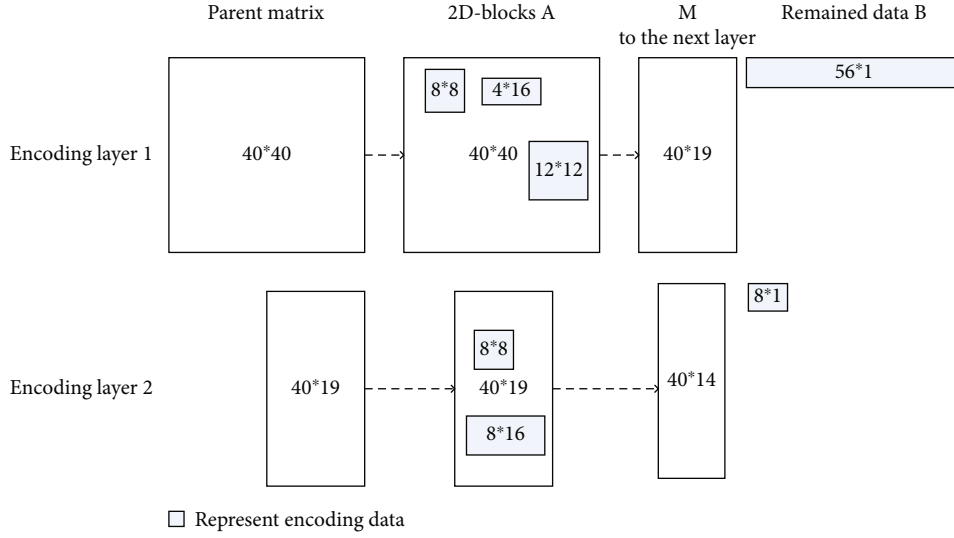


FIGURE 5: Example of extracting 2D blocks.

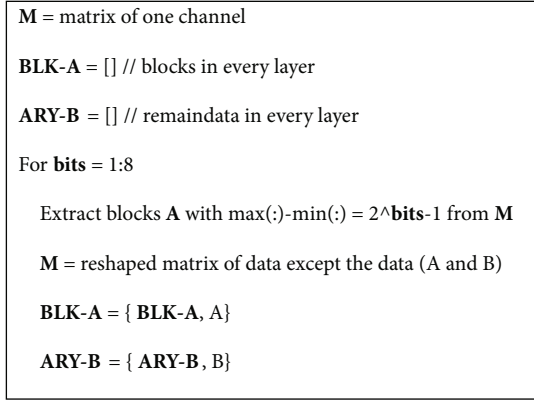


FIGURE 6: Extraction of 2D block.

of contexts, like horizontal edge, vertical edge, or smooth area. Many static predictors can be found in [6, 19]. Median edge detector (MED) used in LOCO-I uses only three causal pixels to determine a type of pixel area which is currently predicted [20]. LOCO-I is further improved and standardized as the JPEG-LS lossless compression algorithm, which has eight different predictive schemes including three one-dimensional and four two-dimensional predictors [21]. To detect edges, Gradient Adjusted Predictor (GAP) embedded in the CALIC algorithm uses local gradient estimation and three heuristic-defined thresholds [22]. Gradient edge detection (GED) predictor combines simplicity of MED and efficiency of GAP [23]. In [19], the prediction errors are encoded using codes adaptively selected from the modified Golomb-Rice code family. To enable processing of images with higher bit depths, a simple context-based entropy coder is presented [6].

LS-based optimization is proposed as an approach to accommodate varying statistics of coding images. To reduce computational complexity, edge-directed prediction (EDP) initiates the LS optimization process only when the prediction error is beyond a preselected threshold [24]. In [25],

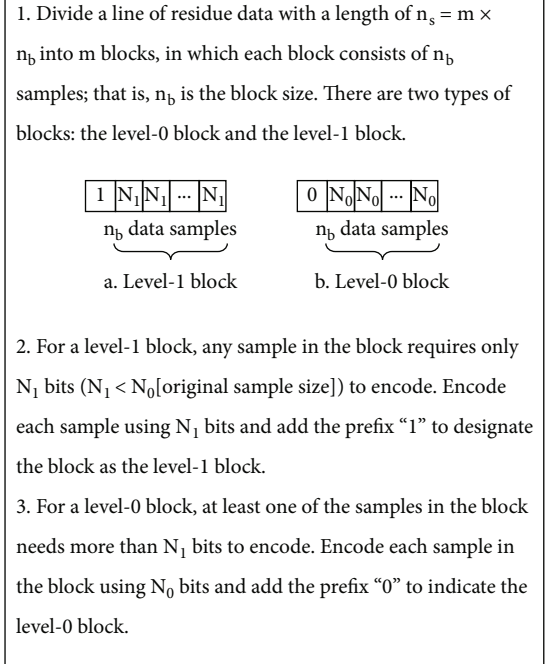


FIGURE 7: Bi-level coding rule.

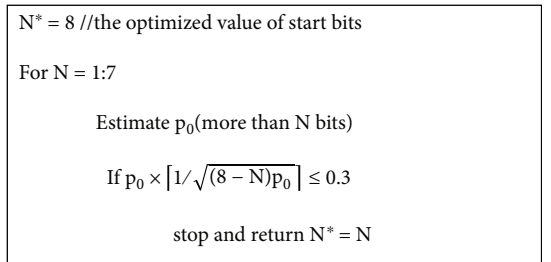


FIGURE 8: Optimization of Start Bits.

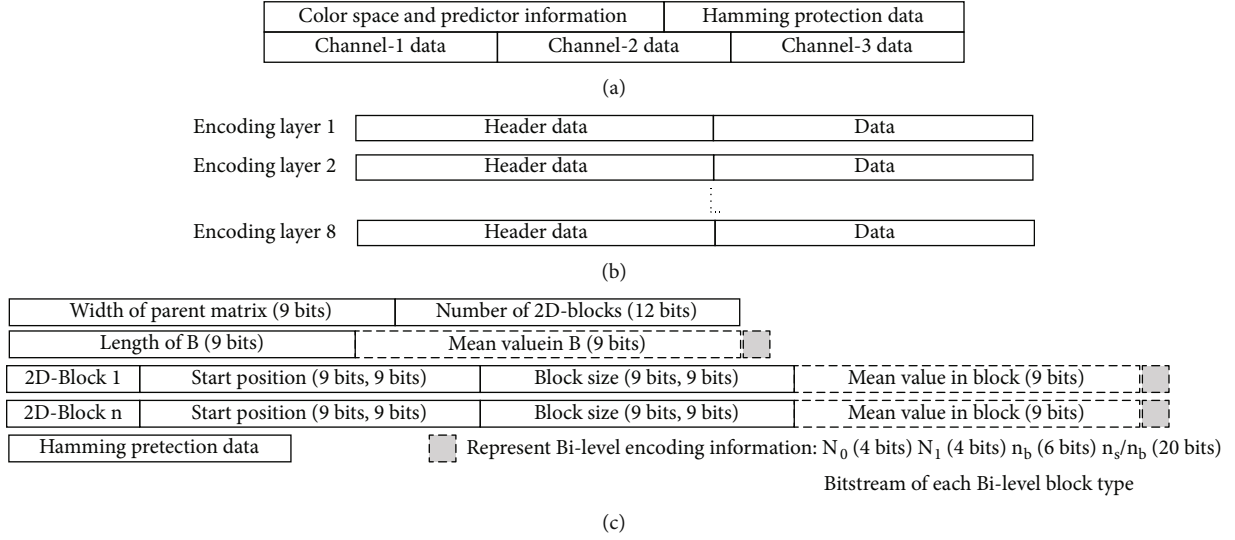














FIGURE 9: 2D-block encoding scheme: (a) data encoding of one image; (b) data encoding in one channel of image; (c) header data encoding of one layer.

TABLE 1: Images from Open Images and CLIC.

FileId	FileName	$th_n$	FileId	FileName	$th_n$
(1)	077a21a70e1b7f86 	102400	(7)	IMG_20160522_165515 	2024
(2)	3632d975bd623167 	10240	(8)	IMG_20170114_204505 	1536
(3)	488d4412c71bf3f1 	20480	(9)	IMG_20170725_123034 	3584
(4)	4e9226ecb7124541 	20480	(10)	IMG_20170730_133144 	20480
(5)	IMG_20131224_110518010_HDR 	20480	(11)	IMG_20170916_004924 	2024
(6)	IMG_20160312_095817 	4096	(12)	c1bfd9c6e7279456 	20480

the LS optimization is processed only when the coding pixel is around an edge or when the prediction error is large. And a switching coding scheme is further proposed that combines the advantages of both run-length and adaptive linear predictive coding [26]. Minimum Mean Square Error (MMSE) predictor uses least mean square principle to adapt  $k$ -order linear predictor coefficients for optimal prediction of the current pixel, from a fixed number of  $m$  causal neighbors [27]. The paper [28] presents a lossless coding method based on blending approach with a set of 20 blended predictors, such as recursive least squares (RLS) predictors and Context-Based Adaptive Linear Prediction (CoBALP+).

