Multimedia Tools and Applications (2019) 78: 8423–8446
https://doi.org/10.1007/s11042-018-6859-7

High capacity and multilevel information hiding algorithm based on pu partition modes for HEVC videos

Yiyuan Yang[1] · Zhaohong Li[2] · Wenchao Xie[2] · Zhenzhen Zhang[3]

Received: 25 July 2018 / Revised: 9 October 2018 / Accepted: 6 November 2018 /
Published online: 26 November 2018

Abstract

In order to realize copyright protection and piracy tracking for ever-increasing videos, video information hiding has gained more and more attention. For the latest video coding standard — High Efficiency Video Coding (HEVC), most of the information hiding algorithms are based on the same characteristics as those in MPEG4 and H.264, such as DCT coefficients, intra prediction modules and motion vectors. However, few algorithms are reported to hide information based on the innovation modules introduced by HEVC. This paper has proposed a high capacity and multilevel information hiding algorithm based on PU partition modes in P-frames, which is considered to be one of the most important innovative features of HEVC. The proposed algorithm consists of two stages: first round calculation and modification process. The first round calculation is to record the PU partition modes selected by HEVC. The modification process is to force each of PU mode into one of the encoded groups according to the embedding binary bits, and then use the modified PU partition modes for the left HEVC encoding process. The experimental results show that the proposed algorithm can achieve high capacity and visual quality, and low bit rate increase, which significantly outperforms state-of-the-art works in embedding capacity.

Keywords Video data hiding · HEVC · High capacity · Multilevel · PU partitioning modes

1 Introduction

With the rapid development of Internet, information transmission has never been as convenient as nowadays. But at the same time, this brings in increasingly prominent information security issues, such as the copyright protection and piracy tracking, identity authentication

✉ Zhaohong Li
zhhli2@bjtu.edu.cn

[1] School of Computer Science and Engineering, Beihang University, Beijing, 100083, China

[2] School of Electronic and Information Engineering, Beijing Jiaotong University, Beijing, 100044, China

[3] School of Information and Engineering, Beijing Institute of Graphic Communication, Beijing, 100026, China

and authenticity identification of digital works. In order to solve these problems, information hiding technology comes into being. It takes audio, image, video and other digital media as the carrier, and embeds imperceptible hiding information into the carrier through a certain algorithm. Moreover, with the arrival of the era of big data, a large number of high-definition digital videos are transmitted on the channel. Compared with other digital media such as text, image and audio, digital video has larger redundancy which can provide large capacity for information hiding. Therefore, video information hiding has received more and more attention.

A large number of information hiding algorithms have been proposed for H.264 and MPEG-4 encoded videos. One common method is to modify the DCT coefficients to achieve the purpose of hiding information [3, 4, 18], which can offer large capacity for embedding information. Modifying intra prediction mode is also a common information hiding method [15, 17], which can embed and extract hiding information easily. In addition, there are some inter prediction based information hiding algorithms [20, 21], which can achieve high visual quality.

In 2013, the Joint Collaborative Team on Video Coding (JCT-VC) released a state-of-the-art video coding standard — high efficiency video coding. The overall architecture of HEVC is almost the same as that of H.264 and MPEG4, so data hiding algorithms proposed for H.264 and MPEG4 can be applied to HEVC videos. In the transform domain, Po-Chun Chang et al. [1, 6] firstly proposed a traditional DCT/DST-based information hiding algorithm for HEVC videos and one quality improvement scheme for the reconstructed intra-coded frames. In the field of intra prediction, Wang et al. [7, 10–12] proposed a series of information hiding algorithms for HEVC videos. Chen et al. [2] proposed an intra prediction based information hiding algorithm by using the improved diamond coding algorithm, which performs better than Wang's methods [7, 10–12] in capacity and visual quality. In the field of inter prediction, Yang and Li [16] firstly proposed an information hiding algorithm by modifying motion vectors in P-frames of HEVC videos. Tew and Wong [9] combined the CB (coding block) size decision based algorithm with odd-even embedding using nonzero coefficients to achieve a high payload. Xie et al. [14] also proposed an information hiding algorithm based on PU partition modes which has high visual quality.

Apart from the same parts as H.264 and MPEG4, HEVC has introduced some innovative technologies including flexible encoding architecture (coding Unit, Predict Unit and Transform Unit), RQT (Residual Quad-tree Transform), and SAO(Sample Adaptive Offset). At present, there are few information hiding algorithms proposed based on these HEVC specific modules, while the information hiding algorithms based on these special modules have great prospective. Inspired by this, an information hiding algorithm based on PU partition modes is proposed in this paper.

According to the video content, HEVC determines the PU partition modes for each CU. Different CU size has different PU partition modes. For 64×64, 32×32 and 16×16 CU size, PU partition modes can be divided into eight modes. For 8×8 CU size, there are only four PU partition modes. Because PU partition modes are so abundant, and lie in different sizes of CU, we can design a multilevel embedding method by modifying PU partition modes in multilevel selected CU instead of all the CU or one type of CU, which can meet different requirements of the embedding capacity, visual quality and bit rate. According to the embedding binary bits, the PU partition modes will be kept or modified into one assigned group, then the modified PU modes are used to finish the left HEVC encoding process. Overall, the proposed algorithm can achieve higher capacity and visual quality, and lower bit rate increase compared with the state-of-the-art works.

