

# REVERSIBLE DATA HIDING BASED ON TWO-DIMENSIONAL HISTOGRAM SHIFTING

Juan Zhao<sup>1</sup> and Zhitang Li<sup>2</sup>

<sup>1</sup>School of Mathematics & Computer Science, Wuhan Polytechnic University, Wuhan 430048, China

<sup>2</sup>Institute of Computer Science and Technology, Huazhong University of Science and Technology, Wuhan 430074, China

## ABSTRACT

*This paper presents a two-dimensional histogram shifting technique for reversible data hiding algorithm. In order to avoid the distortion drift caused by hiding data into stereo H.264 video, we choose arbitrary embeddable blocks from 4×4 quantized discrete cosine transform luminance blocks which will not affect their adjacent blocks. Two coefficients in each embeddable block are chosen as a hiding coefficient pair. The selected coefficient pairs are classified into different sets on the basis of their values. Data could be hidden according to the set which the value of the coefficient pair belongs to. When the value of one coefficient may be changed by adding or subtracting 1, two data bits could be hidden by using the proposed method, whereas only one data bit could be embedded by employing the conventional histogram shifting. Experiments show that this two-dimensional histogram shifting method can be used to improve the hiding performance.*

## KEYWORDS

*Reversible data hiding, Two-dimensional histogram shifting, H.264, Multi-view coding*

## 1. INTRODUCTION

Using data hiding including watermarking and steganography, we embed data into multimedia for covert communication, authentication, annotation, fingerprinting and copyright protection. In some applications such as medical media sharing, military, law enforcement, multimedia file management and remote sensing, the permanent media distortion caused by data hiding is intolerable. Reversible data hiding (RDH) can fully restore the host media after extracting data. Hiding data into the quantized discrete cosine transformation (QDCT) coefficients is the most common practice, but the distortion due to embedded data will spread and accumulate, called distortion drift. By a video RDH algorithm, the value of each QDCT coefficient is restored completely, so important videos without distortion could be treasured and watched. Consequently, there will not be too many network videos with hidden important information so that it is difficult for others to find stego covers. Additionally, some video RDH techniques have been presented for sensitive applications [1] and error concealment [2, 3] recently.

In order to improve the hiding capacity and reduce the distortion, algorithms based on histogram shifting (HS) [2-14], difference expansion [15-18], integer transform [1, 19, 20] and soon have been proposed recently, where HS, difference expansion, and integer transform are three main techniques that could be used for hiding data into H.264 videos.

The peak of the image histogram is used to hide data in the typical HS scheme [6], where each pixel value is changed at most by adding or subtracting 1. When the peak point value is higher, the higher capacity and lower distortion could be obtained. So the difference and the prediction-error HS methods are presented later. Difference image histogram shifting is utilized for multilevel RDH in [8], where a value in a difference image is the difference between two neighboring pixels in an image. Nine-dimensional histogram is applied for prediction strategy in [10], where prediction-error expansion techniques and pixel selection methods are combined with the general framework of HS-based RDH to achieve better hiding performance. In [12], a bidirectional-shift strategy is used to extend the shift able positions in the central zone of the allowable coordinates. QDCT coefficient histogram is similar to difference histogram and prediction-error histogram. So it is an efficient and welcome way to reversibly hide data into QDCT coefficients by using HS in H.264 video RDH schemes [2, 3]. However, in the conventional HS scheme, each coefficient, difference, or prediction error is changed in isolation.

With the rapid development of popular stereo video, traditional RDH technique researchers began to turn their attention to the new cover media. Joint Video Team released multi-view video coding (MVC) standard as an appendix of H.264 standard in 2010. In order to compress efficiently, intra-frame prediction, inter-frame prediction and parallax prediction are used by MVC standard. Hence, embedding information into MVC video may cause visual distortion and serious distortion drift. Therefore, a two-dimensional (2D) HS-based RDH algorithm for MVC video with limited distortion drift is proposed in this paper. Compared with other methods, enhanced 2D HS technique selecting two arbitrary QDCT AC coefficients from every embeddable block for embedding more information with lower distortion.

The rest of the paper is organized as follows. In section 2, the main idea of 2D HS is introduced. Section 3 describes the way of preventing distortion drift for MVC video and depicts the MVC video RDH algorithm based on 2D HS. Experimental results are given in section 4. Last, conclusions are drawn in section 5.

## 2. HS-BASED RDH SCHEME

### 2.1. Conventional HS

One-dimensional (1D) HS proposed by Ni et al. [6] could be used for hiding data into QDCT coefficients of MVC video. The peak of QDCT coefficient histogram is 0. Information is embedded by expansion and shifting as shown in Figure 1 and (1).

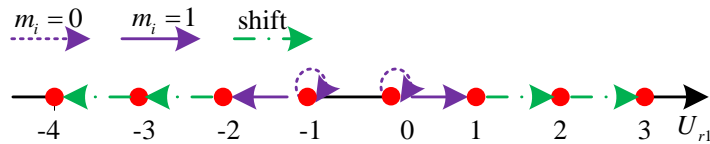


Figure 1. Transformation of QDCT coefficients in conventional 1D HS

$$U'_{r1} = \begin{cases} U_{r1}, & \text{if } (U_{r1}=0 \text{ or } -1) \wedge (m_i=0) \\ 1, & \text{if } (U_{r1}=0) \wedge (m_i=1) \\ -2, & \text{if } (U_{r1}=-1) \wedge (m_i=1) \\ U_{r1} + 1, & \text{if } U_{r1} > 0 \\ U_{r1} - 1, & \text{if } U_{r1} < -1 \end{cases} \quad (1)$$

Where  $U_{r1}$  is a QDCT coefficient,  $U'_{r1}$  is the marked QDCT coefficient after hiding data, and  $m_i \in \{0,1\}$  is a to-be-embedded data bit. When the value of  $U_{r1}$  is 0 or -1, a data bit could be hidden, where the video will not be modified if  $m_i$  is “0”, and the coefficient will become 1 or -2 if  $m_i$  is “1”. Accordingly, the receiver could extract information  $m_i$  and restore the original QDCT coefficient from the marked QDCT coefficient  $U'_{r1}$  as follows:

- A. If the value of the marked coefficient  $U'_{r1}$  is 0 or -1, the extracted data bit  $m_i$  will be 0, the original coefficient  $U_{r1} = U'_{r1}$ , which needs no modification.
- B. If  $U'_{r1}$  is 1, the original value of the coefficient  $U_{r1}$  will be 0, and the bit  $m_i$  will be 1.
- C. If  $U'_{r1}$  is -2, the original value of the coefficient  $U_{r1}$  will be -1, and the bit  $m_i$  will be 1.
- D. If  $U'_{r1} > 1$ , there will be no hidden data bit, and the original coefficient  $U_{r1} = U'_{r1} - 1$ .
- E. If  $U'_{r1} < -2$ , there will be no hidden data bit, and the original coefficient  $U_{r1} = U'_{r1} + 1$ .

In this scheme, the 1D coefficient histogram is defined by

$$h(d_1) = \#\{U_{r1} | U_{r1} = d_1\} \quad (2)$$

Where  $\#$  is the cardinal number of a set, and  $d_1$  is an integer.

