

Lossless data hiding technique reducing cover data size for compressed videos

Xuefei Li[✉],* Seok Kang, and Yuji Sakamoto

Hokkaido University, Faculty of Information Science and Technology, Media Creation Methodology Laboratory, Sapporo, Japan

ABSTRACT. In recent years, multimedia techniques have rapidly advanced. Video discs with massive storage capacities exemplify widely used multimedia storage mediums. However, online multimedia services face strict storage limits, and compression rates impact streaming quality. This poses a dilemma—data hiding risks exceeding storage limits and affecting the quality of video streams. To address this, we propose a lossless data hiding technique focusing on cover data size. The technique embeds new data into compressed cover video data, decreasing data size post-embedding. Extraction requires no key. In addition, the processing time is short enough that the technique can be used in real-time applications. We verified the technique through experiments with MPEG-2 videos. Video data size decreased after hiding data, and quality degradation remained acceptable. Results confirm the technique's efficacy.

© The Authors. Published by SPIE under a Creative Commons Attribution 4.0 International License. Distribution or reproduction of this work in whole or in part requires full attribution of the original publication, including its DOI. [DOI: [10.1117/1.JEI.33.1.011005](https://doi.org/10.1117/1.JEI.33.1.011005)]

Keywords: DCT coefficients; data hiding; video; data size reduction; real-time

Paper 230649SS received Jun. 2, 2023; revised Nov. 18, 2023; accepted Jan. 11, 2024; published Feb. 1, 2024.

1 Introduction

Multimedia, merging different types of media, such as video, audio, context, and interaction, have become a non-splittable part of people's lives. In recent years, multimedia techniques have developed significantly and rapidly. As representative examples of multimedia contents, movies and dramas have been entertaining people for a number of decades. Those contents are usually the composition of video, audio, and sometimes subtitles. Video discs, including capacitance electronic disc (CED), video high density (VHD), video compact disc (VCD), digital versatile disc (DVD), Blu-ray disc (BD), UHD BD (ultra HD Blu-ray disc), etc., are portable storages for multimedia contents. DVDs and BDs are digital storage formats widely used in the world currently, and UHD BDs that support 4K videos are considered the major digital storage formats in the coming years. On the other hand, online video on demand (VOD) services keep massive multimedia data in their storage while streaming services provide streaming data with acceptable quality to users.

Data hiding for videos,^{1,2} by the way, is a developing field of technology that has been emerging recently. Video data hiding techniques and image data hiding techniques that can be utilized for videos have been actively studied^{3,4} while some basic ones have been proposed before.⁵⁻⁷ However, because the video discs and the online services use lossily compressed video data, the additional data embedded into raw data (e.g., YUV sequences) before compression have a great risk of loss after quantization. Thus, it is reasonable to embed data into compressed data. In DVD and BD videos, formats, such as MPEG-2 and H.264/AVC, are commonly used; while in UHD BD videos, H.265/MPEG-H part 2 (HEVC) is used. These formats are also generally used

*Address all correspondence to Xuefei Li, 4everxnow@gmail.com

by online VOD and streaming services. In those formats, orthogonal transformations including discrete cosine transform (DCT) and discrete sine transform (DST) are performed in units of blocks to generate coefficients in frequency domain, then the coefficients are quantized in the purpose of compression. Note that the processes after quantization are totally lossless. Therefore, quantized transform coefficients (qTCs) are possible targets for lossless data hiding in compressed videos. Techniques embedding data using qTCs have also been proposed.^{8–14} According to the reversibility of the cover data (image or video), data hiding techniques can be classified as the reversible type and the irreversible type. In general, reversible data hiding techniques need high redundancy in order to restore the original data and will lead to a significant file-size overhead of the cover data, which is not ideal for video discs because of the strict capacity limit.^{15–18} On the other hand, irreversible data hiding techniques do not need high redundancy and will not lead to significant file-size overhead.^{19–22}

In fact, any existing multimedia storage has a strict capacity limit, and the compression rate of online VOD and streaming services are of importance. Despite of that, data hiding processes for compressed videos generally increase the data size of the contents. Consequently, the increased data size has a risk of exceeding the capacity limit of the storage and affecting the quality of video streaming. Therefore, in order to preserve the integrity of both the contents and the additional data and maintain the quality of video streaming, data hiding techniques that never cause the increase of the data size are expected. Some techniques that never cause the increase of the data size have been proposed before.^{20–24} Least significant bit (LSB) replacement technique as well as its variants is one of the data hiding techniques which can save data size for uncompressed images or videos.^{20–23} Another technique²⁴ losslessly embedding some qTC blocks into others is for compressed images, but it cannot embed additional data except for the cover images themselves (it is actually a compression technique using data hiding).

In this paper, we propose a novel data hiding technique for compressed videos. The proposed technique is lossless and the data size of the videos never increases. Several related works and the proposed technique will be explained in Secs. 2 and 3, respectively, and the experiments with results will be introduced and discussed in Sec. 4.

2 Related Works

In this section, after the introduction of investigation on cover data size, three conventional data hiding techniques—the classical LSB replacement technique, the modulo addition technique, and the modified modulo addition technique will be explained first, and several other conventional techniques will be introduced next.

2.1 Impact on Data Size

Most natural (i.e., not artificial) images have a great spatial redundancy. Therefore, the orthogonal transformations are performed to reduce the effects of the redundancy. As a result, the values of the AC coefficients have the tendency of a Laplacian distribution where the location parameter $\mu = 0$, as shown in Fig. 1.^{25,26} Utilizing the characteristics of the Laplacian distribution, the qTCs will then be coded by applying two entropy coding methods: context adaptive variable length coding (CAVLC) or context adaptive binary arithmetic coding (CABAC). MPEG-2 supports only CAVLC, HEVC supports only CABAC while H.264/AVC supports both. In general, entropy coding methods distribute shorter code to more frequent data.

