

Research Article

High Embedding Capacity Data Hiding Algorithm for H.264/AVC Video Sequences without Intraframe Distortion Drift

**Dinh-Chien Nguyen,^{1,2} Thai-Son Nguyen ,³ Chin-Chen Chang ,¹
Huan-Sheng Hsueh,⁴ and Fang-Rong Hsu¹**

¹Department of Information Engineering and Computer Science, Feng Chia University, Taichung 40724, Taiwan

²Faculty of Information Technology, Hung Yen University of Technology and Education, Dan Tien, Khoai Chau, Hung Yen, Vietnam

³School of Engineering and Technology, Tra Vinh University, Tra Vinh Province, Vietnam

⁴Department of Computer Science and Information Engineering, National Chung Cheng University, Chiayi, Taiwan

Correspondence should be addressed to Thai-Son Nguyen; thaison@tvu.edu.vn

Received 6 December 2017; Revised 5 June 2018; Accepted 18 July 2018; Published 1 August 2018

Academic Editor: Fawad Ahmed

Copyright © 2018 Dinh-Chien Nguyen et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Data hiding is a technique that allows secret data to be delivered securely by embedding the data into cover digital media. In this paper, we propose a new data hiding algorithm for H.264/advanced video coding (AVC) of video sequences with high embedding capacity. In the proposed scheme, to embed secret data into the quantized discrete cosine transform (QDCT) coefficients of *I* frames without any intraframe distortion drift, some embeddable coefficient pairs are selected in each block, and they are divided into two different groups, i.e., the embedding group and the averting group. The embedding group is used to carry the secret data, and the averting group is used to prevent distortion drift in the adjacent blocks. The experimental results show that the proposed scheme can avoid intraframe distortion drift and guarantee low distortion of video sequences. In addition, the proposed scheme provides enhanced embedding capacity compared to previous schemes. Moreover, the embedded secret data can be extracted completely without the requirement of the original secret data.

1. Introduction

With the rapid development of multimedia and network technology, a huge amount of digital media, i.e., images, video, audio, and texts, is transmitted each second in the public network, such as the Internet. Such transmitted media are easily modified or illegally copied by malicious attackers. As a result, the security of private information has become a very important issue. Therefore, many solutions have been proposed in the literature, and they can be divided into two different categories, i.e., encryption and data hiding techniques. Since the encryption technique converts the media into a meaningless form that will emphasize the importance of the media's content, this technique can actually result in attracting the attention of attackers. Data hiding is one promising technique to protect the security and privacy

of the digital media because it embeds the secret data into the cover media. The embedded media that contain the secret data have a meaningful form, which helps avoid attracting the attention of attackers and can guarantee the security and privacy of the secret data.

Many data hiding schemes [1–7] have been proposed for different digital data in the last decade. After Richardson introduced the H.264/AVC video compression standard [8] in 2003, this standard has been used extensively for hiding secret data [9–19]. In 2005, Noorkami et al. [9] proposed a low complexity watermarking algorithm by using the relative change of the DC coefficients of the 4×4 block. In their scheme, a public key is used for determining the embedded data, while the copyright owner possesses a secret key. In 2006, Nguyen et al. [10] proposed a fast watermarking system based on H.264/AVC motion vectors. However, their scheme

offered low embedding capacity; i.e., the average embedding capacity was only about 2,000 bits. Then, to achieve robustness, Zhang et al. [11] proposed a new data hiding scheme with a 2D, 8-bit watermark in the compressed domain. Noorkami and Mersereau [12] introduced a framework for robust watermarking of H.264 encoded video by using the quantized AC coefficient to obtain optimal detection of video watermarking. By using a texture masking-based perceptual model, Gong et al. [13] proposed a fast and robust watermarking scheme for H.264 video. In this scheme, the quantized DC coefficients were used to conceal the watermark. However, in these two schemes [12, 13], the original watermark is required for detection. In 2007, Kapotas et al. [15] proposed blind data hiding in an H.264 stream by using the difference of block sizes during the interframe prediction stage to increase the embedding capacity. However, Kapotas et al.'s scheme had a high bit-rate increment. To solve this issue, Kim et al. [16] embedded the watermark bit into the sign bit of the trailing ones in Context Adaptive Variable Length Coding (CAVLC). However, embedding the watermark in the discrete cosine transform (DCT) coefficients of the I-frames in these previous schemes result in stego videos that have low visual quality, which is caused by the distortion drift in the I-frame prediction. To overcome this shortcoming, in 2010, Ma et al. [17] proposed a new algorithm for data hiding in H.264/AVC based on DCT coefficients. Their scheme used several paired-coefficients of a 4×4 macroblock to embed the secret data to avoid distortion drift. However, this scheme obtained limited visual quality. To improve the visual quality of stego videos, Huo et al. [14] proposed a new data hiding scheme by using the controllable error drift-elimination technique. However, unsatisfactory embedding capacity was obtained by their scheme. To increase the embedding capacity of Ma et al.'s scheme, Lin et al. [18] fully used the remaining luminance blocks to hide the secret data. Their experimental results indicated that their embedding capacity was improved further when the increase of embedding capacity obtained by Lin et al.'s scheme is 0.15 bits per pixel (bpp) more than that of Ma et al.'s scheme. However, the average embedding capacity of these two schemes [17, 18] is still low, i.e., always smaller than 0.68 bpp, because they use a pair of DCT coefficients to hide the secret bit separately. By doing so, as the amount of embedded secret data increases, the the distortion of the video becomes greater. To overcome these shortcomings, we propose in this paper a new data hiding scheme for video sequences without intraframe distortion drift. Instead of using the coefficient pair separately for embedding data, all embeddable coefficient pairs in each luminance block are determined and classified into two different clusters, i.e., the embedding group and the averting group. The secret data are hidden in the embedding group by minimum modification, while the averting group is used to avoid distortion drift. The experimental results showed that the proposed algorithm further improved the embedding capacity while maintaining good visual quality and no distortion drift.

The remainder of this paper is organized as follows. Section 2 provides information about intraframe prediction and introduces the previous data hiding schemes. The proposed scheme is explained in Section 3. In Section 4,

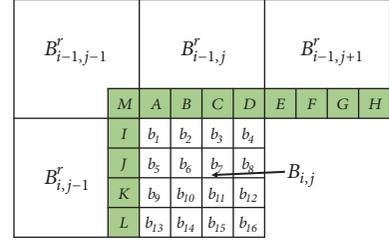


FIGURE 1: Predicted pixels in block, $B_{i,j}$, and the reference pixels in the adjacent region.

experimental results are presented that illustrate the performance of the proposed scheme in comparison with some previous schemes. Our conclusion is presented in Section 5.