The embedding capacity denoted as  $EC$  of the 1D HS hiding scheme is

$$EC = h(0) + h(-1) \quad (3)$$

For QDCT coefficients, the embedding distortion denoted as  $ED$  in terms of  $l^2$ -error can be formulated as

$$ED = \frac{1}{2}h(0) + \frac{1}{2}h(-1) + \sum_{d_1=-\infty}^{-2} h(d_1) + \sum_{d_1=1}^{+\infty} h(d_1) \quad (4)$$

When two QDCT coefficients denoted as  $U_{r1}$  and  $U_{r2}$  are used to embed data, the mapping shown in Figure 1 can be used for hiding data into each coefficient. Therefore, a 2D coefficient histogram defined by (5) could be employed to hide data, as shown in Figure 2, where at most two bits of information denoted by  $m_i m_{i+1}$  could be hidden into one pair of coefficients.

$$t(d_1, d_2) = \#\{(U_{r1}, U_{r2}) | U_{r1} = d_1, U_{r2} = d_2\} \quad (5)$$

where  $d_2$  is an integer.

## 2.2. Proposed 2D HS

In Figure 2, the maximum modification of each QDCT coefficient pair is 2, which may cause obvious distortion. In order to reduce the cost, we primarily seek for different points to record different data with at most one modification. Four points are needed for recording two bits of data “00”, “01”, “10”, and “11”. As illustrated in Figure 3, when the value of coefficient pair  $(U_{r1}, U_{r2})$  is (-1, 0), which can be used for representing “00” with no change, it could be expanded to its neighboring points (-1, 1), (-2, 0), and (-1, -1) to record two data bits 01, 10, and 11 with one modification, respectively. When the value of  $(U_{r1}, U_{r2})$  is (2, 0), it can be expanded to its adjacent points (3,0) and (2,1) for recording a data bit “0” and “1” with one modification,

respectively. According to these mapping rules, the set (denoted as  $D$ ) of all the values of QDCT coefficient pairs, could be divided into sixteen subsets defined as follows:

$$\begin{aligned}
 D_1 &= \{(0, 0)\}, & D_2 &= \{(-1, 0)\}, & D_3 &= \{(-1, -1)\}, & D_4 &= \{(0, -1)\}, & D_5 &= \{(U_{r1}, 0) \mid U_{r1} > 0\}, \\
 D_6 &= \{(U_{r1}, 0) \mid U_{r1} < -1\}, & D_7 &= \{(U_{r1}, -1) \mid U_{r1} > 0\}, & D_8 &= \{(U_{r1}, -1) \mid U_{r1} < -1\}, \\
 D_9 &= \{(0, U_{r2}) \mid U_{r2} > 0\}, & D_{10} &= \{(0, U_{r2}) \mid U_{r2} < -1\}, & D_{11} &= \{(-1, U_{r2}) \mid U_{r2} > 0\}, \\
 D_{12} &= \{(-1, U_{r2}) \mid U_{r2} < -1\}, & D_{13} &= \{(U_{r1}, U_{r2}) \mid U_{r1} > 0, U_{r2} > 0\}, \\
 D_{14} &= \{(U_{r1}, U_{r2}) \mid U_{r1} > 0, U_{r2} < -1\}, & D_{15} &= \{(U_{r1}, U_{r2}) \mid U_{r1} < -1, U_{r2} > 0\}, \\
 D_{16} &= \{(U_{r1}, U_{r2}) \mid U_{r1} < -1, U_{r2} < -1\}.
 \end{aligned}$$

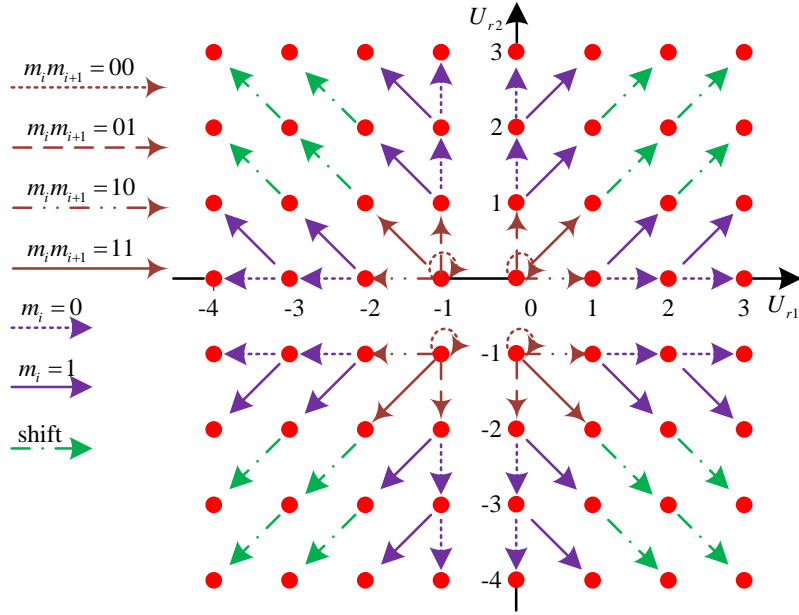


Figure 2. Transformation of QDCT coefficients in conventional 2D HS

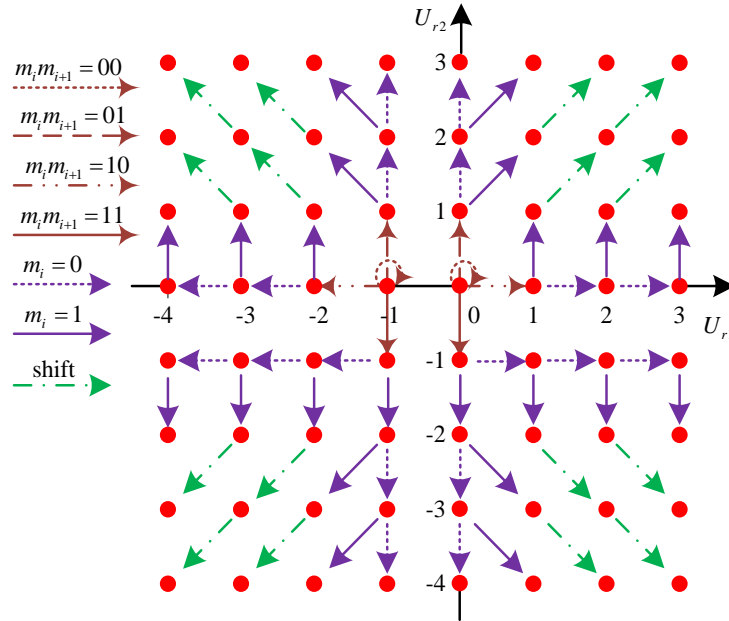


Figure 3. Transformation of QDCT coefficients in the proposed 2D HS

The selected coefficient pairs are classified into different sets on the basis of their values. Information could be hidden according to the set which the value of the coefficient pair belongs to. Correspondingly, the embedding process can be described in the following way.

