IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received July 1, 2020, accepted July 21, 2020, date of publication July 30, 2020, date of current version August 17, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3013033

An Improved High Capacity and Efficient Data Hiding Scheme for 3D Videos Without Distortion Drift

GUANG HUA SONG[1] **AND HUI LIU**[2]
[1]School of Information and Safety Engineering, Zhongnan University of Economics and Law, Wuhan 430073, China
[2]Research Center of Hubei Logistics Development, Hubei University of Economics, Wuhan 430205, China

Corresponding author: Hui Liu (hliu2006@163.com)

This work was supported by the National Social Science Fund of China under Grant 16CXW019.

ABSTRACT Compared to traditional images and 2D videos, 3D videos are more likely to cause distortion drift problems in data hiding. In this paper, an improved scheme for high capacity and efficient data hiding in 3D videos based on Multi-view coding (MVC) standard is proposed, which avoids the problems of distortion drift. To improve the visual quality of data hiding, two selection modes are provided to limit the distortion drift. By modifying the selected coefficients of 4×4 quantized discrete cosine transforms (QDCT) in macroblocks and hiding data into b4-frames, the proposed scheme will effectively prevent distortion drift which caused by intra prediction and inter prediction. Moreover, the proposed algorithm maintains great randomness by using two random sequences. Several benchmark 3D video sequences of different resolutions and diverse contents are used for experimental evaluation. It is experimentally proved that the algorithm has greater embedding capacity and higher efficient than the previous methods.

INDEX TERMS Video data hiding, multi-view video, distortion drift, coefficient modification.

I. INTRODUCTION

With the rapid development of communication technology, especially 5G, and intelligent terminal devices, video data, including 2D and 3D, is increasing explosively and becoming popular application carrier. It is more easily for people to access, download and publish digital video over the Internet, which leads to serious security challenge to the content of the video. Preventing the information in digital video from being intercepted, destroyed or tampered by illegal users has become an important active research area. Thus, several means such as video data hiding is an important research issue in the field of cyberspace security. And, as an important means to achieve covert communication, video data hiding technology has become a hot spot of common concern in the current academic and security circles.

The current video data hiding technique mainly focus on the 2D fields, which aims to achieve the purpose of covert communication and copyright protection [1]–[3]. The researchers generally use a built-in video steganography algorithm based on DCT coefficients. However, considering that H.264 video coding introduces a new predictive coding mechanism, the error caused by modulating the DCT coefficients of the current macroblock will inevitably be transferred to adjacent macroblocks. The resulting error accumulation effect will have an increasing impact on video quality, and even unpredictable consequences [4]. How to control the distortion drift problem in data hiding becomes an important problem to be solved urgently.

On the other hand, because it can provide more realistic and natural visual effects, 3D video has gradually become a research hotspot in the field of video processing and communications, and the research on 3D compression standards is becoming more deeply.

In terms of video coding, the Moving Picture Experts Group (MPEG) under ISO / IEC JTC1 established the 3DAV (3D Audio and 3D Video) ad hoc group in 2001 to study the typical application scenarios of 3DAV and content can be standardized [5]. Considering that the standardization of H.264 video coding has been undertaken by the Joint Video Team (JVT) [6], the standardization of Multiview Video Coding (MVC) is officially handed over to JVT. At the same time, JVT release MVC's Joint Multiview Video Model (JMVM) as the official platform for MVC performance evaluation

The associate editor coordinating the review of this manuscript and approving it for publication was Yun Zhang.

in 2006 [7]. In 2010, JVT officially released the MVC standard based on H.264, hoping to solve the challenges faced by 3D video in stages, and wrote it as Appendix H into the latest H.264 specification [8]. JVT adopts the hierarchical B-frame structure proposed by HHI in the coding, which combines intra-frame prediction, inter-frame prediction and inter-view prediction in predictive coding [9].

The general multi-view structure adopted by the MVC standard includes 8 viewpoints, but the 3D video of the binocular viewpoint structure is currently the mainstream of research and application [10]–[12]. The latest H.264 experimental platform JM19.0 provides a complete support for stereoscopic video processing of binocular viewpoints. In the latest H.264 standard, 3D video coding uses a layered B-frame prediction structure [9]. The proposed hierarchical B-frames prediction structure with two viewpoints is shown in Fig. 1.

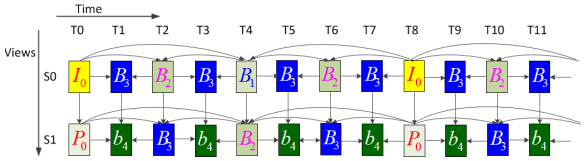


FIGURE 1. Hierarchical B frame structure with two viewpoints [9].

With the gradual rise of 3D video, the researchers pay more attention to the steganography algorithm on the related fields [13], [14]. But these algorithms do not consider the problem of distortion drift. According to the latest H.264 MVC coding specification, a steganography algorithm that can effectively avoid intra-frame distortion drift and inter-frame distortion drift is derived. By modifying the selected 4×4 quantized discrete cosine transforms (QDCT) coefficients of the macroblock, including intra-frame macroblocks and inter-frame macroblocks, will effectively prevent intra-frame distortion drift. And hiding data into b4-frames can completely prevent inter-frame distortion drift.

Compared with the existing similar research, the main contributions and novelty of this paper are highlighted as follows:

1) This algorithm does not need to pre-read the original video, and does not need to obtain the prediction mode and macroblock type of the video in advance. Through the set filtering conditions, the embedding process is implemented during decoding which greatly improves the execution efficiency of the algorithm. It can realize the real-time processing of encoding, embedding, playing and extracting, which can be used for the live broadcast system.

2) Since the embedded carrier can be either the inter-frame macroblock or the intra-frame macroblock, the proposed algorithm is more suitable for 3D H.264 video. Compared with traditional 2D video, 3D video uses more B-frames and P-frames when encoding, and I-frames mainly appears as a key frame. By applying the traditional 2D distortion-free drift algorithm directly to 3D video will inevitably bring huge inter-frame distortion drift, which seriously affects the video quality. On the other hand, there are more inter-frame macroblocks in B-frames and P-frames, and fewer intra-frame macroblocks. The mainstream 2D distortion-free drift algorithm lacks steganography space in 3D H.264 video. This will cause the algorithm's embedding capacity to be limited and even impossible to extract.

3) With loose restrictions, huge embedding capacity, and good versatility, the proposed algorithm can provide a broad space for the study of robustness and reversibility. The current mainstream data embedding algorithms can be combined with the proposed methods to furtherly improve the security and robustness.

The remainder of this paper is organized as follows. Section II reviews the literature related to video data hiding. In Section III, the theory and reasoning about distortion drift avoidance are introduced. Section IV provides the proposed data hiding scheme for 3D videos without distortion drift. In Section V, the experimental results and analysis are presented. The conclusion and future work are drawn in Section VI.

II. PREVIOUS WORK

Data hiding refers to a steganography technology that allows the authorized person to receive and obtain information content by embedding secret information in the carrier, while the unauthorized person cannot perceive the transmission behavior and its content [15]. At present, it has been widely used in the field of image and video. Currently, most video data hiding algorithms have been proposed and reviewed in the literature [2] and [3]. These algorithms can be mainly divided into two categories according to their extraction methods: detectable algorithms and readable algorithms. The former inserts a code that can only be detected, and the latter embeds a readable message [16]–[19]. In recent years, more and more researchers are engaged in the research of video data hiding, and have proposed and implemented various data hiding methods. Extensive research has been conducted using compressed domains according to the encoding stage. The data can be embedded in the compressed domain by using intra prediction [20], motion vector [21], residual coefficient [22] and entropy coding [23].

Part of the research on the H.264 video standard follows the traditional video data hiding theory and technology, and has done a lot of fruitful work. Especially the built-in video data hiding algorithm based on DCT coefficients has made great progress [24]–[27]. However, considering that H.264 video coding introduces a new predictive coding mechanism, the error caused by modulating the DCT coefficients of the current macroblock will inevitably be transferred to the adjacent macroblock, and the resulting error accumulation effect will affect the video quality. Therefore the H.264 video data hiding algorithm that considering distortion drift has attracted more and more researchers' attention [28]–[33].

Aiming at the problem of distortion drift caused by modifying DCT coefficients in H.264 video data hiding, the

literature [4] summarizes the causes and harms of distortion drift, and proposes an algorithm to limit error transmission. The author points out that the distortion caused by information embedding in H.264-encoded video will spread to adjacent blocks due to the presence of intra-frame prediction, and a cumulative effect will result in intra-frame distortion drift. The author uses the DCT transform and selects the modulation position of the corresponding coefficient, so that the error caused by embedding information in some 4×4 blocks will not be transferred to the adjacent blocks.

