A NOVEL ZIGZAG SCANNING CONCEPT FOR H.264/AVC

Myung Han Hyun, Jae Taeg Yu, and Sang Bum Lee
Agency for Defense Development, Daejeon, Korea
drwise@add.re.kr

ABSTRACT

In this paper, a novel zigzag scanning concept of quantized coefficients for H.264/AVC is introduced. In order to scan the quantized coefficients efficiently, the statistical occurrence values of the quantized coefficients after the final mode decision are utilized. We develop a zigzag scanning pattern by reordering the statistical occurrence values in descending order. In addition, we consider the temporal and spatial correlation among the frames to classify the zigzag scanning pattern. In particular, we focus on the macroblock level zigzag scanning so that the proposed method will have the different zigzag scanning pattern based on the macroblock. Experimental results show that the proposed scheme reduces the total bits up to 4.05% and 3.67% while introducing either negligible loss of video quality for intra- and inter mode, respectively.

Keywords : zigzag scanning, quantized coefficients, discrete cosine transform, H.264/AVC.

1. INTRODUCTION

H.264/AVC is the state of the art video coding standard that has been standardized between the ITU-T Video Coding Expert Group (VCEG) and the ISO/IEC Moving Picture Experts Group (MPEG). Currently, due to its high compression performance H.264/AVC has become a promising video compression standard for a wide range of applications, including multimedia streaming and video conferencing [1]. H.264/AVC improves coding performance over previous video coding standards, such as MPEG-2, H.263, and MPEG-4 part 2, by applying more sophisticated coding techniques, such as intra prediction, variable block size motion estimation, rate-distortion optimized mode decision, and entropy coding [2].

During the encoding, transformed coefficients after discrete cosine transform (DCT) are usually quantized and become quantized coefficients. In order to scan the quantized coefficients, a zigzag scanning will be used to employ context-based adaptive variable length coding (CAVLC). However, the conventional zigzag scanning has a fixed way to scan the quantized coefficients so it is not efficient to scan nonzero quantized coefficients because each and every 4×4 block in a macroblock (MB) has a different pattern of the quantized coefficients. To fulfill the run-length coding efficiently, the nonzero quantized coefficients should be skewed in the front part and the zero quantized coefficients should be located in the rear part of a 1D array. Recently, various studies have been carried out to reduce bit rates by changing a zigzag scanning order [3]-[7]. However, the previous studies were applied only to intra mode because the spatial prediction direction was used to determine the zigzag scanning order. In this paper, by adopting the gist of zigzag scanning scheme in [8], we have tried to prove the robustness of the algorithm through not only the various sequences but also the various modes; intra- and inter mode. Furthermore, we have explained the algorithm in detail.

The rest of this paper is organized as follows. In the next Section, we will briefly review the zigzag scanning for CAVLC in H.264/AVC. In Section 3, we will introduce the proposed algorithm. In Section 4, coding performance and analysis of the proposed algorithm will be shown and the paper will be completed with our conclusions presented in Section 5.

2. OVERVIEW OF ZIGZAG SCANNING FOR CAVLC IN H.264/AVC

2.1. CAVLC

CAVLC was originally designed to take advantage of several characteristics of residual data in lossy coding: (1) after transform and quantization, 4×4 blocks typically contain many zeros, especially in high frequency regions; (2) the level of the highest nonzero coefficients tends to be as small as one; and (3) the level of nonzero coefficients tends to be larger toward the low frequency regions [9]. By taking into consideration the above characteristics, CAVLC employs several syntax elements to encode residual data efficiently. Fig. 1 shows a simplified overview of the CAVLC encoding process [10]. In order to encode the syntax elements, the four VLC tables are used. The choice of VLC table depends on the local statistics of nonzero coefficients in the previously coded upper and left sub-blocks. The detailed coding procedure of CAVLC is as follows.