If the coefficient pair  $(U_{r1}, U_{r2}) \in D_1$ , the marked coefficient pair denoted by  $(U'_{r1}, U'_{r2})$  will be

$$(U'_{r1}, U'_{r2}) = \begin{cases} (U_{r1}, U_{r2}), & \text{if } m_i m_{i+1} = 00 \\ (U_{r1}, U_{r2} + 1), & \text{if } m_i m_{i+1} = 01 \\ (U_{r1} + 1, U_{r2}), & \text{if } m_i m_{i+1} = 10 \\ (U_{r1}, U_{r2} - 1), & \text{if } m_i m_{i+1} = 11 \end{cases} \quad (6)$$

If the coefficient pair  $(U_{r1}, U_{r2}) \in D_2$ , the marked coefficient pair  $(U'_{r1}, U'_{r2})$  will be

$$(U'_{r1}, U'_{r2}) = \begin{cases} (U_{r1}, U_{r2}), & \text{if } m_i m_{i+1} = 00 \\ (U_{r1}, U_{r2} + 1), & \text{if } m_i m_{i+1} = 01 \\ (U_{r1} - 1, U_{r2}), & \text{if } m_i m_{i+1} = 10 \\ (U_{r1}, U_{r2} - 1), & \text{if } m_i m_{i+1} = 11 \end{cases} \quad (7)$$

If the coefficient pair  $(U_{r1}, U_{r2})$  belongs to  $D_3$  or  $D_8$ , the marked coefficient pair  $(U'_{r1}, U'_{r2})$  will be

$$(U'_{r1}, U'_{r2}) = \begin{cases} (U_{r1} - 1, U_{r2}), & \text{if } m_i = 0 \\ (U_{r1}, U_{r2} - 1), & \text{if } m_i = 1 \end{cases} \quad (8)$$

If the coefficient pair  $(U_{r1}, U_{r2})$  belongs to  $D_4$  or  $D_7$ , the marked coefficient pair  $(U'_{r1}, U'_{r2})$  will be

$$(U'_{r1}, U'_{r2}) = \begin{cases} (U_{r1} + 1, U_{r2}), & \text{if } m_i = 0 \\ (U_{r1}, U_{r2} - 1), & \text{if } m_i = 1 \end{cases} \quad (9)$$

If the coefficient pair  $(U_{r1}, U_{r2}) \in D_5$ , the marked coefficient pair  $(U'_{r1}, U'_{r2})$  will be

$$(U'_{r1}, U'_{r2}) = \begin{cases} (U_{r1} + 1, U_{r2}), & \text{if } m_i = 0 \\ (U_{r1}, U_{r2} + 1), & \text{if } m_i = 1 \end{cases} \quad (10)$$

If the coefficient pair  $(U_{r1}, U_{r2}) \in D_6$ , the marked coefficient pair  $(U'_{r1}, U'_{r2})$  will be

$$(U'_{r1}, U'_{r2}) = \begin{cases} (U_{r1} - 1, U_{r2}), & \text{if } m_i = 0 \\ (U_{r1}, U_{r2} + 1), & \text{if } m_i = 1 \end{cases} \quad (11)$$

If the coefficient pair  $(U_{r1}, U_{r2}) \in D_9$ , the marked coefficient pair  $(U'_{r1}, U'_{r2})$  will be

$$(U'_{r1}, U'_{r2}) = \begin{cases} (U_{r1}, U_{r2} + 1), & \text{if } m_i = 0 \\ (U_{r1} + 1, U_{r2} + 1), & \text{if } m_i = 1 \end{cases} \quad (12)$$

If the coefficient pair  $(U_{r1}, U_{r2}) \in D_{10}$ , the marked coefficient pair  $(U'_{r1}, U'_{r2})$  will be

$$(U'_{r1}, U'_{r2}) = \begin{cases} (U_{r1}, U_{r2} - 1), & \text{if } m_i = 0 \\ (U_{r1} + 1, U_{r2} - 1), & \text{if } m_i = 1 \end{cases} \quad (13)$$

If the coefficient pair  $(U_{r1}, U_{r2}) \in D_{11}$ , the marked coefficient pair  $(U'_{r1}, U'_{r2})$  will be

$$(U'_{r1}, U'_{r2}) = \begin{cases} (U_{r1}, U_{r2} + 1), & \text{if } m_i = 0 \\ (U_{r1} - 1, U_{r2} + 1), & \text{if } m_i = 1 \end{cases} \quad (14)$$

If the coefficient pair  $(U_{r1}, U_{r2}) \in D_{12}$ , the marked coefficient pair  $(U'_{r1}, U'_{r2})$  will be

$$(U'_{r1}, U'_{r2}) = \begin{cases} (U_{r1}, U_{r2} - 1), & \text{if } m_i = 0 \\ (U_{r1} - 1, U_{r2} - 1), & \text{if } m_i = 1 \end{cases} \quad (15)$$

If the coefficient pair  $(U_{r1}, U_{r2})$  belongs to  $D_{13}$ ,  $D_{14}$ ,  $D_{15}$ , or  $D_{16}$ , no message could be hidden, and the marked coefficient pair will be

$$(U'_{r1}, U'_{r2}) = \begin{cases} (U_{r1} + 1, U_{r2} + 1), & \text{if } (U_{r1}, U_{r2}) \in D_{13} \\ (U_{r1} + 1, U_{r2} - 1), & \text{if } (U_{r1}, U_{r2}) \in D_{14} \\ (U_{r1} - 1, U_{r2} + 1), & \text{if } (U_{r1}, U_{r2}) \in D_{15} \\ (U_{r1} - 1, U_{r2} - 1), & \text{if } (U_{r1}, U_{r2}) \in D_{16} \end{cases} \quad (16)$$

In this way, the value of the marked coefficient pair may be in a new set, which the information could be extracted according to. The value of the coefficient pair could be completely recovered by the reverse process of embedding way. The embedding capacities of the presented 2D HS and the conventional 2D HS, denoted as  $EC_{pre}$  and  $EC_{con}$ , can be computed by (17) and (18).

$$EC_{pre} = 2 \sum_{(U_{r1}, U_{r2}) \in D_1 \cup D_2} t(U_{r1}, U_{r2}) + \sum_{(U_{r1}, U_{r2}) \in D_3 \cup D_4 \cup D_5 \cup D_6 \cup D_7 \cup D_8 \cup D_9 \cup D_{10} \cup D_{11} \cup D_{12}} t(U_{r1}, U_{r2}) \quad (17)$$

$$EC_{con} = 2 \sum_{(U_{r1}, U_{r2}) \in D_1 \cup D_2 \cup D_3 \cup D_4} t(U_{r1}, U_{r2}) + \sum_{(U_{r1}, U_{r2}) \in D_5 \cup D_6 \cup D_7 \cup D_8 \cup D_9 \cup D_{10} \cup D_{11} \cup D_{12}} t(U_{r1}, U_{r2}) \quad (18)$$

According to (17) and (18), we can infer the difference of embedding capacity between the presented 2D HS and the conventional 2D HS by

$$EC_{con} - EC_{pre} = \sum_{(U_{r1}, U_{r2}) \in D_3 \cup D_4} t(U_{r1}, U_{r2}) \quad (19)$$

For QDCT coefficients, the distortion in terms of  $l^2$ -error of the presented 2D HS and the customary 2D HS, denoted as  $ED_{pre}$  and  $ED_{con}$ , can be formulated as (20) and (21).

$$ED_{pre} = \frac{3}{4} \sum_{(U_{r1}, U_{r2}) \in D_1 \cup D_2} t(U_{r1}, U_{r2}) + \sum_{(U_{r1}, U_{r2}) \in D_3 \cup D_4 \cup D_5 \cup D_6 \cup D_7 \cup D_8} t(U_{r1}, U_{r2}) \\ + \frac{3}{2} \sum_{(U_{r1}, U_{r2}) \in D_9 \cup D_{10} \cup D_{11} \cup D_{12}} t(U_{r1}, U_{r2}) + 2 \sum_{(U_{r1}, U_{r2}) \in D_{13} \cup D_{14} \cup D_{15} \cup D_{16}} t(U_{r1}, U_{r2}) \quad (20)$$

$$\begin{aligned}
ED_{con} = & \sum_{(U_{r1}, U_{r2}) \in D_1 \cup D_2 \cup D_3 \cup D_4} t(U_{r1}, U_{r2}) + \frac{3}{2} \sum_{(U_{r1}, U_{r2}) \in D_5 \cup D_6 \cup D_7 \cup D_8 \cup D_9 \cup D_{10} \cup D_{11} \cup D_{12}} t(U_{r1}, U_{r2}) \\
+ 2 & \sum_{(U_{r1}, U_{r2}) \in D_{13} \cup D_{14} \cup D_{15} \cup D_{16}} t(U_{r1}, U_{r2})
\end{aligned} \tag{21}$$

