Pamukkale Univ Muh Bilim Derg, 29(3), 230-237, 2023

Pamukkale Üniversitesi Mühendislik Bilimleri Dergisi

Pamukkale University Journal of Engineering Sciences

Sample-Weighted RDPCM for lossless image compression with HEVC intra-prediction

HEVC iç tahmin ile kayıpsız görüntü sıkıştırma için ağırlıklandırılmış RDPCM yöntemi

Ali AKMAN[1], Serap CEKLİ[2]*

[1]Department of Computer Engineering, Faculty of Engineering, İstanbul Ticaret University, İstanbul, Turkey.
aakman@ticaret.edu.tr

[2]Department of Electrical-Electronic Engineering, Faculty of Engineering, İstanbul University-Cerrahpaşa, İstanbul, Turkey.
serap.cekli@uic.edu.tr

Received/Geliş Tarihi: 28.12.2021 Revision/Düzeltme Tarihi: 15.07.2022 doi: 10.5505/pajes.2022.68740
Accepted/Kabul Tarihi: 19.07.2022 Research Article/Araştırma Makalesi

Abstract

The correlation between pixels isn't completely removed, which degrade the performance of compression because the transform and quantization steps are bypassed in the lossless image compression with high efficiency video coding intra-prediction. To improve this situation, additional methods such as residual differential pulse code modulation is used in conjunction with the intra-prediction. The basic principle of residual differential pulse code modulation is based on the calculation of an enhanced residue of a pixel which is obtained with subtraction of the residue of left or top neighbor pixel from the residue of the pixel. In this study, a new residual differential pulse code modulation method, sampled-weighted residual differential pulse code modulation, considering an average of two or three neighboring pixels with weight is developed. A simple and fast algorithm is also proposed to determine the weight coefficients to be used in calculating the weighted average for the three neighboring pixels case. To obtain their compression performances the proposed sample-weighted residual differential pulse modulation methods are applied to various test images. The compression ratios of the compressed images are obtained.

Keywords: High efficiency video coding, Residual differential pulse code modulation, Lossless image compression.

Öz

Yüksek verimli video kodlama iç tahmin yöntemi ile kayıpsız sıkıştırmada dönüşüm ve kuantalama basamaklarının eksikliği pikseller arasındaki ilişkiyi tam olarak silememekte ve bu sebeple sıkıştırma performansı düşmektedir. Bu durumu iyileştirmek için iç tahmin yöntemine ilave olarak kullanılabilecek farklı yöntemler geliştirilmektedir. Bunlardan birisi de kalıntı diferansiyel darbe kodlama modülasyonu yöntemidir. Temeli bir piksel için bulunan kalıntıdan komşu üst ya da sol pikselin kalıntı değerinin çıkartılmasıyla iyileştirilmiş kalıntı değerinin elde edilmesine dayanır. Bu çalışmada iki ya da üç komşu pikselin kalıntılarının ağırlıklı ortalamasını kullanan yeni bir kalıntı diferansiyel darbe kod modülasyonu yöntemi geliştirilmiştir. İki örneğin kullanıldığı yöntemde, komşu pikseller ve bu piksellerin kalıntılarının ortalaması için kullanılan ağırlık katsayıları ilgili iç tahmin modu için kullanılanlar ile aynı seçilmiştir. Üç örnekli kalıntı diferansiyel darbe kod modülasyonu yönteminde komşu piksellerin kalıntıların ağırlıklı ortalamasının hesaplanmasında kullanılacak katsayıların belirlenmesi için basit ve hızlı bir yöntem önerilmiştir. Önerilen yöntemlerin performanslarını elde etmek için değişik görüntülere yöntem uygulanmıştır. Sıkıştırma oranı ve tepe sinyal gürültü oranı değerleri elde edilmiştir.

Anahtar Kelimeler: Yüksek verimli video kodlama, Kalıntı diferansiyel darbe kodlama modülasyonu, Kayıpsız görüntü sıkıştırma.

1 Introduction

Image and video compression is widely used in many applications such as broadcasting, medical imaging, remote sensing, automotive vision systems, wireless multimedia sensor networks, content distribution in order to improve bit-rate, storage capacity and energy consumption [1]-[6]. Some applications where visual data need to be completely preserved use lossless compression to avoid any distortion [7].

The HEVC standard is a video encoding method that has been widely used recently. It provides a 50% bitrate reduction compared to its predecessor with the same image quality [8]. This performance increase keeps up even with high-resolution images [8], [9]. So, HEVC is indispensable for high-resolution videos or images. HEVC, which can be implemented in lossy or lossless form in accordance with the purpose of application, is also a suitable option for lossless image coding.

In lossy image coding, the image can be compressed effectively after applying the crucial transform stage and the transform coefficient quantization stage, but this situation yields distortion when the image is reconstructed. The lossless image coding operation differs from the lossy image coding process in some respects. For lossless image compression, transformation, quantization, filtering processes are not performed in the HEVC intra prediction stage. In HEVC Standard, it is possible to use transform bypass mode to compress images losslessly. Although the use of this mode is simple; it provides inefficient compression performance. Discrete cosines transform (DCT) is used in the transform stage of the HEVC. If we use DCT in the transform stage, another performance degradation situation occurs. DCT produces numbers with large bit lengths as transform coefficients. These transform coefficients with large bit lengths have negative effects on compression performance. On the other hand, due to the skipping of the transform step, the redundancy cannot be

*Corresponding author/Yazışılan Yazar

efficiently removed after block-based prediction, and the compression performance is negatively affected in this case. For this reason, it is necessary to develop some techniques which can be used instead of the transform step [3],[8],[10].