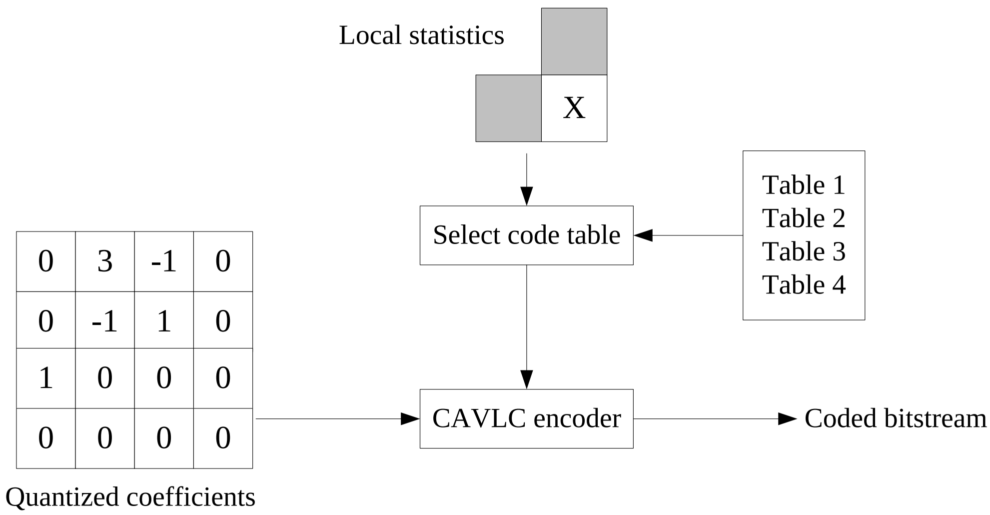


Fig 1. CAVLC encoder overview

Step1 : Both the total number of nonzero coefficients and the number of trailing ones are encoded using a combined codeword (*coeff_token*).

Step 2 : The sign of each trailing one is encoded using a one bit codeword in reverse order (*trailing_ones_sign_flag*).

Step 3 : The absolute value of the level of each remaining nonzero coefficient is encoded in reverse order using one of the predefined VLC tables (*level*).

Step 4 : The number of all zeros before the last nonzero coefficient is encoded (*total_zeros*).

Step 5 : The number of consecutive zeros preceding each nonzero coefficient is encoded in reverse order (*run_before*).

2.2. PREVIOUS ZIGZAG SCANNING METHODS

Because of the characteristic of the DCT, the nonzero elements are squeezed in the top-left area of the block when the frame scan mode; depicted in Fig. 2(a) is selected. However, we may have the nonzero elements placed in the left side area of the block when the field scan mode; depicted in Fig. 2(b) is chosen. By using a conventional zigzag scanning order, the quantized coefficients are serialized, and then the nonzero coefficients (level), the consecutive zeros (run), and other information are coded. We use CAVLC as an entropy coding scheme, thus it is more appropriate that the nonzero coefficients shall be located in the front part of the serialized bit stream to save the bits by reducing the codeword for syntax elements such as *total_zeros* and *run_before*. Even though the conventional zigzag scanning is simple and easy to implement at both encoder and decoder parts because its scanning pattern is fixed; however, it is a little bit illogical for various patterns of quantized coefficients to use the fixed zigzag scanning pattern for variable length coding.

0	1	5	6
2	4	7	12
3	8	11	13
9	10	14	15

(a)

0	2	8	12
1	5	9	13
3	6	10	14
4	7	11	15

(b)

Fig 2. Conventional zigzag scanning orders; (a) frame scan mode, (b) field scan mode

In order to solve the shortcoming of the previous methods, there are various approaches to develop the adaptive zigzag scanning patterns based on the intra prediction modes illustrated in Fig. 3. Lee, *et al* utilized vertical and horizontal prediction direction to make different scanning orders [5]. They classified the zigzag scanning order into three different types. Choi, *et al* extended Lee's algorithm such that energy distribution of the quantized coefficients in frequency domain was exploited to set up six different scanning orders [6]. In their suggestion [7], Wei, *et al* used the average powers of the quantized coefficients after different intra prediction mode to figure out nine different scanning orders. However, the previous studies were applied only to intra mode because the spatial prediction direction or the average powers of the quantized coefficients values are needed to determine the zigzag scanning order.