Therefore, we can infer the difference of hiding distortion between the presented 2D HS and the conventional 2D HS by

$$ED_{con} - ED_{pre} = \frac{1}{4} \sum_{(U_{r1}, U_{r2}) \in D_1 \cup D_2} t(U_{r1}, U_{r2}) + \frac{1}{2} \sum_{(U_{r1}, U_{r2}) \in D_3 \cup D_6 \cup D_7 \cup D_8} t(U_{r1}, U_{r2}) \tag{22}$$

According to (19) and (22), compared with the conventional 2D HS, to get the same quantity of embedding information, the distortion caused by our scheme is lower. When the value of the QDCT coefficient pair  $(U_{r1}, U_{r2})$  belongs to the sets  $D_1, D_2, D_5, D_6, D_7$ , or  $D_8$ , although the same capacity (two data bits or one data bit) can be gotten by the two methods, 2 is the largest cost in the conventional HS, whereas at most one modification is caused by our method. When  $(U_{r1}, U_{r2})$  belongs to the sets  $D_3$  or  $D_4$ , if the same number of data is embedded, the same distortion will be caused. When the value of the QDCT coefficient pair  $(U_{r1}, U_{r2})$  belongs to other sets, the same capacity and distortion will be gotten by the two methods.

### 3. 2D-HS-BASED MVC VIDEO RDH ALGORITHM

The presented RDH algorithm based on 2D HS for MVC video is shown in Figure 4. For the sender, in order to enhance the security and the robustness of the hidden information, the information is encrypted and encoded by binary BCH code before data hiding. The original MVC video is entropy decoded to get QDCT MBs, where some MBs are chosen for hiding data, and the unselected MBs will be entropy encoded directly. According to the approach depicted in Figure 3, several data bits are embedded into each embeddable QDCT coefficient pair, which could be randomly selected from every embeddable  $4 \times 4$  block in a MB. Therefore, for the receiver, the hidden data could be extracted from the marked QDCT coefficient pairs after entropy decoding.

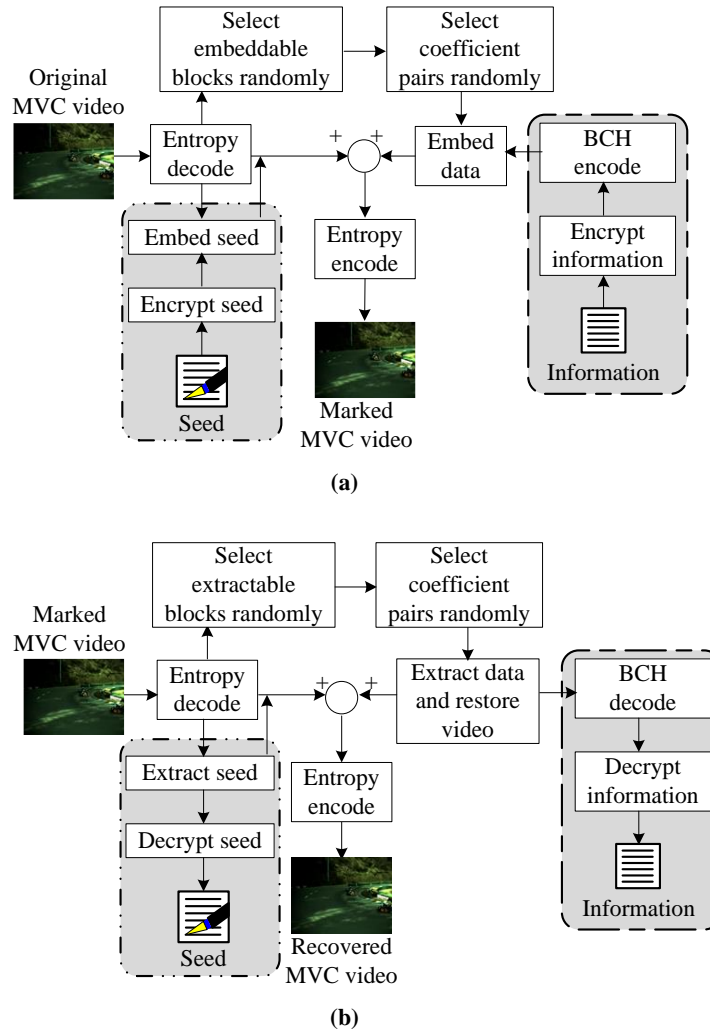


Figure 4. The flowchart of the presented 2D-HS-based RDH algorithm: (a) Information embedding, (b) Information extraction and carrier recovery

### 3.1. Distortion Propagation Preclusion

Figure 5 illustrates the prediction scheme of MVC video with hierarchical B coding and two views. Horizontal arrows indicate inter-frame prediction, and vertical arrows express parallax prediction. Each group of picture (GOP) includes 16 frames, where there are eight frames in each view.  $I_0$  frames and  $P_0$  frames are key frames at the highest level marked by 0. Only intra-frame prediction is used in  $I_0$  frames so that they will not be affected by other frames. Whereas the distortion of one  $I_0$  frame will affect all the frames in the two adjacent GOPs predicted by this  $I_0$  frame. In contrast, parallax distortion drift will be avoided by embedding data into  $P_0$  frames, which do not predict the frames in the left view. The parallax and the inter-frame distortion drift will be eliminated by hiding data into  $b_4$  frames, which are not referred by other frames. But in the internet transfers, it is easy to lose  $b_4$  frames located at the lowest level. By comparison, stronger robustness can be obtained by hiding data into  $P_0$  frames located at the highest level. Additionally, six  $B_3$  frames in each GOP could be used for hiding data, where only  $b_4$  frames may be changed by hiding data into  $B_3$  frames. In short, users could select embeddable frames on demand since different frames have strengths and weaknesses.