2. Related Work

2.1. Intraframe Prediction. Intraframe prediction is a technique in H.264/AVC coding [7] that is used to reduce the spatial redundancies of H.264/AVC intraframes. In H.264/AVC coding, for each block, some previously encoded adjacent blocks are used to predict the pixels of the current block. Figure 1 shows the current 4×4 block, $B_{i,j}$, with its pixels labeled from b_1 to b_{16} . These pixels are predicted based on the reference pixels (labeled from A to M) of four adjacent blocks. These four blocks were encoded previously by using a prediction formula corresponding to the selected optimal prediction mode from nine prediction modes of each 4×4 block in Figure 2.

In H.264/AVC encoding, the current block, $B_{i,j}$, is subtracted from its prediction block, $P_{i,j}$, to obtain the residual block, $R_{i,j} = B_{i,j} - P_{i,j}$. Then, the residual block, $R_{i,j}$, is encoded further by the following processes of H.264/AVC coding, i.e., transformation, quantization, and entropy coding. For simplicity, 4×4 integer DCT transformation and quantization processes are implemented on $R_{i,j}$ to generate the corresponding quantized DCT coefficient matrix $R_{i,j}^q$ as follows:

$$R_{i,j}^q = (C_f R_{i,j} C_f^T) \times \frac{PF}{Q} = \begin{bmatrix} Y_{00} & Y_{01} & Y_{02} & Y_{03} \\ Y_{10} & Y_{11} & Y_{12} & Y_{13} \\ Y_{20} & Y_{21} & Y_{22} & Y_{23} \\ Y_{30} & Y_{31} & Y_{32} & Y_{33} \end{bmatrix}, \quad (1)$$

where $C_f = \begin{bmatrix} \frac{1}{2} & \frac{1}{2} & \frac{1}{2} & \frac{1}{2} \\ \frac{1}{2} & \frac{1}{2} & -\frac{1}{2} & -\frac{1}{2} \\ \frac{1}{2} & -\frac{1}{2} & \frac{1}{2} & -\frac{1}{2} \\ \frac{1}{2} & -\frac{1}{2} & -\frac{1}{2} & \frac{1}{2} \end{bmatrix}$, $PF = \begin{bmatrix} a^2 & ab/2 & a^2 & ab/2 \\ ab/2 & b^2/4 & ab/2 & b^2/4 \\ a^2 & ab/2 & a^2 & ab/2 \\ ab/2 & b^2/4 & ab/2 & b^2/4 \end{bmatrix}$, $a = 1/2$, $b = \sqrt{2/5}$, and Q is the size of the quantization step which is defined by quantization parameter (QP).

After H.264/AVC coding, the block $B_{i,j}$ is represented by the prediction block $P_{i,j}$ and the quantized DCT coefficients $R_{i,j}^q$. To encode the next block, $B_{i,j+1}$, the encoded block, $B_{i,j}$, should be decompressed from $P_{i,j}$ and $R_{i,j}^q$. The dequantization and inverse DCT operations are implemented on $R_{i,j}^q$ to reconstruct the values of $R_{i,j}$, denoted as $R_{i,j}^r$, and the

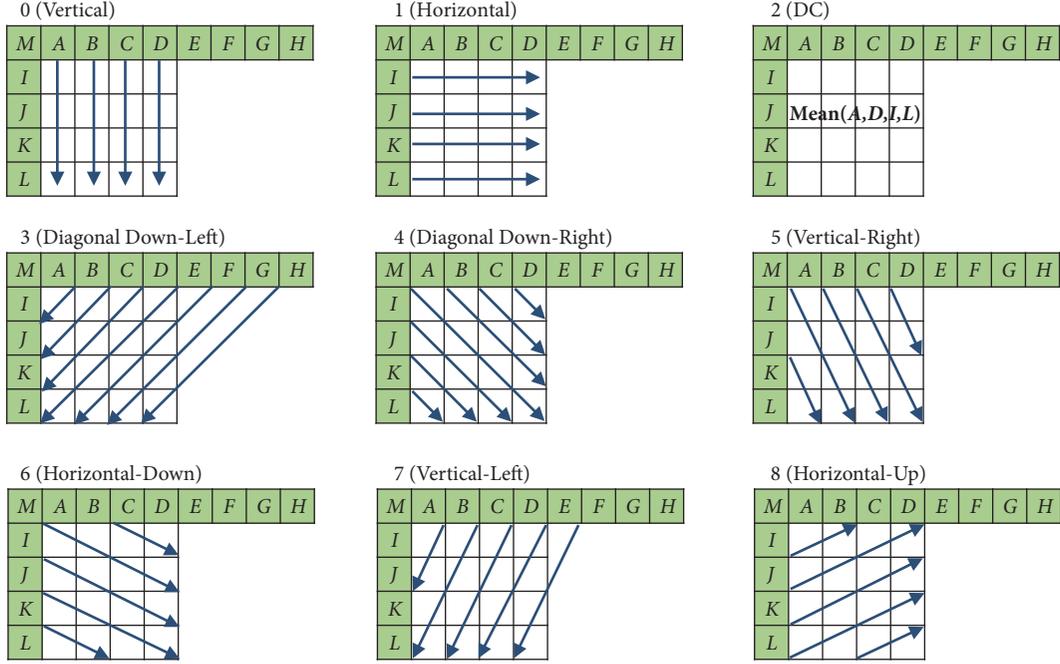


FIGURE 2: Nine prediction modes.

reconstructed block, denoted as $B_{i,j}^r$, is calculated as $B_{i,j}^r = P_{i,j} + R_{i,j}^r$.

In the decoding phase, the residual block, $R_{i,j}^r$, is reconstructed using the dequantization process and the 4×4 integer inverse DCT transformation on $R_{i,j}^q$ by (2), and the reconstructed block is obtained as $B_{i,j}^r = P_{i,j} + R_{i,j}^r$.

$$R_{i,j}^r = C_r^T (R_{i,j}^q \times Q \times PF) C_r, \quad (2)$$

$$\text{where } C_f = \begin{bmatrix} 1 & 1 & 1/2 & 1 \\ 1 & 1/2 & -1/2 & -1 \\ 1 & -1 & -1 & 1 \\ 1/2 & -1 & 1 & -1/2 \end{bmatrix}.$$

2.2. Data Hiding Schemes Based on Intraframe Prediction. In 2010, Ma et al. [17] used three pairs of quantized DCT coefficients in $R_{i,j}^q$ for embedding data into H.264/AVC video sequences. To prevent intraframe distortion drift during the embedding process, they analyzed the use of seven pixels, i.e., $b_4^r, b_8^r, b_{12}^r, b_{13}^r, b_{14}^r, b_{15}^r, b_{16}^r$, in the reconstructed block, $B_{i,j}^r$, for the intraprediction process. Figure 3 shows the four adjacent blocks, i.e., $B_{i,j+1}, B_{i+1,j-1}, B_{i+1,j}$, and $B_{i+1,j+1}$, that are affected directly by the above process. For example, if the selected prediction mode of $B_{i+1,j}$ is 0, as shown in Figure 2, then $b_{13}^r, b_{14}^r, b_{15}^r, b_{16}^r$ are modified by embedding data into some coefficients of $R_{i,j}^q$.