There are two possible ways to improve the lossless compression performance of HEVC. One of them is to use suitable transform at the transform stage. Authors of [11] propose the use of the integer-to-integer (i2i) transform at the transform stage of the HEVC to improve the lossless performance. They obtain about 7.5% performance improvement compared to the transform bypass case. In [12], a new hybrid transform is proposed to be used at the transform stage of HEVC. With this new transform, 7.4% performance improvement is achieved. The transform stage has a high impact on the complexity of the HEVC standard. In all intra mode, 9% of the running time of the HEVC HM reference software is consumed in the transform phase [10]. So, we need simple methods to reduce the complexity instead of the transform stage.

The redundancy is tried to be reduced by applying some approaches to residues which are obtained by subtracting from the original image block after block-based prediction [13]. One of these approaches is the residual differential pulse code modulation (RDPCM) method. DPCM, also known as prediction coding, is used to reduce the spatial correlation between pixels. Many studies have been performed on this subject and several studies have been still carried out. These studies process residues by dealing with their different aspects and try to improve performance [1],[2],[14],[15].

The RDPCM approach is based mainly on subtraction operations between residue values. The subtraction operations between the pixels can be performed in the horizontal axis or vertical axis depending on the prediction mode [16]. The residue value is subtracted from its residue value of upper neighboring pixel in vertical RDPCM while the residue value is subtracted from its residue value of left neighboring pixel in horizontal RDPCM. However, there are some more complicated two-dimensional approaches [17]-[19]. In Secondary RDPCM (SRDPCM), horizontal RDPCM is applied to horizontal modes and vertical RDPCM is applied to vertical modes twice. In Cross RDPCM (CRDPCM) case, vertical RDPCM is applied after horizontal RDPCM process and horizontal RDPCM is applied after having vertical RDPCM operation.

In this study, an RDPCM method, which is suitable for HEVC intra prediction angular modes, is proposed. In the proposed method, a new residue block is obtained via subtracting the weighted average of residue values of adjacent neighboring pixels from a residue value. The selection of neighboring pixels and their weights are determined according to the selected intra prediction mode. To calculate the average value, two or three adjacent neighboring pixels can be used. The method with two reference pixels is based on the method given in [3]. The method with three reference pixels is based on the method presented in [7]. All these methods are proposed to be used with the sample-based angular prediction (SAP). But in this work, we apply these methods to the RDPCM. Also, we propose a simple procedure to calculate the weighting coefficients of reference pixels for the method with three reference pixels. The proposed methods have been applied for compression of grayscale images and the effectivity of the methods are examined. From the results obtained, it has been observed that the proposed methods provide a competitive coding efficiency compared to the ordinary RDPCM and other lossless compression methods. In addition to the coding performance, we get improved encoder run time over the method proposed in [7]. Therefore, we show that the weighted multi-reference method can be applied to the RDPCM efficiently.

The rest of the paper is summarized as follows. The second section presents the proposed method by explaining it in detail. In the third section, the fulfilled experiments by using different sizes of test images, and the obtained results from the tests are given. In the last section, the obtained results are discussed, and concluding remarks are given.

2 Lossless image compression method

The block diagram of the lossless compression method used in this study is given in Figure 1. The first stage of compression is the block-based intra prediction step. Residues are obtained by subtracting the predicted pixel values from the corresponding original image pixel values. In block-based prediction, the prediction is inefficient due to fact that the reference samples are away from the pixels to be predicted. In lossless compression, this inefficiency cannot be compensated because of the absence of the transform stage. To increase the efficiency, the RDPCM method is performed on the residues, and a compressed image is obtained by using the newly obtained coefficients after applying the method.

Coefficient coding is performed similarly to HEVC transform coefficient coding operation. In the decoder implementation, the symmetrical version of the given system in Figure 1 is utilized.

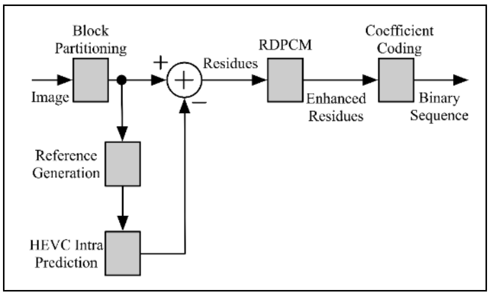

Figure 1. Block diagram of HEVC intra prediction based lossless image compression with RDPCM.

2.1 Block partitioning

Image is partitioned into blocks called coding tree unit (CTU) due to HEVC is a block based technique. This block partitioning process is quite flexible, and this flexibility provides better performance in removing the spatial redundancies in especially more detailed image regions. Each CTU is partitioned into blocks called coding units (CU) in accordance with the quadtree structure. CUs can have the size of 32×32, 16×16, or 8×8. In this study, the image is uniformly partitioned into the CUs, which means all CUs have the same sizes throughout the whole image.

The processing order of CUs is an important property of compression performance. Three different partitioning schemes and various scanning orders are examined to obtain how the partition and the processing order of CUs affect compression performance (Figure 2). In the first method, the partitioning scheme is chosen as: the image is divided into four blocks and then each block is subdivided into four blocks. The process continued until the desired CU size (8, 16, or 32 pixels) is obtained. In the first method, CUs are processed in Z-scan order Figure 2(a).