In the case of CAVLC, qTCs are broken down into level values and zero run-length values before applying CAVLC. Afterwards, the level values and the zero run-length values are coded (as run-level pairs in MPEG-2; separately in H.264/AVC) using the VLC tables. According to the tables, as long as the amount of the zero coefficients in one block remains unchanged, the data size will decrease when the absolute values of the AC coefficients decrease.

In the case of CABAC, only the levels and the positions of the non-zero AC coefficients are coded. Those values are broken down into several syntaxes and then binarized and coded. The same as in the case of CAVLC, as long as the amount of the zero coefficients in one block remains unchanged, the data size will decrease when the absolute values of the AC coefficients decrease.

Consequently, in order to decrease the data size, it is reasonable to decrease the absolute values of the non-zero AC coefficients without reducing the amount of zero AC coefficients.

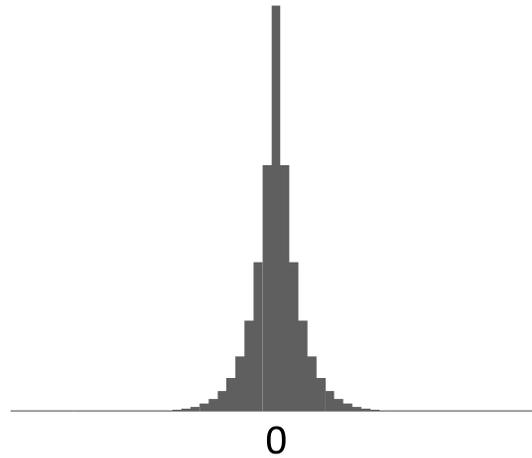


Fig. 1 Laplacian distribution of the AC coefficients.

2.2 Classical LSB Replacement Technique

For one piece of data, its bit plane can be arranged in a from-most-to-least significant order, and the last bit in that order is expressed as an LSB. For example, the rightmost bit of an eight-bit binary number is its LSB. In the classical LSB replacement technique, the LSB of the target of data hiding will be replaced with one bit of additional data, and this process may cause the original value of the target to change by one. The embedded one bit of data can be extracted by directly referring to the LSB of the target. An example of the classical LSB replacement technique is shown in Fig. 2.

2.3 Modulo Addition Technique

For one piece of integer data, the value of it could be expressed as V . The process embedding one bit of additional data, which has a value of A will change V to V_e as follows:

$$V_e = \left\lfloor \frac{V}{2} \right\rfloor \times 2 + A, \quad (1)$$

where $\lfloor \cdot \rfloor$ indicates the rounding down operation.

The embedded data bit A could be extracted as follows:

$$A = V_e \bmod 2, \quad (2)$$

where mod indicates the modulo operation.

In fact, the modulo addition technique can be considered as a variant of the classical LSB replacement technique, in particular the two techniques are sometimes totally the same depending on the representation method of the bit plane of negative numbers.

According to Sec. 2.1, in MPEG-2, H.264/AVC, and HEVC formats, the same as in some other image and video formats using orthogonal transformations, such as JPEG, turning an AC coefficient with a value of 0 into a non-zero value will directly increase the data size. Obviously, the classical LSB replacement technique and the modulo addition technique will inevitably increase the number of non-zero AC coefficients. In addition, according to Eq. (1), if V is positive and is a multiple of 2, the absolute value of V_e will be greater than that of V when $A = 1$.

Those features pose a risk of massive data size increase, which is undesirable in most scenarios of data hiding, e.g., in the case that the remaining capacity of the video disc is insufficient. This technique will be expressed as “MA” since then.

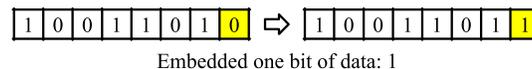


Fig. 2 Example of classical LSB replacement technique.

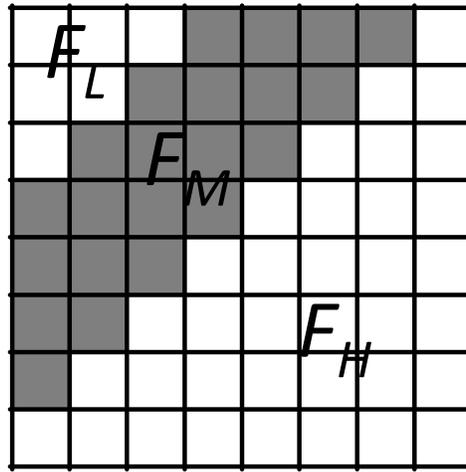


Fig. 3 Various frequency regions in a 8×8 block.

2.4 Modified Modulo Addition Technique

As a modified version of the modulo addition technique, an approach has been proposed before.²⁷ In that approach, in order to avoid the risk of increasing the absolute value of an AC coefficient, the algorithm of embedding was modified as follows:

$$V_e = \text{int}\left(\frac{V-A}{2}\right) \times 2 + A, \quad (3)$$

where $\text{int}(x)$ indicates the integer part of x and can be defined as

$$\text{int}(x) = \begin{cases} \lceil x \rceil & (x > 0) \\ \lfloor x \rfloor & (\text{otherwise}) \end{cases}, \quad (4)$$

where $\lceil \cdot \rceil$ indicates the rounding up operation.

The embedded data bit A could be extracted with Eq. (2) as well.

This technique avoids the risk to increase the absolute value of a non-zero AC coefficient. On the other hand, it changes the value of some AC coefficients from 0 to 1. As a result, the data size of the video may increase after data hiding process. This technique will be expressed as “mMA” since then.