During the embedding process, the seven pixels, $b_4^r, b_8^r, b_{12}^r, b_{13}^r, b_{14}^r, b_{15}^r, b_{16}^r$, and the selected prediction modes of the four adjacent blocks are classified into three different conditions, i.e., Con₁, Con₂, and Con₃, which are defined as follows. Con₁ consists of the prediction modes in {1, 2, 4, 5, 6, 8}. This means the pixels located at the position of b_4^r, b_8^r, b_{12}^r , and, b_{16}^r are referenced for predicting $B_{i,j+1}$. Con₂ consists of the prediction modes in {0, 2, 3, 4, 5, 6, 7}. This indicates that

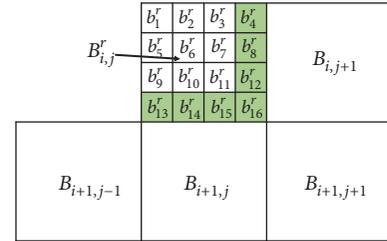


FIGURE 3: The current block and the four blocks directly affected by the encoding process.

the pixels located at the positions of $b_{13}^r, b_{14}^r, b_{15}^r$, and b_{16}^r are referenced for predicting $B_{i+1,j}$ or $B_{i+1,j-1}$. Con₃ consists of the prediction modes in {4, 5, 6}. This means that the pixel located at the position of b_{16}^r is referenced for predicting $B_{i+1,j+1}$.

To take advantage of the relationship of the reference pixels and the selected prediction modes for embedding data, these three conditions also can be classified into the five categories presented in Table 1.

To solve the drift distortion problem during embedding the secret data, Ma et al. classified the current block into three different conditions, i.e., Con₁, Con₂, and Con₃. Then three specific pairs of quantized DCT coefficients are selected for embedding three secret data bits. Take the category Con₂ as an example, three coefficient pairs, i.e., $\{(Y_{01}, Y_{21}), (Y_{02}, Y_{22}), (Y_{03}, Y_{23})\}$, are used for embedding. The main reason is that these three pairs have the same property; i.e., when the values of quantized DCT coefficients are modified in a pair, the modification will be concentrated only on the two middle columns or the two middle rows of the block.

TABLE 1: Five general categories of reference pixels and selected prediction modes.

	Con ₁	Con ₂	Con ₃	Reference pixels
Cat ₁	T	F	X	$b'_4, b'_8, b'_{12}, b'_{16}$
Cat ₂	F	T	X	$b'_{13}, b'_{14}, b'_{15}, b'_{16}$
Cat ₃	F	F	F	Nrp
Cat ₄	F	F	T	b'_{16}
Cat ₅	T	T	X	All

T: true, F: false, X: do not care, Nrp: not reference pixel, and all: all seven reference pixels.

For example, consider the quantized DCT coefficient pair (Y_{03}, Y_{23}) of $R_{i,j}^q$. Assume that Y_{03} is added by the value of

ν to embed the secret data, i.e., $\Delta = \begin{bmatrix} 0 & 0 & 0 & \nu \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$. Then, the modification will be spread to all pixels of the 4×4 block

as $D = \begin{bmatrix} abtQ/4 & -abtQ/2 & abtQ/2 & -abtQ/4 \\ abtQ/4 & -abtQ/2 & abtQ/2 & -abtQ/4 \\ abtQ/4 & -abtQ/2 & abtQ/2 & -abtQ/4 \\ abtQ/4 & -abtQ/2 & abtQ/2 & -abtQ/4 \end{bmatrix}$, which will propagate

to other adjacent blocks. However, in the Ma et al.'s scheme, to embed a hidden bit ν in (Y_{03}, Y_{23}) , the quantized DCT coefficient pair (Y_{03}, Y_{23}) is perturbed to $(Y_{03} + \nu, Y_{23} - \nu)$,

i.e., $\Delta = \begin{bmatrix} 0 & 0 & 0 & \nu \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$. Then, the modification will be $D =$

$$\begin{bmatrix} 0 & 0 & 0 & 0 \\ abtQ/2 & -abtQ & abtQ & -abtQ/2 \\ abtQ/4 & -abtQ & abtQ & -abtQ/2 \\ 0 & 0 & 0 & 0 \end{bmatrix}.$$

In this scenario, it is clear that the modification is concentrated only on the two middle rows, while the modifications in the first and the last rows are zeros. Therefore, the distortion drift is avoided. However, they did not fully explore all of the cases for embedding data, so their average embedding rate was less than 0.45 bpp.

To further improve the embedding capacity of Ma et al.'s scheme while avoiding the distortion drift in the H.264/AVC video sequences, Lin et al. [18] have divided the relationship of the reference pixels and the selected prediction modes into five categories presented in Table 1. Then, for the block belongs to three categories, i.e., Cat₁, Cat₂, and Cat₄, Lin et al. extracted one more coefficient pair for embedding one more secret bit by the same way as was done by Ma et al.'s scheme. In addition, each block belonging to Cat₅ is also used for embedding one more secret bit to increase embedding capacity. As a result, the embedding capacity obtained by Lin et al.'s scheme is 0.15 bits per pixel (bpp) higher than that of Ma et al.'s scheme. However, in these two schemes [17, 18], each pair of quantized DCT coefficients is subsequently perturbed to embed only one secret bit. Thus, to embed n secret bits, n selected pairs of quantized DCT coefficients are modified. This means that the more secret bits are embedded, the more distortion will cause in the video frames, leading to low visual quality of the video frames. In the schemes [17, 18], to guarantee the higher visual quality, if (Y_{mn}, Y_{pq}) is a zero coefficient pair, it is not used to embed any secret data bits. Therefore, their embedding capacity is still low when the average embedding capacity is smaller than 0.68 bpp when QP = 28 is used.

It is obvious that, to maintain the high visual quality of video sequence, the previous schemes [17, 18] have selected quantized DCT coefficient pairs (excluding the zero coefficient pairs) of three categories, Cat₁, Cat₂, and Cat₄ for embedding data. However, their schemes still obtained low embedding capacity, while the visual quality of video sequences is not guaranteed. Therefore, in this paper, to overcome their shortcoming, instead of modifying each coefficient pair for embedding data, the group of coefficients are selected and altered at the same time. In particular, all suitable pairs of quantized DCT coefficients are extracted and classified into two different groups, one group is used for embedding data and the other one is used to prevent distortion drift of video sequences. This means that, in the proposed scheme, the group of n coefficient pairs is modified to embed n secret bits. By modifying by the group, at most $n/2$ coefficient pairs are modified which guarantees the better visual quality of the embedding video sequences. In addition, to increase embedding capacity, in the proposed scheme, zero coefficient pairs are still used for embedding data. The details of the proposed scheme are described in the next section.