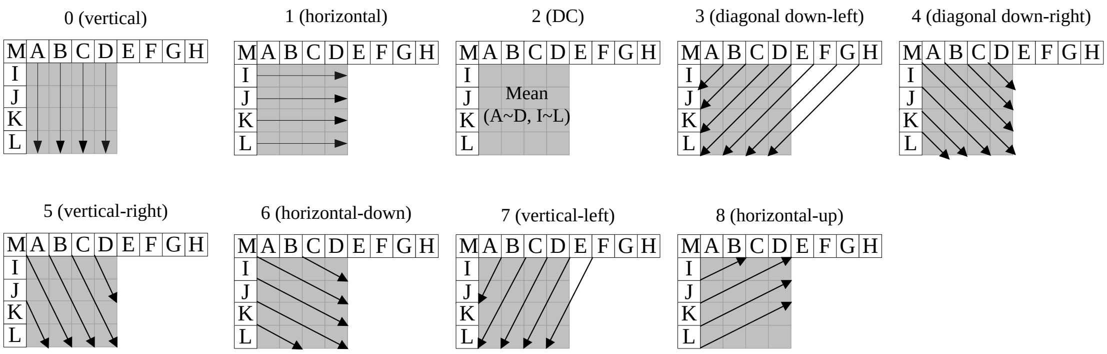


Fig 3. Intra prediction modes

3. PROPOSED METHOD

3.1. BASIC CONCEPT OF THE ALGORITHM

In the proposed method, we focus on how to estimate the location of the nonzero quantized coefficients of the 4×4 block. If we properly predict the location of the nonzero quantized coefficients, we are able to scan the quantized coefficients efficiently for CAVLC. In other words, we pick the nonzero quantized coefficients first, then the zero quantized coefficients such that we can save codeword bits for *total_zeros* and *run_before*. Table 1 shows the example of the performance between the conventional and an ideal method conceptually using the quantized coefficients of 4×4 block in Fig 1. The resulting encoded bit streams of the conventional method and the ideal method are ‘0000100/011/1/0010/111/10/1/1/01’ and ‘0000100/011/1/0010/0101’ [11]. The ideal method scans only the nonzero quantized coefficients so that it does not need the codeword bits for *total_zeros* and *run_before* because we do not have any zero values among the nonzero quantized coefficients.

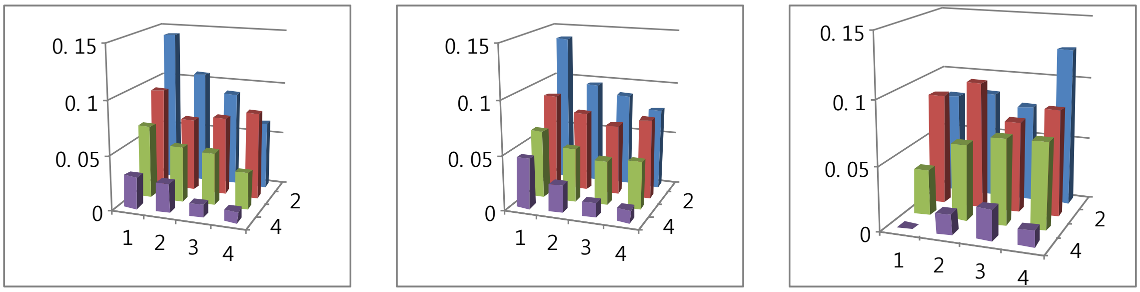
Conventional method			Ideal method		
Reordered coefficients: 0,3,0,1,-1,-1,0,1,0…			Reordered coefficients: 3,1,-1,-1,1,0,0,0…		
Coded SE	SE Value	Code	Coded SE	SE Value	Code
coeff_token	Total coeff:5 Trailing 1s:3	0000100	coeff_token	Total coeff:5 Trailing 1s:3	0000100
Sign_T1s	+,-,-	011	Sign_T1s	+,-,-	011
level	+1	1	level	+1	1
level	+3	0010	level	+3	0010
total_zeros	3	111	total_zeros	0	0101
run_before	1	10			
run_before	0	1			
run_before	0	1			
run_before	1	01			
run_before	1	code not required	*SE=Syntax Element		

Table 1. CAVLC encoding comparison of conventional and ideal method

On the other hand, the ideal method cannot be decoded because it does not have any quantized coefficients scanning information at the decoder. Through the ideal method, we can save bits up to 20% rather than the conventional method by saving bits for *total_zeros* and *run_before*. We can utilize the ideal method as a bit saving measurement by comparing the bitrates. In the following subsection, we are suggesting the method to predict the location of the next frame's nonzero quantized coefficients based on the statistics.