On this basis, the literature [34] devises a video data hiding algorithm without intra-frame distortion drift by studying the DCT transform and quantization operations in the H.264 video encoding process. To effectively control the intra-frame distortion drift, a coefficient compensation algorithm based on coupling coefficient pairs is proposed. During the execution of the algorithm, after embedding information into the selected QDCT coefficients, the corresponding coupling coefficients are compensated to control the intra-frame distortion drift. Combining the coupling coefficient pair method can effectively solve the problem of intra-frame distortion drift. Since then, more researches on intra-frame distortion-free drift algorithms have appeared [35]–[41]. To improve the embedding capacity of the algorithm, literature [35] presents an improved distortion-free drift steganography algorithm based on QDCT coefficient modulation. By introducing a new modulation mechanism, the algorithm doubles the embedding capability of the algorithm in [34] while maintaining similar video quality.

The emergence of intra-frame distortion-free data hiding algorithm for H.264 video effectively solves the problem of error diffusion and accumulation caused by embedded information, and reduces the impact of steganography on video quality [23], [42]–[50]. Nevertheless, most of the existing algorithms without distortion drift embed the data into the DCT coefficients of I-frames by using the intra-frame prediction mode. It is not suitable for the coding structure of 3D videos. And since the prediction mode of all MBs must be achieved before embedding and extracting, current algorithms increase the complexity greatly.

III. RELATED ALGORITHM THEORY AND REASONING

Two theoretical foundations of the proposed algorithm will be discussed in this section.

A. THE INTER PREDICTION STRUCTURE OF 3D VIDEOS AND INTER-FRAME DISTORTION DRIFT AVOIDANCE

JVT adopts the structure of hierarchical B-frames proposed in [9], which combines intra prediction, inter prediction and inter-view prediction. There are only one I-frame and one P-frame in a GOP (Group of Pictures, GOP) in 3D videos with two viewpoints, as is shown in Fig. 1. Among them, the abscissa represents time, the ordinate represents viewpoint, and the length of GOP is 8. Within the same viewpoint, the layered B-frame prediction structure based on 2D video is used, and the main prediction methods are intra prediction and inter prediction.

Compared with the 2D H.264 coding structure, inter-view prediction is increased in 3D video. The prediction direction is from viewpoint S0 to viewpoint S1 and is irreversible. At the same time, the I frame as a key frame only appears in S0 viewpoint, and it is directly or indirectly referenced by the remaining frames. Therefore, distortion drift caused by modifying the I frame far exceeds the 2D video. Because the distortion of the I frame not only affects viewpoint S0, but also spreads the error to viewpoint S1, which causes a larger-scale distortion drift and declines the video quality. As the Fig 2 shows, there is a clear prediction direction in the 3D H.264 video coding structure. A method to avoid distortion drift can be presented based on its prediction structure. It is obvious that distortion caused by modifying viewpoint S0 not only affects its internal frame, but also spreads to viewpoint S1, which causes large-scale distortion drift. On the contrary, the distortion caused by modifying viewpoint S1 will only be transmitted to its internal frame, and not spread to viewpoint S0, which avoid distortion drift between viewpoints.

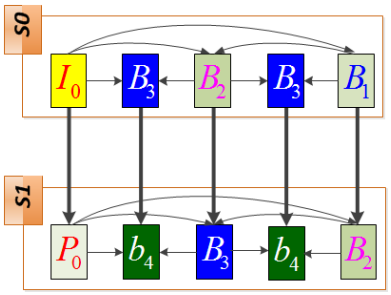


FIGURE 2. The prediction direction of 3D videos with two viewpoints [9].

Furthermore, it is obvious that there is a special frame, i.e. b4-frame, in viewpoint S1, which will not be referenced by any other frame. Using the b4 frame, we can effectively avoid the inter-frame and inter-view distortion drift. In other words, the distortion caused by embedding data into the b4 frame will be strictly limited to this frame, and will not drift to other frames and other viewpoints. On the basis, a proposition can be drawn as ***Prop 1.***

Proposition 1: There is an unreferenced frame named b4-frame in the 3D video coding structure with two viewpoints. The distortion caused by modifying the b4-frame will neither be transmitted to viewpoint S0, nor be transmitted to the other frames of viewpoint S1. Embedding data into b4-frame will completely avoid inter-frame distortion drift.

Proof: As Fig. 2 has shown, the following two conclusions are drawn.

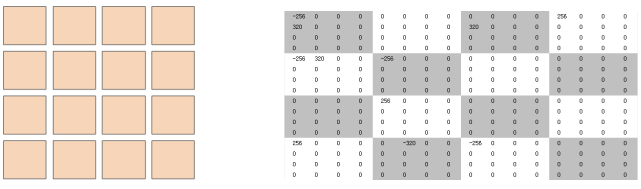
First, since the frame is located in the viewpoint S1, it will not be used as the reference frame for the inter-view prediction. The distortion caused by modifying the b4-frame will not be transmitted to viewpoint S0.

Secondly, the b4-frame can only be predicted by other frames, and cannot be used as a reference for the remaining frames in viewpoint S1. Then, the b4 frame is neither used as a reference frame for inter prediction, nor as a reference frame for inter-view prediction.

Therefore, when the data is hided into b4-frame, it will not cause any inter-frame and inter-view distortion drift. □

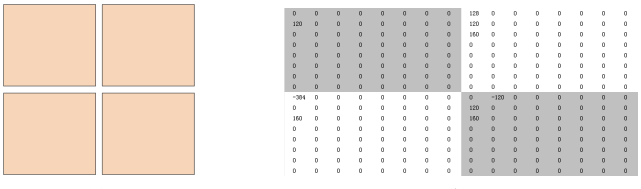
B. THE DCT TRANSFORM AND COEFFICIENT COMPENSATION IN 3D H.264 VIDEOS

When performing integer DCT transformation, 4×4 matrix transformation and 8×8 matrix transformation can be selected according to different motion scenarios. The 4×4 matrix is usually used for areas with intense motion information and more image details, while 8×8 matrix is used for areas with gentle motion and less image details. Considering the visual effect of human, embedding secret data into 8×8 matrix transform block will cause a large decrease in video quality. Therefore, the 4×4 matrix-transformed luminance block is usually selected by an algorithm that uses DCT coefficients for steganography. The different transformation matrices are shown in Fig 3 and Fig 4. The MBs with 4×4 transformation matrix is divided into 4×4 matrix to the integer DCT transform. The residual coefficients after IDCT is stored by 16 Matrix (4×4). On the other hand, the MBs with 8×8 transformation matrix will be divided into 8×8 matrix to the integer DCT transform. The residual coefficients after IDCT is stored by 4 Matrix (8×8). Fig 5 shows that, only the pixel values of the 4×4 IDCT which located in the last row and the last column can be used as reference for adjacent blocks [9]. If an algorithm can ensure that the pixel values of reference coefficient remains unchanged, it will not lead to intra-frame distortion drift.



(a) (b)

FIGURE 3. The 4×4 IDCT (a) and the data format (b) in 3D H.264 coding.



(a) (b)

FIGURE 4. The 8×8 IDCT (a) and the data format (b) in 3D H.264 coding.

Next, we study the DCT process using 4×4 matrix transformation in 3D H.264. For the convenience of expression, the parameters and instructions in the transformation process are given first, as is shown in Table 1.

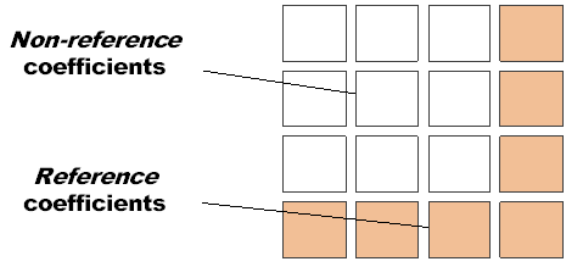


FIGURE 5. The selection of reference coefficients in 4×4 IDCT.

TABLE 1. Parameters and descriptions in DCT transformation.