3. The Proposed Scheme

Figure 4 shows all of the main processes of the proposed embedding phase and extracting phase. In the embedding phase, the original H.264/AVC video sequence is first decoded by entropy coding. Then, the 4×4 quantized DCT blocks meet three cases, i.e., Cat1, Cat2, and Cat4; four secret data bits are embedded by based on group modulation. And if the blocks belong to the category Cat3, each coefficient is used to contain one secret bit. Then, all the quantized DCT coefficients are entropy encoded to get the embedded H.264/AVC video sequences. In this phase, to prevent the distortion drift in the proposed scheme, quantized DCT coefficients of each block are partitioned into two groups, i.e., the embedding group and the averting group. Then, to achieve high embedding capacity and to ensure good image quality, the embedding group is used for embedding data, while the averting group is used for preventing the proliferation of errors. Figure 4(b) shows the detail of the extracting phase. The embedded H.264/AVC video sequence is entropy decoded. Then, the category of the 4×4 quantized DCT blocks is determined. After that according to the determined category, the corresponding secret bits are extracted.

3.1. Category Selection and Coefficient Grouping. To prevent intraframe distortion drift in the proposed scheme, all blocks that belong to the first four categories are selected for embedding data by suitable ways. Therefore, the pixels that are used for intraframe prediction are not used during the embedding process so that the embedding distortion would not affect the other adjacent blocks. Figure 5 shows the percentage of the blocks which meet the conditions of the first four categories of the 14 H.264/AVC test video sequences. It is obvious that most of the blocks in each video sequence belong to Cat₁ and Cat₂. Therefore, the proposed scheme is designed to embed more secret bits into these two categories with small distortion and without distortion drift.

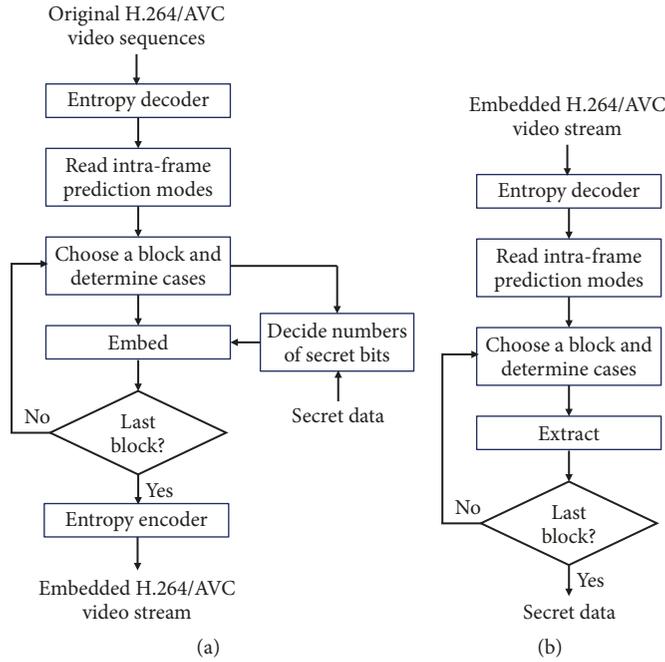


FIGURE 4: Main processes of the proposed scheme: (a) embedding phase and (b) extracting phase.

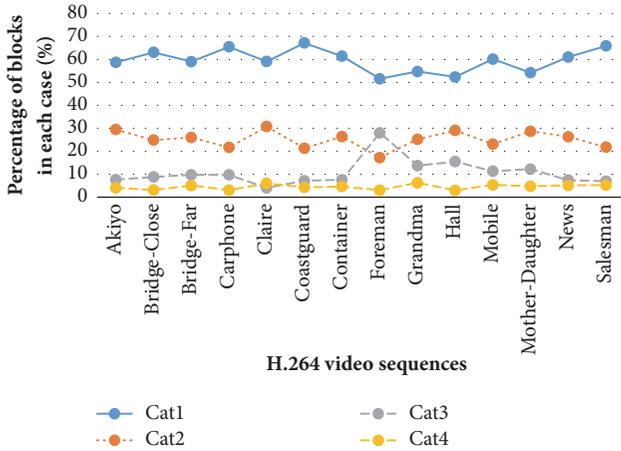


FIGURE 5: Percentage of blocks that meet the first four categories.

In the proposed scheme, four quantized DCT coefficient pairs of these three categories are also selected and divided into two groups, i.e., an embedding group E and an averting group A , to conceal the secret data. Take the block B which belongs to category Cat_1 as an example, four coefficient pairs, $(Y_{li}, Y_{ki}) \in \{(Y_{00}, Y_{02}), (Y_{10}, Y_{12}), (Y_{20}, Y_{22}), (Y_{30}, Y_{32})\}$, are selected for embedding data. Then, all of the first coefficients Y_{li} of each pair are grouped to construct the embedding group, E , which is used to carry the secret bits by modifying, at most, two coefficients, whereas the remaining coefficients Y_{ki} of four pairs are clustered into the averting group A , which is modified to prevent intraframe distortion drift during embedding process. Figure 6 shows an example of the grouping process of Cat_1 .

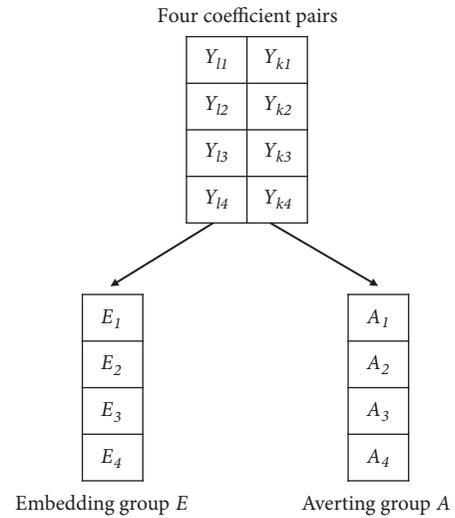


FIGURE 6: Example of coefficient grouping.

3.2. Embedding Phase. In this subsection, the embedding algorithm is described in detail. First, for security reasons, the secret data are encrypted in advance to $S = s_1, s_2, \dots, s_{|S|}$, and $s_r \in \{0, 1\}$. Original video sequences are decoded by entropy decoder to get the intraframe prediction modes and quantized DCT coefficients. Then, if the 4×4 quantized DCT blocks belong to Cat_1, Cat_2, Cat_4 , the secret data are embedded by based on group modulation. Otherwise, if the blocks belong to the category Cat_3 , each coefficient is used to contain one secret bit. Then, all the quantized DCT coefficients are entropy encoded to get the target embedded

video sequences. To make the embedding algorithm clearer, two cases are used for embedding data as an example:

Case 1. If the blocks belong to Cat_1 , Cat_2 , and Cat_4 , the four steps are implemented for embedding n secret bits as follows.

Step 1. Read four appropriate pair-coefficients and classify them into two different groups, i.e., E and A , as was done in Section 3.1.

