A Capacity Controllable Reversible Data Hiding Technology for H.264/AVC Based on Compensational Code Mapping

Jinwoo Kang
Departame of Computer Science
Sangmyung University
Seoul, Rep. of Korea
202032027@sangmyung.kr

Sang-ug Kang
Departame of Computer Science
Sangmyung University
Seoul, Rep. of Korea
sukang@smu.ac.kr

***Abstract*— Reversible data hiding technology is widely used to protect copyright or personal information. Recently, many methods applied to the compressed video domain have been released, but it is still a difficult task to completely control the data hiding of the compressed video domain. In this paper, we propose a method that can control the capacity through a method adaptive to the entropy of the compressed domain. This shows that it is possible to effectively reduce the file size that increases due to the embed payload while maintaining the PSNR based on code mapping.**

Keywords—reversible watermarking, data hiding, image compression, H.264/AVC, CAVLC, compensational code mapping

I. Introduction

The development of Reversible Data Hiding (RDH) technology for video has started to become more active because the importance of video security has increased. For example, a video clip from a car dashboard camera is used as legal evidence and must not be tampered with. In this case, the hash code extracted from the video, that is, the bitstream, can be reversibly hidden and can be used to verify the video integrity when necessary. There are other applications, such as tracking content distribution channels and sanity content tagging. Traditional RDH techniques are typically divided into two types: Difference Expansion (DE) [1] and Histogram Shifting (HS) algorithms [2]. DE doubles the difference between two adjacent pixel values and hides one secret bit from the LSB of the doubled difference. Extending DE, the prediction error expansion (PEE) [3,4] uses the prediction error instead of the pixel difference. Also through [5,6], there is an idea of a suitable location to hide in the macro block within the video.

High capacity and low distortion are traditional performance indicators of RDH, but file size is also important in terms of storage savings. When RDH is applied to raw data, file size is not an issue as it remains the same after data hiding. Considering that the entropy coding rule of Context-Adaptive Variable-Length Coding (CAVLC) in H.264 is based on the pattern occurrence probability of natural and normal video sequences, the artificial coefficient pattern may abruptly increase the bitstream size. So, the method of this paper needs to be able to minimize the bitstream size after data hiding. Many technologies including [6,7,8] selected the mid-frequency locations as RDH places because the coefficient change in low-frequency area is more sensitive in image distortion, and the change in high-frequency area affects the CAVLC entropy coding efficiency more than other areas.

II. Proposed Reversible Data Hiding Method

The proposed method consists of two parts, compensation pairing strategy and compensational code mapping. The data hiding algorithm is based on a compensation algorithm and pairing strategy optimized for 4x4 QDCT coefficients of the H.264 baseline profile, demonstrated in [7]. And the proposed method is to replace the calculated result through a compensation algorithm with a coefficient more familiar to CAVLC entropy coding to insert the secret data into the QDCT block. This replacement method uses the method called code mapping [6].

A. Compensation pairing strategy

To hide the data, the reversible data hiding algorithm of [7] is used. In this method, equations are sequentially applied to (r_7, r_8), (r_8, r_9), (r_9, r_{10}), (r_{10}, r_7) using r_7, r_8, r_9, r_{10} of mid-frequency QDCT coefficients. The succession of hiding through the difference of these two coefficients generates compensation and guarantees more payload.

$$d_h(r_i,r_j) = \left\lceil \frac{r_i - r_j}{2} \right\rceil \tag{1}$$

$$(r_i',r_j') = \begin{cases} (r_i + b, r_j - b) \text{ , } d_h(r_i,r_j) = 0 \\ (r_i + 1, r_j - 1) \text{ , } d_h(r_i,r_j) > 0 \\ (r_i, r_j) \qquad\quad \text{ , } d_h(r_i,r_j) < 0 \end{cases} \tag{2}$$

B. Compensational Code Mapping Algorithm

As a result of the compensation pairing strategy, the coefficients of the macro block are changed. Accordingly, CAVLC entropy coding is also affected. Therefore, we fixed various values for each mid-frequency, investigated which series of coefficients affect the file size, and ranked them. Also, due to the quantization of the image compression process, the probability that the frequency is (0,0,0,0) is generally high. The result table with compensation applied to (0,0,0,0) uses a mapping method with a better coefficient. For example, if secret data (0,1,0,1) is embedded in mid-frequency (0,0,0,0), (r_7, r_8, r_9, r_{10}) becomes (-1,1,-1,1). Because the file size increases by changing the run/level value of CAVLC, map (r_7,r_8,r_9,r_{10}) to (0, 1, 0, 0) to avoid it. (0,1,0,0) is taken from the compensation result of mid-frequency (-1,0,0,0).

TABLE I. When ($\boldsymbol{r}_7$, $\boldsymbol{r}_8$, $\boldsymbol{r}_9$, $\boldsymbol{r}_{10}$) = (0, 0, 0, 0), the existing compensation method table and novel compensational table

(A) Existing compensation method table

e_1	e_2	e_3	e_4	r'_7	r'_8	r'_9	r'_{10}
0	0	0	0	0	0	0	0
0	0	0	1	-1	0	0	1
0	0	1	0	0	0	1	-1
0	0	1	1	-1	0	1	0
0	1	0	0	0	1	-1	0
0	1	0	1	-1	1	-1	1
0	1	1	0	0	1	0	-1
0	1	1	1	-1	1	0	0
1	0	0	0	1	-1	0	0
1	0	0	1	0	-1	0	1
1	0	1		1	-1	1	-1
1	1	0	0	1	0	-1	0
1	1	0	1	0	0	-1	1
1	1	1		1	0	0	-1