The ideal positions to embed additional data in a block have been studied before.²⁸ The qTCs of a 8×8 block could be divided into three regions, which are F_L (low frequencies), F_M (medium frequencies), and F_H (high frequencies) as shown in Fig. 3. The band of frequencies suitable for embedding purpose is F_M .²⁸ However, the mMA technique performs data hiding mainly in F_L . In order to develop this technique, AC coefficients in F_M should be chosen for embedding when the payload is slight.

2.5 Other Related Techniques

Some other data hiding techniques proposed for compressed images and videos have also been proposed.²⁹⁻³² Mobasseri et al. proposed a data hiding technique for MPEG-2 videos using VLC mapping,²⁹ which will not increase the data size of the videos. However, it only supports CAVLC and but also requires detection process beforehand. In addition, the maximum payload size depends on the cover video. Chaumont et al. proposed a DCT-based data hiding technique that spreads the payload data into a group of DCT coefficients.³⁰ The technique can realize a better image quality, but it also leads to a data size increase. Hartung et al. can embed additional data into MPEG-2 videos without data size increase,³¹ yet it has a risk of losing the additional data because it embeds the additional data into the inverse quantized DCT coefficients and quantize them again (the quantization process is lossy). Those techniques are effective respectively in some scenarios, but any one of them has obvious defects in other scenarios, especially in the cases of data hiding into capacity-limited multimedia storages.

F4, the base of F5,³² is a technique that embeds data into nonzero AC coefficients of cover data while reducing the cover data size. However, this technique inevitably turns some 1 and -1 AC coefficients into zeros, thus in order to acquire the embedded data, a unique key for each cover data is needed to be produced to record which zeros are unchanged, which are changed from 1 and which are changed from -1 . This key has to be merged with any other keys for data extraction of each cover data and the key-sharing is costly in most data hiding situations. F5 is an improved version of F4 and also has this characteristic.

3 Proposed Technique

3.1 Overview

In MPEG-2, H.264/AVC, and HEVC, frames (pictures) are classified as I (intra) pictures, P (predictive) pictures, and B (bidirectionally predictive) pictures. Each frame is divided into blocks and the blocks are classified roughly as intra-blocks and inter-blocks of luminance and chrominance. Orthogonal transformations and quantization are then performed within each block and qTCs are acquired. Because inter-blocks are basically predicted from intra-blocks, the degradation of the intra-blocks will be propagated to inter-blocks and accumulated.³³ At the same time, human eyes are more sensitive to direct current (DC) coefficients than alternating current (AC) coefficients. Thus, it is conceivable that AC ingredients of qTCs in intra-blocks are proper targets of data hiding process. The flowcharts of the proposed processes are shown in Fig. 4.

3.2 Details of the Proposed Technique

In this section, the proposed polarized LSB replacement technique will be explained and discussed.

What we have already discussed is that the video data size may increase after data hiding process in MA and mMA, and key-sharing is costly in F4. Here, we propose a novel technique, polarized LSB replacement, to avoid data size increase and key-sharing, which will be expressed as “PLSB-r” since then. Before the algorithm is explained, a partial lookup table including several original qTCs and the corresponding modified qTCs for MA, mMA, and PLSB-r is shown as Table 1. The changes of the qTC values are shown in Fig. 5: where the red arrows refer to the changes which increase the absolute value of qTCs, green arrows for changes which decrease the absolute value of qTCs, and white ones for no change of the absolute value. From Table 1 and Fig. 5, we can conclude that different from MA and mMA, the proposed technique never increases the absolute value of any qTC. Further, as long as any qTC which has a value greater than 1 or less than -1 exists (this condition is basically satisfied in all natural images and videos with sufficient quality), the video data size decreases due to the decrease of the absolute value of those qTCs.

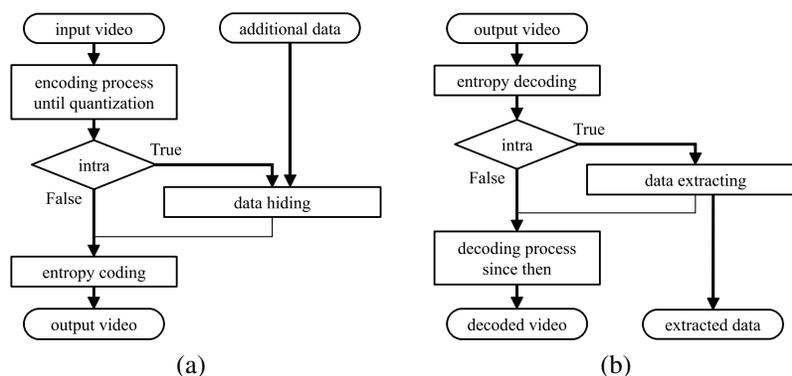


Fig. 4 Flowcharts of the proposed processes: (a) embedding and (b) extracting.

Table 1 Modified qTC lookup table.

Additional data	0				1			
	Technique							
Original qTC	MA	mMA	F4	PLSB-r	MA	mMA	F4	PLSB-r
4	4	4	4	4	5	3	3	3
3	2	2	2	2	3	3	3	3
2	2	2	2	2	3	1	1	1
1	0	0	0	-1	1	1	1	1
0	0	0	—	—	1	1	—	—
-1	-2	0	-1	-1	-1	-1	0	1
-2	-2	-2	-1	-1	-1	-1	-2	-2
-3	-4	-2	-3	-3	-3	-3	-2	-2
-4	-4	-4	-3	-3	-3	-3	-4	-4

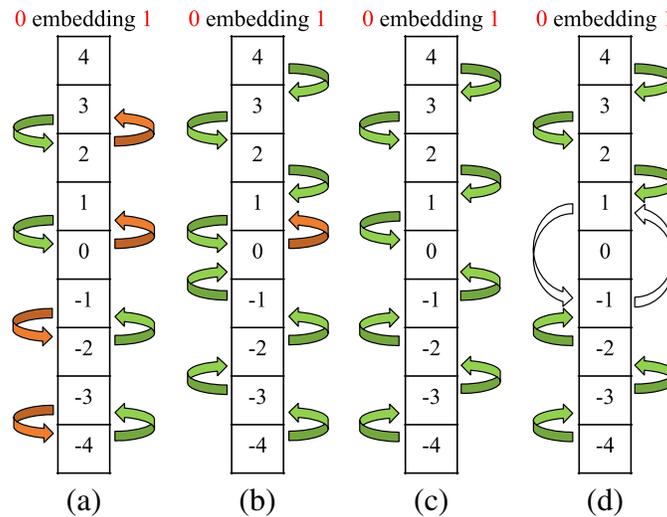


Fig. 5 Changes of qTC values: (a) MA, (b) mMA, (c) F4, and (d) PLSB-r.