Symbol	Descriptions
$X_{i,j}^{p}$	Residual coefficient matrix of 4x4 luma block
$X_{i,j}^{p'}$	Transformed residual coefficient matrix
$Y_{i,j}^{QDCT}$	Quantized DCT residual coefficient matrix
$Y_{i,j}^{QDCT'}$	Transformed QDCT residual coefficient matrix
Q	Quantization step
$\Delta Y_{i,j}^{QDCT}$	Quantized DCT coefficient difference matrix
$\Delta X_{i,j}^{r}$	Transformed residual coefficient difference matrix

Based on the H.264 coding standard, transform process and quantization of the MBs' residuals, the 4×4 IDCT and quantization operation are applied to MBs' residuals $X_{i,j}^{p}$ and QDCT coefficient matrix of $Y_{i,j}^{QDCT}$, i.e.$Y_{i,j}^{QDCT}$ is derived as (1):

$$Y_{i,j}^{QDCT} = (C_f X_{i,j}^{p} C_f^{T}) \bigotimes (E_f / Q) = \begin{bmatrix} Y_{00} & Y_{01} & Y_{02} & Y_{03} \\ Y_{10} & Y_{11} & Y_{12} & Y_{13} \\ Y_{20} & Y_{21} & Y_{22} & Y_{23} \\ Y_{30} & Y_{31} & Y_{32} & Y_{33} \end{bmatrix} \quad (1)$$

where:

$$C_f = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 2 & 1 & -1 & -2 \\ 1 & -1 & -1 & 1 \\ 1 & -2 & 2 & -1 \end{bmatrix},$$

$$E_f = \begin{bmatrix} a^2 & ab/2 & a^2 & ab/2 \\ ab/2 & b^2/4 & ab/2 & b^2/4 \\ a^2 & ab/2 & a^2 & ab/2 \\ ab/2 & b^2/4 & ab/2 & b^2/4 \end{bmatrix}, a = 1/2, b = \sqrt{2/5}$$

In decoding stage, dequantization operation and 4×4 integer inverse DCT are applied to$Y_{i,j}^{QDCT}$ which obtains a residual block$X_{i,j}^{r}$is derived as (2):

$$X_{i,j}^{r} = C_r^{T}(Y_{i,j}^{QDCT} \times Q \bigotimes E_r) \quad (2)$$

where:

$$C_r = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1/2 & -1/2 & -1 \\ 1 & -1 & -1 & 1 \\ 1/2 & -1 & 1 & -1/2 \end{bmatrix},$$

$$E_r = \begin{bmatrix} a^2 & ab & a^2 & ab \\ ab & b^2 & ab & b^2 \\ a^2 & ab & a^2 & ab \\ ab & b^2 & ab & b^2 \end{bmatrix}$$

The residual matrix after transformation and quantization is defined as (3):

$$\Delta Y_{i,j}^{QDCT} = Y_{i,j}^{QDCT'} - Y_{i,j}^{QDCT} \tag{3}$$

The residual matrix change after inverse quantization and inverse transformation is defined as (4):

$$\Delta X_{i,j}^{r} = X_{i,j}^{r'} - X_{i,j}^{r} \tag{4}$$

To limit the distortion caused by the embedded data to the current macroblock, we only need to ensure that the reference coefficient of the current macroblock does not change before and after embedding data. Then an embedding algorithm with coefficient compensation is proposed by the ***Prop 2***, which can ensure that the reference coefficient of the current macroblock remains unchanged after embedding the data. In this way, the distortion caused by the embedded data will only affect the current macroblock, and will not drift to adjacent macroblock. Therefore, in order to solve the problem of intra-frame distortion drift we need the following proposition.

Proposition 2: One bit will be embedded into each 4×4 residual coefficients without intra-frame distortion drift after IDCT by

$$(Y_{00}, Y_{02}, Y_{20}, Y_{22}) \rightarrow (Y_{00}+1, Y_{02}-1, Y_{20}-1, Y_{22}+1)$$

Or

$$(Y_{00}, Y_{02}, Y_{20}, Y_{22}) \rightarrow (Y_{00}-1, Y_{02}+1, Y_{20}+1, Y_{22}-1)$$

Proof: After converting $(Y_{00}, Y_{02}, Y_{20}, Y_{22})$ in $Y_{i,j}^{QDCT}$ to $(Y_{00}+1, Y_{02}-1, Y_{20}-1, Y_{22}+1)$ in $Y_{i,j}^{QDCT'}$, the change between $Y_{i,j}^{QDCT}$ and $Y_{i,j}^{QDCT'}$ is given by (5):

$$\begin{aligned} \Delta Y_{i,j}^{QDCT} &= Y_{i,j}^{QDCT'} - Y_{i,j}^{QDCT} \\ &= \begin{bmatrix} 1 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 \\ -1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \end{aligned} \tag{5}$$

Then, change between the decoded residual block $X_{i,j}^{r'}$ and the perturbed residual block $X_{i,j}^{r}$ is calculated by (6):

$$\begin{aligned} \Delta X_{i,j}^{r} &= X_{i,j}^{r'} - X_{i,j}^{r} = C_r^T(\Delta Y_{i,j}^{QDCT} \times Q \bigotimes E_r)C_r \\ &= C_r^T(\begin{bmatrix} 1 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 \\ -1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \times Q \bigotimes E_r)C_r \\ &= Q \times \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 4a^2 & 4a^2 & 0 \\ 0 & 4a^2 & 4a^2 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \end{aligned} \tag{6}$$

It can be seen by (6), the reference pixels have not been changed. Therefore, this operation will not cause intra-frame distortion drift.

Analogously, after converting $(Y_{00}, Y_{02}, Y_{20}, Y_{22})$ in $Y_{i,j}^{QDCT}$ to $(Y_{00}-1, Y_{02}+1, Y_{20}+1, Y_{22}-1)$ in $Y_{i,j}^{QDCT'}$, the change between $Y_{i,j}^{QDCT}$ and $Y_{i,j}^{QDCT'}$ is given by (7):

$$\begin{aligned} \Delta Y_{i,j}^{QDCT} &= Y_{i,j}^{QDCT'} - Y_{i,j}^{QDCT} \\ &= \begin{bmatrix} -1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \end{aligned} \tag{7}$$

Then, change between the decoded residual block $X_{i,j}^{r'}$ and the perturbed residual block $X_{i,j}^{r}$ is calculated by (8):

$$\begin{aligned} \Delta X_{i,j}^{r} &= X_{i,j}^{r'} - X_{i,j}^{r} = C_r^T(\Delta Y_{i,j}^{QDCT} \times Q \bigotimes E_r)C_r \\ &= C_r^T(\begin{bmatrix} -1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \times Q \bigotimes E_r)C_r \\ &= Q \times \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & -4a^2 & -4a^2 & 0 \\ 0 & -4a^2 & -4a^2 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \end{aligned} \tag{8}$$

It is obvious from (8) that the reference pixels have also not been changed. Therefore, this operation will not cause intra-frame distortion drift.

In summary, when the proposed method in Prop 2 is used to embed data, the reference coefficient of the current macroblock has not changed. Therefore, the distortion caused by the embedded information is strictly limited to the current macroblock without drifting to adjacent macroblocks. It will effectively avoid the problem of intra-frame distortion drift. □

IV. PROPOSED ALGORITHM FOR DATA HIDING

According to ***Prop 1*** and ***2***, an improved high capacity and efficient data hiding scheme for 3D videos without distortion drift is proposed. When embedding and extracting hidden data, the algorithm does not need to pre-read the prediction mode and macroblock type of the original video, and does not need to distinguish between intra-frame macroblocks and inter-frame macroblocks. The hiding data is embedded and extracted only according to the integer DCT transform of 4×4.

Compared with algorithms in [34] and [35], the proposed method will embed data into all residual coefficient with 4×4 transformation matrix, including inter-MBs and intra-MBs. As long as they use 4×4 transformation matrix when performing integer DCT, both intra-MBs and inter-MBs can be used to hide data without intra-frame distortion drift. This greatly improves the embedding capacity of the algorithm, and it is more suitable for the coding structure of 3D video.

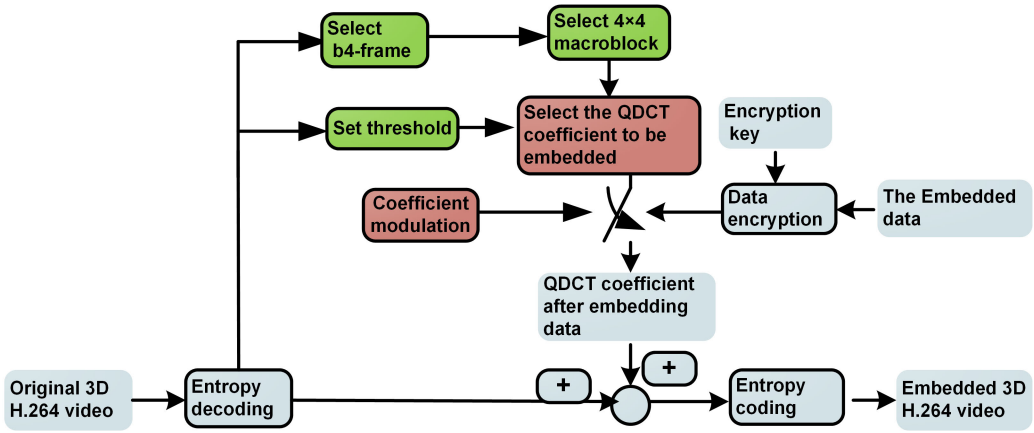


FIGURE 6. Proposed data hiding method diagram in embedding.