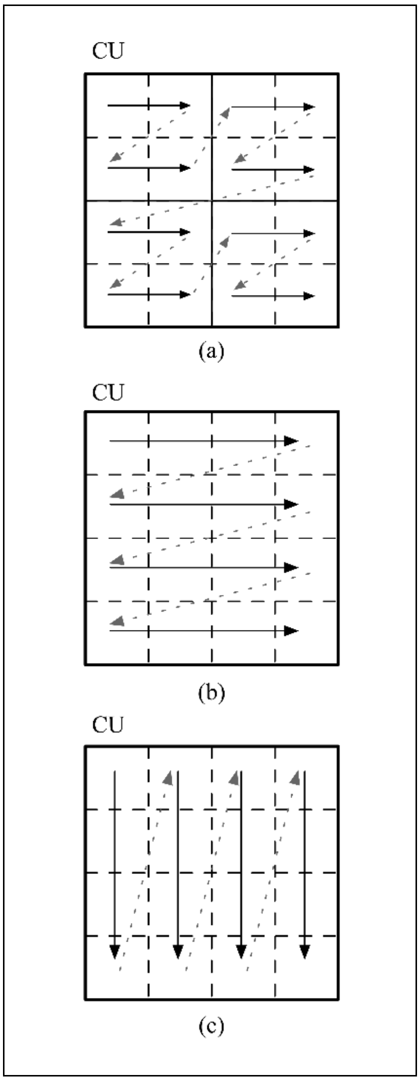

Figure 2. CU processing order. (a): Z scan order. (b): horizontal scan order (H-scan), and (c): Vertical scan order (V-scan).

In the second and third methods, the image is partitioned into CUs to form a grid. In the second method, CUs are processed horizontally through the row of the grid starting first CU of the row Figure 2(b). In the third method, CUs are processed vertically through the column of the grid starting the first CU of the column Figure 2(c).

2.2 Intra prediction

The intra prediction is used to eliminate spatial redundancies of the image. For this process, the image is partitioned into coding blocks (CU). In intra prediction, the prediction process is performed at the level of prediction units (PU). CU and PU sizes can be equal, or PU can be formed by subdividing each CU into four units. In this study, CU size and PU size are chosen the same, which means each CU is subdivided into one PU.

Each PU has a different prediction mode. HEVC intra prediction has 35 prediction modes comprising of DC mode, planar mode, and 33 angular modes. The angular modes designed complying with modeling the directional properties in the image provide a good compromise between coding complexity and coding efficiency due to having many angular directions. DC and planar prediction modes are used to predict smooth and gradually varying image regions.

A pixel value of PU is predicted via interpolation of two reference samples, in angular modes. These reference samples are determined by means of prediction angle, and projection from that pixel to reference samples, and a graphical representation is given in Figure 3.

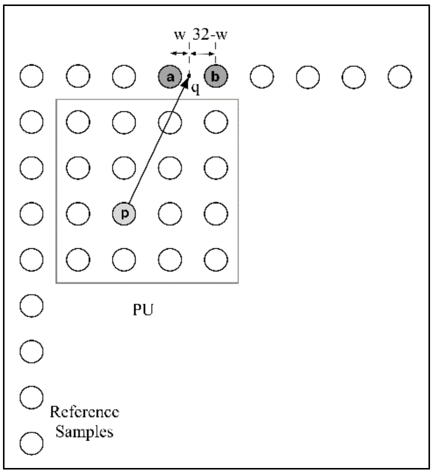

Figure 3. Block-based angular intra prediction in HEVC.

In the angular modes, a pixel value is predicted by using the following equation:

$$P(i,j) = ((32-w).a + w.b + 16) \gg 5 \quad (1)$$

In (1), a and b represent reference samples. 32-w and w are the 5-bit interpolation weights which are determined with prediction angle. Reference samples are obtained from the left, top left, and top neighboring blocks which are reconstructed, and a representation is given in Figure 3. If there are unavailable pixels in the reference samples, in this case these pixels are replaced by the available ones.

2.3 Residual differential pulse code modulation (RDPCM)

RDPCM is employed to eliminate the possible correlation between residues after HEVC intra prediction stage. The fundamental principle of the method is to subtract the neighboring residue value of a pixel from the residue value of that pixel on account of finding the enhanced residue value. There are two kinds of RDPCM methods as vertical and horizontal RDPCM, according to the selection of neighboring pixels. RDPCM is applied for each PU. The enhanced residual coefficients for the N×N sized PU is calculated with horizontal RDPCM using the following equation:

$$r_H(i,j) = R(i,j) - R(i-1,j)\,, \quad 0 \le i < N, 0 \le j < N \quad (2)$$

As for r_H is the enhanced residue coefficients calculated with horizontal RDPCM, and $S(i,j)$ represents the pixel values of the original image are given, then $R(i,j) = S(i,j) - P(i,j)$ is the calculated residue values obtained after intra prediction. If index i is 0, the index of R is negative. Negative indexed R values are obtained from residue reference samples, and a representation is given in Figure 4(a). At the decoder, intra prediction residue values are calculated from enhanced residue values by using the following equation:

$$R(i,j) = r_H(i,j) + R(i-1,j), \quad 0 \le i < N, 0 \le j < N \quad (3)$$

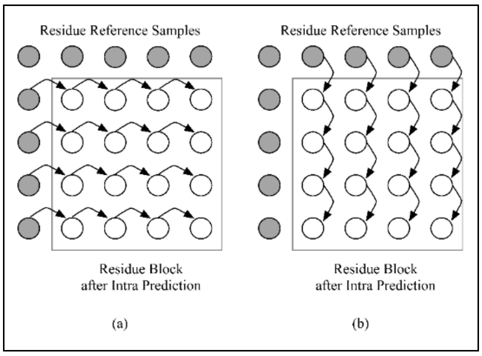

Figure 4. In NxN sized PU, pixel by pixel RDPCM:
(a): Horizontal. (b): Vertical.