The algorithm of the proposed PLSB-r technique is explained below. First, only in the cases where $V \neq 0$, V_e is acquired as follows:

$$V_e = \begin{cases} \text{int}(\frac{V-A}{2}) \times 2 + A & (V > 0) \\ \text{int}(\frac{V-A+1}{2}) \times 2 + A & (V < 0) \end{cases} \quad (5)$$

Next, if $V_e \leq 0$, it will be reduced by 1 as follows:

$$V'_e = \begin{cases} V_e & (V_e > 0) \\ V_e - 1 & (V_e \leq 0) \end{cases} \quad (6)$$

The value V of one qTC is changed to V'_e finally. Note that $V'_e \neq 0$, the additional data bit A could be extracted as follows:

$$A = \begin{cases} (V'_e) \bmod 2 & (V'_e > 0) \\ (V'_e + 1) \bmod 2 & (V'_e < 0) \end{cases} \quad (7)$$

In other words, the same as F4, “1” is extracted if the modified qTC has a positive odd value or a negative even value; “0” is extracted if the modified qTC has a positive even value or a negative odd value. In this way, the original video is not required for the extraction. Different from F4, zero AC coefficients are skipped so that the amount of zero AC coefficients remains the same, thus no unique key for each cover data is required for data extraction.

In the proposed technique, the sign of a qTC with a value of 1 or -1 may invert, i.e., -1 becomes 1; 1 becomes -1 , and the parity of the positive modified qTCs and negative modified qTCs conflict when the same data bit is embedded. The histogram for AC coefficients of cover data remains symmetrical and nearly remains a Laplacian distribution, therefore whether the cover data carry hidden data is difficult to detect by attackers. These characteristics make the proposed technique totally different from conventional LSB replacement and modulo addition techniques, in which the parity of the positive modified qTCs and negative modified qTCs are generally the same. Due to that feature of the proposed technique, although the robustness of the proposed technique is relatively weak, it is more effective to use it in steganography situations, in which the lack of detectability is rather important, or enrichment situations, in which the robustness can be disrespected to a certain extent.

The time-complexity of both the data hiding process and the data extraction process is $O(n)$. The processing time of the proposed technique is generally negligible, which is $< 1/100$ of the video coding or decoding time in our experiments. Thus, the proposed technique is suitable for real-time situations where an information timeline is important. The scheme of the proposed technique is simple so that it can be combined with other data hiding techniques that enhance security and other aspects.

As defect of the proposed technique, the maximum payload will be less than MA and mMA because embedding process is only performed to non-zero qTCs, and the quality will be slightly worse than F4 because the change of the absolute value of 1 and -1 AC coefficients is greater.

As shown in Table 1, when $n = 1$, the absolute value of an AC coefficient will never increase after the embedding process is performed.

4 Experiments and Results

4.1 Overview

To verify the effectiveness of the proposed technique, we conducted several experiments with the MPEG-2 format.

In the experiments, the 15 videos we used were uncompressed raw videos in YUV format.³⁴ The first frames of the experimental videos are shown in Fig. 6. The 15 raw videos were compressed to common DVD and BD formats that are also generally used in the videos of online services with the specifications shown in Table 2, respectively. The compressed videos with no data embedded will be represented as baseline videos since then. In order to compare the videos with data embedded and the baseline videos objectively, fixed quantization parameter (QP) values are used to keep the coding process the same for the comparison objects. The QP values of

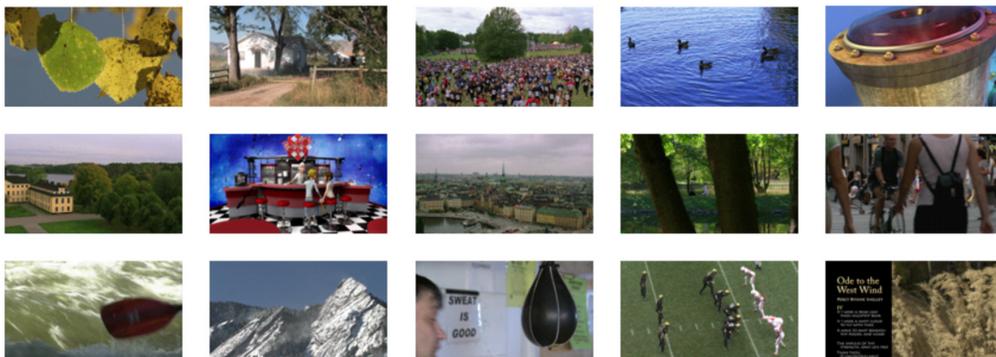


Fig. 6 First frames of the experimental videos. Horizontally from top left: a, aspen; co, controlled burn; cr, crowd run; d, ducks take off; f, factory; i, in to tree; l, life; o, old town cross; pa, park joy; pe, pedestrian area; r, red kayak; sn, snow mnt; sp, speed bag; t, touchdown pass; w, west wind easy.

Table 2 Details of the experimental videos.