Although individual prediction is favored, the morphology of 2D context region would be destructed accordingly and inherent statistical correlation among the correlated region gets obscure. As an alternative, spatial structure has been considered to compensate the pixel-wise prediction [29]. In [14], quadtree-based variable block-size partitioning is introduced into the adaptive prediction technique to remove spatial redundancy in a given image and the resulting prediction errors are encoded using context-adaptive arithmetic coding. Inspired by the success of prediction by partial matching (PPM) in sequential compression, the paper [30] introduces the probabilistic modeling of the encoding symbol based on

TABLE 2: Comparison with prediction-based methods.

Image	Ref. [17]			Ref. [16]			Best CR(7,4)	Our	
	Calic	jpeg2000	jpeg-xr	Ref. method	MRP	CoBaLP2			LOCO-I
Lena	13.1787	13.5848	14.0942	13.6461	11.872	12.399	13.173	1.155179053	1.3876
Peppers	13.8661	14.8	15.3245	15.2102				0.989051407	1.2764
Mandrill	18.1511	18.0939	18.2553	18.5305	16.041	17.039	17.822	0.854952043	1.13
Barbara	14.9567	11.1612	12.1408	11.4575				1.228746525	1.2406
Airplane					10.121		11.06	1.355032676	1.5302
Average								1.116592	1.31296

TABLE 3: Comparison with engineering and deep learning methods.

	Our CR	Bilevel	PNG(7,4)	Jpeg2000(7,4)	WebP(7,4)	FLIF(7,4)	L3C(7,4)
(1)	1.5141	1.4958	1.201423	1.195981	1.209832	1.26147	1.201423
(2)	1.8985	1.9171	1.589528	1.591757	1.629015	1.697848	1.589528
(3)	2.0597	1.9877	1.762672	1.724237	1.767789	1.811668	1.762672
(4)	1.8084	1.7509	1.350212	1.362216	1.343821	1.373226	1.350212
(5)	1.4849	1.4478	1.098277	1.11419	1.104521	1.131249	1.098277
(6)	1.7418	1.7397	1.578617	1.500231	1.544284	1.571556	1.578617
(7)	1.4761	1.4772	1.122815	1.098124	1.086112	1.100787	1.122815
(8)	1.7614	1.7707	1.667265	1.566052	1.611617	1.709558	1.667265
(9)	1.295	1.2774	0.918406	0.904853	0.911615	0.928585	0.918406
(10)	1.5852	1.5504	1.19649	1.176481	1.216922	1.201159	1.19649
(11)	1.807	1.8237	1.740748	1.670904	1.780918	1.795483	1.740748
(12)	1.7631	1.7602	1.528354	1.461382	1.506129	1.576181	1.528354
Average	1.682933	1.66655	1.396234	1.363867	1.392715	1.429898	1.396234

its previous context occurrences. In [15], superspatial structure prediction is proposed to find an optimal prediction of the structure components, e.g., edges, patterns, and textures, within the previously encoded image regions instead of the spatial causal neighborhood. The paper [17] presents a lossless color image compression algorithm based on the hierarchical prediction and context-adaptive arithmetic coding. By exploiting the decomposition and combinatorial structure of the local prediction task and making the conditional prediction with multiple max-margin estimation in a correlated region, a structured set prediction model with max-margin Markov networks is proposed [29]. In [16], the image data is treated as an interleaved sequence generated by multiple sources and a new linear prediction technique combined with template-matching prediction and predictor blending method is proposed. Our method uses a variable-size 2D-block extraction and encoding method with built-in bi-level to improve the compression rate.

**2.2. Engineered Lossless Compression Algorithms.** PNG remove redundancies from the RGB representation with autoregressive filters and then the deflate algorithm based on the LZ77 algorithm and Huffman coding is used for data compression. Lossless WebP compression uses many types of transformation including spatial transformation, color transformation, green subtraction transformation, color

indexing transformation, and color cache coding and then performs the entropy coding which uses a variation of LZ77 Huffman coding [8]. FLIF use Adam7 interlacing and YCoCg interleaving to traverse the image and perform entropy coding with “meta-adaptive near-zero integer arithmetic coding” (MANIAC) based on context-adaptive binary arithmetic coding CABAC [9].

**2.3. Deep Learning-Based Lossless Compression.** Huffman, arithmetic coding, and asymmetric numeral systems are the algorithms for implementing lossless compression, but they do not cater for latent variable models, so bits back with asymmetric numeral systems (BB-ANS) are proposed to solve the issue [10]. But BB-ANS become inefficient when the number of latent variables grows, to improve its performance on hierarchical latent variable models, Bit-Swap is proposed [11]. In contrast to these works focusing on smaller datasets, a fully parallel hierarchical probabilistic model (termed L3C) to enable practical compression on superresolution images [12].

### 3. Our Proposed Method

The proposed method is shown in Figure 1. In color image data, R, G, and B are highly correlated. So, their straightforward encoding is not efficient. Therefore, a linear transformation from RGB to the YCrCb color space is used for the

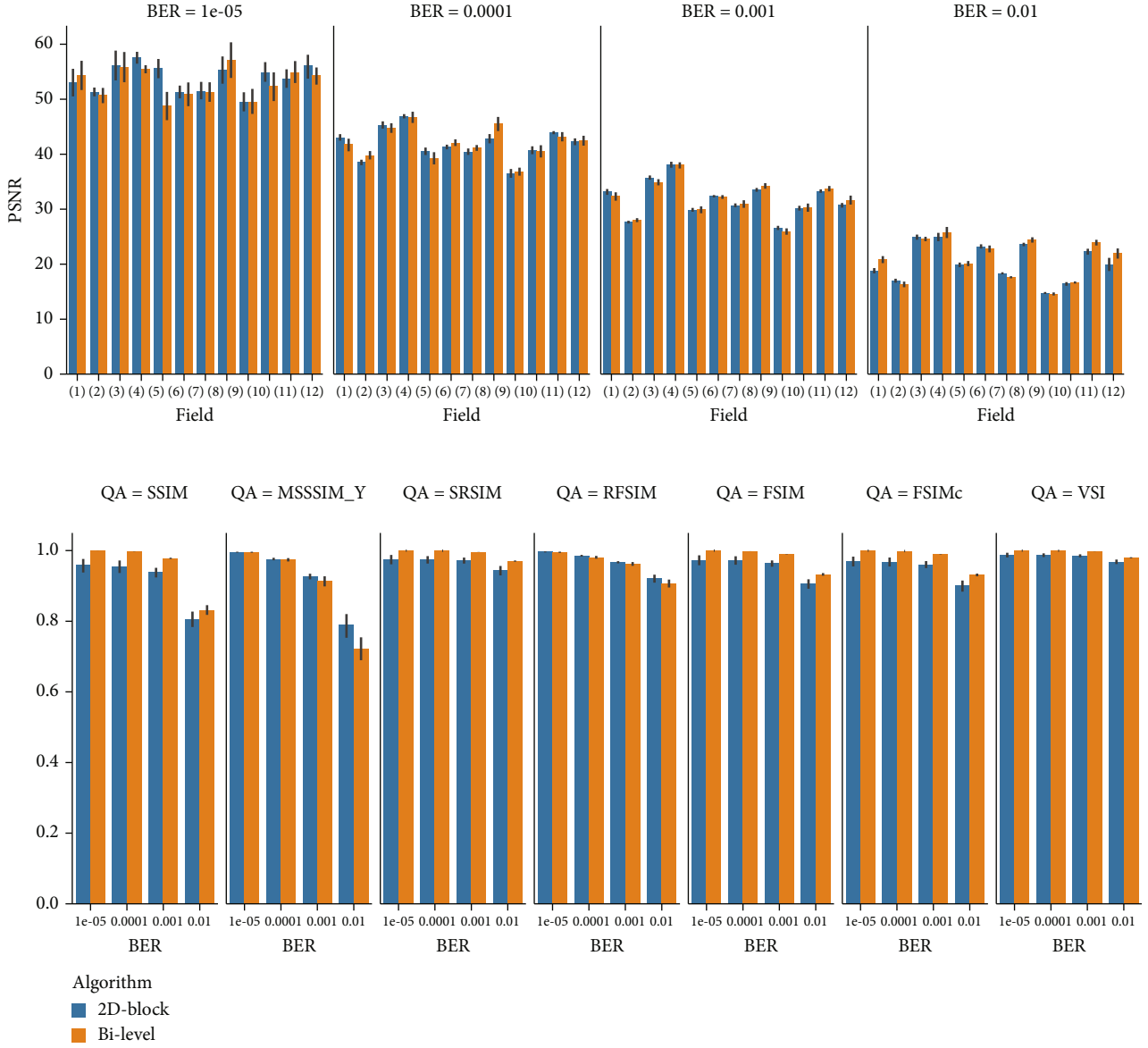


FIGURE 10: Comparison of recovered image quality: (a) PSNR; (b) other assessment methods.