Step 2. Read n secret bits, s_1, s_2, \dots, s_n , from the encrypted message and embed them into group E . For embedding, the weight value W of group E is calculated by (3). In this paper, four coefficient pairs are used for embedding; therefore, the value of n could be set to at most 4.

$$W = \left(\sum_{i=1}^n E_i \times i \right) \bmod (2^n) \quad (3)$$

And, the difference value, d , is calculated by

$$d = dec(s_1 s_2 \dots s_n) - W, \quad (4)$$

where $dec(s_1 s_2 \dots s_n)$ is the decimal value of n secret bits, $s_1 s_2 \dots s_n$.

Step 3. If the value of d is 0, all elements of group E are kept unchanged. Otherwise, we can arbitrarily increase or decrease the elements of group E by 1. If E_i increases by 1, the values of W will be increased by 1 to 4. Whereas, if E_i decreases by 1, the values of W will be decreased by $2^n - 4$ to $2^n - 1$. It can be observed that, in the proposed scheme, at most, two elements in group E are modified by the value 1. Therefore, we can alter E to E' by changing; at most, two elements in E to satisfy $W' \equiv dec(s_1 s_2 \dots s_n) \bmod (2^n)$.

Step 4. Preventing the drift of intraframe distortion, the inverse operations of modifying the group E are performed on the corresponding elements of the group A . For example, if the element E_i increases by 1, the element A_i will decrease by 1 and vice versa.

Similarly, when the block $B_{i,j}$ satisfies one of two categories, i.e., Cat_2 and Cat_4 , four coefficient pairs, $(Y_{ij}, Y_{ki}) \in \{(Y_{00}, Y_{20}), (Y_{01}, Y_{21}), (Y_{02}, Y_{22}), \text{ and } (Y_{03}, Y_{23})\}$, are selected to generate the embedding and averting groups, E and A ; then n secret bits are embedded into the group by the same manner as was done for Cat_1 .

Case 2 (the block belongs to category Cat_3). In such blocks, each coefficient is used to embed one secret bit because this category does not have any effect on other adjacent blocks during the encoding process. Then, the stego coefficient, Y'_{ij} , is calculated by

$$Y'_{ij} = \begin{cases} Y_{ij} + 1 & \text{if } (Y_{ij} \text{ is even, } s_r = 1, \text{ and } Y_{ij} \geq 0) \\ & \text{or } (Y_{ij} \text{ is odd, } s_r = 0, \text{ and } Y_{ij} \geq 0) \\ Y_{ij} - 1 & \text{if } (Y_{ij} \text{ is even } s_r = 1, \text{ and } Y_{ij} < 0) \\ & \text{or } (Y_{ij} \text{ is odd, } s_r = 0, \text{ and } Y_{ij} < 0) \\ Y_{ij} & \text{otherwise} \end{cases} \quad (5)$$

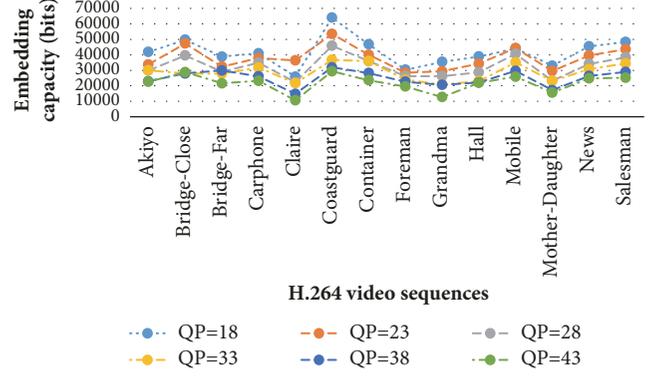


FIGURE 7: Embedding capacity of the proposed scheme for different QPs.

where Y_{ij} is one of 16 quantized DCT coefficients in $R_{i,j}^q$. Here, if the block $B_{i,j}$ belongs to Cat_5 , we leave it without embedding any secret bits.

3.3. Extracting Phase. In this subsection, the extracting algorithm is used to extract the secret data from the embedded H.264/AVC video sequences. If the current block belongs to Cat_1 , Cat_2 , or Cat_4 , the embedding group E will be reconstructed as was done in the embedding phase. Then, n embedded bits are extracted by

$$(s_1 s_2 \dots s_n) = bin \left(\left(\sum_{i=1}^n E_i \times i \right) \bmod (2^n) \right). \quad (6)$$

For the block that belongs to Cat_3 , each coefficient is checked to determine the embedded bit, s_r , by using

$$s_r = \begin{cases} 1 & \text{if } Y_{ij} \text{ is odd} \\ 0 & \text{otherwise.} \end{cases} \quad (7)$$

4. Experimental Results

In this section, we describe the experimental evaluation of the performance of the proposed scheme. Fourteen video sequences, i.e., Akiyo, Bridge-Close, Bridge-Far, Carphone, Claire, Coastguard, Container, Foreman, Grandma, Hall, Mobile, Mother-Daughter, News, and Salesman, were used as test samples. The size 30 of the group of pictures (GOP) and the structure of "IBBPB" were used in the experiment. Six different quantization steps (QP), i.e., 18, 23, 28, 33, 38, and 43, were checked for the 14 video sequences mentioned above. In principle, in H.264/AVC coding, if a small value of QP is used, the better visual quality of video sequences is obtained and the more encoded bits are required.

4.1. Embedding Capacity Evaluation. Figure 7 shows the performance of the proposed scheme, in terms of embedding capacity, using six QPs. It is apparent that using a smaller value of QP resulted in a higher embedding capacity.

Table 2 compares the embedding capacity of the proposed scheme and two state-of-the-art schemes [17, 18]. As shown

TABLE 2: Comparison of embedding capacity of the proposed scheme and two previous schemes [17, 18] in terms of embedded bits per 4×4 block.

QP	Ma et al.	Lin et al.	Proposed	Improvement rate	
				Ma et al.	Lin et al.
18	0.97	1.32	1.46	51%	11%
23	0.69	0.94	1.33	93%	41%
28	0.49	0.67	1.14	133%	70%
33	0.24	0.31	1.00	317%	223%
38	0.10	0.13	0.87	770%	569%
43	0.03	0.04	0.76	2433%	1800%
Average	0.42	0.57	1.09	160%	91%

TABLE 3: Embedding capacity comparison of the proposed scheme and two previous schemes [17, 18] for QP = 28.