Target	DVD	BD
Codec	MPEG-2	
Resolution	720 × 480	1920 × 1080
Aspect ratio	16:9	
Frame rate	29.97 fps	
Number of frames	300	
Number of frames in one GOP	18	12
Scan type	Progressive	
Color space	YUV	
Chroma subsampling	4:2:0	
Average bit rate	4.69 Mbps	19.78 Mbps

different videos vary in order to keep the bit rates similar for different videos as shown in Table 3. The bit rates of the baseline videos and the average of them are shown in Fig. 7. The additional data to be embedded are a randomly generated bit-sequence S and all experiments use the same S .

To compare the proposed technique with the conventional ones, we performed MA, mMA, and PLSB-r to the experimental videos. Two conventional techniques using qTCs were also performed with the goal of comparison,^{13,14} which are expressed as Adap and Psy since then, respectively. For the purpose of comparing the techniques with similar payload values, in MA and mMA, four bits of additional data are embedded in the F_M of one block, while in PLSB-r, all non-zero quantized AC coefficients in intra-blocks are used for embedding, in Adap, the payload is adjusted to be similar as the proposed technique. However, Psy cannot achieve a similar payload as the proposed technique, so we maximized the payload in Psy.

Table 3 QP of the experimental videos.

Video	Target	
	DVD	BD
a	6	5
co	11	10
cr	12	12
d	13	16
f	2	2
i	5	6
l	3	2
o	5	6
pa	12	16
pe	13	14
r	8	9
sn	8	8
sp	1	3
t	3	4
w	9	7

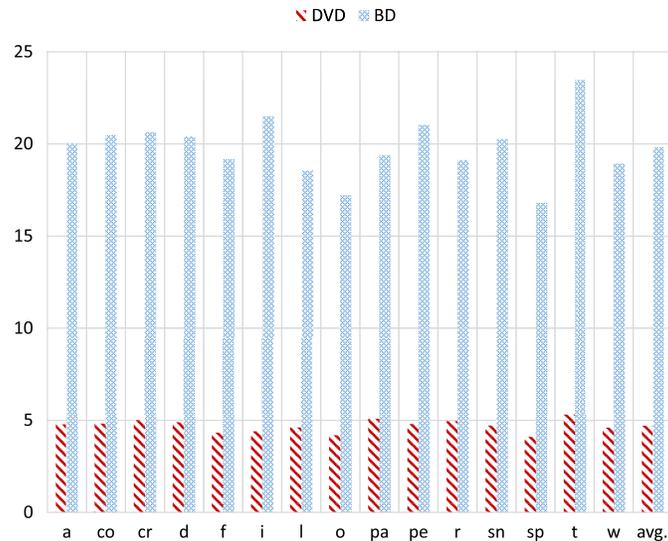


Fig. 7 Bit rates of the experimental videos (Mbps).

4.2 Losslessness Verification Experiment

To verify the losslessness of the proposed technique, we calculated the mean squared error (MSE) value of the extracted data and S . The MSE value is calculated as follows:

$$\text{MSE} = \frac{1}{n} \sum_{i=1}^n (b_{ei} - b_i)^2, \quad (8)$$

where b_{ei} indicates the i 'th bit of the extracted data and b_i indicates the i 'th bit of S .

The MSE values were all 0, which corroborated that the data hiding processes of the three techniques are lossless.

4.3 Effectiveness Verification Experiment

For the purpose of verifying the effectiveness of the proposed techniques, we evaluated the cover videos in viewpoints of quality, payload, and file-size overhead. P (payload) stands for the size of embedded data. O (file-size overhead) stands for the increase of the video data size and is defined as: $O = F_2 - F_1$, where F_2 is the data size after data hiding and F_1 is the data size before data hiding.

The quality is evaluated using peak signal-to-noise ratio (PSNR) and structural similarity (SSIM),³⁵ which are both calculated between the enriched videos and the baseline videos. Both PSNR value and SSIM value for a pair of videos are the average of all frames.

The payload and file-size overhead results are represented as payload per second and overhead per second in order to help the users of the proposed system to understand them in a kind of practical manner.

The overhead-to-payload ratios are calculated as $\text{ratio} = O/P$. The overhead-to-payload ratio could be considered as a payload efficiency evaluation index of data hiding in the viewpoint of overhead. The file-size increase rates are calculated as $\text{FR} = O/F$ and the payload rates are calculated as $\text{PR} = P/F$, where F stands for the file-size of the baseline video. Negative file-size increase rate values indicate that the file-size of the cover video decreases after data hiding process.

The average values of the experiment results of all the 15 experimental videos for both DVD and BD are shown in Figs. 8–10, and the comparison of the proposed technique with MA, mMA, Adap, and Psy are shown in Table 4. As an example, the output videos of Adap, Psy, and PLSB-r for “aspen,” “ducks take off,” “factory,” “pedestrian area,” and “west wind easy” after data hiding process are shown in Fig. 11 with the baseline video.

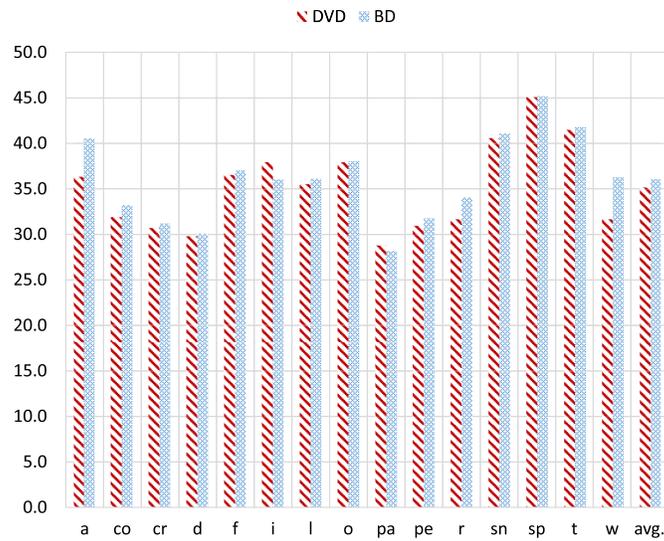


Fig. 8 PSNR results of the experimental videos (dB).