Algorithm 1 Proposed Data Hiding Method in Embeddings

Input: M, GOP, T, KEY
Output: GOP'
begain:
$GOP' = \varnothing$
encrypt:$M \rightarrow M' = (m_i)$
foreach decoded frame in GOP $F = (f_i)$
  if $F \in \{b4\}$
    decoded MB in frame $CUR\_MB = (mb_i)$
    if $CUR\_MB \in \{inter\_MB \quad OR \quad intra\_MB\} \quad AND \quad Trans = 4 \times 4$ then
      decoded sub_MB in MB $sub\_MB = (sub\_mb_i)$ do
      Obtain QDCT coefficients $Y_{i,j}^{QDCT}$
        if $|Y_{00}^i| > T$ then
          if $(|Y_{00}^i \%2) \bigoplus m_i == 1 \quad AND \quad Y_{00}^i \geq 0$ then
            $(Y_{00}, Y_{02}, Y_{20}, Y_{22}) \rightarrow (Y_{00}+1, Y_{02}-1, Y_{20}-1, Y_{22}+1)$
          if $(|Y_{00}^i \%2) \bigoplus m_i == 1 \quad AND \quad Y_{00}^i < 0$ then
            $(Y_{00}, Y_{02}, Y_{20}, Y_{22}) \rightarrow (Y_{00}-1, Y_{02}+1, Y_{20}+1, Y_{22}-1)$
          else $(Y_{00}, Y_{02}, Y_{20}, Y_{22}) \rightarrow (Y_{00}, Y_{02}, Y_{20}, Y_{22})$
        else goto $next \quad sub\_mb_i$
      else goto $next \quad mb_i$
    else goto $next \quad f_i$
  until M is completely embedded
return GOP'
end

A. DATA EMBEDDING ALGORITHM AND PROCESS

The carrier selected by this algorithm is still the qualified QDCT coefficient in the 3D H.264 code stream. Firstly, we obtain the b4 frame that meets the condition of Prop 1, and further obtain the sub-macroblock in which the 4×4 matrix transformation is used. To ensure video quality, a threshold is set to select a sub-macroblock with a larger residual value as a candidate carrier. Secondly, the data is embedded using a random LSB method [15]. After the embedding is completed, the modulated QDCT coefficients are entropy coded to obtain a secret video. Then the secret information embedding operation is completed. Fig. 6 depicts the embed process structure of the proposed method.

To achieve the purpose of maintaining video quality and security, the idea of threshold and random embedding are introduced. In the selection of embedded macroblocks, the AC coefficient (Y_{00}) is used as the threshold judgment basis, and the threshold is set as the limiting condition [34]. At the same time, in the selection of embedded macroblocks, the idea of randomness is introduced. The embedding position, random seed and threshold are shared as keys. The specific algorithm flow is presented in **Algorithm 1**.

TABLE 2. Parameters and descriptions in Algorithm 1.

Symbol	Descriptions
M	The embedded data
$M' = (m_i)$	Encrypted binary sequence of the embedded data
GOP	Original video sequence group
T	Candidate sub-macroblock threshold
GOP'	Embedded video sequence group
$F = (f_i)$	Decoded frame sequence
$\{b4\}$	Collection of b4 frames
$CUR\_MB = (mb_i)$	Decoded current macroblock sequence
$\{inter\_MB \quad OR \quad intra\_MB\}$	Collection of inter and intra macro blocks
$sub\_MB = (sub\_mb_i)$	Current macroblock and its sub-macroblock sequence

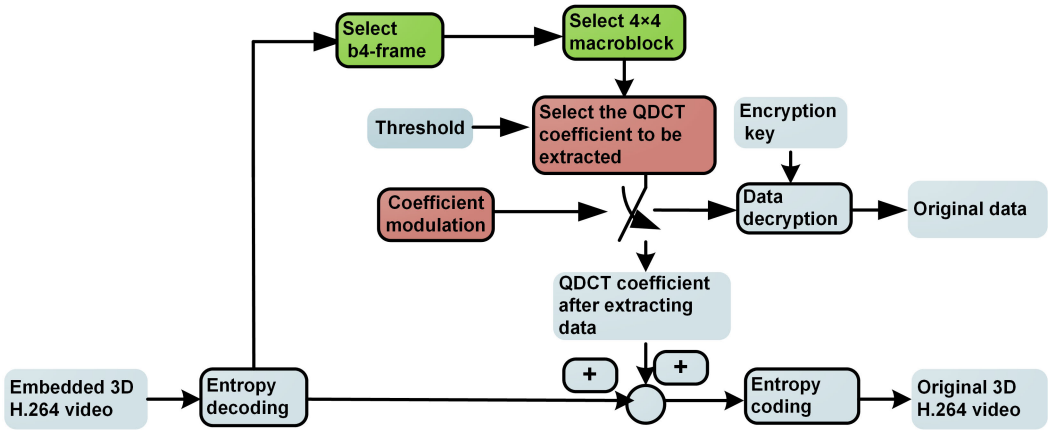


FIGURE 7. Proposed data hiding method diagram in extracting.

The parameters and explanations involved are illustrated in Table2.

B. DATA EXTRACTION ALGORITHM AND PROCESS

The algorithm extracts hidden information from the qualified QDCT coefficients of the dense 3D H.264 stream. The specific process is shown in Fig 7. And the specific algorithm flow is illustrated in Algorithm 2.

C. SECURITY ANALYSIS AND DISCUSSION

In this subsection, we analyze the security of the proposed scheme in detail. To improve the security of the algorithm, the idea of a random sequence is employed. We have taken advantage of two random sequences (RS_1, RS_2) to select the frames, MBs and coefficients for hiding data respectively. Then we will discuss the range of the two random sequences.

Through the study of Section II, it is obtained that there are four b4-frames in a GOP (Length=8, Total frames=16 with two viewpoints). This means that one quarter frames are b4-frames in the 3D MVC videos. Supposing the total GOP of the videos is N, then the total number of the b4-frames is $4N$.

If one frame is selected for hiding data, we can draw that:

$$|RS_1|_1 = C_{4N}^1$$

If two frames are selected for hiding data, we can obtain that

$$|RS_1|_2 = C_{4N}^2$$

And so forth, if k frames are selected for hiding data, we can get that

$$|RS_1|_k = C_{4N}^k$$

The total length of the random sequence is calculated as (9):

$$\begin{aligned} |RS_1| &= C_{4N}^1 + C_{4N}^2 + \cdots + C_{4N}^{4N} + \cdots + C_{4N}^{4N} \\ &= 2^{4N} - 1 \end{aligned} \tag{9}$$

Next,we need to select the sub-MBs with randomly. All of the 4×4 sub-MBs can be used to embed data. Then, we have:

$$\begin{aligned} |RS_2| &= C_{16}^1 + C_{16}^2 + \cdots + C_{16}^{16} \\ &= 2^{16} - 1 \end{aligned} \tag{10}$$

Algorithm 2 Proposed Data Hiding Method in Extracting

Input: GOP', T, KEY
Output: M
begain:
$M = \varnothing$
foreach decoded frame in $GOP'F = (f_i)$ do
  if $F \in \{b4\}$
    decoded MB in frame $CUR\_MB = (mb_i)$ do
    if $CUR\_MB \in \{inter\_MB \quad OR \quad intra\_MB\} \quad AND \quad Trans = 4 \times 4$ then
      decoded sub_MB in MB $sub\_MB = (sub\_mb_i)$ do
      Obtain QDCT coefficients $Y_{i,j}^{QDCT}$
        else goto $next \quad sub\_mb_i$
      else goto $next \quad mb_i$
    else goto $next \quad f_i$
  until M' is completely embedded
return M'
decrypt $M' \rightarrow M$
end

Analogously, the overall complexity is calculated as (11):

$$\begin{aligned} |RS| &= |RS_1| \times |RS_2| \\ &= (2^{4N} - 1) \times (2^{16} - 1) \end{aligned} \tag{11}$$

Then, it can be concluded that Algorithm 1 maintains great randomness through two random sequences. More importantly, data is hided with LSB randomly, which remains the statistical distribution of the concerned DCT coefficients almost unchanged after embedding.

V. EXPERIMENTAL RESULTS AND DISCUSSIONS

To verify the effectiveness of the proposed algorithm, this section embeds and extracts the algorithm for 9 groups of videos, and conduct detailed analysis of the its performance. All experiments are performed on a computer with a CPU of Intel Core i5-9600K 3.70GHz and a memory of 16.00GB. The test platform uses the latest H.264 official recommended platform JM19.0, which has shown positive support for multi-view 3D video. The video sequences of Akko& Kayo, Ballroom, Crowd, Exit, Flamenco, Objects, Race, Rena and Vassar are used for the experiments. To ensure the accuracy of the test results, the selected test video covers typical sequences such as high-speed motion, medium-speed motion, slow-speed motion, and quasi-static motion. The length of each GOP sequence is 8, the video is a 3D H.264 video with two viewpoints, and the coding structure is a standard layered B-frame structure, as is shown in Fig 1. The resolution of all test sequences is 640 × 480, that is, each frame contains 1200 macroblocks. The quantization parameter used in encoding and decoding is 28, the frame rate is 30 frames per second, and the encoding length is 300 frames. Moreover, compared with the literature [34], corresponding evaluation indexes are introduced to measure the performance of the algorithm. All the evaluation indicators and corresponding explanations are given in Table 3. Besides using PSNR to measure the quality of the embedded video frames, the well-known SSIM index [51] is adopted to measure the visual quality.