The rest of this paper is organized as follows. Section 2 analyzes the distribution of PU partition modes of HEVC videos at different resolutions. Section 3 presents the information hiding algorithm based on PU partition modes. Section 4 gives out the experimental results and analysis. Finally, conclusions are drawn in Section 5.

2 PU partition modes in HEVC

2.1 Block structures and prediction units in HEVC

In HEVC, a video sequence will be divided into several GOPs (groups of picture). In each GOP, pictures will be named as I (Intra), P (Predicted) and B (Bi-directionally predicted)-frames according to the order of them. In each frame, HEVC adopts block coding structure, and divides each frame into a number of coding tree units (CTU) which are added to the corresponding grammatical elements. In CTU, a quadtree structure is established. Every leaf node of CTU is called a CU and CUs at the same level must be 4 square code blocks of the same size. In addition, the optional sizes of CU are 64×64, 32×32, 16×16 and 8×8. Figure 1 gives an example for the partitioning of a 64×64 CTU and its quadtree.

CU is the basic coding unit and the lowest coding level of HEVC. In the prediction stage, HEVC will allocate prediction unit (PU) for each CU. A CU can be split into one or more PUs, and only one layer is allowed. At the same time, each CU also determines its own prediction mode and PU partition mode. There are two different prediction modes in HEVC: intra prediction and inter prediction. Intra prediction can be applied in all frames while inter prediction only can be applied in P and B-frames. In intra prediction, the entire CU can be coded as a single PU known as $2N\times2N$, or can be divided into four equal-sized square PUs known as $N\times N$. In inter prediction, the mode of PU partition is more diverse. As shown in Fig. 2, there are eight PU partition modes for inter prediction, including four symmetric PU partition modes ($2N\times2N$, $2N\times N$, $N\times2N$ and $N\times N$) and four asymmetrical PU partition modes ($2N\times nU$, $2N\times nD$, $nL\times2N$ and $nR\times2N$). The partition mode of PU can be selected from any of the eight modes in a 16×16 or greater CU. But for a 8×8 CU, in order to reduce

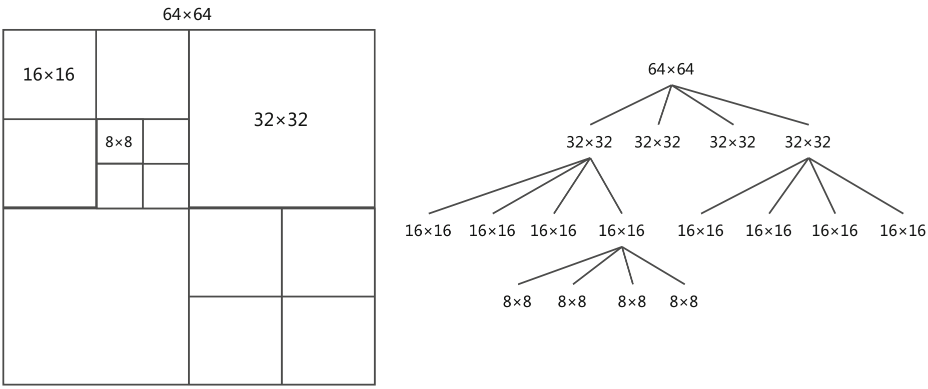


Fig. 1 Example for the partitioning of a 64×64 CTU into CUs of different sizes. The quadtree is shown on the right

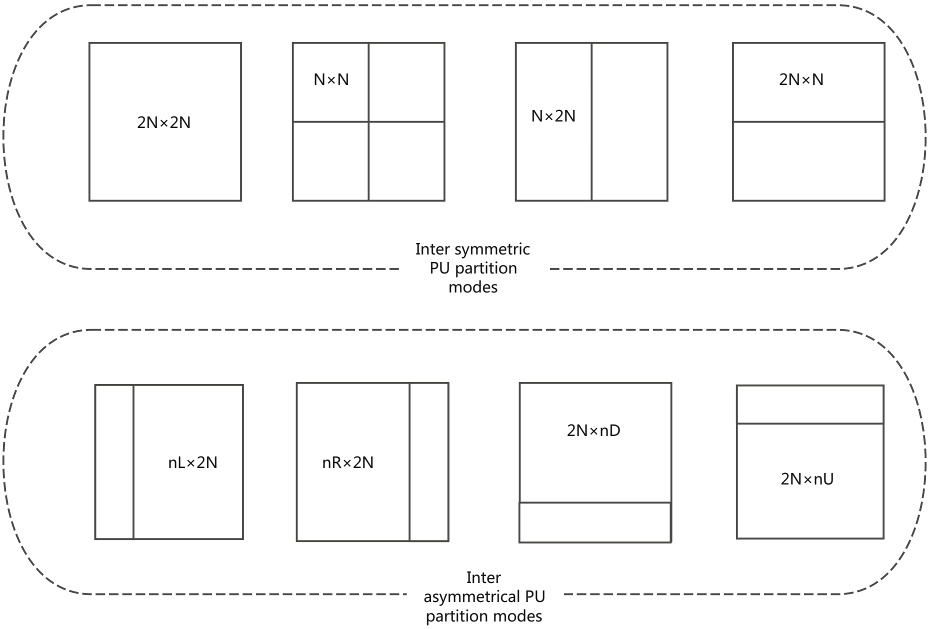


Fig. 2 Inter PU partition modes

the complexity of the algorithm, only four symmetric PU partition modes(2N×N, 2N×N, N×2N and N×N) are available.