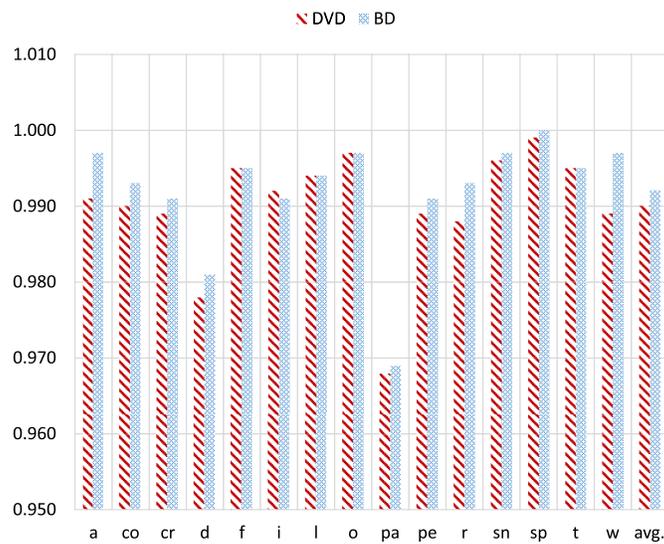


Fig. 9 SSIM results of the experimental videos.

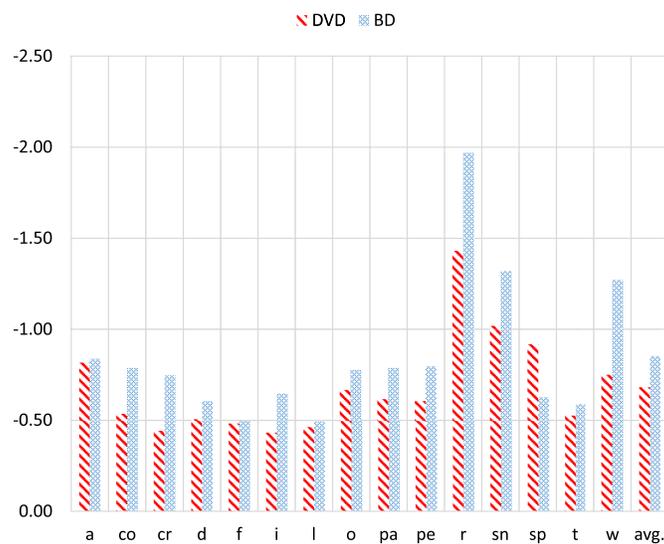


Fig. 10 File-size increase rate results of the experimental videos (%).

Table 4 Average results of the comparison between conventional and proposed techniques.

Technique	PSNR (dB)	SSIM	Payload per second (Kbps)	Overhead per second (Kbps)	Ratio	FR (%)	PR (%)
(a) DVD videos							
MA	44.0	0.999	68.6	51.1	0.75	1.05	1.43
mMA	43.4	0.999	68.6	37.7	0.55	0.77	1.43
Adap	35.6	0.989	69.4	30.5	0.44	0.63	1.45
Psy	46.8	0.999	20.9	7.4	0.35	0.15	0.43
PLSB-r	35.1	0.990	66.6	-33.3	-0.50	-0.68	1.39
(b) BD videos							
MA	43.7	0.999	676.1	580.3	0.86	2.92	3.34
mMA	43.6	0.999	676.1	497.0	0.74	2.49	3.34
Adap	37.0	0.992	432.8	285.0	0.66	1.42	2.14
Psy	45.8	0.999	120.6	90.4	0.75	0.43	0.60
PLSB-r	36.0	0.992	446.1	-170.3	-0.38	-0.85	2.20