TABLE 3. Parameters and descriptions in experimental evaluation.

Symbol	Descriptions
$PSNR1$	The comparison between the original YUV and the corresponding decoded YUV.
$PSNR2$	The comparison between the original YUV and the corresponding decoded YUV of data hiding.
$SSIM$	Structural similarity
$Capacity$	The average number of embedded bits per frame
$Decodingtime$	The average time for an operation to decode a frame

A. ANALYSIS OF THE INFLUENCE OF PARAMETERS ON THE ALGORITHM

1) THE STATISTICAL PROPERTIES OF THE B4-FRAMES

The embedding space selected by the algorithm is the macroblock of b4-frame. Inevitably the statistical characteristics of b4-frame need to be analyzed first. The main prediction mode in b4-frame is inter prediction, so there will be more inter macroblocks and few intra macroblocks in b4-frame (including skip macroblocks without any information). For 9 sets of test videos, the average macroblock type distribution of each frame is counted, as shown in Fig 8.

It can be found that most of the macroblocks in b4-frame are Skip macroblocks, accounting for about 76.6% of all macroblocks. Since they do not contain any residual coefficients, the corresponding coefficients are only obtained through prediction relationships. They are generally not suitable for using as the carrier for data hiding. Except for Skip macroblocks, the remaining macroblocks are inter-frame macroblocks and intra-frame macroblocks. The former accounts for

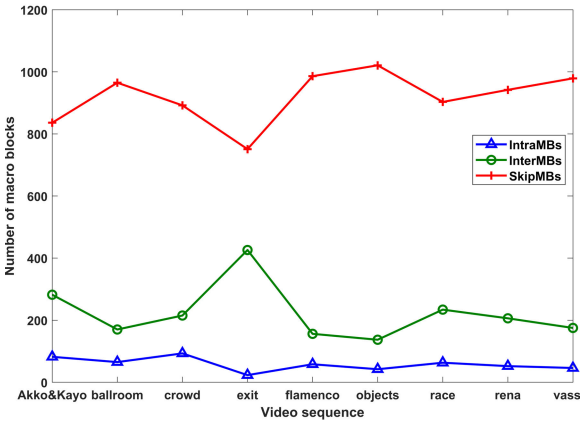


FIGURE 8. The statistical properties of the b4-frame.

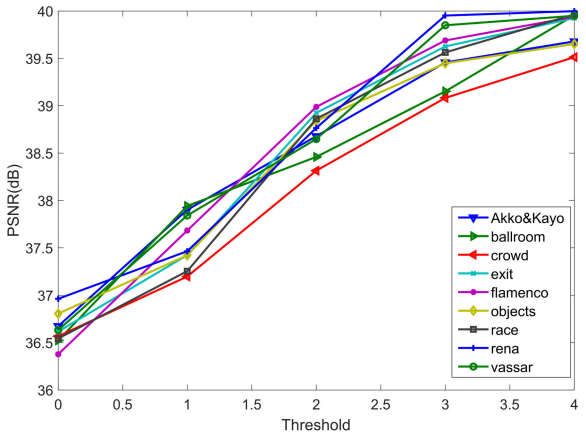


FIGURE 9. The PSNR value under different thresholds.

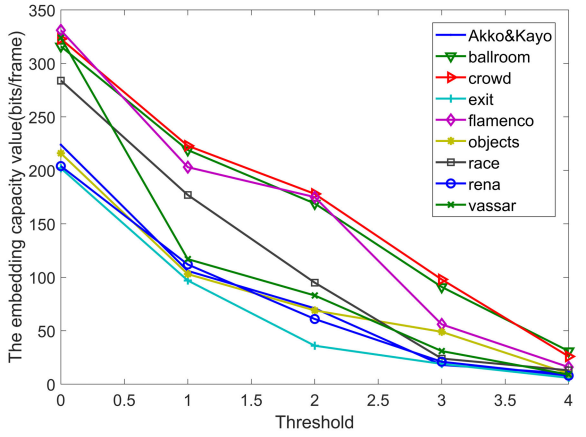


FIGURE 10. The embedding capacity value under different thresholds.

approximately 18.6% of all macroblocks, while intra-frame macroblocks only account for approximately 4.8% of the total number of macroblocks. Then, the traditional 2D video distortion-free steganography algorithm will encounter a large embedding capacity problem in the b4-frame, and too little embedding space will bring hidden dangers to steganography. Using our algorithm, data can be embedded into all MBs with 4×4 transformation matrix, which greatly improve the embedding capacity. This also confirms that the algorithm proposed in this paper is more suitable for 3D H.264 video from another aspect.

2) THE EFFECT OF THRESHOLD ON THE QUALITY OF EMBEDDED VIDEO

Another aspect that affects the performance of the algorithm is the selection of thresholds. A smaller threshold will bring greater embedding capacity, but it will lead to a decrease in video quality. On the contrary, a larger threshold will result in a reduction in embedding capacity, but it will cause less degradation in video quality after embedding. Therefore, there is a trade-off between embedding capacity and video quality.

To this end, the effect of threshold on video quality is analyzed through experiments. The average PSNR of the video after embedding information under different thresholds is shown in Fig 9. It is obvious that with the increasing threshold, the quality of the video becomes higher and higher. When the threshold is greater than or equal to 2, the PSNR value after embedding data is basically greater than 38dB, and the human eye can no longer perceive the decline in video quality. When the threshold is greater than or equal to 4, the PSNR value after embedding data exceeds 39dB, and the data hiding has little effect on the video quality.

3) THE RELATIONSHIP BETWEEN THRESHOLD AND INFORMATION EMBEDDING CAPACITY

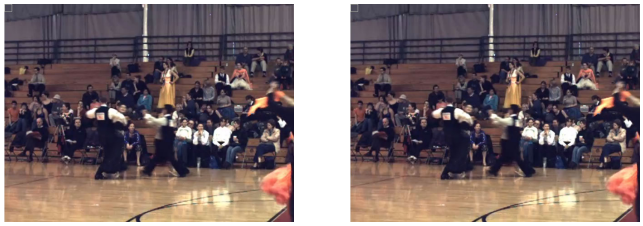
In addition, the embedding capacity under different thresholds is counted through experiments. For 9 sets of test videos, the thresholds are specified as $0, 1, 2, 3, 4$ respectively. We execute the algorithm, count the embedded capacity information, and calculate the average embedded capacity of each frame. Results in Fig 10 show that, with the increasing threshold, the embedded capacity of the video becomes lower and lower. When the threshold is greater than or equal to 4, the average number of embedded bits per frame is less than 20. When the threshold is less than or equal to 2, the embedding capacity of each frame of the video is about 50 bits.

Combined with Fig 9, it can be concluded that, as the threshold increases, the capacity of information embedding decreases rapidly, but the quality of the video becomes higher and higher. On the other hand, as the threshold decreases, the capacity of information embedding increases significantly, but it will degrade the video quality. Therefore, in practical applications, the threshold should be selected reasonably according to specific needs to achieve a balance between embedding capacity and video quality.

4) EXPERIMENTAL PERFORMANCE OF VISUAL QUALITY

To analyze and visually evaluate the change of video quality before and after embedding, this section discusses the change of the contrast image before and after embedding information

for a given video, further confirming the invisibility of the algorithm. According to the above discussion, let $T = 2$ as the threshold, and the embedding position is chosen randomly. Then we execute the algorithm, save the same frame image before and after video embedding for comparison. To test the accuracy of the results, four groups of videos, Ballroom, Crowd, Rena, and Akko & Kayo, are selected as carriers according to the degree of exercise urgency. Select the first b4-frame of each group of videos as the comparison image. The results are depicted in Fig.11-14.

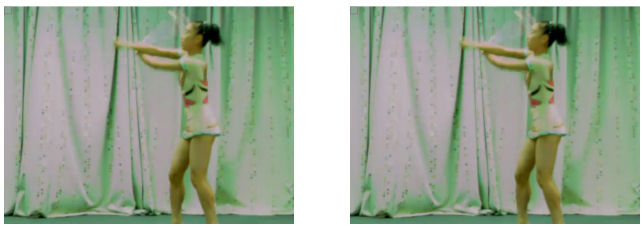


(a) (b)

FIGURE 11. Original video frames of and embedded video frames in Ballroom sequence by (a) Original video frames and (b) embedded video frames.

(a) (b)

FIGURE 12. Original video frames of and embedded video frames in Crowd sequence by (a) Original video frames and (b) embedded video frames.



(a) (b)

FIGURE 13. Original video frames of and embedded video frames in Rena sequence by (a) Original video frames and (b) embedded video frames.