3.2. ANALYSIS OF THE STATISTICAL CHARACTERISTICS OF THE QUANTIZED COEFFICENTS

Because of the temporal redundancy in video sequences, the previous frame's *nth* MB seems to highly correlate to the current frame's *nth* MB with respect to the texture characteristics. It shall be assumed that the quantized coefficients are very similar among the corresponding MBs in the sequential video frame. As can be seen Fig. 4(a) and Fig. 4(b), it is noteworthy that both patterns' probability distributions of the quantized coefficients have some similarities. Fig. 4(a) and Fig. 4(b) represent the probability distributions of 10^{th} and 20^{th} frames in 207^{th} MB ('Paris' sequence, QP=24), respectively. On the contrary, it is noticed from Fig. 4(a) and Fig. 4(c) that even if MBs are adjacent, the patterns' probability distributions of the quantized coefficients are different. Fig. 4(c) represents the probability distributions of 10^{th} frame in 208^{th} MB ('Paris' sequence, QP=24). To this end, we consider the temporal-correlation characteristics among the corresponding MBs of the video sequences in particular, the proposed algorithm borrows the idea in [3] to develop a representative scanning pattern based on the statistics of the quantized coefficients of all 16 4×4 blocks in each MB.



(a) (b) (c)

Fig 4. Probability distributions of quantized coefficients for 'Paris' with QP=24; (a) 10^{th} frame, 207^{th} MB, (b) 20^{th} frame, 207^{th} MB, (c) 10^{th} frame, 208^{th} MB

3.3. SCANNING PATTERN GENERATION

In order to scan the quantized coefficients using the representative zigzag scanning pattern for *nth* MB, 4×4 matrixes are needed to be defined as follows:

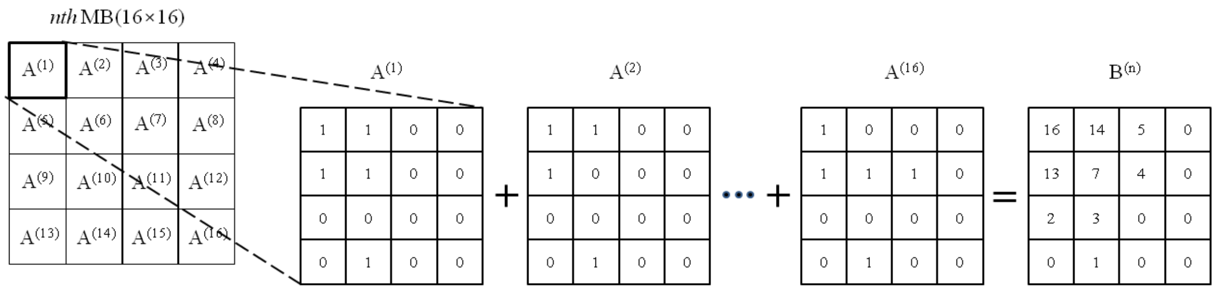
$$A^{(l)} = \left[a_{ij}\right]^{(l)}, \qquad B^{(n)} = \left[b_{ij}\right]^{(n)}, \qquad 1 \le i, j \le 4 \tag{1}$$

where l is a 4×4 block number(1~16), and n is a MB number. Matrix A represents 16 4×4 blocks in the MB. We increase the count value of the 4×4 block's *(i, j)th* element if the *(i, j)th* element is not zero. Matrix B represents the representative 4×4 block of *nth* MB as storage for the accumulated quantized coefficients. If the final mode is determined, the final mode's count value for the nonzero quantized coefficients of 16 4×4 matrix *As* are summed to matrix B in accordance with the corresponding position of element in the 4×4 block and the procedure is defined as follows:

$$B^{(n)} = \left[b_{ij}\right]^{(n)} = \otimes\sum_{l=0}^{16} A^{(l)} = \otimes\sum_{l=0}^{16} \left[a_{ij}\right]^{(l)}, \qquad 1 \le i, j \le 4 \tag{2}$$

where l and n are already defined in equation 1, and $\otimes\sum$ is defined as an operator that adds elements according to the corresponding position of the matrix. After the summation in equation 2, we sort the elements of matrix B in descending order and then make a column vector defined in equation 3 that conserves the zigzag scanning order of matrix B in accordance with their level size. In other word, each element in vector C indicates the position of element itself in matrix B. In Fig. 5, we present an example of the proposed scanning order generation.