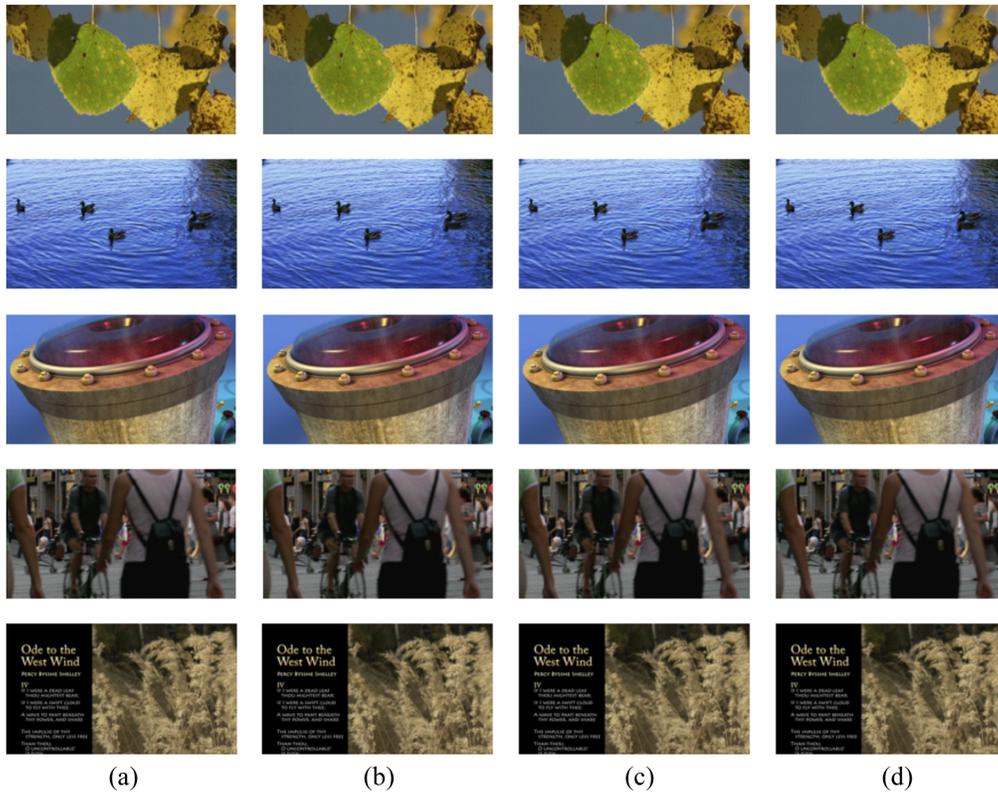


Fig. 11 Baseline video and data hiding results for “aspen,” “ducks take off,” “factory,” “pedestrian area,” and “west wind easy”: (a) baseline, (b) Adap, (c) Psy, and (d) PLSB-r, from top to bottom.

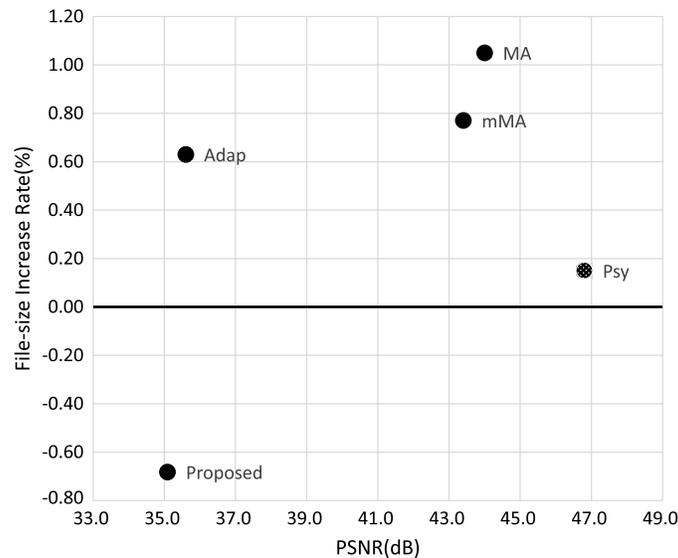


Fig. 12 PSNR versus file-size increase rate.

4.4 Discussion

From Figs. 8–11, we can conclude that the quality of the output videos with data hidden in them are acceptable. From Fig. 10, we can conclude that the data sizes of all the videos decrease after data hiding.

On the other hand, as shown in Fig. 12, note that Psy has much smaller payload which suggests that it is not suitable for comparison. Compared with the rest three techniques with similar payload (in the case of DVD videos, 68.6 Kbps for MA and mMA, 69.4 Kbps for Adap, and 66.6 Kbps for PLSB-r), the proposed technique realizes data size decrease in exchange for a certain degree of the video quality degradation. However, the PSNR values are around 35 dB while the SSIM values are around 0.99, due to which we can consider that the video quality is sufficient. Furthermore, Fig. 11 indicates that the video quality degradation is visually insensitive.

5 Conclusion

In this paper, we proposed a novel data hiding technique for videos using qTCs. In the proposed system, newly added data are embedded into AC ingredients of qTCs of the cover data. The quality degradation of the cover data is controlled to be acceptable, and the video data size decreases after the data hiding process. No key is needed for extraction of embedded data.

However, even though the proposed technique is suitable for other formats using qTCs such as H264/AVC and HEVC theoretically, the experimental proof is required. In the future, the experiments for H264/AVC, HEVC, and other formats using qTCs with the proposed technique will be a topic for us.

Disclosures

We have no relevant financial interests in the manuscript and no other potential conflicts of interest to disclose.

Code and Data Availability

In the proposed technique, the experiments are conducted with FFMPEG <https://ffmpeg.org/> 2.4.2, where our code is implemented in two source c files: mpegvideo_enc.c and mpeg4videodec.c in the libavcodec folder before compiling FFMPEG. However, as the version of the FFMPEG we used is out of date, we recommend installing up-to-date versions of FFMPEG or other up-to-date tools to implement the proposed technique. If the files we used in our experiments are needed, contact us for them.

Acknowledgments

This paper is under no support or fund.