It can be seen from Fig.11 that, for the Ballroom video with relatively intense motion, the video quality before and after information embedding has not decreased significantly. The human eye can hardly able to recognize the video change. This shows that the algorithm has good invisibility, and the embedded information will not significantly change the video quality. Meanwhile, Fig 12-14 further verify this conclusion from more test sequences with different degrees of exercise relief.

(a) (b)

FIGURE 14. Original video frames of and embedded video frames in Akko&Kayo sequence by (a) Original video frames and (b) embedded video frames.

B. ALGORITHM COMPARISON AND DISCUSSION

To objectively evaluate the performance of the algorithm, a comparative experiment is conducted between the proposed algorithm and the algorithms in [34] and [35]. The evaluation parameters used are given in Table 3. The embedding thresholds selected by the three algorithms are all $T = 2$. The quantization step size $QP = 28$, and the length of each GOP sequence is 8. The evaluation and analysis of the algorithm is carried out from the following four aspects.

1) EXPERIMENTAL PERFORMANCE OF VISUAL QUALITY WITH PSNR

According to the definition in [51], the corresponding two peak signal-to-noise ratios are used to evaluate the performance of the three algorithms. The specific results are shown in Table 4.

Table 4 provides the comparison of the PSNR increase performance between the proposed method and the methods in [34] and [35]. Results in Table 4 show that, after embedding data, the video quality of our algorithm is not significantly reduced compared with the algorithms in [34], [35]. The reason is that, the embedding capacity of our algorithm far exceeds the above two algorithms under the same conditions. Because more information is embedded than the first two algorithms, the video quality is slightly reduced. And the average decrease in PSNR is less than 0.0005dB, which is almost negligible. More importantly, the average PSNR of the video after the algorithm embedded exceeds 38dB, and the human visual system cannot detect the change in video quality. Therefore, compared with other algorithms, our algorithm also has good concealment and achieves the purpose of safe embedding.

2) EXPERIMENTAL PERFORMANCE OF VISUAL QUALITY WITH SSIM

To further verify the performance of the algorithm, structural similarity is introduced, and comparative experiments are conducted between the proposed algorithm and algorithms in [34], [35]. For better comparison, the experiments in this section also count the PSNR value after embedding data. The specific performance of the three algorithms is shown in Table 5.

TABLE 4. The comparison of PSNR between the algorithm in [34], [35] and the proposed algorithm for QP = 28 (dB).

Video sequence	PSNR(dB)	The methods in [34]	The methods in [35]	The proposed method
Akko&Kayo	PSNR1	39.679	39.679	39.679
	PSNR2	39.6784	39.6783	39.6751
	DPSNR	0.0006	0.0007	0.0039
Ballroom	PSNR1	36.955	36.955	36.955
	PSNR2	36.9544	36.9542	36.9459
	DPSNR	0.0006	0.0008	0.0091
Crowd	PSNR1	35.8218	35.8218	35.8218
	PSNR2	35.8216	35.8213	35.8153
	DPSNR	0.0002	0.0005	0.0065
Exit	PSNR1	38.4256	38.4256	38.4256
	PSNR2	38.4255	38.4254	38.4248
	DPSNR	0.0001	0.0002	0.0008
Flamenco	PSNR1	39.3892	39.3892	39.3892
	PSNR2	39.3889	39.3886	39.3841
	DPSNR	0.0003	0.0006	0.0051
Objects	PSNR1	37.8566	37.8566	37.8566
	PSNR2	37.8565	37.8564	37.855
	DPSNR	0.0001	0.0002	0.0016
Race	PSNR1	37.564	37.564	37.564
	PSNR2	37.5635	37.5632	37.5626
	DPSNR	0.0005	0.0008	0.0014
Rena	PSNR1	41.466	41.466	41.466
	PSNR2	41.4659	41.4658	41.4651
	DPSNR	0.0001	0.0002	0.0009
Vassar	PSNR1	36.6493	36.6493	36.6493
	PSNR2	36.6491	36.6489	36.6481
	DPSNR	0.0002	0.0004	0.0012

It can be seen from Table 5 that, compared with the algorithms in [34], [35], our algorithm has a certain degree of decline in the video after embedding information. The reason for this situation is mainly because the embedding ability of the algorithm is much larger than the other two algorithms, but it has similar visual effects to the two algorithms. In the next section, we will discuss the comparison of embedding capacity, which can be seen more clearly.

3) EXPERIMENTAL PERFORMANCE OF EMBEDDING CAPACITY

This section discusses the comparative testing of the three algorithms in terms of embedded capacity. As a very important indicator of video data hiding, the importance of embedding capacity for the steganography algorithm is self-evident. For 9 groups of videos, given the same threshold and quantization step size, Table 6 shows the comparison test of the three algorithms in terms of embedding capacity.

As it is shown in Table 6, compared with the algorithms in [34], [35], the proposed algorithm has greatly improved the embedding capacity while maintaining similar visual effects. The average rate of improvement per frame is close to 200%. This indicates that the algorithms in [34], [35] are not suitable for 3D videos with hierarchical B-frames prediction structure. The embedding capacity in the proposed algorithm can be improved because we take full advantage of the MBs with 4×4 transformation matrix, including not only intra-frame macroblocks, but also inter-frame macroblocks. Therefore, the proposed algorithm shows better performance in terms of embedding capacity, and can maintain the similar human visual effect with the algorithms in [34], [35].

TABLE 5. The comparison of SSIM between the algorithm in [34], [35] and the proposed algorithm for QP = 28.

Video sequence	PSNR(dB) and SSIM	The methods in [34]	The methods in [35]	The proposed method
Akko&Kayo	PSNR	39.6784	39.6783	39.6751
	SSIM	0.9941	0.9932	0.993
Ballroom	PSNR	36.9544	36.9542	36.9459
	SSIM	0.9848	0.9846	0.9841
Crowd	PSNR	35.8216	35.8213	35.8153
	SSIM	0.9735	0.9734	0.9731
Exit	PSNR	38.4255	38.4254	38.4248
	SSIM	0.9936	0.9934	0.9931
Flamenco	PSNR	39.3889	39.3886	39.3841
	SSIM	0.9926	0.9925	0.9921
Objects	PSNR	37.8565	37.8564	37.855
	SSIM	0.9941	0.9932	0.993
Race	PSNR	37.5635	37.5632	37.5626
	SSIM	0.9768	0.9765	0.976
Rena	PSNR	41.4659	41.4658	41.4651
	SSIM	0.9956	0.9954	0.9951
Vassar	PSNR	36.6491	36.6489	36.6481
	SSIM	0.9739	0.9736	0.9732
Average	PSNR	38.2004	38.2002	38.1973
	SSIM	0.9866	0.9862	0.9859

TABLE 6. The comparison of embedding capacity between the algorithm in [34], [35] and the proposed algorithm for QP = 28 (bits/frame).

Video sequence	The methods in [34]	The methods in [35]	The proposed method
Akko&Kayo	32	45	263
Ballroom	65	108	367
Crowd	128	190	509
Exit	95	159	416
Flamenco	111	196	525
Objects	67	132	405
Race	157	231	619
Rena	56	106	592
Vassar	201	306	596

4) EXPERIMENTAL PERFORMANCE OF EFFECTIVENESS

Execution efficiency is another major indicator to measure steganography algorithms. To comparison with the literature [34], [35], the decoding time of performing the embedding operation is employed to measure the efficiency of the algorithm execution. The algorithms in [34] and [35] must perform a complete pre-reading of the video before performing embedding to obtain data, such as macroblock type and prediction mode. Therefore, it is necessary to calculate the decoding time with the pre-reading time and without the pre-reading time, separately. Table 7 shows the comparison result of the three algorithms in decoding time.

The execution efficiency of the proposed algorithm is superior to the algorithms in [34], [35]. The reason is that the algorithms in [34], [35] need to pre-read the complete video during the execution process, and do coefficient compensation according to the prediction mode to limit the distortion drift. Therefore, the algorithm has higher complexity and lower execution efficiency, which is reflected in the longer decoding time used when embedding data. The average decoding time of the algorithm is about 40% of the two algorithms. Even if the read-ahead time is not considered, the average time of our algorithm is still much lower than the algorithms proposed by the literature [34], [35], which is about 55%. It basically meets the needs of real-time processing.

5) THE ANALYSIS AND EXPERIMENTAL PERFORMANCE OF BIT-RATE INCREASE

As an important indicator to check the invisibility of the embedding algorithm, the change of video bit rate before and after data embedding can help us further verify the effectiveness of the algorithm. The lower bit-rate increase means

TABLE 7. The comparison of decoding time between the algorithm in [34], [35] and the proposed algorithm for QP = 28 (ms/frame).