Sequences	Ma et al.	Lin et al.	Proposed	Improvement rate	
				Ma et al.	Lin et al.
Akiyo	0.36	0.47	1.04	189%	121%
Bridge-Close	0.58	0.76	1.39	140%	83%
Bridge-Far	0.14	0.20	1.02	629%	410%
Carphone	0.44	0.59	1.22	177%	107%
Claire	0.19	0.26	0.78	311%	200%
Coastguard	1.00	1.27	1.61	61%	27%
Container	0.49	0.65	1.26	157%	94%
Foreman	0.38	0.56	0.92	142%	64%
Grandma	0.30	0.43	0.92	207%	114%
Hall	0.55	0.74	1.05	91%	42%
Mobile	1.16	1.27	1.42	22%	12%
Mother-Daughter	0.27	0.36	0.81	200%	125%
News	0.56	0.74	1.19	113%	61%
Salesman	0.55	0.76	1.34	144%	76%
Average	0.49	0.67	1.14	133%	70%

in the table, the average embedding capacity of the proposed scheme was considerably higher than those of the two state-of-the-art schemes. The average improvement in embedding capacity of our proposed scheme over the schemes of Ma et al. [17] and Lin et al. [18] were 160 and 91%, respectively. The main reason for this improvement over Ma et al.'s scheme was that they only selected three coefficient pairs for carrying secret bits, which resulted in low embedding capacity. Lin et al.'s scheme also had a lower embedding capacity than the proposed scheme because they did not use coefficient pairs with value of zero for embedding data to avoid significant distortion in the video intraframes. In our proposed scheme, all of the blocks that belonged to the first four categories were used to embed secret data with a small modification. As a result, the proposed scheme achieved higher embedding capacity. Specifically, Table 3 shows the embedding capacity of the three schemes for 14 test video sequences when the value of QP was 28. As Table 3 shows, the gain in embedding capacity of the proposed scheme ranged from 22 to 629% better than that of Ma et al.'s scheme and from 12 to 410% better than that of Lin et al.'s scheme. The value of the improvement rate was different for the 14 video sequences

TABLE 4: Comparison of bit rate increment ratio of the proposed scheme and two previous schemes [17, 18].

QP	Ma et al.	Lin et al.	Proposed
18	0.24%	0.65%	0.64%
23	0.82%	1.22%	1.12%
28	1.71%	2.06%	1.78%
33	2.29%	2.53%	2.71%
38	2.19%	2.28%	2.96%
43	1.39%	1.33%	2.87%
Average	1.44%	1.68%	2.01%

because the selected prediction modes of blocks were based on the content of each video sequence.

4.2. Bit-Rate Increment Ratio. Table 4 shows the ratio of the bit-rate increment of different QPs. The average ratios of the bit-rate increment of Ma et al.'s scheme, Lin et al.'s scheme, and the proposed scheme were 1.44%, 1.68%, and 2.01%, respectively. These results indicated that the degradation was quite small in all three schemes.

TABLE 5: Comparison of visual quality (PSNR: dB) of the proposed scheme and two previous schemes [17, 18].

QP	Ma et al.	Lin et al.	Proposed
18	42.66	42.21	42.32
23	39.22	38.74	38.21
28	35.71	35.21	35.26
33	32.94	32.65	31.57
38	30.07	29.91	29.27
43	27.53	27.49	25.68
Average	34.69	34.37	33.71

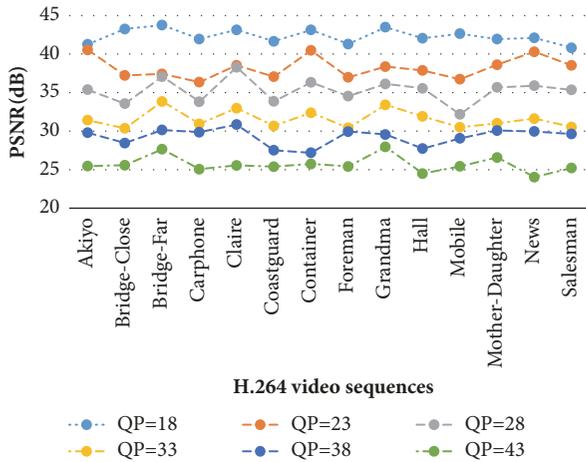


FIGURE 8: Visual quality of the proposed scheme for different QPs.

4.3. Visual Quality Evaluation. To evaluate the visual quality of video frames in the three schemes, the peak signal-to-noise ratio (PSNR) [20] is calculated by comparing the original frame to the embedded frame. Figure 8 shows the performance of the proposed scheme in the visual quality with different value of QP. It clear to see that the higher PSNR is obtained when the smaller QP is used.

Table 5 shows comparison results of the proposed scheme, Ma et al.'s scheme, and Lin et al.'s scheme in terms of visual quality of the video frames. It can be observed that the PSNR of the proposed scheme was slightly smaller than that of other two schemes [17, 18]. However, the proposed scheme can improve embedding capacity significantly; i.e., the average improvement rates were 160 and 91% over Ma et al.'s scheme and Lin et al.'s scheme, respectively. Table 6 compares the visual quality of the three schemes for QP = 28, corresponding to the embedding capacity in Table 3. Compared with Ma et al.'s scheme, the average degradation of the proposed scheme was larger than 0.45 dB. However, the proposed scheme provided better visual quality of video frames than Lin et al.'s scheme.

For a fair comparison, both values of PSNR and the structural similarity (SSIM) [21] index are used to evaluate the visual quality of the proposed scheme and two other schemes [17, 18]. Here, the max embedding capacity obtained by Lin et al.'s scheme [17] is embedded into the proposed scheme.

TABLE 6: Comparison of visual quality (PSNR: dB) of the proposed scheme and two previous schemes [17, 18] for QP = 28.

Sequences	Ma et al.	Lin et al.	Proposed
Akiyo	37.32	36.77	35.38
Bridge-Close	34.01	33.55	33.57
Bridge-Far	37.58	37.20	37.08
Carphone	36.05	35.57	33.82
Claire	39.48	39.02	38.30
Coastguard	33.45	32.96	33.86
Container	35.25	34.72	36.33
Foreman	35.73	35.19	34.54
Grandma	36.17	35.67	36.13
Hall	35.70	35.04	35.56
Mobile	32.35	31.67	32.18
Mother-Daughter	36.47	36.41	35.68
News	35.74	35.12	35.89
Salesman	34.67	34.08	35.35
Average	35.71	35.21	35.26

Table 7, Figures 9 and 10 show that the proposed scheme successfully preserved the high visual quality of the video sequences. The average PSNR of the proposed scheme was better than those of Ma et al.'s and Lin et al.'s schemes, at 3.39 and 3.89 dB, respectively. As can be seen in Figure 9, the PSNR obtained by the proposed scheme is always better than that of two previous schemes [17, 18]. In terms of SSIM, in Figure 10, the proposed schemes showed better performance in some video sequences, i.e., Bridge-Close, Bridge-Far, and Coastguard, while two previous schemes [17, 18] obtained higher values of SSIM in some other video sequences, i.e., Carphone and Claire. However, the SSIM obtained by the proposed scheme also was higher than those of the two previous schemes [17, 18]. It can be concluded that, based on coefficient grouping for embedding data, the proposed scheme prevented intraframe distortion drift and improved embedding capacity further while maintaining good visual quality of the embedded video sequences. In addition, Figure 11 shows the visual effect in video sequences by three schemes. It is obvious that three schemes reveal the same visual quality observation.