$$C^{(n)} = \left[c_1 \quad c_2 \quad c_3 \quad \ldots \quad c_{16}\right]^T \tag{3}$$



$C^{(n)}=[c_1\ c_2\ c_3\ \ldots\ c_{16}]^T = [b_{11}\ b_{12}\ b_{21}\ b_{22}\ b_{13}\ b_{23}\ b_{32}\ b_{31}\ b_{42}\ \ldots\ b_{44}]^T = [16\ 14\ 13\ 7\ 5\ 4\ 3\ 2\ 1\ 0\ 0\ 0\ 0\ 0\ 0\ 0]^T$

Fig 5. Example of the proposed scanning order generation

To decode the encoded bit stream, the first P frame should be encoded by using the conventional zigzag scanning method. As we decode the first P frame with the conventional zigzag scanning method, the quantized coefficients are accumulated to $B^{(n)}$. By reordering the elements of $B^{(n)}$ in descending order, the representative zigzag scanning pattern ($C^{(n)}$) in each MB for the second P frame will be initialized. During the decoding process, we continuously update $B^{(n)}$ by accumulating the quantized coefficients of the final mode to make the representative zigzag scanning pattern ($C^{(n)}$) for the corresponding MB of the next frame. Since only the previously coded frames' quantized coefficients are utilized for the proposed algorithm, the encoder does not need to transmit additional information to the decoder to indicate the zigzag scanning order. That is, the decoder can generate the zigzag scanning order by using the already decoded frames' quantized coefficients. The aforementioned procedure can be summarized as follows:

Step 1 : Load the representative scanning pattern ($C^{(n)}$) of the previously coded frame's corresponding *nth* MB.

Step 2 : Perform inter/intra prediction, DCT, and quantization. Using the representative scanning pattern ($C^{(n)}$), fulfill the entropy coding, and then RD cost should be calculated to find out the final mode.

Step 3 : The quantized coefficients of the final mode are accumulated into the representative 4×4 block ($B^{(n)}$).

Step 4 : Update the representative scanning pattern ($C^{(n)}$) for the *nth* MB of the next frame by sorting the representative 4×4 block ($B^{(n)}$) in descending order.

4. EXPERIMENTAL RESULTS

The proposed algorithm is implemented on Joint Model 11.0(JM 11.0) [12]. We have used a variety of QCIF and CIF video sequences, adopted as test sequences in the MPEG standard. The simulation conditions are as follows:

(a) Baseline profile is used and RD optimization is enabled.
(b) MV search range is 32 and the number of reference frame is 5.
(c) The number of frames in a sequence is 100 and CAVLC is used.
(d) GOP structures are IPPP (Only the first frame is I) for inter- and III for intra mode.
(e) Quantization parameters (QPs) are (12, 16, 20, 24) and (28, 32, 36, 40) for inter- and intra mode, respectively.

Comparisons were made in terms of distortion differences and percentage total bits differences with respect to the previous method (JM 11.0 and the method proposed in [3]). The changes are calculated as follows:

$$\Delta PSNR(dB) = PSNR_{<proposed \ or \ method[3]>} - PSNR_{JM11.0} \quad (4)$$

$$\Delta Totalbits(\%) = \frac{Totalbits_{<proposed \ or \ method[3]>} - Totalbits_{JM11.0}}{Totalbits_{JM11.0}} \times 100(\%) \quad (5)$$