decorrelation of color components [31]. To approximate the original color transform well enough, the lossless color transform with equations (1) and (2) in [32] is adopted in our algorithm. As [7, 19] mentioned, the prediction residues have reduced amplitudes and are assumed to be statistically independent with an approximate Laplacian distribution. Therefore, a predictor in Figure 1 is employed to further remove data correlation in  $Y$ ,  $Cb$ , and  $Cr$  channels, respectively. The predictor value of  $X_p$  can be obtained with equation (3), where  $A_p$ ,  $B_p$ , and  $C_p$  are the pixel value and their location is illustrated in Figure 2. After the prediction step, variable-size 2D blocks are extracted and key information about the blocks are encoded with Hamming code. Finally, these 2D blocks are separately encoding with built-in bi-level coding to make use of the sparsity property of Laplacian distribution and achieve better signal quality and robust to bit errors [7].

$$\begin{cases} Y = G + \left[ \frac{86}{256}Cr + \frac{29}{256}Cb \right], \\ Cr = R - G, \\ Cb = B - \left[ \frac{87}{256}R + \frac{169}{256}G \right], \end{cases} \quad (1)$$

$$\begin{cases} G = Y - \left[ \frac{86}{256}Cr + \frac{29}{256}Cb \right], \\ R = Cr + G, \\ B = Cb + \left[ \frac{87}{256}R + \frac{169}{256}G \right], \end{cases} \quad (2)$$

$$\text{Predictor}(X_p) = \frac{3A_p + 3B_p - 2C_p}{4}. \quad (3)$$



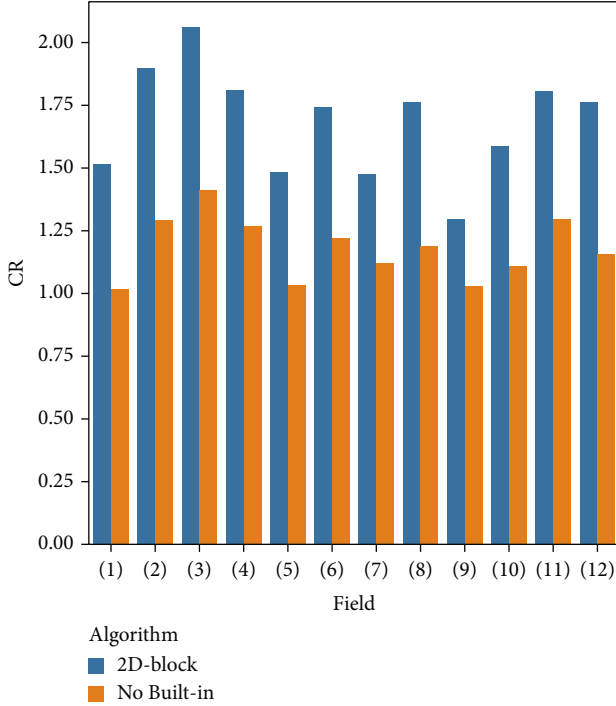


FIGURE 11: Effect of bi-level on CR (BER = 0.001, optimal  $th_n$ ).

The procedure of 2D-block extraction and encoding is further shown in Figure 3; the 2D-layer-block extraction method is used to keep the Laplacian or Gaussian distribution of data in Layer-1~ $n$  or 2D blocks, which have the sparsity property and are suitable for bi-level coding.  $n$  in Layer- $n$  represents  $n$  bits required to encode in the extracted blocks, and the remaining data not belonging to any blocks is left to the next layer for extraction. The 2D-block encoding method with built-in bi-level is used to improve CR and keeps robust to bit errors. The built-in bi-level procedure split the 2D block into many one-dimension signals, and each signal is encoding separately. It is because the bi-level method has the maximum encoding length, which is normally the same as the width of image.

### 3.1. 2D-Layer-Block Extraction Method

**3.1.1. Principle of the Extraction Method.** In the proposed algorithm, to keep the data distribution have the sparsity property and be suitable for bi-level coding, a novel 2D-layer-block extraction method is proposed to keep the Laplacian or Gaussian distribution of data in 2D blocks. In addition, the extraction method can rearrange the order of data encoded and the entropy of data is decreased, so CR can be improved. The principle of the method is introduced as follows.

For encoding residues, if a two-dimension block, called 2D block, can be encoded with  $n$  bits per residue, all of these datum  $x$  in the block must be satisfied with condition shown in (4). Therefore, it is feasible to find these blocks for  $n = 1$  bits, then  $n = 2, \dots, 8$ .

$$\max(x) - \min(x) \leq 2^n - 1. \quad (4)$$

Let us consider all these datum  $x$  in the block governed by a probability density  $f(x)$ , and the entropy is calculated by (6) [33].

$$f(x) = \frac{1}{2\pi\sigma} e^{-((x-\mu)/2\sigma)^2}, \quad (5)$$

$$H(x) = - \int_{-\infty}^{\infty} f(x) \log f(x) dx. \quad (6)$$

By inserting (5) into (6), the entropy for a Gaussian distribution is expressed as

$$H(x) = \log \sqrt{2\pi e \sigma^2} \text{ (Gaussian)}. \quad (7)$$

Since the residue sequence with Gaussian distribution has maximum entropy, the following inequality holds in general.

$$H(x) \leq \log \sqrt{2\pi e \sigma^2} \text{ (in general)}. \quad (8)$$

According to (4), the fixed standard deviation  $\sigma$  is less than  $(2^n - 1)/2$  and (9) can be deduced when we assume  $\mu = 0$ ;  $L$  is the sample size in all of these datum  $x$  in one block. By substituting (9) to (8), equation (10) can be obtained. When blocks for  $n$  bits are found starting from 1 to 8, the entropy of these data in blocks is increased later and later according to (10). So, the entropy is decided by  $n$  and it is possible to improve the compression ratio with this method.

$$\sigma = \sqrt{\frac{1}{L} \sum_{i=1}^L (x_i - \mu)^2} \leq \sqrt{\frac{1}{L} \sum_{i=1}^L x_i^2} \leq \frac{(2^n - 1)}{2}, \quad (9)$$

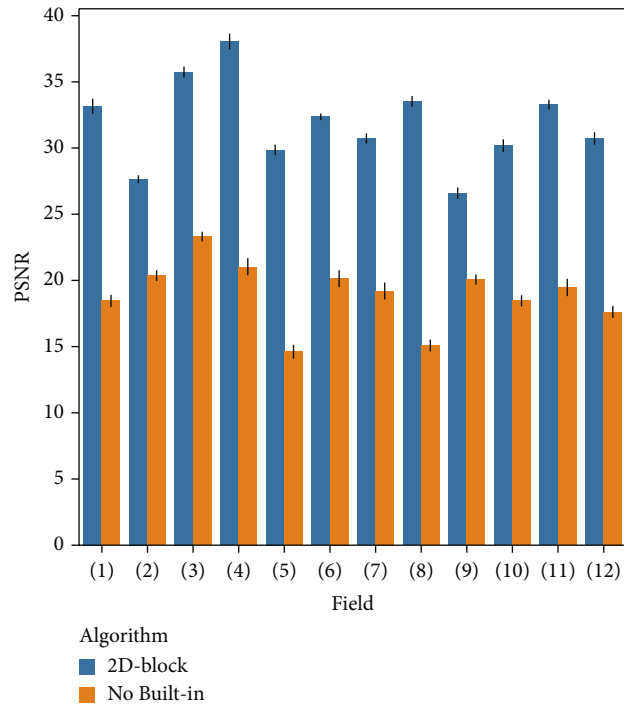
$$H(x) \leq \log \sqrt{2\pi e} \frac{(2^n - 1)}{2}. \quad (10)$$

According to the discussion above, we assume  $\mu = 0$ . After performing prediction and making it zero-mean by removing the average, many residue values are close to zero and the residues follow a Laplacian distribution as shown in Figure 4(a). That is, all the data in one of these encoding 2D blocks will satisfy (11). Note that the sample size  $L$  in the block is above a threshold value of  $th_n$  and data in the block possess a Laplacian or Gaussian distribution approximately.