References

1. I. Cox et al., *Digital Watermarking and Steganography*, Morgan Kaufmann Publishers Inc., San Francisco, California (2008).
2. Y. Tew and K. Wong, "An overview of information hiding in H.264/AVC compressed video," *IEEE Trans. Circuits Syst. Video Technol.* **24**(2), 305–319 (2014).
3. S. Roop, M. Himanshu, and P. Raju, "Optimal keyframe selection-based lossless video-watermarking technique using IGSA in LWT domain for copyright protection," *Complex Intell. Syst.* **8**(2), 1047–1070 (2021).
4. K. Sahib et al., "A modulo function-based robust asymmetric variable data hiding using DCT," *Symmetry* **12**(10), 1659 (2020).
5. P. W. Wong, "A watermark for image integrity and ownership verification," in *Image Process., Image Quality, Image Capture Syst. Conf.* (1998).
6. J. Fridrich, M. Goljan, and R. Du, "Lossless data embedding: new paradigm in digital watermarking," *EURASIP J. Adv. Signal Process.* **2002**(2), 185–196 (2002).
7. T. Kalker and F. Willems, "Capacity bounds and constructions for reversible data-hiding," in *14th Int. Conf. on Digit. Signal Process. Proc. DSP 2002 (Cat. No. 02TH8628)* (2002).
8. G. Xuan et al., "Reversible data hiding for JPEG images based on histogram pairs," *Lect. Notes Comput. Sci.* **4633**, 715–727 (2007).
9. P. Zheng, B. Zhao, and M. Liu, "An efficient method to hide information in MPEG video sequences," in *Int. Conf. Multimedia Inf. Netw. and Security*, IEEE (2009).
10. Z. Shahid, M. Chaumont, and W. Puech, "Considering the reconstruction loop for data hiding of intra- and inter-frames of H.264/AVC," *Signal Image Video Process.* **7**(1), 75–93 (2011).
11. S. Gujjunoori and B. B. Amberker, "A DCT based near reversible data embedding scheme for MPEG-4 video," in *Proc. Fourth Int. Conf. Signal and Image Process.*, Springer India, pp. 69–79 (2013).
12. F. Huang et al., "Reversible data hiding in JPEG images," *IEEE Trans. Circuits Syst. Video Technol.* **26**(9), 1610–1621 (2016).
13. Y. Chen et al., "Adaptive video data hiding through cost assignment and STCS," *IEEE Trans. Depend. Secure Comput.* **18**(3), 1320–1335 (2019).
14. F. Muhammad and E. Ferda, "Video steganography based on DCT psychovisual and object motion," *Bull. Electr. Eng. Inf.* **9**(3), 1015–1023 (2020).
15. Z. Ni et al., "Reversible data hiding," *IEEE Trans. Circuits Syst. Video Technol.* **16**(3), 354–362 (2006).
16. J. Tian, "Reversible watermarking by difference expansion," in *Proc. Workshop on Multimedia and Security*, ACM, Vol. **19** (2002).
17. L. R. Mathews and V. A. C. Haran, "Histogram shifting based reversible data hiding using block division and pixel differences," in *Int. Conf. on Control, Instrum., Commun. and Computat. Technol. (ICCICCT)*, IEEE (2014).
18. Z. Shahid and W. Puech, "A histogram shifting based RDH scheme for H.264/AVC with controllable drift," *Proc. SPIE* **8665**, 86650S (2013).
19. S. M. Kim et al., "Data hiding on H.264/AVC compressed video," *Lect. Notes Comput. Sci.* **4633**, 698–707 (2007).
20. Y. K. Lee and L. H. Chen, "High capacity image steganographic model," in *IEE Proc. – Vis., Image, and Signal Process.*, Vol. **147**, p. 288 (2000).
21. M. U. Celik et al., "Lossless generalized-LSB data embedding," *IEEE Trans. Image Process.* **14**(2), 253–266 (2005).
22. C. Lalengmawia, A. Bhattacharya, and A. Datta, "Image steganography using advanced encryption standard for implantation of audio/video data," in *Int. Conf. Recent Trends in Inf. Technol. (ICRTIT)*, IEEE (2016).
23. M. T. Al-Bayati and M. M. Al-Jarrah, "DuoHide: a secure system for hiding multimedia files in dual cover images," in *9th Int. Conf. Dev. in eSystems Eng. (DeSE)*, IEEE (2016).
24. R. Jafari, D. Ziou, and A. Mammeri, "Increasing compression of JPEG images using steganography," in *IEEE Int. Symp. Rob. and Sens. Environ. (ROSE)*, IEEE (2011).
25. E. Y. Lam and J. W. Goodman, "A mathematical analysis of the DCT coefficient distributions for images," *IEEE Trans. Image Process.* **9**(10), 1661–1666 (2000).
26. N. Kamaci and G. AlRegib, "Improved DCT coefficient distribution modeling for h.264-like video coders based on block classification," in *18th IEEE Int. Conf. on Image Process.*, IEEE (2011).
27. Y. Miura et al., "Additional data hiding technique for specified regions of JPEG image," *IEICE Tech. Rep.*, 116(34), 1–6, IEICE (2016).
28. S. A. Parah et al., "Hiding in encrypted images: a three tier security data hiding technique," *Multidimens. Syst. Signal Process.* **28**(2), 549–572 (2015).

29. B. G. Mobasseri and M. P. Marcinak, "Watermarking of MPEG-2 video in compressed domain using VLC mapping," in *Proc. 7th Workshop on Multimedia and Security*, ACM (2005).
30. M. Chaumont and W. Puech, "A DCT-based data-hiding method to embed the color information in a JPEG grey level image," in *14th Eur. Signal Process. Conf.* (2006).
31. F. Hartung and B. Girod, "Digital watermarking of MPEG-2 coded video in the bitstream domain," in *IEEE Int. Conf. on Acoust., Speech, and Signal Process.*, IEEE Computer Society Press (1997).
32. A. Westfield, "F5: a steganographic algorithm," *Lect. Notes Comput. Sci.* **2137**, 289–302 (2001).
33. Y. Nemoto et al., "A study on video scrambling considering inter-frame prediction," in *IEICE Tech. Rep.*, Image Engineering 105(500), pp. 207–212, IEICE (2006).
34. Xiph open source community, "Derf's test media collection," 2023, <https://media.xiph.org/video/derf/>.
35. Z. Wang et al., "Image quality assessment: from error visibility to structural similarity," *IEEE Trans. Image Process.* **13**(4), 600–612 (2004).

Xuefei Li received his BS degree in engineering from Zhejiang University in 2013 and his MS degree in information science and engineering from Hokkaido University in 2015. He is now in the doctor course of Graduate School of Information Science and Technology, Hokkaido University. His research interests are scrambling and data hiding techniques in images and videos.

Seok Kang received his BS degree in engineering from Soongsil University in 1994 and his PhD in the field of electronics and information engineering from Hokkaido University in 2000. He is now with the Graduate School of Information Science and Technology, Hokkaido University.

Yuji Sakamoto received his BS degree in engineering from Hokkaido University in 1983 and his PhD in information science and engineering from Hokkaido University in 1988. He is now with the Graduate School of Information Science and Technology, Hokkaido University.