5. Conclusion

In this paper, a high-quality data hiding algorithm based on H.264/AVC is proposed without intraframe distortion drift. In the proposed scheme, the quantized coefficients are clustered into two different groups, i.e., the embedding group and the averting group. Then, the embedding group is used to carry secret bits to preserve high visual quality and further improve embedding capacity. In addition, the averting group is used to avoid the intraframe distortion drift. The experimental results demonstrated that the embedding capacity increased significantly in the proposed scheme while guaranteeing the good visual quality of embedded video sequences.

TABLE 7: Comparison of visual quality (PSNR and SSIM) of the proposed scheme and two previous schemes [17, 18] for QP = 28.

Sequences	Capacity	Ma et al.		Capacity	Lin et al.		Proposed	
		PSNR	SSIM		PSNR	SSIM	PSNR	SSIM
Akiyo	0.36	37.32	0.990	0.47	36.77	0.989	41.41	0.990
Bridge-Close	0.58	34.01	0.965	0.76	33.55	0.963	37.18	0.982
Bridge-Far	0.14	37.58	0.965	0.20	37.20	0.964	47.89	0.977
Carphone	0.44	36.05	0.992	0.59	35.57	0.991	36.44	0.983
Claire	0.19	39.48	0.994	0.26	39.02	0.994	44.21	0.985
Coastguard	1.00	33.45	0.935	1.27	32.96	0.931	35.40	0.975
Container	0.49	35.25	0.958	0.65	34.72	0.957	40.28	0.982
Foreman	0.38	35.73	0.989	0.56	35.19	0.987	35.37	0.980
Grandma	0.30	36.17	0.982	0.43	35.67	0.980	37.74	0.974
Hall	0.55	35.70	0.990	0.74	35.04	0.988	37.19	0.975
Mobile	1.16	32.35	0.981	1.27	31.67	0.979	33.91	0.991
Mother-Daughter	0.27	36.47	0.985	0.36	36.41	0.985	43.07	0.982
News	0.56	35.74	0.988	0.74	35.12	0.987	39.37	0.976
Salesman	0.55	34.67	0.976	0.76	34.08	0.974	37.96	0.986
Average	0.49	35.71	0.978	0.65	35.21	0.976	39.10	0.981

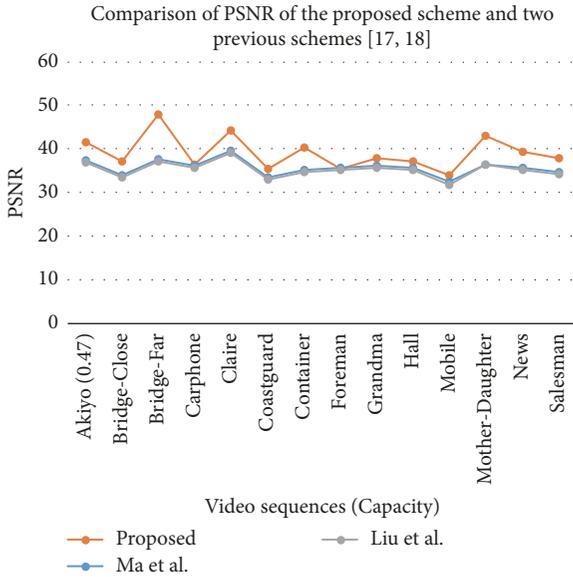


FIGURE 9: Visual quality (PSNR) performances of the proposed scheme and two previous schemes [17, 18] with different video sequences.

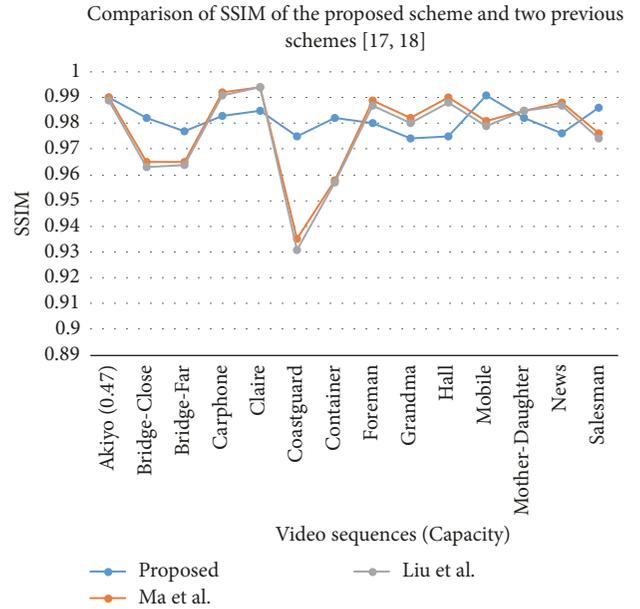


FIGURE 10: Visual quality (SSIM) performances of the proposed scheme and two previous schemes [17, 18] with different video sequences.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This research is funded by Vietnam National Foundation for Science and Technology Development (NAFOSTED) under

Grant no. 102.01-2016.06. The authors also would like to thank the Ministry of Science and Technology of the Republic of China for financially supporting this research under Contract nos. MOST 106-2218-E-035-011 and MOST 106-2627-M-035-007.