Enhanced residue coefficients are found in a similar way for the vertical RDPCM method. In this case, the direct upper neighbors of that pixel are used, and the related representation is given in Figure 4b. The enhanced residue values are calculated via Equation (4).

$$r_V(i,j) = R(i,j) - R(i,j-1), \quad 0 \le i < N, 0 \le j < N \tag{4}$$

where, r_V is the enhanced residue coefficients calculated by vertical RDPCM, and $R(i,j)$ represents residue values obtained after intra prediction. Negative indexed R values are obtained from the residue reference samples as given in Figure 4b. At the decoder, the intra prediction residue values are computed from enhanced residue values, and the related equation is given as follows:

$$R(i,j) = r_V(i,j) + R(i,j-1), \quad 0 \le i < N, 0 \le j < N \tag{5}$$

In Figure 4, the points outside the PU shown in a dark color are the residue reference samples. These samples are obtained from the residue coefficients of the left, upper left, and upper neighbor PUs. In the case of Horizontal RDPCM, the residues of the left neighbor PU, and the residues of the upper neighbor PU in the case of vertical RDPCM are used. If one of these neighbors is unavailable, then that residue value is generated from available residue samples. In the absence of both neighbors, all residue reference samples have been selected as 1.

2.4 Sample-weighted RDPCM

Residue value of the pixel in PU correlated with residue values of the neighboring pixels. Correlation between the left and upper neighboring pixels is reduced with horizontal and vertical RDPCM, respectively. The possible correlation which may occur with other neighboring pixels is not considered. In this study, an RDPCM method that conforms to the selected intra prediction mode for reducing correlation that can occur between a pixel and its neighboring pixels is proposed. The use of the weighted average of residue values of two or three pixels is proposed to find the enhanced residue coefficient of a pixel. Enhanced residue coefficients are calculated using the following equation:

$$r_E(i,j) = R(i,j) - A(i,j), \quad 0 \le i < N, 0 \le j < N \tag{6}$$

Where, $r_E(i,j)$ is the enhanced residue coefficient of (i,j) pixel, $R(i,j)$ is the residue value of the pixel obtained after intra prediction stage, and $A(i,j)$ is the sample-weighted average of neighboring pixels calculated depending on selected intra prediction mode. At the decoder, the residue coefficients are calculated using Equation (7).

$$R(i,j) = r_E(i,j) + A(i,j), \quad 0 \le i < N, 0 \le j < N \tag{7}$$

The efficiency of the enhanced residual coefficients is affected by the selected neighboring pixels and their weights. In the first proposed method, it is suggested that the use of two neighboring pixels as it is in the intra-prediction stage as given in Figure 5. In this case, the weighted average is found via Equation (8) as given in the following form.

$$A(i,j) = ((32 - w).a + w.b) \gg 5 \tag{8}$$

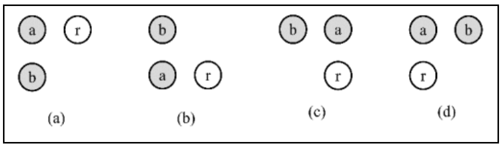

Figure 5. Neighboring pixels used to calculate the weighted average for RDPCM with 2-neighboring pixels. (a): Modes 2-9. (b): Modes 10-18. (c): Modes 19-26, and (d): Modes 27-34.

Where, a and b are the residue values of two neighboring pixels of pixel (i,j) as given in Figure 5, $32 - w$ and w are the weights calculated in depending on selected intra prediction mode. These weights are calculated in the same manner as intra prediction.

In the second proposed method, three neighboring pixels are employed. These three neighboring pixels have been selected considering the method given in [7]. The neighboring pixels are shown in Figure 6 with the prediction mode computed for the PU. In the three-pixel Sample-weighted RDPCM method, the weighted average is calculated by using the following equation:

$$A(i,j) = (\rho_1.a + \rho_2.b + \rho_3.c) \gg 5 \tag{9}$$

where a, b and c are the residue values of neighboring pixels shown in Figure 6. ρ_1, ρ_2 and ρ_3 are the weighting coefficients.

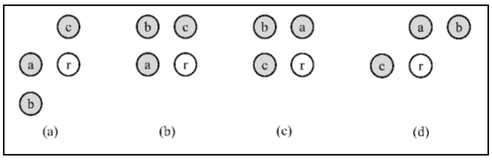

Figure 6. Neighboring pixels used to calculate the weighted average for RDPCM with 3-neighboring pixels. (a): Modes 2-9. (b): Modes 10-18. (c): Modes 19-26, and (d): Modes 27-34 [7].