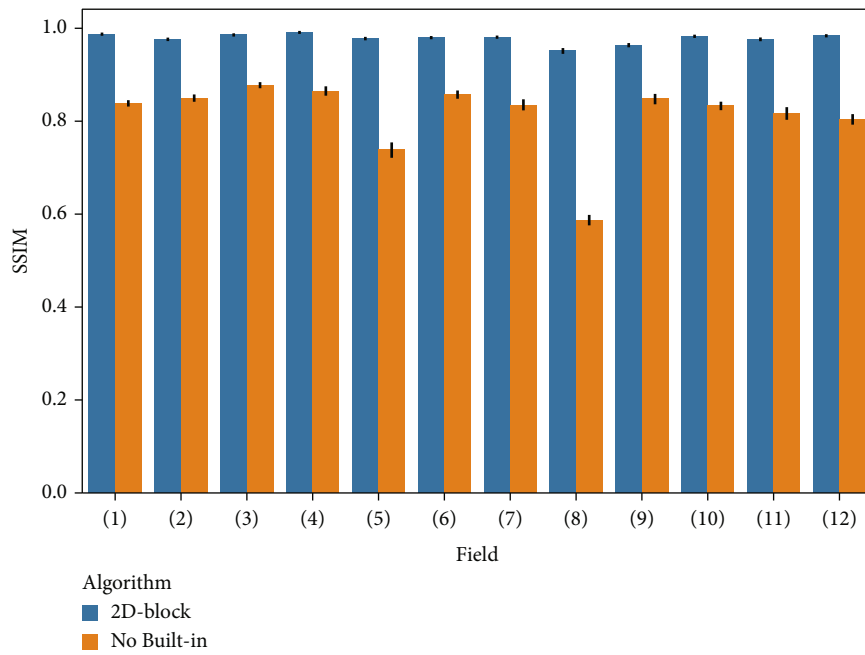
$$\text{mean}(x) \approx 0, L \geq th_n. \quad (11)$$

To proceed, if all of these 2D blocks with  $n = 1$  bits are found, they will be extracted from residue data. The rest of the residue data consists of three portions. The first portion has values bigger than  $(2^n - 1)/2$ , and the second portion has values smaller than  $-(2^n - 1)/2$ , while the third portion contains data which size is smaller than  $th_n$ . It is noted that after the residues  $\in [-(2^n - 1)/2, (2^n - 1)/2]$  shown in Figure 4(a) are extracted, the rest of the residue data will nearly keep the Laplacian distribution. When the extraction is



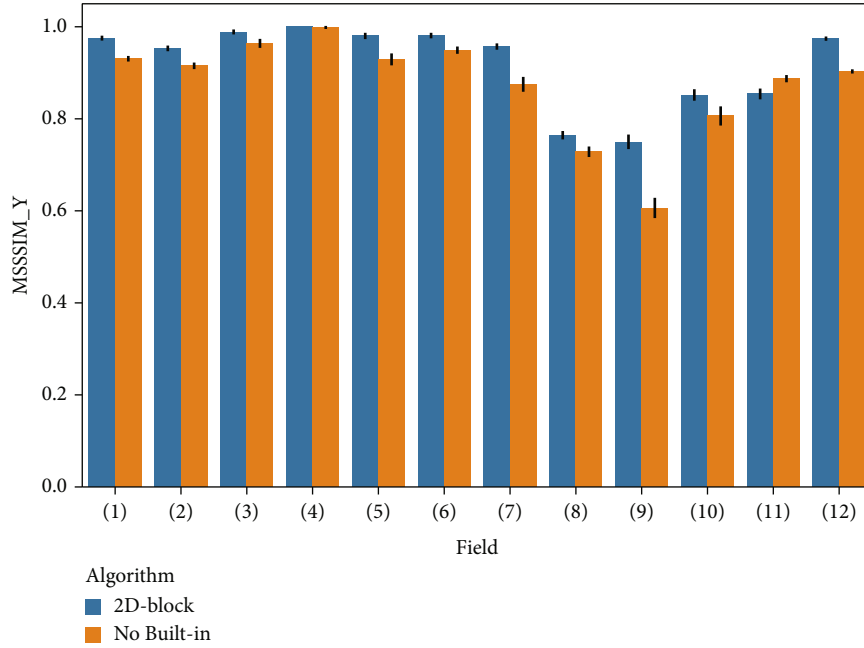


(a)

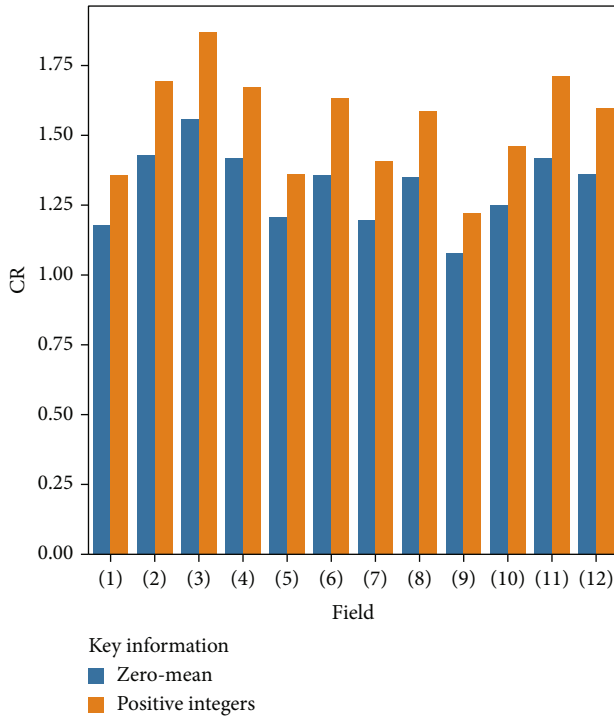


(b)

FIGURE 12: Continued.



(c)

FIGURE 12: Effect of bi-level on PSNR (BER = 0.001, optimal  $th_n$ ).FIGURE 13: Effect of zero-mean on CR (BER = 0.001,  $th_n = 128$ , no bi-level).

repeated from  $n = 1$  to 8, the Laplacian distribution of the remained residue data will change with decreasing the probability density around zeros as shown in Figure 4(b). In addition, the Laplacian or Gaussian distribution in these 2D blocks will be flattened as depicted in Figure 4(c) because of

increasing value of  $(2^n - 1)$ . In this paper, the procedure is called as layer extraction.

**3.1.2. Procedure of the Extraction Method.** According to (11), 2D-layer blocks each having a sample size above the threshold of  $th_n$  are extracted repeatedly. For example, in Figure 5, image residue data is given as a matrix  $40 * 40$  and many 2D blocks belonging  $A$  are extracted. The data which are not included in blocks are reshaped as a matrix  $M$  with the same height as the original residue image, while other remaining data are collected as an array of  $B$ . After the first layer is finished, matrix  $M$  is processed similarly in the next layer. With the extraction and matrix reshape operations, many edge values will be merged with other data and have less effect on compression [24]. The pseudocode of block extraction procedure is shown in Figure 6.

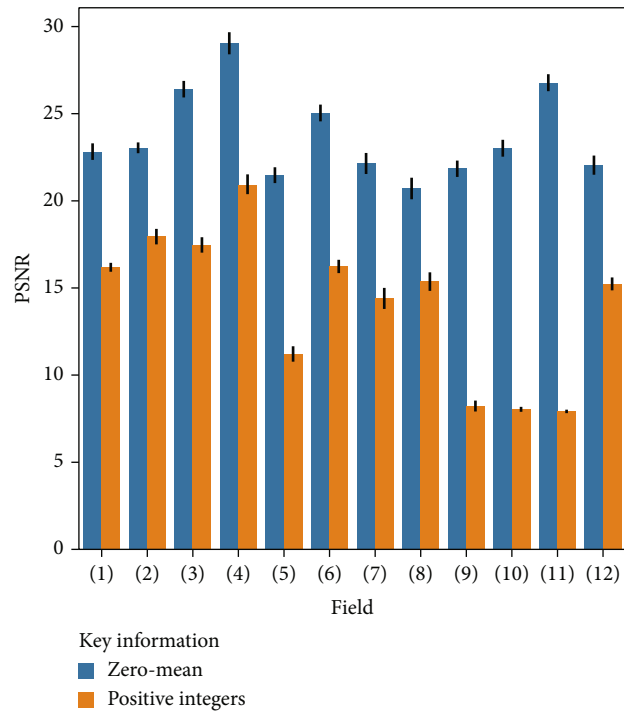
### 3.2. Built-In Bilevel Coding

**3.2.1. Bilevel Coding.** As most of these data in one of 2D blocks has a sparse distribution discussed above, a bi-level coding scheme proposed by our previous works [7, 34] in Figure 7 can be applied.