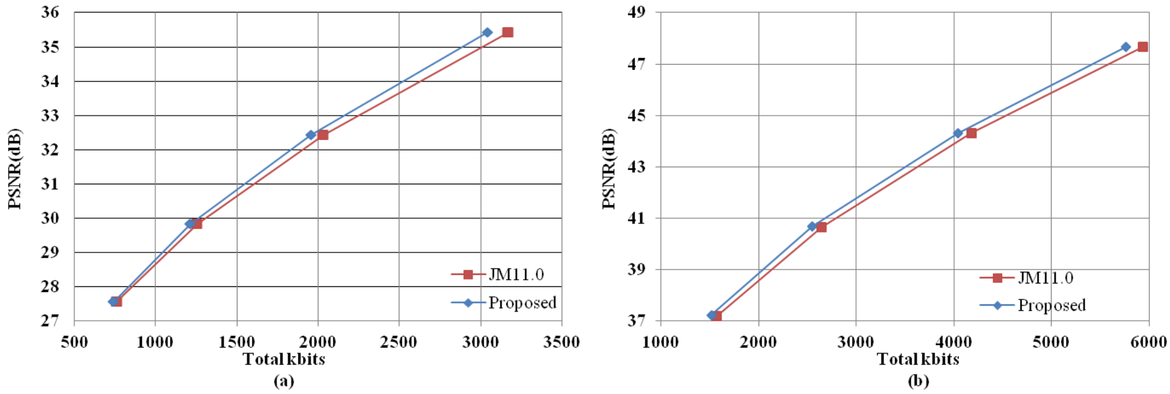


Fig 7. RD curves; (a) container (qcif) intra mode, (b) coastguard (qcif) intra mode

Experimental results are listed in Table 2 and Table 3. Table 2 shows the intra-coding results of the proposed algorithm and the method proposed in [3]. Table 3 compares with the inter-coding results between the proposed algorithm and JM 11.0. The proposed algorithm achieves total bits saving up to 4.05% and 3.67% while introducing either negligible loss of video quality for intra- and inter-coding, respectively. The RD curves for both the JM11.0 and the proposed method are illustrated in Fig. 7. Experimental results show that the proposed method gives consistent performance gain over the JM11.0.

In order to speculate the bit saving trait according to the various QPs, we considered the ideal method aforementioned in Section 3.1. Comparing with the ideal method, the proposed method has estimated the location of the nonzero quantized coefficients of the 4×4 block suitably for both intra- and inter mode when we observed the variation of total bits. Moreover, we have noticed that the bit saving tendency according to the various sequences was also similar between the proposed method and ideal method. Thus it is certain that the proposed method is appropriate method for enhancing the compression performance. In general, a lot of residual data are generated by using smaller QPs. It means we have more zero values among the nonzero coefficients. Therefore, we can save the encoding bits by using the ideal method. From the experimental results of Table 4, however, we noticed that we have saved more bits even though we used bigger QPs. In this case, we regarded that the bits for *total_zeros* and *run_before* have played a key role for bit saving. In other words, the bits portion for levels (nonzero coefficients) is getting smaller rather than that of *total_zeros* and *run_before* when QPs are getting bigger.

Test sequence	QP	JM11.0		Method in [3]		Proposed method		ΔPSNR (dB)	ΔTotal bits (%)
		PSNR (dB)	Total bits (Kbits)	PSNR (dB)	Total bits (Kbits)	PSNR (dB)	Total bits (Kbits)		
Carphone (QCIF)	28	38.34	2006.17	38.33	1952.18	38.33	1954.86	-0.01	-2.56
	32	35.44	1391.46	35.46	1360.01	35.46	1358.46	0.02	-2.37
	36	32.69	959.57	32.68	939.26	32.68	937.81	-0.01	-2.27
	40	29.92	660.68	29.92	653.79	29.92	650.11	0.00	-1.60
Container (QCIF)	28	37.26	2530.98	37.27	2463.00	37.27	2448.77	0.01	-3.25
	32	34.39	1746.49	34.38	1697.50	34.38	1691.38	-0.01	-3.16
	36	31.59	1166.55	31.57	1133.02	31.60	1125.64	0.01	-3.51
	40	28.93	792.53	28.92	773.05	28.91	765.25	-0.02	-3.44
Coastguard (QCIF)	28	35.42	3171.28	35.43	3035.42	35.43	3042.86	0.01	-4.05
	32	32.42	2033.54	32.43	1959.92	32.43	1955.71	0.01	-3.83
	36	29.83	1258.10	29.85	1227.59	29.83	1216.21	0.00	-3.33
	40	27.56	760.15	27.57	745.83	27.55	742.17	-0.01	-2.37
Paris (CIF)	28	36.73	13908.14	36.71	13484.56	36.72	13462.06	-0.01	-3.21
	32	33.53	9836.06	33.50	9526.11	33.51	9491.82	-0.02	-3.50
	36	30.53	6718.78	30.51	6512.08	30.52	6466.59	-0.01	-3.75
	40	27.72	4551.02	27.71	4424.98	27.70	4375.39	-0.02	-3.86
Akiyo (CIF)	28	40.41	4512.51	40.41	4410.05	40.40	4400.81	-0.01	-2.48
	32	37.81	3138.18	37.81	3080.07	37.81	3084.38	0.00	-1.71
	36	35.31	2181.92	35.31	2151.70	35.20	2137.26	-0.11	-2.05
	40	32.76	1549.74	32.77	1536.44	32.74	1529.21	-0.02	-1.32