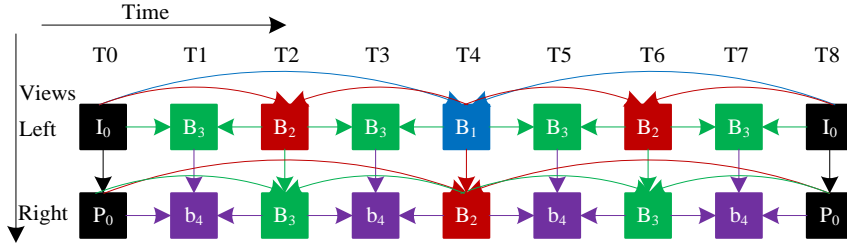


Figure 5. Prediction structure of MVC video with hierarchical B coding and two views

At the MVC encoder, after the prediction block is subtracted from the original pixel block in the YUV video, the consequent residual block, denoted by  $L^{R0}$ , undergoes  $4 \times 4$  (or  $8 \times 8$ ) DCT and quantization as reckoned in:

$$U = \text{round} \left[ (C_f L^{R0} C_f^T) \otimes (E_f / Q) \right] \quad (23)$$

Where  $U$  is a QDCT block with 16 QDCT coefficients numbered by zigzag scan,

$$C_f = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 2 & 1 & -1 & -2 \\ 1 & -1 & -1 & 1 \\ 1 & -2 & 2 & -1 \end{bmatrix}, E_f = \begin{bmatrix} a^2 & ac/2 & a^2 & ac/2 \\ ac/2 & c^2/4 & ac/2 & c^2/4 \\ a^2 & ac/2 & a^2 & ac/2 \\ ac/2 & c^2/4 & ac/2 & c^2/4 \end{bmatrix}, a=1/2, c=\sqrt{2/5}, \text{ the matrix } C_f^T \text{ is the transpose of } C_f, Q \text{ is the quantization step size, } \otimes \text{ means a mathematical operation, which expresses that each element in the preceding matrix is multiplied by the value at the corresponding position in the latter matrix.}$$

If data is hidden into one QDCT block  $U$  by changing some QDCT coefficients, the block  $U$  will be altered to a marked QDCT block denoted by  $U'$ , and the deviation (denoted by  $\Delta U$ ) caused by hiding data is

$$\Delta U = U' - U \quad (24)$$

At the decoder, the residual pixel block, denoted as  $L^R$ , is gained by lossy decompression (inverse quantization and  $4 \times 4$  inverse DCT) as shown in:

$$L^R = \text{round} \left[ C_u^T (U \otimes E_u) C_u \right] \quad (25)$$

$$\text{Where } C_u = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1/2 & -1/2 & -1 \\ 1 & -1 & -1 & 1 \\ 1/2 & -1 & 1 & -1/2 \end{bmatrix}, E_u = \begin{bmatrix} a^2 & ac & a^2 & ac \\ ac & c^2 & ac & c^2 \\ a^2 & ac & a^2 & ac \\ ac & c^2 & ac & c^2 \end{bmatrix}.$$

If data is hidden by changing some QDCT coefficients, the residual pixel block  $L^R$  will be turned into the marked pixel block denoted as  $L^{R'}$ , and the mutation denoted as  $\Delta L^R$  is

$$\Delta L^R = L^{R'} - L^R = \text{round} \left[ C_u^T (\Delta U \cdot Q \otimes E_u) C_u \right] \quad (26)$$

Take the QDCT coefficient  $U_{15}$  as an instance to prove the distortion caused by hiding data. Assume an integer denoted as  $z$  is added to the value of  $U_{15}$ , i.e., the change of the QDCT block

$$\Delta U = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & z \end{bmatrix}$$

for hiding data is  $\Delta L^R = \frac{1}{4} Qc^2 z$ , then the difference of the corresponding pixel block in YUV video is

$$\Delta L^R = \frac{1}{4} Qc^2 z \begin{bmatrix} 1 & -2 & 2 & -1 \\ -2 & 4 & -4 & 2 \\ 2 & -4 & 4 & -2 \\ -1 & 2 & -2 & 1 \end{bmatrix} \quad (27)$$

In order to hide information, only  $z$  is added to the QDCT coefficient  $U_{15}$ , but the whole  $4 \times 4$  pixel block in the corresponding YUV video is altered. Similarly, the modification of one QDCT coefficient in an  $8 \times 8$  transformation block will change the whole  $8 \times 8$  pixel block, whose distortion range is bigger than  $4 \times 4$  block. Additionally, only two kinds of transformations,  $4 \times 4$  transformation and  $8 \times 8$  transformation, are used in MVC standard. Thus a  $4 \times 4$  transformation block is chosen to hide information in this paper.

Accordingly, the boundary pixels denoted as  $o_0, o_1, \dots, o_{12}$ , shown in Figure 6 (In mode 2 that is not depicted in Figure 6(b) and Figure 6 (c), all elements are predicted with the average of upper pixels denoted as H and left pixels denoted as V, i.e., Mean (H+V) ), may be changed by embedding data into some QDCT coefficients of the four blocks denoted by  $L_{x,y-1}$ ,  $L_{x-1,y-1}$ ,  $L_{x-1,y}$ , and  $L_{x-1,y+1}$  (integers  $x$  and  $y$  present the position of a block). If intra-frame prediction is employed in the current block  $L_{x,y}$ , its prediction block will be calculated by the pixels  $o_0 \dots o_{12}$ . So the distortion of the blocks  $L_{x,y-1}$ ,  $L_{x-1,y-1}$ ,  $L_{x-1,y}$ , and  $L_{x-1,y+1}$  will drift to the block  $L_{x,y}$ . Otherwise, if parallax prediction or inter-frame prediction is applied in the block  $L_{x,y}$ , whose prediction block is gotten by referring other frames as shown in Figure 5, the block  $L_{x,y}$  will not be affected by any modification of its neighboring blocks.

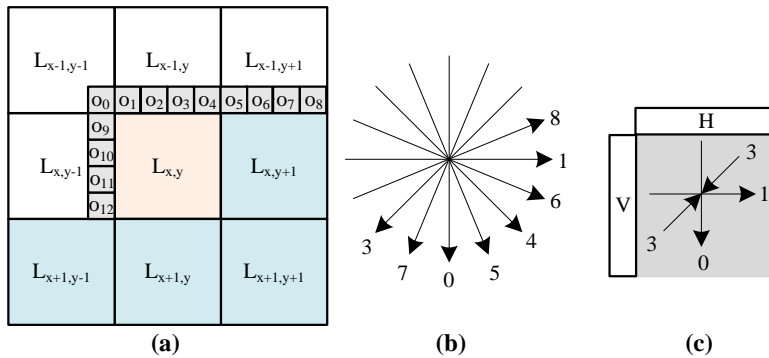


Figure 6. Intra-frame prediction modes: (a) Position, (b) Predictive directions of  $4 \times 4$  blocks, (c) Predictive directions of  $16 \times 16$  blocks.

Let inter-MB be a  $16 \times 16$  MB with inter-frame prediction or parallax prediction. The nine  $4 \times 4$  blocks numbered by  $0 \dots 8$  are not located at the rightmost column or the bottom row of the inter-MB shown in Figure 7. If the block  $L_{x,y}$  is such a block, its neighboring blocks  $L_{x,y+1}$ ,  $L_{x+1,y+1}$

and  $L_{x+1,y}$  will be in the current inter-MB. In addition, the neighboring block  $L_{x+1,y-1}$  may be in either the current inter-MB or one of the blocks numbered by 9...11 in the encoded MB at the encoder (or a decoded MB at the decoder). Thus these adjacent blocks  $L_{x,y+1}$ ,  $L_{x+1,y+1}$ ,  $L_{x+1,y}$  and  $L_{x+1,y-1}$  will not be affected by the current block  $L_{x,y}$ . Therefore, the nine  $4 \times 4$  blocks numbered by 0...8 in each inter-MB can be chosen as embeddable blocks for hiding data without intra-frame distortion drift.

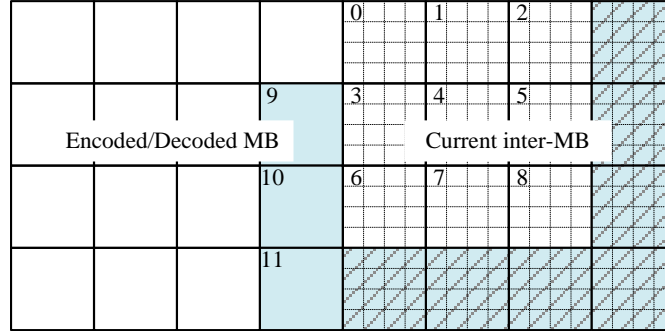


Figure 7.  $4 \times 4$  embeddable blocks without intra-frame distortion drift

### 3.2. Data Embedding

The embedding flowchart at the encoder is shown in Figure 4(a). There are five main steps to complete the data-hiding process: processing the to-be-hidden information, embedding the seed denoted by  $K$ , choosing embeddable blocks, selecting coefficient pairs, and data embedding.