Video sequence	The methods in [34](ms/frame)		The methods in [35](ms/frame)		The proposed method(ms/frame)
	With pre-reading time	Without pre-reading time	With pre-reading time	Without pre-reading time	
Akko&Kayo	36.928	27.563	37.662	28.326	17.236
Ballroom	37.659	28.365	38.026	28.596	17.265
Crowd	37.862	28.486	38.236	28.96	17.521
Exit	37.713	28.399	38.215	28.886	17.428
Flamenco	37.807	28.411	38.225	28.803	17.506
Objects	37.509	28.302	37.987	28.487	17.242
Race	38.219	28.901	38.563	29.021	18.391
Rena	37.556	28.32	37.952	28.401	18.088
Vassar	38.401	28.963	38.964	29.113	18.561

that the data embedding operation has a lower impact on the original video, and the algorithm has better invisibility.

Compared to similar algorithms, the proposed algorithm performs well in bit-rate increase, which mainly due to the following three aspects. First, the proposed algorithm is performed in the entropy encoding stage, which is the last step of video compression. Thus, the algorithm is very efficient because it does not involve complex operations such as rate-distortion optimization (RDO) or full decoding. Second, the embedding method we use is coefficient modulation and compensation. It is the same as simple constant bit replacement, which greatly reduces the chance of bit-rate increase. Last but not least, the proposed algorithm strictly limits the problem of distortion drift, and the error caused by the embedded data only exists in the current macroblock, which further reduces the bit-rate increase caused by video encoding. Therefore, compared to other algorithms, the proposed algorithm has a lower bit-rate increase after hiding data.

We further verified this through comparative experiments. Tables 8 provides the comparison of the increase in bit-rate between the proposed method and the methods in [34] and [35] when embedding the same data. The average increase in bitrate of the proposed method is 0.85%, compared with 0.90% of methods in [34] and 0.96% in [35]. It shows that the proposed scheme can obtain lower increase in bitrate after hiding data when compared with related schemes. And the average increase in bit rate is less than 1%, which indicates that the proposed algorithm has high invisibility.

6) ROBUSTNESS ANALYSIS AND DISCUSSION

As with the methods in literature [34] and [35], the proposed algorithm accurately extracts hidden data without any attack. However, the embedded data will not be extracted correctly after some attacks, such as the unreliable transmission situations of error bits and re-quantization attacks. These attacks, such as re-encoding, will lead to changes in prediction mode and DCT coefficients during the video decoding process.

TABLE 8. The comparison of bit-rate increase performance between the algorithm in [34], [35] and the proposed algorithm for QP = 28 (%).

Video sequence	The methods in [34](%)	The methods in [35](%)	The proposed method(%)
Akko&Kayo	0.76	0.82	0.81
Ballroom	0.86	0.93	0.85
Crowd	1.03	1.06	0.94
Exit	0.66	0.77	0.65
Flamenco	0.99	1.02	0.92
Objects	0.87	0.91	0.82
Race	1.06	1.08	0.91
Rena	0.85	0.92	0.88
Vassar	1.02	1.09	0.84
Average	0.9	0.96	0.85

This will make it impossible to determine the embedding and extraction positions, and thus fail to extract data correctly. Therefore, the proposed algorithm is not robust, which means that the embedded data may be lost when hitting by various attacks. To further improve the robustness, various redundant codes, such as BCH, convolutional codes, etc., can be used before embedding information [19], [22], [25], [30], [36] and [47].

Although it is not robust, with high capacity and efficiency, the proposed algorithm can be applied to video copyright protection, covert communication and other fields. Next, we will work on devising methods to further improve the robustness and security of the algorithm, and the literatures [19], [22], etc., provide a good source of reference for our future work.

VI. CONCLUSION AND FUTURE WORK

Based on the integer DCT transform and quantization process in the 3D H.264 encoding process, this paper derives an intra-frame distortion-free steganography algorithm based on

full coefficient compensation. Combined with the nature of b4-frames, the algorithm can effectively avoid intra-frame and inter-frame distortion drift. Compared with the existing similar research, the algorithm has the following contributions: First, the algorithm does not need to pre-read the original video, and does not need to obtain the video prediction mode and macroblock type information in advance, which greatly improves the execution efficiency of the algorithm. In addition, the proposed scheme takes full advantage of inter-MBs and all MBs with 4×4 IDCT to hide data. Moreover, data is hided into MBs directly without the data of MB-type and prediction mode by the proposed algorithm. This greatly reduces the complexity of the embedding operation, and is more suitable for real-time processing. Second, since the embedded carrier can be either the inter-frame macroblock or the intra-frame macroblock, the proposed algorithm is more suitable for 3D H.264 video. With loose restrictions, huge embedding capacity, and good versatility, the proposed algorithm can provide a broad space for the study of robustness and reversibility.

Since the operation position of the coefficient modulation selected by the proposed algorithm is relatively fixed, and in the case of high security requirements, it is necessary to introduce stronger randomness to control the embedding position. The proposed algorithm uses the LSB algorithm during data embedding and extraction, and it can randomly select frames, macroblocks, sub-macroblocks, etc. Therefore, the randomness of the algorithm is increased, and the anti-steganography ability of the algorithm is not lower than that of the random LSB algorithm. Future work will focus on further improving the security and robustness of the algorithm.

REFERENCES

[1] J. J. Chae and B. S. Manjunath, "Data hiding in video," in *Proc. Int. Conf. Image Process.*, Kobe, Japan, 1999, pp. 311–315.

[2] M. Asikuzzaman and M. R. Pickering, "An overview of digital video watermarking," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 28, no. 9, pp. 2131–2153, Sep. 2018.

[3] Y. Tew and K. Wong, "An overview of information hiding in H.264/AVC compressed video," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 24, no. 2, pp. 305–319, Feb. 2014.

[4] X. Ma, Z. Li, J. Lv, and W. Wang, "Data hiding in H.264/AVC streams with limited intra-frame distortion drift," in *Proc. Int. Symp. Comput. Netw. Multimedia Technol.*, Wuhan, China, Dec. 2009, pp. 1–5.

[5] A. Smolic, K. Mueller, P. Merkle, C. Fehn, P. Kauff, P. Eisert, and T. Wiegand, "3D video and free viewpoint Video–technologies, applications and MPEG standards," in *Proc. IEEE Int. Conf. Multimedia Expo*, Toronto, ON, USA, Jul. 2006, pp. 2161–2164.

[6] *Draft I. Recommendation and Final Draft International Standard of Joint Video Specification*, document ITU-TRec.H.264| ISO/IEC 14496-10 AVC, JVTG050, Joint Video Team (JVT) of ISO/IEC MPEG and ITU-T VCEG, 2003.

[7] *Vetro A, Su Y, Kimata H, Joint multiview video model JMVM 2.0. ITU-T and ISO/IEC Joint Video Team*, document JVT-U207, 2006, pp. 1–36.

[8] A. Vetro, T. Wiegand, and G. J. Sullivan, "Overview of the stereo and multiview video coding extensions of the H.264/MPEG-4 AVC standard," *Proc. IEEE*, vol. 99, no. 4, pp. 626–642, Apr. 2011.

[9] P. Merkle, A. Smolic, K. Muller, and T. Wiegand, "Efficient prediction structures for multiview video coding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 17, no. 11, pp. 1461–1473, Nov. 2007.

[10] K. Muller, H. Schwarz, D. Marpe, C. Bartnik, S. Bosse, H. Brust, T. Hinz, H. Lakshman, P. Merkle, F. H. Rhee, G. Tech, M. Winken, and T. Wiegand, "3D high-efficiency video coding for multi-view video and depth data," *IEEE Trans. Image Process.*, vol. 22, no. 9, pp. 3366–3378, Sep. 2013.

[11] Y. Fu, Y. Guo, Y. Zhu, F. Liu, C. Song, and Z.-H. Zhou, "Multi-view video summarization," *IEEE Trans. Multimedia*, vol. 12, no. 7, pp. 717–729, Nov. 2010.

[12] P. Merkle, K. Muller, D. Marpe, and T. Wiegand, "Depth intra coding for 3D video based on geometric primitives," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 26, no. 3, pp. 570–582, Mar. 2016.

[13] J. Zhao, Z. T. Li, and B. Feng, "A novel two-dimensional histogram modification for reversible data embedding into stereo H.264 video," *Multimedia Tools Appl.*, vol. 75, no. 10, p. 5959–5980, 2016.

[14] Y. L. Chen, H. Wang, Y. Hu, and A. Malik, "Intra-frame error concealment scheme using 3D reversible data hiding in mobile cloud environment," *IEEE Access*, vol. 6, pp. 77004–77013, 2018.

[15] W. Bender, D. Gruhl, N. Morimoto, and A. Lu, "Techniques for data hiding," *IBM Syst. J.*, vol. 35, nos. 3–4, pp. 313–336, 1996.

[16] G. J. Sullivan, J.-R. Ohm, W.-J. Han, and T. Wiegand, "Overview of the high efficiency video coding (HEVC) standard," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 22, no. 12, pp. 1649–1668, Dec. 2012.