Table 2. Experimental results comparison of intracoding performance

Test sequence	QP	JM11.0		Proposed method		ΔPSNR (dB)	ΔTotal bits(%)
		PSNR (dB)	Total bits (Kbits)	PSNR (dB)	Total bits (Kbits)		
Carphone (QCIF)	12	48.51	2998.65	48.50	2970.08	-0.01	-0.95
	16	45.84	1737.62	45.84	1721.78	0.00	-0.91
	20	43.00	980.01	43.03	974.91	0.03	-0.52
	24	40.10	556.60	40.16	557.65	0.06	0.19
Container (QCIF)	12	48.10	2023.52	48.13	1956.14	0.03	-3.33
	16	44.85	1112.92	44.90	1078.76	0.05	-3.07
	20	41.65	562.26	41.70	547.02	0.05	-2.71
	24	38.70	259.71	38.77	260.46	0.07	0.29
Coastguard (QCIF)	12	47.67	5936.76	47.67	5762.72	0.00	-2.93
	16	44.29	4176.55	44.29	4038.22	0.00	-3.31
	20	40.64	2641.48	40.67	2544.56	0.03	-3.67
	24	37.18	1567.42	37.23	1510.26	0.05	-3.65
Paris (CIF)	12	47.81	14255.18	47.83	14186.92	0.02	-0.49
	16	44.65	8317.92	44.67	8271.54	0.02	-0.56
	20	41.50	4987.78	41.52	4949.97	0.02	-0.76
	24	38.41	3129.72	38.44	3102.84	0.03	-0.87
Akiyo (CIF)	12	49.62	3987.10	49.63	3954.66	0.01	-0.81
	16	46.85	2111.14	46.86	2086.58	0.01	-1.16
	20	44.45	1079.64	44.47	1072.91	0.02	-0.62
	24	42.11	582.97	42.13	580.79	0.02	-0.37

Table 3. Experimental results comparison of intercoding performance

Test sequence	Intra mode					Inter mode				
	QP	PSNR (dB)	Total bits (Kbits)	ΔPSNR (dB)	ΔTotal bits(%)	QP	PSNR (dB)	Total bits (Kbits)	ΔPSNR (dB)	ΔTotal bits(%)
Carphone (QCIF)	28	38.38	1758.98	0.04	-12.32	12	48.64	2446.98	0.13	-18.40
	32	35.51	1245.95	0.07	-10.46	16	46.04	1446.46	0.20	-16.76
	36	32.75	879.16	0.06	-8.38	20	43.28	841.50	0.28	-14.13
	40	29.99	623.33	0.07	-5.65	24	40.47	505.78	0.37	-9.13
Container (QCIF)	28	37.31	2156.14	0.05	-14.81	12	48.47	1570.12	0.37	-22.40
	32	34.42	1491.54	0.03	-14.60	16	45.36	907.85	0.51	-18.43
	36	31.66	1006.90	0.07	-13.69	20	42.07	473.22	0.42	-15.84
	40	29.02	703.66	0.09	-11.21	24	39.06	239.79	0.36	-7.67
Coastguard (QCIF)	28	35.49	2594.93	0.07	-18.17	12	47.73	4501.04	0.06	-24.18
	32	32.50	1701.14	0.08	-16.35	16	44.40	3146.68	0.11	-24.66
	36	29.90	1101.83	0.07	-12.42	20	40.84	2001.70	0.20	-24.22
	40	27.62	693.54	0.06	-8.76	24	37.48	1236.30	0.30	-21.13
Paris (CIF)	28	36.77	11830.77	0.04	-14.94	12	48.14	11536.42	0.33	-19.07
	32	33.55	8382.28	0.02	-14.78	16	45.08	6935.89	0.43	-16.62
	36	30.57	5786.56	0.04	-13.87	20	41.89	4113.63	0.39	-17.53
	40	27.76	4000.06	0.04	-12.11	24	38.73	2589.30	0.32	-17.27
Akiyo (CIF)	28	40.47	4089.74	0.06	-9.37	12	49.79	3352.75	0.17	-15.91
	32	37.86	2892.10	0.05	-7.84	16	47.04	1803.51	0.19	-14.57
	36	35.36	2049.58	0.05	-6.07	20	44.63	962.33	0.18	-10.87
	28	32.81	1487.89	0.05	-3.99	24	42.37	545.35	0.26	-6.45