The PU partition modes have an important impact on the compression quality of HEVC. The smaller the PU is subdivided, the more accurate the prediction of the motion of the object, and the higher the visual quality would be. But the improvement of visual quality is at the cost of the transmission bandwidth and storage space. Therefore, it is necessary for HEVC to find an optimal partition mode to minimize the expense of the transmission bandwidth and storage space and ensure high visual quality at the same time. In order to find the best PU partition mode, the method of recursion traversal is adopted in HEVC. For a CU, it traverses all the possible PU partition modes, and compares these PU partition modes to find the best PU partition mode by measuring the cost space, visual quality and so on.

2.2 The distribution of different PU partition modes

As mentioned before, there is only intra prediction in I-frames, in which PU partition modes are much less than those in P and B-frames. Since we hide information by changing the PU partition modes and use different modes to represent different information, P and B-frames are more suitable to hide information for their variable PU partition modes. Hence, the algorithm proposed in this paper is aimed at hiding information based on P and B-frames. However, when one PU partition mode is inappropriately modified, the entire CU architecture will be affected, which will ultimately affect the entire compression process and lead to some serious problems such as visual quality problems. Therefore, in order to ensure the efficiency of the algorithm proposed in this paper, in this section, we will take a in-depth and detailed discussion on the distribution of different PU partition modes at different resolutions.

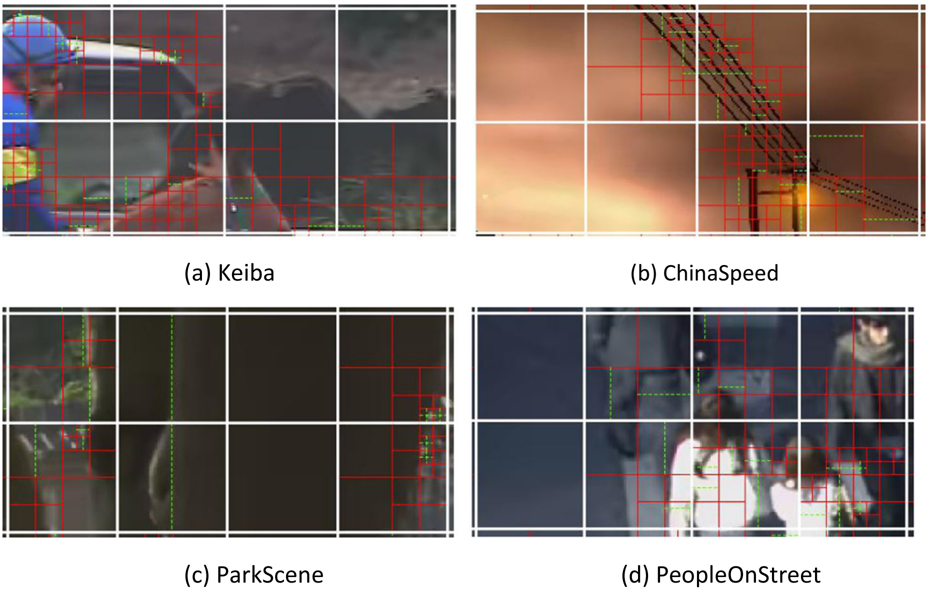


Fig. 3 Example of different CU structures and PU partition modes at different resolutions. The resolution of figure **a**, figure **b**, figure **c** and figure **d** are 832×480, 1024×768, 1920×1080 and 2560×1600 respectively

In a video sequence, the correlation in time domain between adjacent frames is very strong, and the difference between frames lies in the residual of moving objects, so the redundancy of inter frame information is very high. The flexible multiple PU partition modes proposed by HEVC is to remove redundant information between video frames and improve coding efficiency. For large static blocks in the video, such as the background, the CU structure and the PU partition modes are very simple, because their motion residuals are almost 0. For the moving objects, their boundary movement residuals are large so the more detailed CU and PU partitions are needed to adapt the more accurate shape of the moving target in the frame. Figure 3 lists CU and PU partition examples at different resolutions, in which the white line represents the boundary of the CTU, the red line represents the boundary of the CU, and the green line represents the PU partition boundary which is different form the size of the CU. It can be seen from Fig. 3 that although their resolutions are different, the background parts of the block, such as the sky in figure b, have simple CU structures and PU partitions. However, as for the boundary of moving objects, such as the people in figure d, CU and PU partitions are more detailed to fit the boundary shape of moving objects. Thus, the partition modes of CU and PU in a video are closely related to the video content.

Next, we will discuss the specific distribution of each CU and PU partition mode. It is acknowledged that resolution, compression parameters and video content will affect the distribution of PU partition modes. Therefore we use four different videos under different resolutions with QP = 32, one video with different QPs and two videos under the same resolution with the same QP to get the distribution of CU and PU partition modes in six frames, which are shown in Figs. 4, 5 and 6.