A simple algorithm is proposed in this study to calculate the values of these weighting coefficients. The flow diagram of the proposed algorithm is shown in Figure 7. In this algorithm, the calculation of the coefficients is based on the distance between a neighboring pixel and scaled prediction point which is positioned between p pixel and q point (Figure 3) on the line that lies in the direction of the prediction mode angle. The scale factor is defined as the ratio of the distance between p pixel and scaled prediction point to the distance between p pixel and q point. In the first step of the algorithm, the distances of the pixels a, b, and c to the scaled prediction point are calculated. The weighting coefficient values of the pixels are inversely proportional to their distances. The raw weighting coefficient of the i-th neighboring pixel is calculated as

$$\rho_i' = round\left(32 \cdot (1/d_i) \Big/ \left(\sum_{j=1}^{3} 1/d_j\right)\right), \quad i = 1,2,3 \tag{10}$$

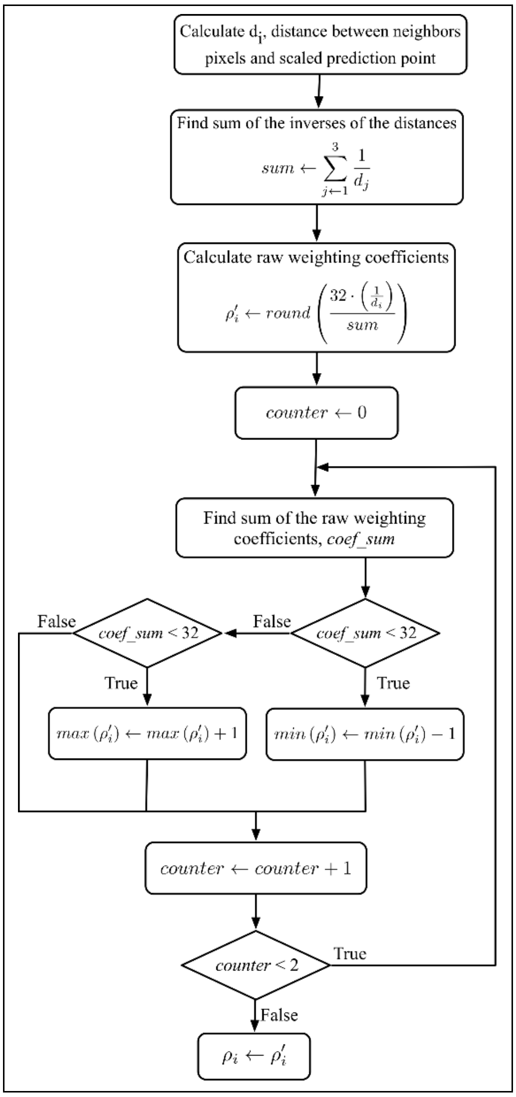

Figure 7. Flow chart of the algorithm for calculating the weighting coefficients.

Where d_i is the distance between i-th neighboring pixel and the scaled prediction point and ρ_i' is the raw weighting coefficient. If the sum of the raw weighting coefficients is less than 32, the coefficient with the largest value is increased by one. When this sum is greater than 32, the coefficient with the smallest value is decreased by one. This procedure is repeated two times to ensure that the sum of the weighting coefficient is 32 (Figure 7). The weighting coefficients obtained by the algorithm are given in Table 1 with the prediction mode.

Table 1. Weighting coefficients (for 0.6 scale factor).

Modes	ρ_1	ρ_2	ρ_3	Modes	ρ_1	ρ_2	ρ_3
2, 34	15	12	5	11, 25	18	7	7
3, 33	13	14	5	12, 24	18	7	7
4, 32	12	15	5	13, 23	17	8	7
5, 31	10	16	6	14, 22	16	9	7
6, 30	9	17	6	15, 21	15	9	8
7, 29	8	18	6	16, 20	13	10	9
8, 28	8	18	6	17, 19	12	11	9
9, 27	7	19	6	18	10	12	10
10, 26	19	7	6				

In both the proposed methods, neighboring pixels can be outside of PU. In this case, reference samples are used for a, b and c residue values. Residue reference samples are generated using the residual values of the left, top left and top PUs. In case of the absence of neighboring PUs, residue reference samples are obtained from available reference samples.

2.5 Residue coefficient coding and binarization

Residual coefficient coding and binarization process are applied to the non-binary residual coefficient to obtain binary bit sequence. A new approach for residue coding and binarization is proposed in this study. This new approach is performed at the CU level. Each CU is divided into subunits called binarization units (BU). These binarization units can be 32, 16, 8, and 4 pixels in size depending on the size of CU. The residue coding and binarization process for a CU is performed according to the package structure given in Figure 8.

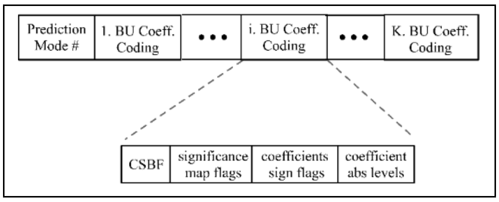

Figure 8. Residue coefficient coding and binarization.

The package contains two sections, the section for the number of the prediction mode of CU and the section for residue coding and binarization packages of the BUs that make up the CU. BUs contain enhanced residue coefficients calculated with RDPCM. BU coefficient coding and binarization are performed as shown in Figure 8.

The package structure of the BU residue coding and binarization contains four sections. One of the sections is the one-bit-sized CSBF (Coefficient sub block flag) section whose value can be either one or zero. If all residue coefficients of a BU are zero, then CSBF takes the value of zero and the BU coding and binarization process are skipped. When a BU has non-zero coefficients, the value of CSBF section is one and for the BU coding and binarization process, the following procedure is performed.