Table 4. Experimental results of ideal zigzag scanning method

5. CONCLUSIONS

In this paper, we proposed a novel zigzag scanning concept of quantized coefficients for H.264/AVC. To develop the novel zigzag scanning, we considered not only the statistical occurrence values of the quantized coefficients after the final mode decision but also the temporal and spatial correlation among the macroblocks. Therefore, we developed the zigzag scanning pattern variously for every macroblock. The proposed method allowed us to save the total bits up to 4.05% and 3.67% while maintaining or improving the image quality for intra- and inter mode, respectively.

REFERENCES

[1] Wiegand, T., Sullivan, G. J. Bjontegaard, G., and Luthra, A., "Overview of the H.264/AVC Video Coding Standard," IEEE Trans. Circuits Syst. Video Technol., Vol. 13, No. 7, July 2003, pp. 560-576.

[2] ITU-T Rec. H.264|ISO-IEC 14496-10 AVC, "Draft ITU-T Recommendation and Final Draft International Standard of Joint Video Specification," JVT Doc. JVT-G050, 2003.

[3] Yoo, Y. J., Jeong, S. Y., and Han, J. K., "Adaptive scan pattern for quantized coefficients in intra coding of H.264, " IEICE Trans. Inf. & Syst., Vol.E92-D, No.4, April 2009, pp. 750-752.

[4] Jia, J., Jung, E. K., and Kim, H. K., "Adaptive transform coefficient scan for H.264 intra coding," IEICE Trans. Inf. & Syst., Vol.E90-D, No.10,Oct. 2007, pp. 1709-1711.

[5] Lee, Y. L., Han, K. H., Sim, D. G., and Seo, J. I., "Adaptive scanning for H.264/AVC intra coding," ETRI Journal, Vol. 28, No. 5, October 2006, pp. 668-671.

[6] Choi, B. D., Kim, J. H., and Ko, S. J., "Adaptive coefficient scanning based on the intra prediction mode," ETRI Journal, Vol.29, No.5, Oct. 2007, pp. 694-696.

[7] Wei, Y. C. and Yang, J. F., "Adaptive mode-dependent scan for H.264/AVC intracoding," Journal of Electronic Imaging, Vol. 19, No. 3, Jul.-Sep. 2010, 033008 pp. 1-12.

[8] Hyun, M. H., Yu, J. T., and Lee, S. B., "Macroblock-dependent zigzag scanning algorithm for H.264/AVC," Electronics Letters, Vol. 48, No. 4, February 2012, pp. 213-215.

[9] Heo, J., Kim, S. H., and Ho, Y. S., "Improved CAVLC for H.264/AVC lossless intra coding," IEEE Trans. Circuits Syst. Video Technol., Vol. 20, No. 2, 2010, pp. 213-222.

[10] Richardson, I. E., The H.264 advanced video compression standard, 2nd ed., Wiley 2010.

[11] Tian, X., Le, T. M., and Lian, Y., Entropy coders of the H.264/AVC standard algorithms and VLSI architectures, Springer 2011.

[12] H.264/AVC Reference Software Version 11.0, http://iphome.hhi.de/suehring/tml/download/.