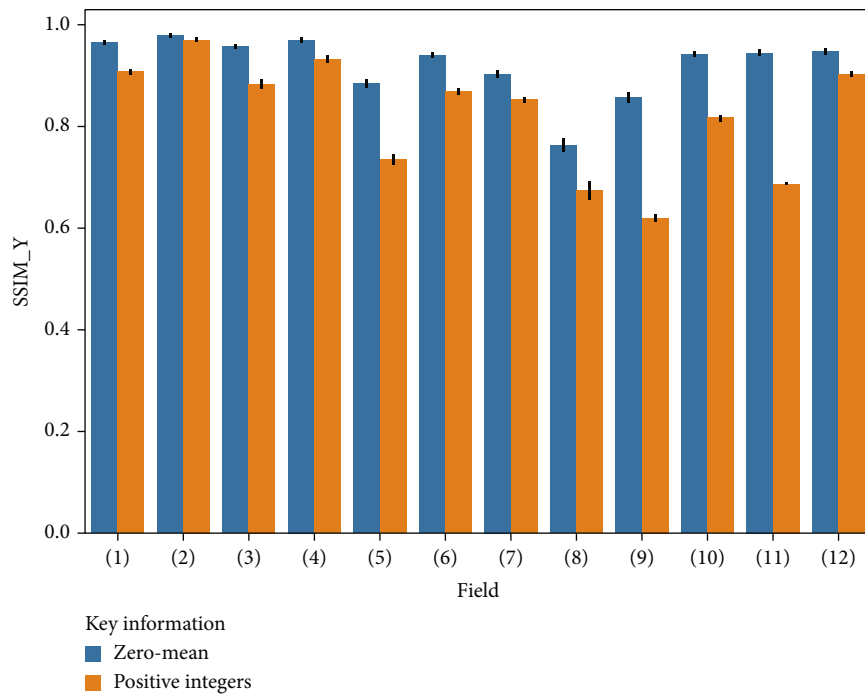
Let  $p_0$  as the probability of a data sample requiring more than  $N_1$  bits and less or equal to  $N_0$  bits to encode. Assuming that  $n_b p_0 \leq 0.3$  [34], the average total length is expressed in the following:

$$L_{ave} = \frac{n_s}{n_b} + n_s N_1 + (N_0 - N_1) n_s n_b p_0, \quad n_b p_0 \leq 0.3. \quad (12)$$

For a given 2D block for  $n$  bits,  $N_0 = n$ , the original total length is  $N_0 * n_s$ . When bi-level block coding is applied, the compression ratio will be improved according to (12).

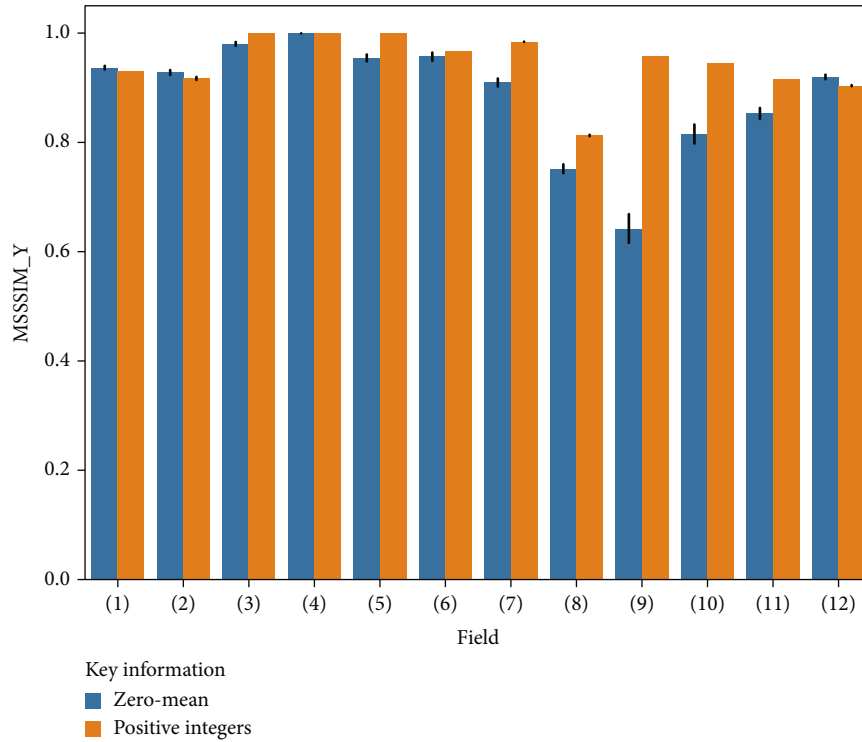


(a)



(b)

FIGURE 14: Continued.



(c)

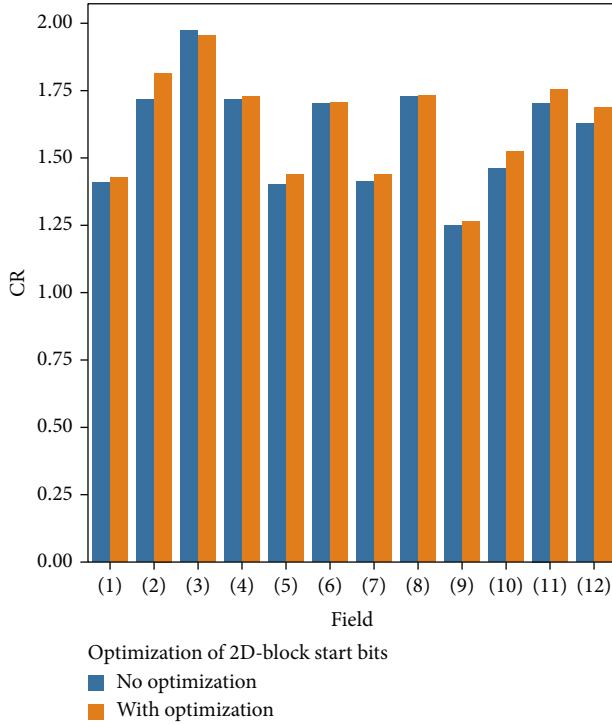
FIGURE 14: Effect of zero-mean on PSNR (BER = 0.001,  $th_n = 128$ , no bi-level).

File Id	Original	Built-in Bi-level (zero mean)	No Bi-level (zero mean)	No Bi-level (positive integer)
(4)				
(9)				

FIGURE 15: Effects on images with different methods.

3.2.2. *Optimization of 2D-Block Start Bits.* According to (12), for an 8-bit gray image data,  $n_s$  is a constant,  $N_0 = 8$ . Given  $N_1$ ,  $p_0$ , which can be estimated, optimal  $n_b$  can be determined to achieve a minimum length of  $L_{ave}$ . By taking the derivative of (12) and setting it equal to zero, the optimized

block size  $x$  can be calculated by equation (13). So, the minimum  $N_1$  satisfying  $n_b p_0 \leq 0.3$  can be found through Figure 8. Finally, the “bits” value in Figure 7 can start from the minimum  $N_1$  and the efficiency of 2D-block extraction will be improved.