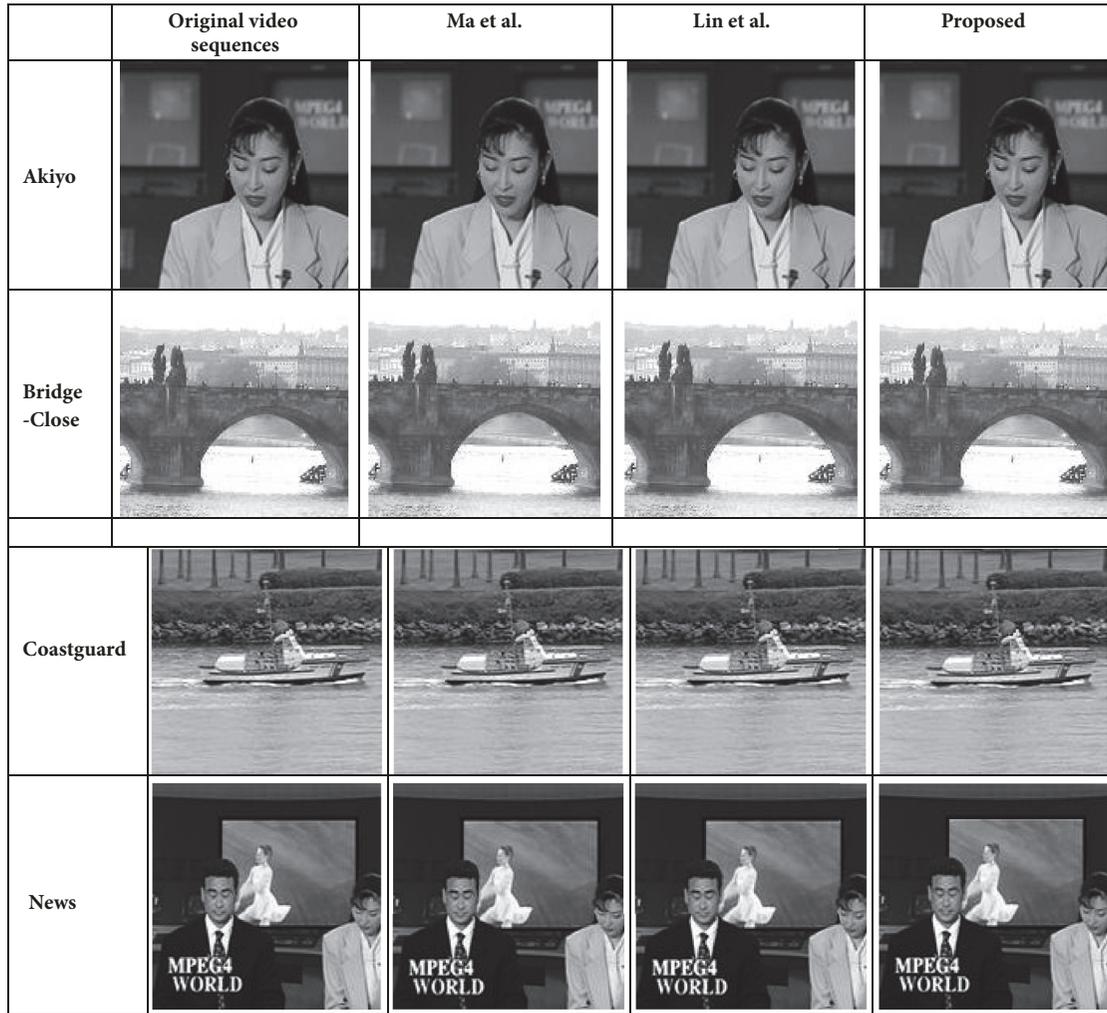


FIGURE II: Visual effect of the resultant intraframes by the proposed scheme and two previous schemes [17, 18].

References

- [1] Y. Liu, X. Qu, and G. Xin, "A ROI-based reversible data hiding scheme in encrypted medical images," *Journal of Visual Communication and Image Representation*, vol. 39, pp. 51–57, 2016.
- [2] W. Zhang, H. Wang, D. Hou, and N. Yu, "Reversible data hiding in encrypted images by reversible image transformation," *IEEE Transactions on Multimedia*, vol. 18, no. 8, pp. 1469–1479, 2016.
- [3] J. Bai, C.-C. Chang, T.-S. Nguyen, C. Zhu, and Y. Liu, "A high payload steganographic algorithm based on edge detection," *Displays*, vol. 46, pp. 42–51, 2017.
- [4] Z. Pan and L. Wang, "Novel reversible data hiding scheme for Two-stage VQ compressed images based on search-order coding," *Journal of Visual Communication and Image Representation*, vol. 50, pp. 186–198, 2018.
- [5] K. Chen and D. Xu, "An Efficient Reversible Data Hiding Scheme for Encrypted Images," *International Journal of Digital Crime and Forensics*, vol. 10, no. 2, pp. 1–22, 2018.
- [6] C.-C. Chang and T.-S. Nguyen, "A reversible data hiding scheme for SMVQ indices," *Informatica (Netherlands)*, vol. 25, no. 4, pp. 523–540, 2014.
- [7] X. P. Zhang, "Separable reversible data hiding in encrypted image," *IEEE Transactions on Information Forensics and Security*, vol. 7, no. 2, pp. 826–832, 2012.
- [8] I. E. G. Richardson, *H.264 and MPEG-4 video compression: video coding for next-generation multimedia*, Wiley, Chichester, U.K, 2003.
- [9] M. Noorkami and R. Mersereau, "Compressed-domain video watermarking for H.264," in *Proceedings of the International Conference on Image Processing*, pp. II–890, Genova, Italy, September 2005.
- [10] C. Nguyen, D. B. Tay, and G. Deng, "A Fast Watermarking System for H.264/AVC Video," in *Proceedings of the APCCAS 2006 - 2006 IEEE Asia Pacific Conference on Circuits and Systems*, pp. 81–84, Singapore, December 2006.
- [11] J. Zhang, A. T. S. Ho, G. Qiu, and P. Marziliano, "Robust video watermarking of H.264/AVC," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 54, no. 2, pp. 205–209, 2007.
- [12] M. Noorkami and R. M. Mersereau, "A framework for robust watermarking of H.264-encoded video with controllable detection performance," *IEEE Transactions on Information Forensics and Security*, vol. 2, no. 1, pp. 14–23, 2007.

- [13] X. Gong and H.-M. Lu, "Towards fast and robust watermarking scheme for H.264 video," in *Proceedings of the 10th IEEE International Symposium on Multimedia (ISM '08)*, pp. 649–653, Berkeley, Calif, USA, December 2008.
- [14] W. Huo, Y. Zhu, and H. Chen, "A controllable error-drift elimination scheme for watermarking algorithm in H.264/AVC stream," *IEEE Signal Processing Letters*, vol. 18, no. 9, pp. 535–538, 2011.
- [15] S. K. Kapotas, E. E. Varsaki, and A. N. Skodras, "Data hiding in H. 264 encoded video sequences," in *Proceedings of the 9th IEEE Workshop on Multimedia Signal Processing (MMSP '07)*, pp. 373–376, October 2007.
- [16] S. M. Kim, S. B. Kim, Y. Hong, and C. S. Won, "Data hiding on H," in *Proceedings of the 4th International Conference on Image Analysis and Recognition*, pp. 698–707, 2007.
- [17] X. Ma, Z. Li, H. Tu, and B. Zhang, "A data hiding algorithm for h.264/AVC video streams without intra-frame distortion drift," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 20, no. 10, pp. 1320–1330, 2010.
- [18] T.-J. Lin, K.-L. Chung, P.-C. Chang, Y.-H. Huang, H.-Y. M. Liao, and C.-Y. Fang, "An improved DCT-based perturbation scheme for high capacity data hiding in H.264/AVC intra frames," *The Journal of Systems and Software*, vol. 86, no. 3, pp. 604–614, 2013.
- [19] Y. Liu, Z. Li, X. Ma, and J. Liu, "A robust without intra-frame distortion drift data hiding algorithm based on H.264/AVC," *Multimedia Tools and Applications*, vol. 72, no. 1, pp. 613–636, 2014.
- [20] VQEG, "Final Report From the Video Quality Experts Group on the Validation of Objective Models of Video Quality Assessment," <http://www.vqeg.org/>.
- [21] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: from error visibility to structural similarity," *IEEE Transactions on Image Processing*, vol. 13, no. 4, pp. 600–612, 2004.



Hindawi

Submit your manuscripts at
www.hindawi.com