- A. **Processing the to-be-hidden information** To ensure safety, the original information is encrypted into cypher text with a RSA public key. Then the encrypted information is encoded by BCH code before data hiding to guarantee the robustness.
- B. **Embedding the seed  $K$**  To improve the security, random function and the seed  $K$  are used to select embeddable blocks and coefficients. We encrypt the seed  $K$  with a RSA public key and hide the seed  $K$  into some QDCT MBs of  $I_0$  frames with a public RDH algorithm.
- C. **Choosing embeddable blocks** It is necessary to limit the distortion region since large region of distortion is obvious. Random block numbers are generated to select embeddable  $4 \times 4$  blocks (not at the bottom row or the rightmost column, as shown in Figure 7), which could be used for hiding data without intra-frame distortion drift. Additionally, a positive integer is set to generate a random threshold denoted by  $G$  so that embeddable blocks could be chosen randomly according to  $|U_0| \geq G$ . When the threshold  $G$  is bigger, the fewer embeddable blocks will be found, and the region of the distortion will be less.
- D. **Selecting coefficient pairs** Two QDCT coefficients ( $U_{r1}$ ,  $U_{r2}$ ) are randomly chosen from each embeddable blocks. There are 15 ways to choose  $U_{r1}$  from 15 AC coefficients, and 14 ways to select  $U_{r2}$  from the rest 14 AC coefficients. Thus the optional of selecting ( $U_{r1}$ ,  $U_{r2}$ ) is  $15 \times 14 = 210$ . Given a marked block, the probability for directly guessing the hidden data bit is  $1/210 \approx 0.00476$ . In this way, the application of arbitrary embedding positions including blocks and coefficients can be used to reduce the modification of statistical histogram and improve the undetectability of RDH algorithm.
- E. **Data embedding** Some data bits are hidden into each coefficient pair by following Figure 3.

### 3.3. Data Extracting and Video Restoring

Figure 4(b) displays the procedure of information extraction and video restoration. There are four main steps to complete this process: extracting the seed  $K$ , choosing extractable blocks and coefficient pairs, data extracting and video restoring, processing the extracted information.

- A. **Extracting the seed  $K$**  After the entropy decoding of the MVC video, we extract the seed  $K$  from  $I_0$  frame with the public RDH algorithm and decrypt the seed  $K$  with the RSA private key.
- B. **Choosing extractable blocks and coefficient pairs** The same random seed can be used to generate the same random sequence. Thus the communication parties can use the same random seed  $K$  to choose the same embeddable/ extractable blocks and coefficient pairs, where information could be hidden without intra-frame distortion drift.
- C. **Data extracting and video restoring** According to the reverse process of Figure 3, we can extract the hidden information and completely recover the MVC video.
- D. **Processing the extracted information** Finally, we use BCH code technique to recover the information and decrypt the hidden data.

## 4. EXPERIMENTAL RESULTS AND DISCUSSION

The presented algorithm has been effectuated in the H.264 reference software version JM18.4[21]. Test samples are the nine video sequences [22] (the size of each frame is  $640 \times 480$ ) shown in Figure 8. Two YUV files are encoded into a MVC video with 233 frames (at 30 frames/second) and two views, left view and right view, which has 30 P0 frames. The parameters intra-period and quantization step are set to 8 and 28, respectively. The capacity of a sequence is the average number of bits embedded into each P0 frame of the 30 P0 frames in that sequence. The peak signal-to-noise ratio (PSNR) value compared to the original YUV files is the average of all the frames in two views. The structural similarity (SSIM) compared to the original YUV files is the average of all the frames. The embedding efficiency  $e$  is defined as

$$e = N_{hide} / \sum_{i=1}^{i=N_m} Lm_i \quad (28)$$

where  $N_{hide}$  is the number of embedded bits,  $Lm_i$  is the changed size of the  $i$ th modified QDCT coefficient,  $N_m$  is the number of modified QDCT coefficients.

### 4.1. Data Hiding Performance

The capacity, PSNR, SSIM, bit-rate increase and embedding efficiency always depend on the video content and the embeddable block selection conditions. First, different embedding performances can be achieved by hiding data into different videos. Second, lower embedding capacity, better video quality, lower bit-rate increase, and less embedding efficiency can be obtained by data hiding with stricter block selection condition (higher threshold). When the threshold  $G$  increases, the increases in PSNR and SSIM values are very gentle, whereas the embedding capacity, the bit-rate increase and the embedding efficiency decline sharply.

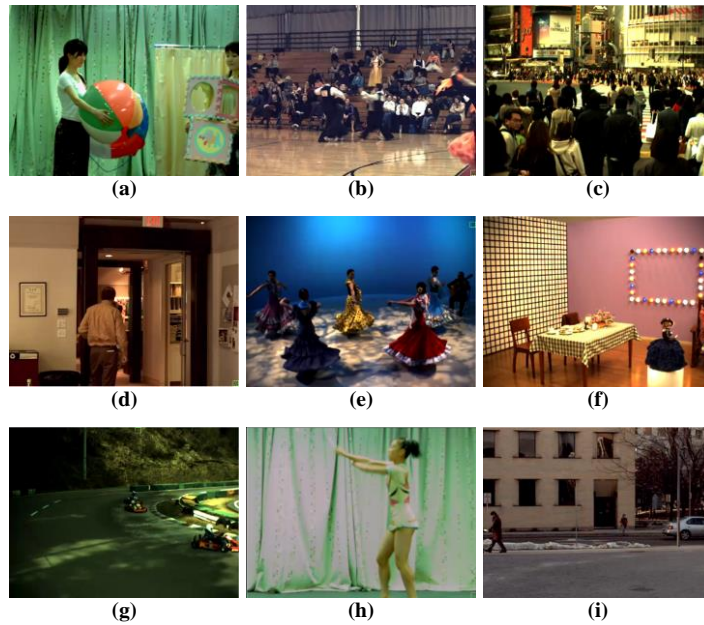
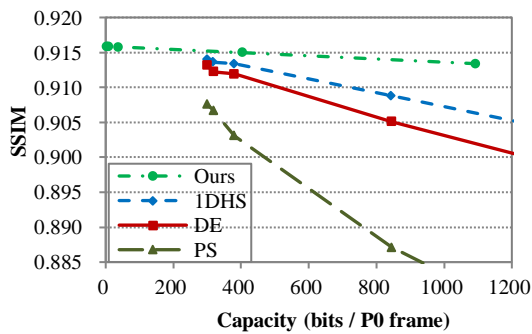
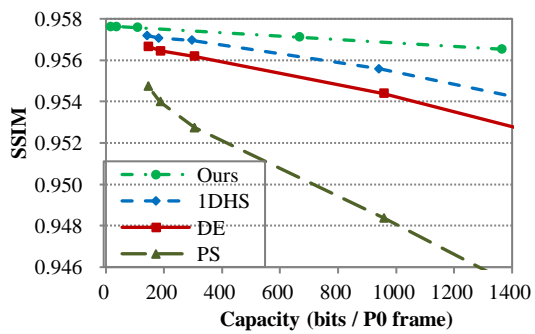