Figure 4 shows the number distribution of CU with different sizes under different resolutions, and the specific number distribution of different PU partition modes under CU of each size. We can see that, at any resolution, the PU partition mode of 2N×2N size is the

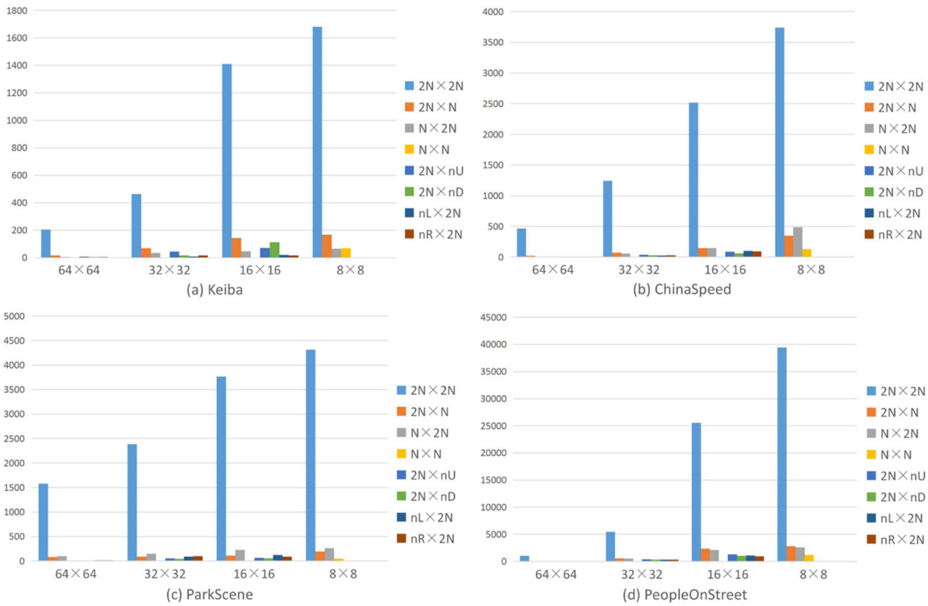


Fig. 4 Example of the distribution of different PU partition modes at different resolutions. Where the x axis represents CU of different sizes, and the y axis represents the number of PU. The resolutions of videos used in figure a, figure b, figure c and figure d are 832×480, 1024×768, 1920×1080 and 2560×1600 respectively

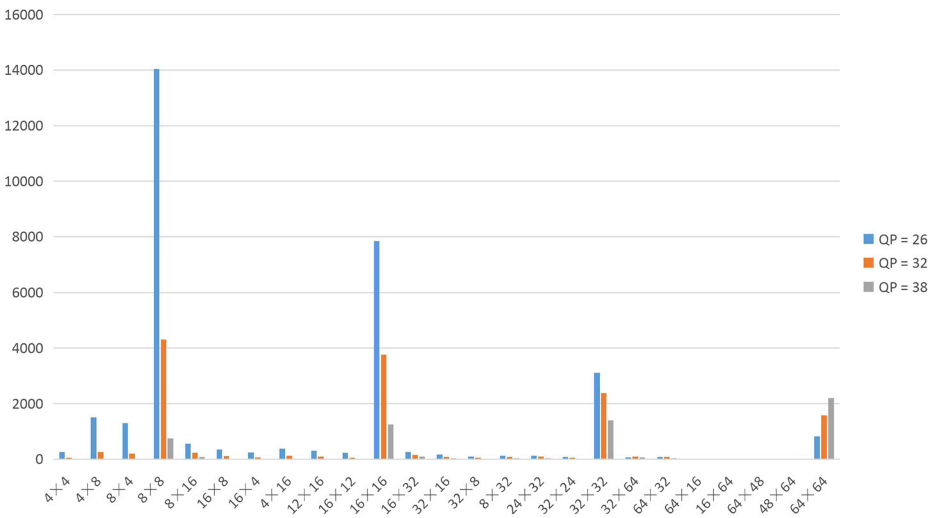


Fig. 5 Example of the distribution of different PU partition modes with different QPs. Where the x axis represents different PU partition modes, and the y axis represents the number of PU. The used video is ParkScene and its resolution is 1920×1080