BU residue coding and binarization process involve three steps. In the first step, a significance map is determined. The significance map is obtained by scanning the BU residue data. The significance map scan starts at the top-left coefficient of the BU, and continues in the horizontal scan order until the bottom-right coefficient of the BU is reached. The significance map flag corresponding to a residue coefficient is set to one if the value of the residue coefficient is non-zero. Otherwise, it is set to zero. In the second step, the sign of each nonzero residue coefficient is coded. In the last step, absolute levels of all non-zero residue coefficients of the BU are converted to a binary stream. As a binarization scheme, k-th order Exp-Golomb coding is selected.

3 Experiments and results

To investigate the efficiency of the proposed method, the tests have been performed by using different design parameters via HEVC intra prediction lossless image compression software prepared in MATLAB platform. CU size, block scan order, Exp-Golomb order and BU size were used as design parameters. The variation of compression ratio with respect to these parameters was examined by arranging eight different tests. The first test is the arrangement that RDPCM residue improvement does not perform, and this test is used for control. At the second test arrangement, only horizontal RDPCM is applied to residue values obtained after intra prediction (H-RDPCM). At the third test, only vertical RDPCM is applied (V-RDPCM). At the fourth test, firstly horizontal RDPCM, and after that vertical RDPCM are applied to residues (HV-RDPCM). At the fifth test, firstly vertical RDPCM, and after that horizontal RDPCM are applied to residues (VH-RDPCM). At the sixth test, horizontal RDPCM is applied for blocks, which planar, DC and horizontal angular modes selected at intra prediction, and vertical RDPCM is applied for blocks, which vertical angular modes selected (H/V-RDPCM). At the seventh test, 2 sample-weighted RDPCM (2 SW-RDPCM) and at the eighth test, the proposed 3-sample-weighted RDPCM (3 SW-RDPCM) are applied to residues.

512x512 pixel size monochrome airplane, barb, boat, cameraman, goldhill, house, lena_gray, man and peppers_gray test images used for the tests [20]. The variation of the mean value of compression ratio results with respect to CU size obtained from these images are given in Table 2 by choosing as CU scan order Z-scan, BU size 4 and Exp-Golomb order k = 0 and k = 1.

Table 2. Compression rate of the RDPCM method with Exp-Golomb order (k).

RDPCM Method	k=0			k=1		
	CU Size			CU Size		
	8x8	16x16	32x32	8x8	16x16	32x32
Not Applied	1.4329	1.2721	1.1200	1.4790	1.3294	1.1904
H-RDPCM	1.4859	1.4529	1.4177	1.5213	1.4938	1.4732
V-RDPCM	1.5147	1.4820	1.4500	1.5496	1.5231	1.5075
HV-RDPCM	1.5343	1.5676	1.5756	1.5577	1.5867	1.6052
VH-RDPCM	1.5373	1.5685	1.5753	1.5600	1.5873	1.6050
H/V-RDPCM	1.5423	1.5192	1.4883	1.5726	1.5547	1.5395
2 SW-RDPCM	1.5412	1.5269	1.4970	1.5727	1.5625	1.5475
3 SW-RDPCM	1.5711	1.5449	1.5077	1.5972	1.5768	1.5527

When the results given in Table 2 are compared, it is observed that the compression ratio is better for k = 1. At HV-RDPCM and VH-RDPCM methods, values are higher because of RDPCM applied to residues two times. Proposed 3 SW-RDPCM yields the best compression ratio values among methods applied only once. With 2 SW-RDPCM method, we get compression performance improvement between 6% and 34% compared to the method without RDPCM depending on the value of the Exp-Golomb order and the size of CU. With 3 SW-RDPCM method, the performance improvement varies between 8% and 35%.

We also investigate the complexity of the proposed methods based on the encoding and decoding times. Since the execution time of any program depends on factors such as the hardware on which the program is run, the programming language, and the compiler, we focus on the percentage increase in the encoding and decoding times. To calculate the percentage increase, we take the execution time of the method without RDPCM as a reference execution time. Because we write our encoder's and decoder's program in Matlab, we use Matlab's Profiler to get the execution time of the encoder and decoder. The encoding and decoding times are normalized with respect to the reference execution time. The results are given in Table 3 and compared with the encoding and decoding times of the method proposed in [7]. We can notice from the table that the encoding time of the 2 SW-RDPCM method is increased by 2.8% with respect to the method without RDPCM. But, for the 2 SW-RDPCM method, the increase in the compression rate varies between 6% and 34%. The encoding time of the 3 SW-RDPCM method is increased by 3.1%, but the increase in the compression rate varies between 8% and 35%. As a result, we can say that more compression performance is achieved despite a smaller increase in the encoding time. The increase in decoding time is higher than encoding time, but it is still below the increase in the compression performance. Also, we can see in Table 3 that the increase in encoding time with our proposed 3 SW-RDPCM method is less than that of the method proposed in [7]. This is because we use a simple procedure for the calculation of weighting coefficients. But the increase in decoding time with our proposed methods is greater than that of the method proposed in [7].

Table 3. Percentage increase in the average encoding and decoding times.