(B) NOVEL COMPENSATIONAL TABLE

e_1	e_2	e_3	e_4	r'_7	r'_8	r'_9	r'_{10}
0	0	0	0	0	0	0	0
0	0	0	1	-1	0	0	1
0	0	1	0	**1**	**0**	**0**	**0**
0	0	1	1	**-1**	**0**	**0**	**0**
0	1	0	0	0	1	-1	0
0	1	0	1	**0**	**1**	**0**	**0**
0	1	1	0	**0**	**-1**	**0**	**0**
0	1	1	1	**0**	**0**	**1**	**0**
1	0	0	0	**0**	**0**	**-1**	**0**
1	0	0	1	**0**	**0**	**0**	**1**
1	0	1		**0**	**0**	**0**	**-1**
1	1	0	0	**0**	**-1**	**1**	**0**
1	1	0	1	**0**	**1**	**1**	**0**
1	1	1		1	0	0	-1

III. Experimental Results

The proposed method is implemented based on the H.264/AVC JM-18 reference software. We used 352 × 288-sized video sequences at 300 frames in 4 CIFs (Common Intermediate Format), including Bridge(close), Coastguard, Hall monitor, and Mobile for comparison of experiments. We also take the peak signal-to-noise-ratio (PSNR) and the file increase per payload (FPP) index from [8]. FPP is used to measure the effect of file growth after data embedding, and it is divided by the amount of embedding payload by subtracting the file size before and after data hiding. The smaller the FPP, the better the data hiding efficiency, which is the most important point of this study.

From the experimental results, the proposed method and [7] have the same compensation pairing strategy, so the applied mid-frequency coefficients are also the same. Therefore, the payload is omitted because the result is the same. The main thing to check is FPP, and our research method showed a smaller increase in file size even though the same payload was included in the same video. Also, compensational code mapping showed a slightly better or maintained result compared to the PSNR of [7].

IV. Conclusion

In this paper, we proposed an compensational code mapping for adaptively CAVLC of H.264 video sequence. Through this, the capacity of the video can be adjusted and the file size can be reduced. Furthermore, the more familiar and adaptive mapping tables are made in CAVLC, the better results can be obtained in FPP.

Acknowledgment

This research was supported by Basic Science Research Program through the National Research Foundation(NRF) of Korea funded by the Ministry of Science, ICT & Future Planning (NRF-2015R1C1A1A02037777).

References

[1] J. Tian, "Reversible data embedding using a difference expansion," IEEE Trans. Circ Syst. Video Technol. 2003, 13, 890–896.

[2] Z. Ni, Y.Q. Shi, N. Ansari, W. Su, "Reversible data hiding," IEEE Trans. Circuits Syst. Video Technol. 2006, 16, 354–362.

[3] V. Sachnev, H. Kim, J. Nam, S. Suresh, and Y.Q. Shi, "Reversible watermarking algorithm using sorting and prediction," IEEE Trans. Circuits Syst. Video Technol., 2009, vol. 19, no. 7, pp. 989–999.

[4] S. Kang, H. Hwang, H. Kim, "Reversible watermark using an accurate predictor and sorter based on payload balancing," ETRI J. 2012, 34(3):410–420.

[5] Z. Shahid, W. Puech, "A histogram shifting based RDH scheme for H. 264/AVC with controllable drift," In Proceedings of the Media Watermarking, Security, and Forensics 2013 SPIE, Burlingame, CA, USA, 5–7 February 2013, pp. 267–275.

[6] S. Bouchama, H. Aliane, L. Hamami, "Reversible data hiding scheme for the H.264/AVC codec," In Proceedings of the 2013 International Conference on Information Science and Applications (ICISA), Suwon, Korea, 24–26 June 2013, pp. 1–4.

[7] H. Kim, S. Kang, "Genuine reversible data hiding technology using compensation for H.264 bitstreams," Multimed Tools Appl, 2018, 77, 8043–8060.

[8] J. Kang, H. Kim, S. Kang, "Genuine Reversible Data Hiding Technique for H.264 Bitstream Using Multi-Dimensional Histogram Shifting Technology on QDCT Coefficients," Appl. Sci., 2020, 10, 6410.

TABLE II. COMPARISON BETWEEN OUR PROSED METHOD AND THE METHODS OF [7]

Video	*Performance*	*Method*	*Quantization Parameter*				*Average*
			24	*22*	*20*	*18*	
Bridge(close)	PSNR (dB)	Proposed	31.51	31.95	36.54	40.69	35.17
		[7]	31.08	32.29	36.24	40.38	35.00
	FPP	Proposed	2.35	2.34	2.40	2.48	2.39
		[7]	2.98	2.96	2.93	2.90	2.94
Coastguard	PSNR (dB)	Proposed	30.72	32.99	35.70	38.06	34.37
		[7]	30.49	32.74	35.09	38.06	34.09
	FPP	Proposed	2.34	2.45	2.57	2.61	2.49
		[7]	2.91	2.95	3.00	3.01	2.97
Hall monitor	PSNR (dB)	Proposed	29.55	31.84	32.89	36.34	32.65
		[7]	29.59	31.64	33.91	36.56	32.92
	FPP	Proposed	2.23	2.23	2.22	2.19	2.22
		[7]	2.92	2.93	2.91	2.82	2.90
Mobile	PSNR (dB)	Proposed	33.58	35.32	38.28	39.84	36.76
		[7]	32.95	35.16	38.09	39.80	36.50
	FPP	Proposed	2.33	2.34	2.35	2.31	2.33
		[7]	2.87	2.85	2.85	2.81	2.85