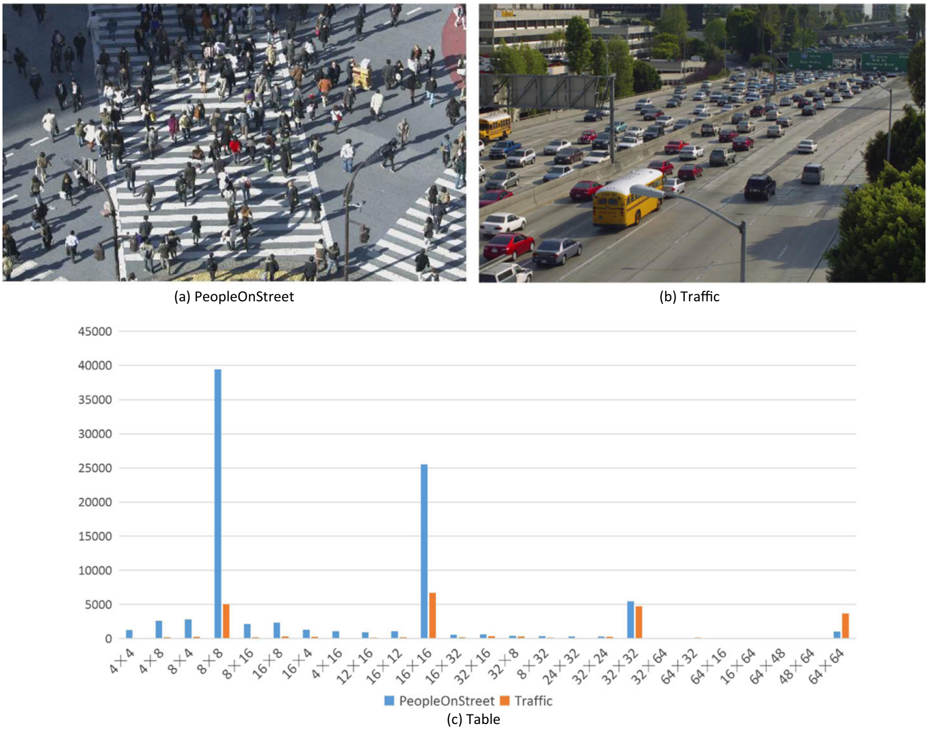


Fig. 6 Example of the distribution of different PU partition modes with different video contents. The resolutions of videos used in figure a and figure b are both 2560×1600

most in each CU size, while the PU partition mode of N×N size is the least. This is because each frame in the video is spatially redundant, that is to say, the content of each frame is similar in a certain range, and the boundary is only a very small part of each frame. The PU partition is to fit the boundary of the moving object, so the PU partition mode of 2N×2N size accounts for the vast part of the image. On the contrary, N×N PU partition mode is the least. Because there is only one PU partition mode for every CU, it can be known from Fig. 4 that CU with 8×8 size is dominant in any given resolution, while CU with 64×64 size is the least. Although each frame in the video is spatially redundant, a large piece of the same content such as the size of 64×64 is still limited. In addition, a 64×64 CU can be divided into 4 CUs of 32×32 size, and one 32×32 CU can be divided into 4 CUs of 16×16 size, and each 16×16 CU can be divided into 4 CUs of 8×8 size, thus accumulating down, CUs of these small partition sizes are increased by four times. Therefore, although each frame has large spatial redundancy, CU with 64×64 size has the least composition, opposing to the number of CU with 8×8 size. Now we analyze the difference between the CU and PU partition modes under different resolutions. Although CU with 64×64 size is the least in the same frame, the largest CTU known as 64×64 is the same at any resolution, so the total number of CTU, PU and CU increases considerably at high resolution. Therefore, the absolute number of CU with 64×64 size at high resolution is much greater than that at low resolution, and even can reach the absolute number of CU with 8×8 size at low resolution.

Figure 5 shows the specific number distribution of different PU partition modes with different QPs. It suggests that with the increase of QP, the number of each PU partition mode declines except 64×64 PU partition mode. Because lager QP means less details in video, which means smaller PU partition modes would decrease while larger PU partition modes would increase.

Figure 6 shows the specific number distribution of different PU partition modes with different video contents. We can see that the number of 64×64 PU partition mode in video PeopleOnStreet are smaller than that in video Traffic while the number of other PU partition modes are lager especially PU partition modes smaller than 16×16. It is because that the content in video PeopleOnStreet are more complex than that in video Traffic as shown in Figure a and b of Fig. 6. So there are more details in video PeopleOnStreet, which means more small PU partition modes are needed in compression. For a large number of similar smooth blocks in video Traffic, large PU partition modes are enough to have a good quality of compression.

According to these characteristics of the PU partition and CU partition analysed in this section, the hiding information algorithm by modifying PU partition modes is proposed. Since under different resolutions, the characteristics of PU partition modes are different, it is reasonable to propose multiple information hiding levels based on different video resolutions. In the next section, we will describe the multilevel information hiding algorithm in detail.

3 The proposed method

3.1 Information hiding algorithm

For all sizes of CUs, 2N×2N PU partition mode is dominant (see Section 2.2). Therefore, if this PU partition mode is modified, although it will have considerable capacity, the bit rate will increase seriously and hidden information can be easily detected. Considering these costs, we propose an algorithm that do not modify the 2N×2N PU partition mode for all sizes of CUs. Moreover, for different sizes of CUs, the PU partition modes are different (see Section 2.1). Thus the information hiding algorithm will be introduced in detail in two parts based on different CU sizes – 16×16, 32×32 and 64×64 sizes and 8×8 size.

For CU of 16×16 and larger size, it has eight PU partition modes in total. Except 2N×2N PU partition mode, there are other seven PU partition modes that can be used to hide information. However, because N×N PU partition mode only appears in 8×8 CU, we also do not consider to modify this PU partition mode to hide information. In the remaining six PU partition modes, there are three horizontal PU partition modes and three vertical PU partition modes. When modifying the horizontal PU partition mode to the vertical PU partition mode, the visual quality will be greatly affected for the inaccurate identification of the shape of a moving object. Therefore, we classify the six PU partition modes into three groups to represent different information, each of which contains a horizontal partition mode and a vertical partition mode. For each PU partition mode, m_1 to m_8 are used to represent 2N×2N, 2N×N, N×2N, N×N, 2N×nU, 2N×nD, nL×2N and nR×2N respectively. As shown in Figure 7, m_2 and $m_3 \in Group1$, which represents binary bit '10', m_6 and $m_7 \in Group2$, which represents two binary bits '10', and m_5 and $m_8 \in Group3$, which represents two binary bits '11'. The process for modifying the PU partition modes in 16×16 and larger CU is shown in Algorithm 1.