Compression Method	Encoder Time	Decoder Time
H-RDPCM	1.6%	1.6%
V-RDPCM	1.9%	1.8%
HV-RDPCM	2.4%	4.8%
VH-RDPCM	2.6%	5.0%
H/V-RDPCM	1.5%	2.2%
2 SW-RDPCM	2.8%	6.7%
3 SW-RDPCM	3.1%	7.5%
3 Tab Filtering[7]	9.7%	-12.7%

To compare the performance of proposed sample-weighted RDPCM methods in this study, compression performance values given in Table 9 in [7], and also mean of compression performance values obtained from all test images for CU size=8, BU size=8 and k=1 are given in Table 4. When the given values are examined, it is seen that the mean compression performance values, which are obtained by the proposed 2 SW-RDPCM and 3 SW-RDPCM methods, are comparable with the other methods given in the literature. Table 4 shows the compression performances of different methods with our proposed methods. We can see in Table 4 that LOCO-I, CALIC, and MRP have lesser bit-per-pixel values. In this study, we propose methods to be used in conjunction with HEVC intra prediction. So, our proposed methods have a compression performance more than that of HEVC and have a compression performance comparable with that of SAP-based methods. Since HEVC produces more efficient compression performance results in compression of high-resolution images than its predecessor [8], [9], our proposed methods also produce better results in high-resolution images. To test this situation, we apply our 3 SW-RDPCM compression method to another set of images. The size of images in the set is 1024x1024 pixels and the entropy of images in the set is about 7. We obtain 4.45 bit/pixel compression performance value with this set of images.

Also, some tests have been performed to observe the effect of CU scan order over compression ratio. The mean of compression rate values obtained from test images is given in Table 5 when CU size and BU size are 8 and 4 respectively, and k=1. It can be seen that the best compression ratio is obtained by horizontal scan.

Selection of BU size is also effective on compression performance when coding of residue coefficients. While higher the number of zero coefficients better the performance at small.

Table 4. Comparison of compression performance of several methods with proposed sample-weighted RDPCM.

Compression Method	Compression Performance (bit/pixel)
HEVC [7]	5.23
SAP [3]	4.92
3-tap Filtering [7]	4.84
LOCO-I [21]	4.75
CALIC [22]	4.59
MRP [23]	4.32
2 SW-RDPCM	5.04
3 SW-RDPCM	4.96

Table 5. Compression rate with CU scanning method for k=1 and CU Size=8.

CU Partition & Scan Type	RDPCM Method		
	Not Applied	2 SW-RDPCM	3 SW-RDPCM
Z - Scan	1.4790	1.5727	1.5972
H - Scan	1.4800	1.5733	1.5979
V - Scan	1.4786	1.5715	1.5963

BU sizes in lossy compression, some tests were also carried out by considering the fact that the effect of variation of BU size in lossless compression could be different. CU scan order is chosen as Z-scan and k = 1 in these tests. BU size is chosen as both 4 and 8 when CU size is 8 for all test images and compression ratios are obtained. The average values of these test are given in Table 6. BU size alternated as 4, 8 and 16 successively, for CU size 16 and as 4, 8 16 and 32 successively, for CU size 32. Obtained compression ratio values are given in Table 6. It can be concluded that when CU size is 8 the best BU size is 8, and when CU size is 16 or 32 the best BU size is 16.

Table 6. Compression rate with BU size for k=1.

CU Size	BU Size	Not Applied	2 SW-RDPCM	3 SW-RDPCM
8	4 x 4	1.4790	1.5727	1.5972
	8 x 8	1.4916	1.5871	1.6121
16	4 x 4	1.3294	1.5625	1.5768
	8 x 8	1.3396	1.5767	1.5913
	16 x 16	1.3412	1.5785	1.5932
32	4 x 4	1.1818	1.5354	1.5527
	8 x 8	1.1899	1.5489	1.5577
	16 x 16	1.1914	1.5506	1.5594
	32 x 32	1.1904	1.5475	1.5564

Finally, the effects of entropy of images on compression performances are examined. The results are given in Table 7. The entropies of the images are obtained using Matlab's built-in function entropy(). The compression rates of the images are obtained with the case CU size = 8, BU size = 8, k = 1, and Z-scan. When the results are examined, although there seems to be no linear relationship between entropy and compression rate, we may say that the compression rate is higher for low entropy values. The highest compression rate was obtained for the "house" image that has the lowest entropy.

Table 7. Entropies and compression performanses of the test images.

Test Image	Entropy	Compression Rate	
		2 SW-RDPCM	3 SW-RDPCM
airplane	6.71	1.56	1.60
barb	7.47	1.35	1.35
boat	7.12	1.44	1.46
cameraman	7.05	1.80	1.85
goldhill	7.48	1.32	1.34
house	5.75	2.50	2.59
lena_gray	7.45	1.49	1.52
man	7.24	1.38	1.40
peppers_gray	6.76	1.43	1.41

When the compression rates given in the tables are examined, it is seen that both the proposed sample-weighted RDPCM methods provide better compression rates in accordance with the method which uses only HEVC intra prediction. This improvement ensures up to 35 % increase in compression rate with the expense of a 3.1% and 7.5% increase in encoding and decoding times, respectively.

4 Conclusion

In this study, the use of RDPCM has been investigated to improve the performance of lossless image compression with HEVC Intra prediction. For this purpose, the sample-weighted RDPCM method is proposed, which uses the weighted average of two and three neighboring pixel residues. In these proposed methods, the enhanced residue value of a pixel has been obtained by subtracting the weighted average of neighboring pixels' residues calculated by taking into account the selected mode in intra prediction stage from the residue coefficient of that pixel. In order to find out the effectiveness of the proposed methods, the tests have been performed by using different design parameters such as CU size, BU size and Exp-Golomb order. Other RDPCM methods such as vertical and horizontal RDPCM have been included in the tests. As a result of these tests, the compression rates of the images have been found. When the results are examined, it is seen that the proposed enhanced RDPCM methods provide better compression performance, according to the situation where RDPCM method is not utilized and the other RDPCM methods.