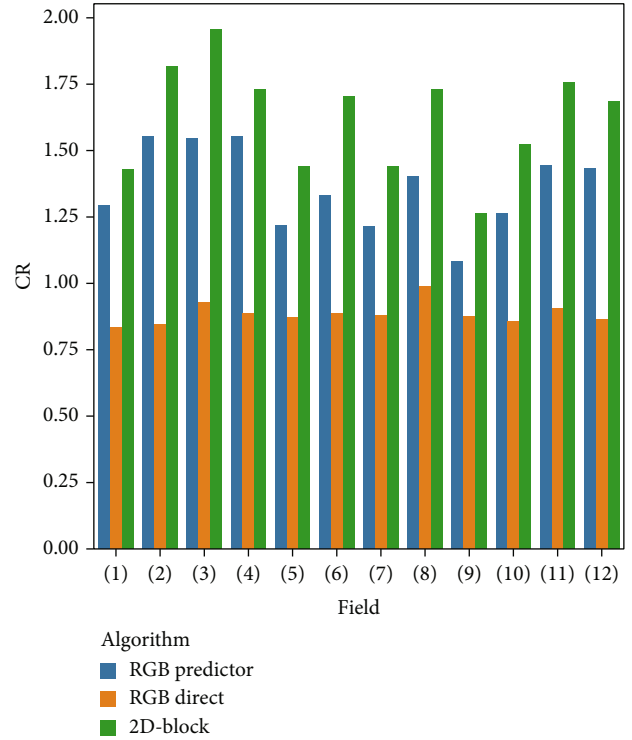
FIGURE 16: Effects on CR with optimization (BER = 0.001,  $th_n = 128$ ).

$$x^* = \frac{1}{\sqrt{(N_0 - N_1)p_0}} = \frac{1}{\sqrt{(8 - N_1)p_0}}. \quad (13)$$

**3.3. 2D-Block Encoding.** Figure 9(a)–9(c) show the details of the encoding scheme. When one color image is given, three channels are separately encoded and the head information including the color space, predictor information, and their hamming coding in (a). In each channel, 2D blocks are extracted with extraction method layer by layer so encoding is implemented recursively layer by layer as well and each layer is encoded separately. In each layer of (b), head data and image data are separately encoded. (c) shows the encoding scheme of head data. Width of parent matrix \* the height of image is the size of  $M$ , and length of  $B$  is the length of the remaining data in Figures 5 and 6. Every block has start position  $(x, y)$ , its size  $(w, h)$ , mean value of the data in block, the maximum bits used of each data in block, and the key information of built-in bi-level coding including  $N_0$ ,  $N_1$ ,  $n_b$ , the number of block  $n_s/n_b$ , and bitstream of block type. Particularly, the mean value of data in block has two functions. One is used to improve the capability of robust to bit error because the mean value keeps the key information of one block. Another is used to ensure zero-mean of the block data, which is the feature in bi-level coding.

## 4. Experiments

To validate our proposed algorithm, Open Images from [http://data.vision.ee.ethz.ch/mentzerf/validation\\_sets\\_lossless/val\\_oi\\_500\\_r.tar.gz](http://data.vision.ee.ethz.ch/mentzerf/validation_sets_lossless/val_oi_500_r.tar.gz), CLIC mobile dataset from

FIGURE 17: Comparison with different color space and predictor (BER = 0.001,  $th_n = 128$ ).

[https://data.vision.ee.ethz.ch/cvl/clic/mobile\\_valid\\_2020.zip](https://data.vision.ee.ethz.ch/cvl/clic/mobile_valid_2020.zip) shown in Table 1, and many classic images from <http://sipi.usc.edu/database/> and <http://homepages.cae.wisc.edu/~ece533/images/> are used.

In the extraction of 2D blocks, an optimal threshold of sample size  $th_n$  is given in Table 1. All these data in one layer will be split into many pieces with 512 samples, which is the same as the width of image. To evaluate the effect of 2D-block encoding, built-in bi-level coding and color space, etc., experiments with different combinations are implemented. All the results are the average values from 10 runs. A bit-error rate (BER) is default set as 0.001. The bi-level method is applied for RGB color image [7].

To evaluate the performances for the bit-error environment, the Peak Signal to Noise Ratio (PSNR in dB), Structural Similarity (SSIM) as (16) [35], Multi-Scale SSIM(MSSSIM\_Y), Spectral Residual-based Similarity (SRSIM), Riesz-transform-based Feature SIMilarity (RFSIM), Feature Similarity (FSIM), FSIMc, and Visual Saliency-based Index (VSI) [36, 37] are used as error metrics to measure the recovered image quality. In (15),  $I(x, y)$  represents the original pixel, while  $\tilde{I}(x, y)$  is the recovered pixel. In (17) and (18),  $K_1 = 0.01$ ,  $K_2 = 0.03$ , and  $L = 255$ .

$$\text{PSNR} = 20 \times \log_{10} \left( \frac{255}{\text{RMSE}} \right), \quad (14)$$

$$\text{RMSE} = \sqrt{\frac{1}{M \times N} \sum_{x=1}^M \sum_{y=1}^N [I(x, y) - \tilde{I}(x, y)]^2}, \quad (15)$$

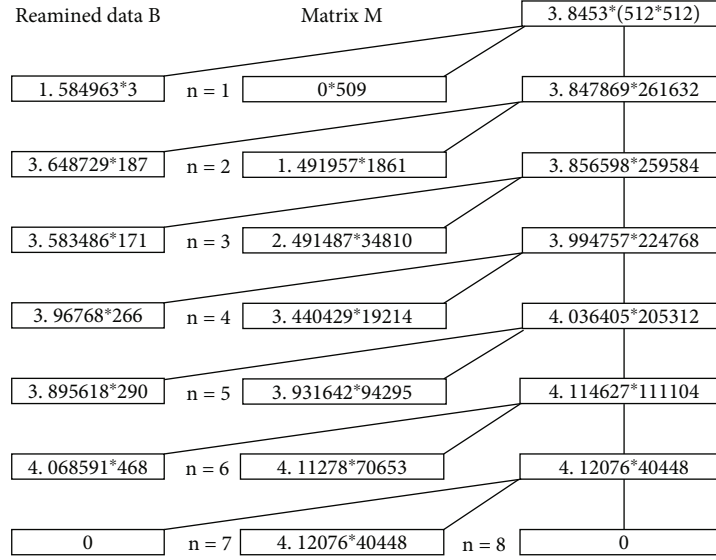


FIGURE 18: The entropy of 2D blocks for  $n$  bits (entropy \* number of samples,  $th_n = 128$ ). Left is the entropy of blocks extracted.

$$SSIM(x, y) = \frac{(2\mu_x\mu_y + C_1)(2\sigma_{xy} + C_2)}{(\mu_x^2 + \mu_y^2 + C_1)(\sigma_x^2 + \sigma_y^2 + C_2)}, \quad (16)$$

$$C_1 = (K_1L)^2, \quad (17)$$

$$C_2 = (K_2L)^2. \quad (18)$$

**4.1. Comparison.** In this experiment, our proposed method is compared with many state-of-the-art methods from Refs. [16, 17], engineered lossless compression algorithms including PNG, Jpeg2000, WebP, FLIF, and deep learning-based lossless compression algorithm L3C. As all of these methods are not suitable to be applied in bit-error situation, these methods with hamming code (7,4) are supposed a solution robust to bit-error environment. The results are given in Tables 2 and 3 and Figure 10.

In Table 2, the results are taken from Refs. [15, 16] and the best results of CR with hamming code are listed in the last second column. The average CR of our method is 1.31296 and better than 1.116592. In Table 3, the results of Jpeg2000, WebP, and FLIF are achieved through the compression tools including OpenJpeg, WebP from Google, and FLIF from Cloudinary. As a deep learning method, the result of L3C is achieved by using the neuron model trained with Open Images dataset to compress images. The average CR of our method is 1.682933 and better than others such as bi-level (1.66655), FLIF (1.429898), and L3C (1.396234). In addition, it is noticed that CR of L3C with images (5) and (9) from the CLIC dataset get the worst results, which are 1.098277 and 0.918406. One reason that leads to the result is that the neuron model used to compress is trained with Open Images dataset and L3C do not perform well on the images from a different dataset CLIC.

Figure 10 shows the image quality assessment results. PSNR, SSIM and MSSSIM\_Y can better reflect the situation

of bit-error channel. Therefore, only these three assessment results are discussed in the late section.

According to the comparison results in Tables 2 and 3 and Figure 10, the compression ratio of our proposed method is higher than bi-level coding although 2D-block encoding requires more header bits to encode the information about the position and size of block. And similar image quality with bi-level coding is kept in Figure 10. The reason is that the 2D-layer-block extraction method rearranges the data order to decrease the entropy and the data distribution of one-layer blocks nearly keeps as Laplacian distribution which is suitable for bi-level coding as discussed before. In addition, the analysis will be further discussed in Section 4.6–4.7.

**4.2. The Effect of Built-In Bilevel Encoding.** To investigate the advantage of the bi-level coding method, two experiments including “built-in bi-level” and “no bi-level” are implemented. The results are shown in Figures 11 and 12.

When the center of the Laplacian distribution is located at zero, bi-level coding can require less bits to encode. So, the built-in bi-level encoding method achieves the best compression ratio as shown in Figure 11. As bi-level coding is proposed for noisy channel [7], it gets higher PSNR, SSIM, and MSSSIM\_Y and maintains a better image quality just as shown in Figure 12.