Figure 8. Test videos (the size of each frame is 640×480) (a) Akko&Kayo, (b) Ballroom, (c) Crowd, (d) Exit, (e) Flamenco, (f) Objects, (g) Race, (h) Rena and (i) Vassar

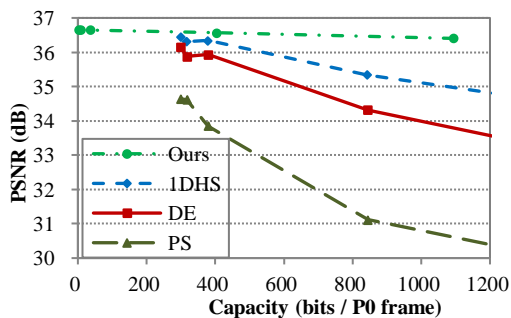
Figure 9 shows the performance comparison of four different RDH algorithms. In the method presented in [17] denoted by DE, the algorithm presented in [1] denoted by PS, and the proposed scheme, data is hidden into QDCT coefficient pair  $(U_2, U_5)$ , where  $U_2$  is used by [14], denoted by 1DHS.



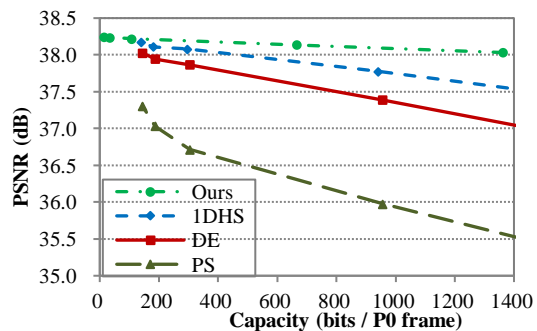
(a) Vassar



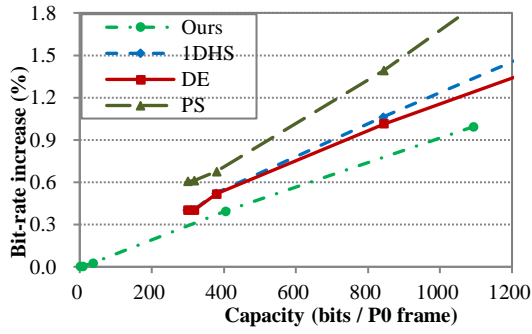
(b) Average of 9 videos



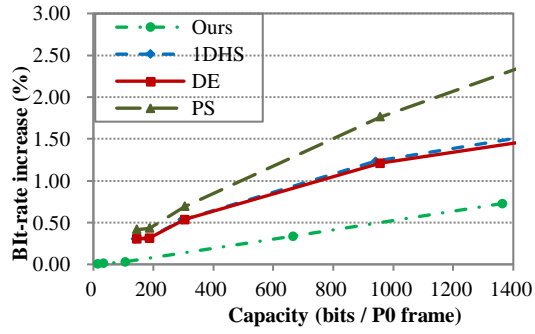
(c) Vassar



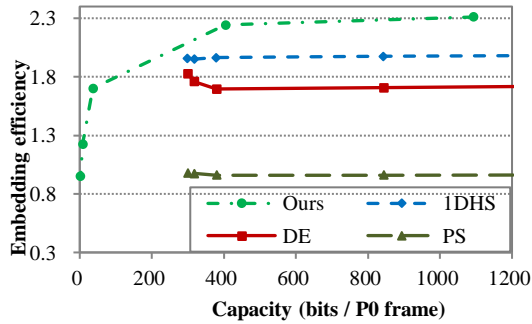
(d) Average of 9 videos



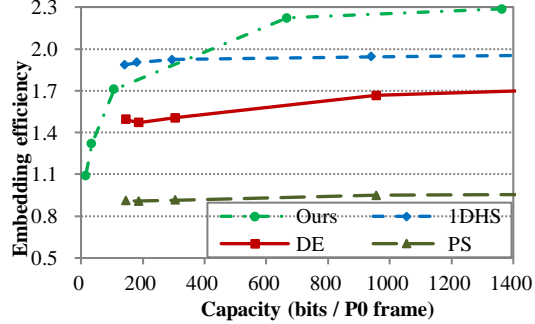
(e) Vassar



(f) Average of 9 videos



(g) Vassar



(h) Average of 9 videos

Figure 9. Comparison of hiding performance on different schemes

The experimental results show that when the same number of data is hidden, the best SSIM, PSNR, embedding efficiency, and the least bit-rate increase can be obtained by using the presented algorithm. The best SSIM and PSNR show that the best video quality can be gained by using our scheme. The highest embedding efficiency means that if the same modification is made, the most data bits could be hidden by using our method. According to the lower bit-rate increase, we can know that the application of the proposed method has minimal impact on the coding efficiency of MVC.

In order to further compare these methods, the threshold  $G$  is set to be 0 and 1000 bits of data are hidden into one  $P_0$  frame of Race on average, as shown in Table 1. Compared with some state-of-the-art RDH methods, our RDH algorithm is superior by enhancing PSNR, SSIM, and embedding efficiency with 0.19dB, 0.0007, and 0.5 at least, respectively.

Table 1. Comparison of embedding performance between the proposed algorithm and four methods for hiding 1000 bits of information into one  $P_0$  frame of Race on average

	<b>Ours</b>	<b>1DHS</b>	<b>DE</b>	<b>PS</b>
PSNR(dB)	37.62	37.43	37.38	36.87
SSIM	0.9614	0.9607	0.9605	0.9593
e	2.21	1.71	1.55	0.95

Accordingly, parts of the marked frames of Race are shown in Figure 10. It is easy to find apparent distortion on the trees in the frame (b). By contrast, little distortion could be seen in the

frame (c). The experimental results verify that superior visual quality can be achieved by using the proposed method to embed information.

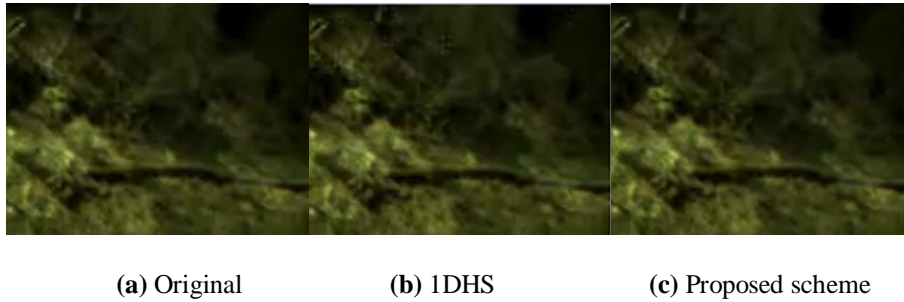


Figure 10. The marked frames of Race generated by 1DHS and the proposed 2D HS with 1000 bits of information

## 4.2. Discussions

The computational efficiency of the proposed RDH scheme is related to the frame number denoted by  $L_V$  and the information length  $N_{hide}$ , so the computational complexity of the proposed algorithm could be denoted as  $O(L_V \times N_{hide})$ . All the experiments were run on a machine with 24 processors (2.5GHz) and 32GB of RAM. It took 162ms on average to run the data hiding procedure for each frame of the test videos. The video decoding procedure without data hiding for each frame of the test videos cost 161ms on average. The duration caused by hiding data for each frame is 1ms on average. Therefore, the presented method with low complexity can be used for real-time application.