5 Author contribution statements

Ali AKMAN contributed to the study by developing the compression algorithm, implementing the algorithm, testing the compression software, obtaining and interpreting the results, and writing the paper. Serap CEKLİ contributed to the study by developing the compression algorithm, searching for literature, and writing the paper.

6 Ethics committee approval and conflict of interest statement

There is no need to obtain permission from the ethics committee for the article prepared.

There is no conflict of interest with any person/institution in the article prepared.

7 References

[1] Kim K, Jeon G, Jeong J. "Improvement of implicit residual DPCM for HEVC". *Tenth International Conference on Signal-Image Technology and Internet-Based Systems*, Marrakech, Morocco, 23-27 November 2014.

[2] Sanchez V, Aulf-Llinas F, Serra-Sagrista J. "DPCM-based edge prediction for lossless screen content coding in HEVC". *IEEE Journal on Emerging and Selected Topics in Circuits and Systems*, 6(4), 497-507, 2016.

[3] Zhou M, Gao W, Jiang M, Yu H. "HEVC lossless coding and improvements". *IEEE Transactions on Circuits and Systems for Video Technology*, 22(12), 1839-1843, 2012.

[4] Senturk A, Kara R, Ozcelik I. "Fuzzy logic and image compression based energy efficient application layer algorithm for wireless multimedia sensor networks". *Computer Science and Information Systems*, 17(2), 509-536, 2020.

[5] Göçeri E, Yaldır AK. "JPEG2000 standardının yeni özelliklerini destekleyen bir görüntü işleme uygulaması geliştirilmesi". *Pamukkale Üniversitesi Mühendislik Bilimleri Dergisi*, 13(3), 337-346, 2007.

[6] Sayıt M, Tunalı ET. "H.264 SP tipi çerçeveler ile uyarlamalı video akışlandırma". *Pamukkale Üniversitesi Mühendislik Bilimleri Dergisi*, 18(2), 73-83, 2012.

[7] Alvar SR, Kamisli F. "On lossless intra coding in HEVC with 3-tap filters". *Signal Processing: Image Communication*, 47(9), 252-262, 2016.

[8] Sullivan GJ, Ohm JR, Han WJ, Wiegand T. "Overview of the high efficiency video coding (HEVC) standard". *IEEE Transactions on Circuits and Systems for Video Technology*, 22(12), 1649-1668, 2012.

[9] Bossen F, Bross B, Suhring K, and Flynn D, "HEVC Complexity and Implementation Analysis". *IEEE Transactions on Circuits and Systems for Video Technology*, 22(12), 1685-1696, 2012.

[10] Abhilash AA, Sreelekha G. "HEVC-based lossless intra coding for efficient still image compression". *Multimedia Tools and Applications*, 76, 1639-1658, 2017.

[11] Kamisli F, "Lossless intra coding in HEVC with integer to integer DST". *24th European Signal Processing Conference*, Budapest, Hungary, 28 August-2 September 2016.

[12] Chen F, Zhang J, and Li H, "Hybrid transform for HEVC-based lossless coding," *IEEE International Symposium on Circuits and Systems (ISCAS)*, Melbourne, Australia, 1-5 June 2014.

[13] Sanchez V, Llinàs FA, Sagristà JS. "Piecewise mapping in HEVC lossless intra-prediction coding". *IEEE Transactions on Image Processing*, 25(9), 4004-4017, 2016.

[14] Jeon G, Kim K, Jeong J. "Improved residual DPCM for HEVC lossless coding". *27th SIBGRAPI Conference on Graphics, Patterns and Images*, Rio de Janeiro, Brazil, 26-30 August 2014.

[15] Tan YH, Yeo C, Li Z. "Residual DPCM for lossless coding in HEVC". *IEEE International Conference on Acoustics, Speech and Signal Processing*, Vancouver, BC, Canada, 26-31 May 2013.

[16] Lee YL, Han KH, Sullivan GJ. "Improved lossless intra coding for H.264/MPEG-4 AVC". *IEEE Transactions on Image Processing*, 15(9), 2610- 2615, 2006.

[17] Hong SW, Kwak JH, Lee YL. "Cross residual transform for lossless intra-coding for HEVC". *Signal Processing: Image Communication*, 28(10), 1335-1341, 2013.

[18] Cai X, Lim JS. "Adaptive residual DPCM for lossless intra coding". *Proceedings Visual Information Processing and Communication VI*, 9410(94100A), 1-8, 2015.

[19] Kwak JH, Lee YL. "Secondary residual transform for lossless intra coding in HEVC". *Journal of Broadcast Engineering*, 17(5), 734-741, 2012.

[20] Image Database. "Standard Test Images". https://www.imageprocessingplace.com/root_files_V3/image_databases.htm (28.12.2021).

[21] Weinberg MJ, Seroussi G, Sapiro G. "The loco-i lossless image compression algorithms: principles and standardization into jpeg-ls". *IEEE Transactions on Image Processing*, 9(8), 1309-1324, 2000.

[22] Wu X, Memon N. "Context-based, adaptive, lossless image coding". *IEEE Transactions on Communication*, 45(4), 437-444, 1997.

[23] Matsuda I, Mori H, Itoh S. "Lossless coding of still images using minimum-rate predictors". *International Conference on Image Processing*, Vancouver, BC, Canada, 10-13 September 2000.