Algorithm 1 the modification process of PU partition modes in 16×16 and lager CU

Input: the present PU partition mode decided by HEVC in one 16×16 or larger CU m_0, hidden binary bits B;
Output: the modified PU partition mode $m_0^{'}$;

```
if m_0 == m_1 or m_0 == m_4 then
    m_0' = m_0
else
    if B == '0' then
        if m_0 in Group1 then
            m_0' = m_0
        else if m_0 == m_5 or m_0 == m_6 then
            m_0' = m_2
        else
            m_0' = m_3
        end if
    end if
    if B == '10' then
        if m_0 in Group2 then
            m_0' = m_0
        else if m_0 == m_5 or m_0 == m_2 then
            m_0' = m_6
        else
            m_0' = m_7
        end if
    end if
    if B == '11' then
        if m_0 in Group3 then
            m_0' = m_0
        else if m_0 == m_2 or m_0 == m_6 then
            m_0' = m_5
        else
            m_0' = m_8
        end if
    end if
end if
```

For CU of 8×8 size, it has four PU partition modes in total. Like other sizes of CU, we do not modify the 2N×2N PU partition mode for its cost. Furthermore, since the size of 8×8 CU is small, the influence of modifying the horizontal PU partition mode to the vertical PU partition mode on visual quality is limited. Hence we divide the rest three PU partition modes into three groups. For each PU partition mode, m_1 to m_4 are used to represent 2N×2N, 2N×N, N×2N and N×N respectively. As shown in Fig. 8, $m_4 \in Group1$, which represents binary bit '0', $m_3 \in Group2$, which represents two binary bits '10', and $m_2 \in Group3$, which represents tow binary bits '11'. The process for modifying the PU partition modes in 8×8 CU is shown in Algorithm 2.

Algorithm 2 the modification process of PU partition modes in 8×8 CU

Input: the present PU partition mode decided by HEVC in one 8×8 CU m_0, hidden binary bits B;
Output: the modified PU partition mode $m_0^{'}$;

if $m_0 == m_1$ **then**
    $m_0^{'} = m_0$
else
    if $B ==$ '0' **then**
        if m_0 in $Group1$ **then**
            $m_0^{'} = m_0$
        else
            $m_0^{'} = m_4$
        end if
    end if
    if $B ==$ '10' **then**
        if m_0 in $Group2$ **then**
            $m_0^{'} = m_0$
        else
            $m_0^{'} = m_3$
        end if
    end if
    if $B ==$ '11' **then**
        if m_0 in $Group3$ **then**
            $m_0^{'} = m_0$
        else
            $m_0^{'} = m_2$
        end if
    end if
end if

As for the whole information hiding process, the PU partition modes decided by HEVC will be recorded at the first round calculation, then PU partition modes in qualified CU will be modified according to the Algorithm 1 and Algorithm 2, after that, the modified PU partition modes will be used for the second HEVC encoding to finish the whole information hiding in videos. The specific process of embedding information is given below.

Step1. Record the division depth and the selected PU partition modes of each CU in the process of determining the optimal CU structure for each CTU by HEVC coding.
Step2. According to the division depth, determine what the optimal CU structure contains. According to the size of CU, judge the corresponding group it belongs to.
Step3. Perform PU partition mode judgement for each CU. If the PU partition mode of the CU does not belong to any of the three groups, no changes will be made to this CU. If the PU partition mode of CU belongs to one of the three groups, then whether the PU partition mode would be modified is determined according to the to-be-embedded binary information.
Step4. Implement the second calculation. If there is no change in PU partition modes, the CTU will be compressed by default procedure of HEVC. If it is necessary to adjust the PU partition modes, the CU structure and the PU partition modes of the CTU need

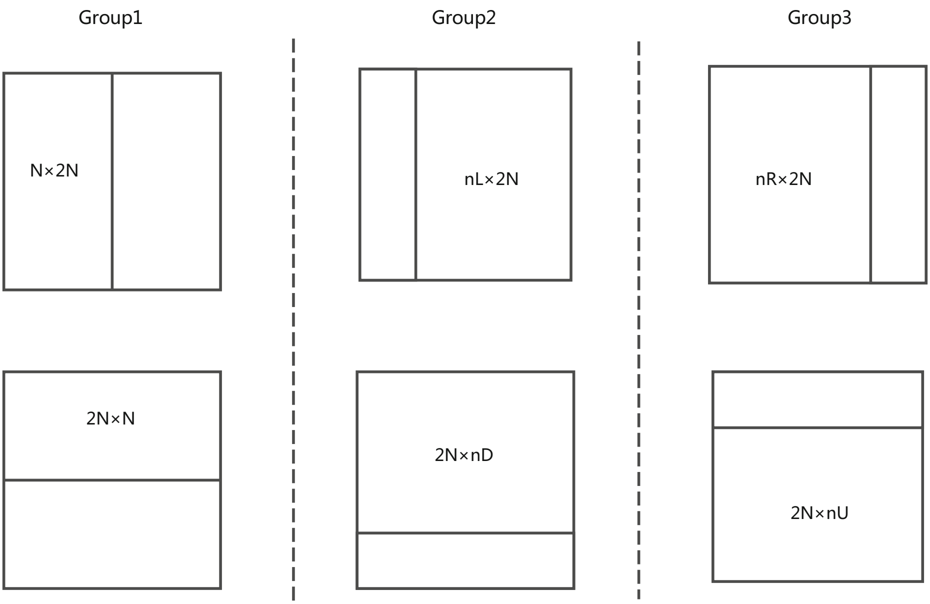


Fig. 7 Three groups of different PU partition modes for 16×16, 32×32 and 64×64 CU sizes