[17] Y.-H. Chen and V. Sze, "A deeply pipelined CABAC decoder for HEVC supporting level 6.2 high-tier applications," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 25, no. 5, pp. 856–868, May 2015.

[18] J.-R. Ohm, G. J. Sullivan, H. Schwarz, T. Keng Tan, and T. Wiegand, "Comparison of the coding efficiency of video coding Standards—Including high efficiency video coding (HEVC)," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 22, no. 12, pp. 1669–1684, Dec. 2012.

[19] R. J. Mstafa, K. M. Elleithy, and E. Abdelfattah, "A robust and secure video data hiding method in DWT-DCT domains based on multiple object tracking and ECC," *IEEE Access*, vol. 5, pp. 5354–5365, 2017.

[20] D. Xu, R. Wang, and J. Wang, "Prediction mode modulated data-hiding algorithm for H.264/AVC," *J. Real-Time Image Process.*, vol. 7, no. 4, pp. 205–214, Dec. 2012.

[21] H. A. Aly, "Data hiding in motion vectors of compressed video based on their associated prediction error," *IEEE Trans. Inf. Forensics Security*, vol. 6, no. 1, pp. 14–18, Mar. 2011.

[22] Y. Liu, M. Hu, X. Ma, and H. Zhao, "A new robust data hiding method for H.264/AVC without intra-frame distortion drift," *Neurocomputing*, vol. 151, pp. 1076–1085, Mar. 2015.

[23] T. Stutz, F. Autrusseau, and A. Uhl, "Non-blind structure-preserving substitution watermarking of H.264/CAVLC inter-frames," *IEEE Trans. Multimedia*, vol. 16, no. 5, pp. 1337–1349, Aug. 2014.

[24] P.-C. Chang, K.-L. Chung, J.-J. Chen, C.-H. Lin, and T.-J. Lin, "A DCT/DST-based error propagation-free data hiding algorithm for HEVC intra-coded frames," *J. Vis. Commun. Image Represent.*, vol. 25, no. 2, pp. 239–253, Feb. 2014.

[25] R. J. Mstafa and K. M. Elleithy, "A DCT-based robust video steganographic method using BCH error correcting codes," in *Proc. IEEE Long Island Syst., Appl. Technol. Conf. (LISAT)*, Apr. 2016, pp. 1–6.

[26] H. Zhang, Y. Cao, X. Zhao, H. Yu, and C. Liu, "Data hiding in H.264/AVC videos using the coded block pattern," in *Proc. Int. Workshop Digit. Watermarking*, vol. 2016, pp. 588–600.

[27] S. K. Kapotas, E. E. Varsaki, and A. N. Skodras, "Data hiding in H. 264 encoded video sequences," in *Proc. IEEE 9th Workshop Multimedia Signal Process.*, 2007, pp. 373–376.

[28] S. Lian, Z. Liu, Z. Ren, and H. Wang, "Commutative encryption and watermarking in video compression," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 17, no. 6, pp. 774–778, Jun. 2007.

[29] S. Chen and H. Leung, "A temporal approach for improving intra-frame concealment performance in H.264/AVC," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 19, no. 3, pp. 422–426, Mar. 2009.

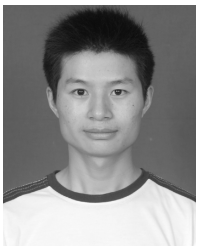
[30] M. Noorkami and R. M. Mersereau, "A framework for robust watermarking of H.264-encoded video with controllable detection performance," *IEEE Trans. Inf. Forensics Security*, vol. 2, no. 1, pp. 14–23, Mar. 2007.

[31] 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," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 20, no. 11, pp. 1643–1647, Nov. 2010.

[32] D. Xu, "Commutative encryption and data hiding in HEVC video compression," *IEEE Access*, vol. 7, pp. 66028–66041, 2019.

[33] B. Guan, D. Xu, and Q. Li, "An efficient commutative encryption and data hiding scheme for HEVC video," *IEEE Access*, vol. 8, pp. 60232–60245, 2020.

[34] 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 Trans. Circuits Syst. Video Technol.*, vol. 20, no. 10, pp. 1320–1330, Oct. 2010.
[35] 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," *J. Syst. Softw.*, vol. 86, no. 3, pp. 604–614, Mar. 2013.
[36] Y. Liu, H. Zhao, S. Liu, C. Feng, and S. Liu, "A robust and improved visual quality data hiding method for HEVC," *IEEE Access*, vol. 6, pp. 53984–53997, 2018.
[37] Y. Yao, W. Zhang, and N. Yu, "Inter-frame distortion drift analysis for reversible data hiding in encrypted H.264/AVC video bitstreams," *Signal Process.*, vol. 128, pp. 531–545, Nov. 2016.
[38] Y.-Q. Shi, X. Li, X. Zhang, H.-T. Wu, and B. Ma, "Reversible data hiding: Advances in the past two decades," *IEEE Access*, vol. 4, pp. 3210–3237, 2016.
[39] T. Shanableh, "Data hiding in MPEG video files using multivariate regression and flexible macroblock ordering," *IEEE Trans. Inf. Forensics Security*, vol. 7, no. 2, pp. 455–464, Apr. 2012.
[40] M. Fallahpour, S. Shirmohammadi, M. Semsarzadeh, and J. Zhao, "Tampering detection in compressed digital video using watermarking," *IEEE Trans. Instrum. Meas.*, vol. 63, no. 5, pp. 1057–1072, May 2014.
[41] W. Huo, Y. Zhu, and H. Chen, "A controllable error-drift elimination scheme for watermarking algorithm in H.264/AVC stream," *IEEE Signal Process. Lett.*, vol. 18, no. 9, pp. 535–538, Sep. 2011.
[42] T. Dutta and H. P. Gupta, "An efficient framework for compressed domain watermarking in p frames of high-efficiency video coding (HEVC)-encoded video," *ACM Trans. Multimedia Comput., Commun., Appl.*, vol. 13, no. 1, pp. 1–24, Jan. 2017.
[43] D. Xu, R. Wang, and Y. Q. Shi, "Data hiding in encrypted H.264/AVC video streams by codeword substitution," *IEEE Trans. Inf. Forensics Security*, vol. 9, no. 4, pp. 596–606, Apr. 2014.
[44] D. Xu, R. Wang, and Y. Q. Shi, "An improved scheme for data hiding in encrypted H.264/AVC videos," *J. Vis. Commun. Image Represent.*, vol. 36, pp. 229–242, Apr. 2016.
[45] Y. Liu, C. Yang, and Q. Sun, "Enhance embedding capacity of generalized exploiting modification directions in data hiding," *IEEE Access*, vol. 6, pp. 5374–5378, 2018.
[46] H. Mareen, J. De Praeter, G. Van Wallendael, and P. Lambert, "A novel video watermarking approach based on implicit distortions," *IEEE Trans. Consum. Electron.*, vol. 64, no. 3, pp. 250–258, Aug. 2018.
[47] Y. Zhou, C. Wang, and X. Zhou, "An Intra-Drift-Free robust watermarking algorithm in high efficiency video coding compressed domain," *IEEE Access*, vol. 7, pp. 132991–133007, 2019.
[48] H. Mareen, M. Courteaux, J. De Praeter, M. Asikuzzaman, G. Van Wallendael, and P. Lambert, "Rate-Distortion-Preserving forensic watermarking using quantization parameter variation," *IEEE Access*, vol. 8, pp. 63700–63709, 2020.
[49] Y. Xue, J. Zhou, H. Zeng, P. Zhong, and J. Wen, "An adaptive steganographic scheme for H.264/AVC video with distortion optimization," *Signal Process., Image Commun.*, vol. 76, pp. 22–30, Aug. 2019.
[50] T. Luo, L. Zuo, G. Jiang, W. Gao, H. Xu, and Q. Jiang, "Security of MVD-based 3D video in 3D-HEVC using data hiding and encryption," *J. Real-Time Image Process.*, vol. 17, no. 4, pp. 773–785, Aug. 2018.
[51] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: From error visibility to structural similarity," *IEEE Trans. Image Process.*, vol. 13, no. 4, pp. 600–612, Apr. 2004.



GUANG HUA SONG received the M.S. degree from the School of Mathematics and Statistics, Huazhong University of Science and Technology, China, in 2007, and the Ph.D. degree from the School of Computer Science and Technology, Huazhong University of Science and Technology, China, in 2014. He is currently an Assistant Professor with the School of Information and Safety Engineering, Zhongnan University of Economics and Law, China. His research interests include information security, data analysis, and artificial intelligence.

HUI LIU received the M.S. degree from the School of Mathematics and Statistics, Huazhong University of Science and Technology, China, in 2007, and the Ph.D. degree from the School of Management, Huazhong University of Science and Technology, in 2014. She is currently an Associate Professor with the Research Center of Hubei Logistics Development, Hubei University of Economics, China. Her research interests include artificial intelligence and intelligent optimization algorithm.