**4.3. Comparison between Zero-Mean and Positive Integers.** As we know, min value can also be as the key information to improve image quality just as mean value. In this experiment, positive integer values by removing min value are encoded. The results are shown in Figures 13 and 14.

As discussed in Section 4.2, the compression ratio with built-in bi-level coding is higher when zero-mean values are used. While without the bi-level coding used, the compression ratio with positive integer values by removing the min value is higher than with zero-mean values in

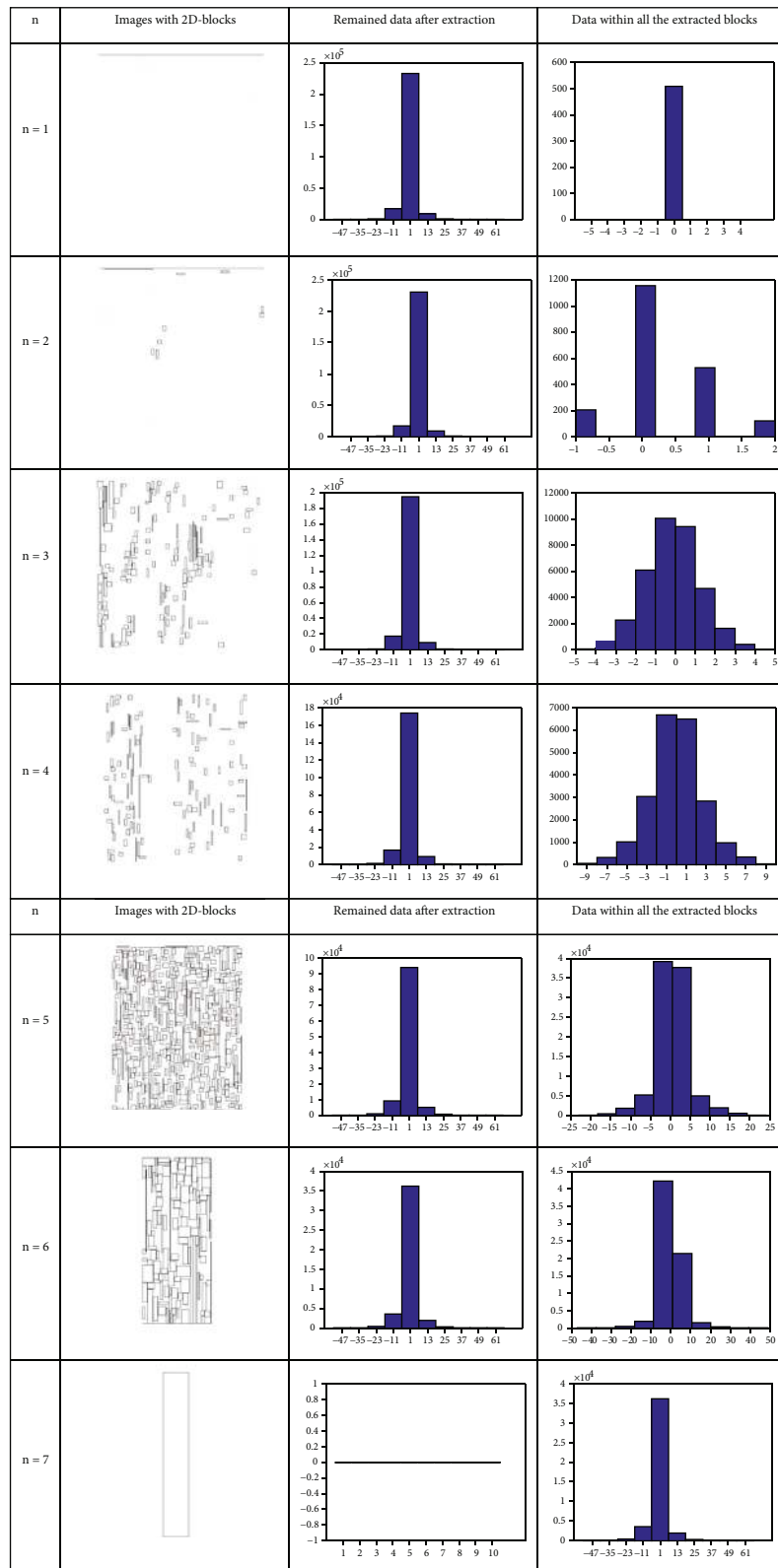


FIGURE 19: Change of data distribution ( $Th_n = 128$ ).

Figure 13. The reason is that encoding with zero-mean values requires the sign bit. But the usage of sign bit can result in that encoding with zero-mean values has smaller

amplitude around the mean value, so encoding with zero-mean values has less effect on PSNR and SSIM than with positive integer values. Consequently, according to the



PSNR and SSIM shown in Figures 14(a) and 14(b), encoding with zero-mean values can achieve better PSNR and SSIM than with positive integer, which is consistent with Section 3.3.

Figure 15 gives the reconstructed images in different methods. According to the results of image (9) at the last row, encoding with “positive integer” values shows worse image quality than others. But in Figure 14(c), the MSSSIM value of “positive integer” is higher than “zero-mean.” Therefore, the MSSSIM result cannot be consistent with the real image quality in some cases.

**4.4. Evaluation with Optimization of 2D-Block Start Bits.** In the 2D-block coding method, the excessive number of blocks can lead to a decrease of compression ratio. So, the experiment based on optimization is conducted and the results are given in Figure 16. It is observed that the optimization does work and improves the compression ratio.

**4.5. Evaluation with Different Color Spaces and Predictors.** In Figure 17, “RGB Direct” denotes that RGB image is directly encoded without a predictor while “RGB Predictor” designates that the RGB residue with a predictor is used. It shows that color space YCrCb in “2D block” performs better than color space RGB and “RGB Direct” is the worst, which validates that the difference between pixel values and their predictions is generally smaller numbers than the pixel values themselves [13].

**4.6. The Decrease of Entropy.** In Figure 18, the original entropy of gray image of (4) is 3.8453 while the average entropy with the 2D-block encoding method is 3.772696. It is indicated that the compression ratio is improved and less than the value according to the information theory, which is coincident with the principle in Section 3.1.

**4.7. The Change of Data Distribution in Every Layer.** Figure 19 shows all the images with 2D blocks and the data distribution of remained data or all the data in blocks from gray image of (4) without the optimization of 2D-block start bits. The images indicated reshape operator has changed the distribution of edge cared about by many predictors [24]. The remained data distribution is close to the Laplacian distribution shown in the left histogram, and the data distributions in blocks are close to Gaussian or Laplacian distribution shown in the right histogram. All of these are coincident with the analysis in Section 3.1.

**4.8. Discussion.** Through these experiments, the results of comparison proved that our method performs better than state-of-the-art methods, engineering lossless compression algorithms and deep learning methods under bit-error situation. There are four main reasons.

First, the 2D block extraction method extracts the data encoded with smaller bits layer by layer; thus, the entropy is decreased as Section 4.6 show.

Second, the edge data always cause poor compression rate but the 2D-block extraction method has changed the edge data distribution. And the data distribution of each

layer block nearly keeps as Laplacian distribution which is suitable for bi-level coding as Figure 19 of Section 4.7.

Third, built-in bi-level coding with zero-mean value can preserve high image quality under bit-error environment as Section 4.2 and 4.3 discussed.

At last, optimization of 2D-block start bits and color space used in “2D block” is an important mechanism to improve the compression rate as the discussion in Sections 4.4 and 4.5.

## 5. Conclusions

When image data is transferred through wireless communication systems, bit errors may occur and will cause corruption of image data. To reducing the bit-error effect, a bit-aware lossless image compression algorithm based on bi-level coding can be applied. But bi-level coding is one of the one-dimension coding methods and has not considered the inherent statistical correlation in 2D context region. So, to resolve this shortage, a novel 2D-layer-block extraction and encoding method with built-in bi-level coding is proposed to improve the compression ratio. With the layer extraction method, the data distribution is close to the Laplacian distribution after each layer extraction, which is suitable for bi-level coding. For color image, a lossless color transformation from RGB to the YCrCb color space is used for the decorrelation of color components. Through experiments, it is demonstrated that our proposed method obtains the better lossless compression ratio and keeps the same image quality with the bi-level method under noise transmission channel. Although it is not as efficient when compared to state-of-the-art methods in terms of lossless compression ratio sometimes, it is more robust to bit errors caused by noisy channel. Furthermore, after applying the feed-forward error control scheme, different predictor, and coding method, we can achieve better compression efficiency, since the bi-level block coder requires a smaller number of bits by the bit-error protection algorithm than the amount required by the entropy coder. Also, it is noted that deep learning methods are trained with Open Images dataset but perform poor on the images from a different dataset. Therefore, the generalization ability of deep learning methods is required to be improved in the future.