to be calculated again. In the recalculation process, the PU partition modes of each CU are directly specified based on the data recorded during the first calculation and the modification result in step 3.

Step5. Once the recalculation process finished, continue the compression according to the default process of HEVC.

By accomplishing the process above, part of the PU modes will be adjusted according to the hidden binary bits. Thus, the hidden information is represented by these PU partition modes.

When extracting secret bits, assuming that m_n is the PU partition mode that is now read. If $m_n \in Group1$, the hidden binary bit is '0', if $m_n \in Group2$, the hidden binary bits are '10', and if $m_n \in Group3$, the hidden binary bits are '11'.

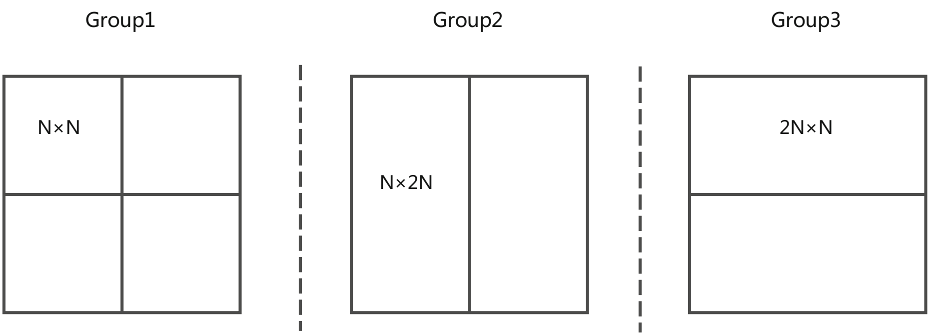


Fig. 8 Three groups of different PU partition modes for 8×8 CU size

3.2 Multiple information hiding levels

In Section 3.1, we have proposed different modification methods for CUs with different sizes. Because these modification methods only revolves the size of CU, and the program of modifying PU partition modes in different CU sizes can be carried out separately, the way of modifying PU partition modes in different CU sizes can be combined arbitrarily. That is to say, we can design different embedding levels according to the specific needs. For example, we can just modify PU partition modes in 8×8 CU, or PU partition modes in both 8×8 CU and 16×16 CU.

In this paper, multiple information hiding levels will be proposed based on the video resolution and the requirement of capacity, visual quality and bit rate. First of all, we divide the video into two situations: high resolution and low resolution. In low resolution videos, the absolute number of 64×64 CU size is small, which has limited capacity contribution and will increase bit rate when modifying PU partition modes in CU with this size. But in high resolution videos, the number of 64×64 CU size increases, so modifying PU partition modes in 64×64 CU is acceptable. Thus, in low resolution videos, the PU partition modes in 64×64 CU will not be modified, while it is on contrary in high resolution videos.

In Section 2.2, we have analyzed that the number of 8×8 CU size is the most in a video. When modifying PU partition modes in this CU size, the need for the embedding capacity can be satisfied, and it is hard to be perceived because its size is small. Hence it is reasonable to take the modification of PU partition modes in 8×8 CU as the basic level, and add the modification of PU partition modes in other larger sizes of CU when considering higher capacity.

Therefore, for videos with low resolution, three information hiding levels are designed — modifying PU partition modes in 8×8 CU, modifying PU partition modes in 8×8 CU and 16×16 CU and modifying PU partition modes in 8×8, 16×16 and 32×32 CU. For videos with high resolution, four information hiding levels are designed — modifying PU partition modes in 8×8 CU, modifying PU partition modes in 8×8 CU and 16×16 CU, modifying PU partition modes in 8×8, 16×16 and 32×32 CU and modifying PU partition modes in all sizes of CU.