Furthermore, in our paper [23], we have tested that BCH(7,4,1) is the most powerful code in terms of error correction capability if 1 random error bit is corrected. Otherwise, BCH(63,7,15) is the most powerful code in terms of error correction capability, where the hidden data bits can be recovered 100% when the frame loss rate is no more than 15%. So high robustness could be achieved by using the proposed scheme shown in Figure 4.

## 5. CONCLUSIONS

A novel 2D HS is proposed in this paper. Utilizing this scheme, two data bits can be hidden with at most one modification, whereas two will be cost in the conventional HS. Compared with some state-of-the-art methods, the proposed scheme has superior hiding performance. In our future work, we will generalize the proposed scheme for hiding over 2 bits of data with at most one modification, and extend this method to image RDH such as difference HS and prediction-error HS.

## ACKNOWLEDGMENT

This work is supported by the Natural Science Foundation of Hubei Province under Grant 2017CFB306.

The authors are heartily grateful to the reviewers for their valuable comments improving the quality of the original manuscript.

**REFERENCES**

- [1] S. Maiti and M. P. Singh, "A novel reversible data embedding method for source authentication and tamper detection of H.264/AVC video," presented at the 5th International Conference on Information Processing, ICIP 2011, Bangalore, India, August 5, 2011 - August 7, 2011, 2011. Available: <Go to ISI>://WOS:000306579700044
- [2] K. L. Chung, Y. H. Huang, P. C. Chang, and H. Y. M. Liao, "Reversible data hiding-based approach for intra-frame error concealment in H.264/AVC," (in English), *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 20, no. 11, pp. 1643-1647, Nov 2010.
- [3] D. W. Xu, R. D. Wang, and Y. Q. Shi, "An improved reversible data hiding-based approach for intra-frame error concealment in H.264/AVC," (in English), *Journal of Visual Communication and Image Representation*, Article vol. 25, no. 2, pp. 410-422, Feb 2014.
- [4] J. Zhao, Z. Li, and B. Feng, "A novel two-dimensional histogram modification for reversible data embedding into stereo H.264 video," *Multimedia Tools and Applications*, vol. 75, no. 10, pp. 5959-5980, May 2016.
- [5] J. Zhao and Z. Li, "Three-dimensional histogram shifting for reversible data hiding," *Multimedia Systems*, vol. 24, no. 1, pp. 95-109, Feb 2018.
- [6] Z. C. Ni, Y. Q. Shi, N. Ansari, and W. Su, "Reversible data hiding," (in English), *IEEE Transactions on Circuits and Systems for Video Technology*, Article vol. 16, no. 3, pp. 354-362, Mar 2006.
- [7] W. F. Qi, X. L. Li, T. Zhang, and Z. M. Guo, "Optimal Reversible Data Hiding Scheme Based on Multiple Histograms Modification," *Ieee Transactions on Circuits and Systems for Video Technology*, vol. 30, no. 8, pp. 2300-2312, Aug 2020.
- [8] C.-C. Lin, W.-L. Tai, and C.-C. Chang, "Multilevel reversible data hiding based on histogram modification of difference images," *Pattern Recognition*, vol. 41, no. 12, pp. 3582-3591, Dec 2008.
- [9] L. L. An, X. B. Gao, X. L. Li, D. C. Tao, C. Deng, and J. Li, "Robust reversible watermarking via clustering and enhanced pixel-wise masking," *IEEE Transactions on Image Processing*, vol. 21, no. 8, pp. 3598-3611, Aug 2012.
- [10] X. L. Li, B. Li, B. Yang, and T. Y. Zeng, "General framework to histogram-shifting-based reversible data hiding," (in English), *IEEE Transactions on Image Processing*, Article vol. 22, no. 6, pp. 2181-2191, Jun 2013.
- [11] C. Qin, C. C. Chang, Y. H. Huang, and L. T. Liao, "An inpainting-assisted reversible steganographic scheme using a histogram shifting mechanism," (in English), *IEEE Transactions on Circuits and Systems for Video Technology*, Article vol. 23, no. 7, pp. 1109-1118, Jul 2013.
- [12] H. Yao, F. Y. Mao, Z. J. Tang, and C. Qin, "High-fidelity dual-image reversible data hiding via prediction-error shift," *Signal Processing*, vol. 170, May 2020, Art. no. 107447.
- [13] B. Ou and Y. Zhao, "High Capacity Reversible Data Hiding Based on Multiple Histograms Modification," *Ieee Transactions on Circuits and Systems for Video Technology*, vol. 30, no. 8, pp. 2329-2342, Aug 2020.
- [14] X. Z. Xie, C. C. Chang, and Y. C. Hu, "An adaptive reversible data hiding scheme based on prediction error histogram shifting by exploiting signed-digit representation," *Multimedia Tools and Applications*, vol. 79, no. 33-34, pp. 24329-24346, Sep 2020.
- [15] J. Tian, "Reversible data embedding using a difference expansion," (in English), *IEEE Transactions on Circuits and Systems for Video Technology*, Article vol. 13, no. 8, pp. 890-896, Aug 2003.
- [16] Z. W. Zhang, M. J. Zhang, and L. Y. Wang, "Reversible Image Watermarking Algorithm Based on Quadratic Difference Expansion," *Mathematical Problems in Engineering*, vol. 2020, Jan 2020, Art. no. 1806024.
- [17] C. Y. Weng, "DWT-based reversible information hiding scheme using prediction-error-expansion in multimedia images," *Peer-to-Peer Networking and Applications*, vol. 13, no. 2, pp. 514-523, Mar 2020.
- [18] W. G. He and Z. C. Cai, "An Insight Into Pixel Value Ordering Prediction-Based Prediction-Error Expansion," *Ieee Transactions on Information Forensics and Security*, vol. 15, pp. 3859-3871, 2020.
- [19] D. Coltuc, "Low distortion transform for reversible watermarking," *IEEE Transactions on Image Processing*, vol. 21, no. 1, pp. 412-417, Jan 2012.
- [20] Y. Q. Qiu, Z. X. Qian, H. Q. Zeng, X. D. Lin, and X. P. Zhang, "Reversible data hiding in encrypted images using adaptive reversible integer transformation," *Signal Processing*, vol. 167, Feb 2020, Art. no. 107288.



- [21] K. Sühring. (Aug). H.264/AVC software coordination (JM 18.4 ed.). Available: <http://iphome.hhi.de/suehring/tml>
- [22] (Feb). Video test sequences. Available: <http://blog.csdn.net/do2jiang/article/details/5499464>
- [23] 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, Sep. 2014.

## AUTHORS

**Juan Zhao** received her B.S. degree from Henan Normal University, Xinxiang, China, in 2007, and PhD degree from Huazhong University of Science and Technology, Wuhan, China, in 2015. She is currently a lecturer in School of Mathematics & Computer Science, Wuhan Polytechnic University. Her research interests include data hiding, network security and multimedia security.



**Zhitang Li** received his M.E. degree in Computer Architecture from Huazhong University of Science and Technology, Wuhan, China, 1987, and PhD degree in Computer Architecture from Huazhong University of Science and Technology, Wuhan, China, 1992. His research interests include computer architecture, network security, and P2P networks. He was the director of China Education and Research Network in Central China. He was a vice president of Department of Computer Science and Technology, Huazhong University of Science and Technology, China. He has published more than one hundred papers in the areas of network security, computer architecture, and P2P networks.