## Data Availability

The [CLIC mobile dataset], [Open Images] and other classic images used to support the findings of this study are included within the article.

## Ethical Approval

If the images or other third-party material in this article are not included in the article’s Creative Commons license, it is required to obtain permission directly from the copyright holder.

## Conflicts of Interest

The authors declare that they have no conflicts of interest.

## Acknowledgments

This work was supported in part by the National Natural Science Foundation of China (No. 61502423), Zhejiang Provincial Natural Science Foundation (No. LY16G020012), and Zhejiang Province Public Welfare Technology Application Research Project (Nos. LGF19F010002, LGN20F010001, LGF20F010004, and LGG21F030014).

## References

- [1] J. Ballé, D. Minnen, S. Singh, S. J. Hwang, and N. Johnston, "Variational image compression with a scale hyperprior," 2018, arXiv preprint arXiv:1802.01436.
- [2] J. Lee, S. Cho, and M. Kim, *An End-to-End Joint Learning Scheme of Image Compression and Quality Enhancement with Improved Entropy Minimization*, 2019, <https://arxiv.org/pdf/1912.12817>.
- [3] F. Jiang, W. Tao, S. Liu, J. Ren, X. Guo, and D. Zhao, "An end-to-end compression framework based on convolutional neural networks," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 28, no. 10, pp. 3007–3018, 2018.
- [4] Z. Cheng, H. Sun, M. Takeuchi, and J. Katto, "Learned image compression with discretized Gaussian mixture likelihoods and attention modules," in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Seattle, WA, USA, June 2020.
- [5] K. N. Shruthi, B. M. Shashank, Y. S. Saketh et al., "Comparison analysis of a biomedical image for compression using various transform coding techniques," in *2016 IEEE 6th International Conference on Advanced Computing (IACC)*, pp. 297–303, Bhimavaram, India, February 2016.
- [6] A. Avramović and G. Banjac, "On predictive-based lossless compression of images with higher bit depths," *Telfor Journal*, vol. 4, no. 2, pp. 122–127, 2012.
- [7] L. Tan and L. Wang, "Bit-error aware lossless image compression," *International Journal of Modern Engineering*, vol. 11, no. 2, pp. 54–59, 2011.
- [8] "WebP image format," <https://developers.google.com/speed/webp/>.
- [9] J. Sneyers and P. Wuille, "FLIF: free lossless image format based on MANIAC compression," in *2016 IEEE International Conference on Image Processing (ICIP)*, Phoenix, AZ, USA, September 2016.
- [10] J. Townsend, T. Bird, and D. Barber, "Practical lossless compression with latent variables using bits back coding," *The International Conference on Learning Representations*, Louisiana, USA, 2019, arXiv preprint arXiv:1901.04866.
- [11] F. Kingma, P. Abbeel, and J. Ho, "Bit-swap: recursive bits-back coding for lossless compression with hierarchical latent variables," in *In International Conference on Machine Learning*, pp. 3408–3417, California, USA, 2019.
- [12] F. Mentzer, E. Agustsson, M. Tschannen, R. Timofte, and L. Van Gool, "Practical full resolution learned lossless image compression," in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Long Beach, CA, USA, June 2019.
- [13] R. Jovanovic and R. A. Lorentz, "Adaptive lossless prediction based image compression," *Applied Mathematics & Information Sciences*, vol. 8, no. 1, pp. 153–160, 2014.
- [14] I. Matsuda, N. Ozaki, Y. Umezu, and S. Itoh, "Lossless coding using variable block-size adaptive prediction optimized for each image," *EUSIPCO*, pp. 818–821, 2005.
- [15] X. Zhao and Z. He, "Lossless image compression using super-spatial structure prediction," *IEEE Signal Processing Letters*, vol. 17, no. 4, pp. 383–386, 2010.
- [16] T. Strutz, "Context-based predictor blending for lossless color image compression," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 26, no. 4, pp. 687–695, 2016.
- [17] S. Kim and N. Cho, "Hierarchical prediction and context adaptive coding for lossless color image compression," *IEEE Transactions on Image Processing*, vol. 23, no. 1, pp. 445–449, 2014.
- [18] H. S. Malvar, G. J. Sullivan, and S. Srinivasan, "Lifting-based reversible color transformations for image compression," *Proceedings of SPIE-The International Society for Optical Engineering*, 2008.
- [19] R. Starrosolski, "Simple fast and adaptive lossless compression algorithm," *Software: Practice and Experience*, vol. 37, no. 1, pp. 65–91, 2007.
- [20] M. Wienberger, G. Seroussi, and G. Sapiro, "LOCO-I: a low complexity, context-based, lossless image compression algorithm," in *Proceedings of the Conference on Data Compression*, pp. 140–148, Snowbird, UT, USA, 31 March–3 April 1996.
- [21] S. D. Rane and G. Sapiro, "Evaluation of JPEG-LS, the new lossless and near-lossless still image compression standard for compression of high-resolution elevation data," *IEEE Transactions of Geosciences and Remote Sensing*, vol. 39, no. 10, pp. 2298–2306, 2001.
- [22] X. Wu and N. Memon, "CALIC—a context based adaptive lossless image codec," in *1996 IEEE International Conference on Acoustics, Speech, and Signal Processing Conference Proceedings*, pp. 1890–1893, Atlanta, GA, USA, May 1996.
- [23] A. Avramović and B. Reljin, "Gradient edge detection predictor for image lossless compression," in *In Proceedings ELMAR-2010*, pp. 131–134, Zadar, Croatia, September 2010.
- [24] L. Xin and M. T. Orchard, "Edge-directed prediction for lossless compression of natural images," *IEEE Transactions on Image Processing*, vol. 10, no. 6, pp. 813–817, 2001.
- [25] L. Kau and Y. Lin, "Adaptive lossless image coding using least-squares optimization with edge-look-ahead," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 52, no. 11, pp. 751–755, 2005.
- [26] L. Kau and Y. Lin, "Least-squares-based switching structure for lossless image coding," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 54, no. 7, pp. 1529–1541, 2007.
- [27] F. Hsieh and K. Fan, "A high performance lossless image coder," in *in Conference on Computer Vision & Graphic Image Processing*, Taipei, Taiwan, 2005.
- [28] G. Ulacha and R. Stasinski, "Performance optimized predictor blending technique for lossless image coding," in *In 2011 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 1541–1544, Prague, Czech Republic, 2011.
- [29] W. Dai, H. Xiong, J. Wang, and Y. Zheng, "Large discriminative structured set prediction modeling with max-margin Markov network for lossless image coding," *IEEE Transactions on Image Processing*, vol. 23, no. 2, pp. 541–554, 2014.

- [30] Y. Zhang and D. A. Adjeroh, "Prediction by partial approximate matching for lossless image compression," *IEEE Transactions on Image Processing*, vol. 17, no. 6, pp. 924–935, 2008.
- [31] M. Domanski and K. Rakowski, "Color transformations for lossless image compression," in *IEEE Signal Processing Conference*, pp. 1–4, Tampere, Finland, 2000.
- [32] S. C. Pei and J. J. Ding, "Improved reversible integer-to-integer color transforms," in *IEEE International Conference on Image Processing*, pp. 473–476, Cairo, Egypt, 2009.
- [33] P. J. Coles, M. Berta, M. Tomamichel, and S. Wehner, "Entropic uncertainty relations and their applications," *Reviews of Modern Physics*, vol. 89, no. 1, pp. 1–58, 2017.
- [34] L. Tan, J. Jiang, and Y. Zhang, "Bit-error aware lossless compression of waveform data," *IEEE Signal Processing Letters*, vol. 17, no. 6, pp. 547–550, 2010.
- [35] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: from error measurement to structural similarity," *IEEE Trans. on Image Processing*, vol. 13, no. 4, pp. 1–14, 2004.
- [36] H. Amirpour, A. M. G. Pinheiro, M. Pereira, and M. Ghanbari, "Reliability of the most common objective metrics for light field quality assessment," in *ICASSP 2019-2019 IEEE international conference on acoustics, speech and signal processing (ICASSP)*, pp. 2402–2406, Brighton, United Kingdom, 2019.
- [37] <https://sse.tongji.edu.cn/linzhang/IQA/IQA.htm>.