4 Experimental results

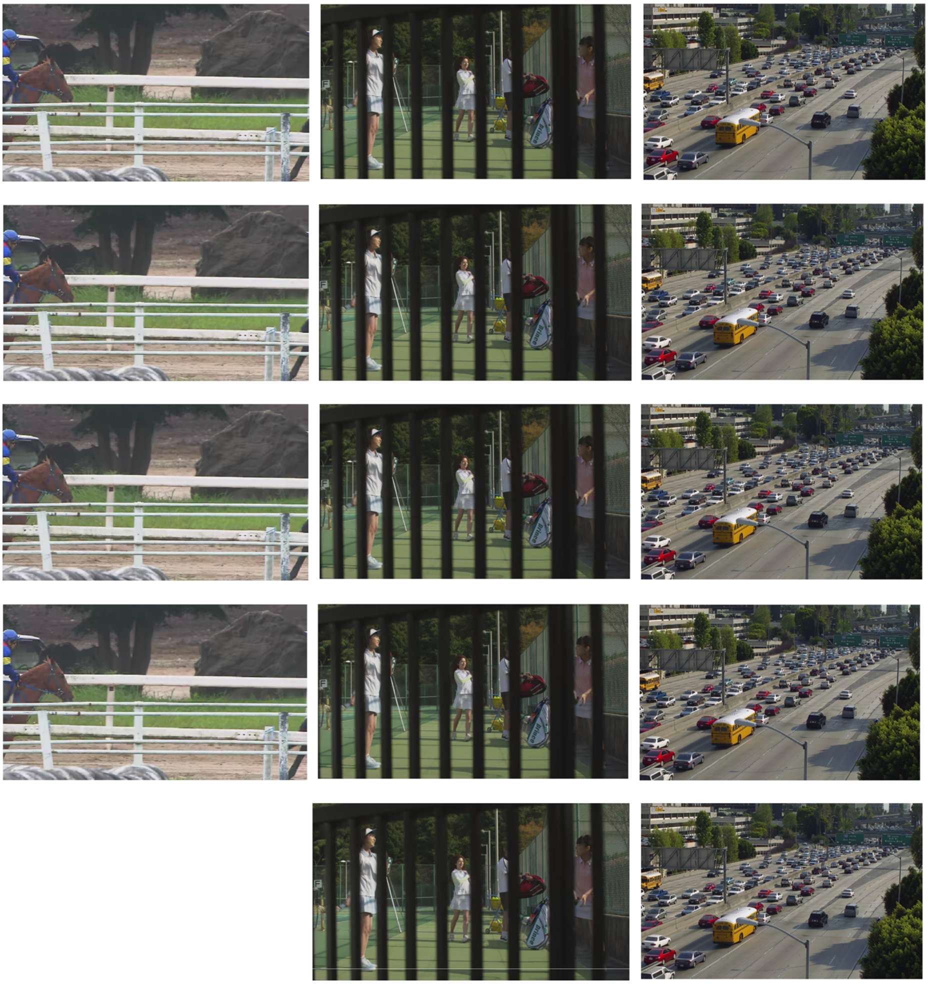
4.1 Configurations

In this section, the experimental results of the proposed algorithm are given. Twelve common test video sequences will be used to carry out the experiments, which consist of 2 sequences with resolution 2560×1600: Traffic and PeopleOnStreet; 6 sequences with resolution 1920×1080: BasketballDrive, BQTerrace, Cactus, Kimono1, ParkScene and Tennis; 1 sequence with resolution 1024×768: ChinaSpeed; 3 sequences with resolution 832×480: Kerba, PartyScene and RaceHorses. The video codec platform for HEVC is HM 16.15, these video sequences are encoded with frame rate 25fps, the GOP size is 4 and the structure of it is "IPPP", and the quantization parameters (QP) are set as 26, 32 and 38, respectively. The remaining parameters are set to the HM default configuration.

Since there are multiple information hiding levels proposed in this paper, we will use Level1 to represent the way modifying PU partition modes in 8×8 CU for all resolutions, Level2 to represent the way modifying PU partition modes in 8×8 CU and 16×16 CU for all resolutions, Level3 to represent the way modifying PU partition modes in 8×8, 16×16 and 32×32 CUs for all resolutions, and Level4 to represent the way modifying PU partition modes in all sizes of CU for high resolution, where high resolution refers to a video resolution greater or equal to 1920×1080.

4.2 Subjective visual quality

In this subsection, we will estimate video visual quality in subjective aspect. Subjective assessment is to judge video visual quality by human eyes, and to see whether the hidden information can be perceived. Figure 9 shows some original and information hidden video frames of different resolutions. In Fig. 9, it is difficult to visually find the difference between



(a) Keiba (b) Tnnis (c) Traffic

Fig. 9 Subjective performance evaluation of the six P-frames. Original frames are laid on the first line. The information hidden frames with Level1, Level2, Level3 and Level4 are placed in the second line, third line, fourth line and fifth line respectively. The resolution of **a**, **b** and **c** are 832×480, 1920×1080 and 2560×1600

Table 1 The PSNR with multilevel information hiding with QP = 32

Sequence	Schemes	PSNR			
		Y	U	V	Average
Traffic	Original	36.2004	38.2371	40.5311	36.9914
	Level1	36.1939	38.2327	40.5272	36.9851
	Level2	36.1757	38.2299	40.5232	36.9698
	Level3	36.1431	38.2277	40.5194	36.9431
	Level4	36.1293	38.2220	40.5114	36.9311
PeopleOnStreet	Original	33.9676	41.2460	41.9162	35.3562
	Level1	33.9362	41.1894	41.8781	35.3232
	Level2	33.8807	41.1688	41.8632	35.2710
	Level3	33.8379	41.1595	41.8564	35.2311
	Level4	33.8399	41.1517	41.8593	35.2329
BasketballDrive	Original	36.5805	42.2522	42.0479	37.7729
	Level1	36.5796	42.2587	42.0350	37.7717
	Level2	36.5808	42.2334	42.0132	37.7700
	Level3	36.5637	42.2430	41.9818	37.7537
	Level4	36.5500	42.2333	41.9938	37.7416
BQTerrace	Original	32.8403	39.1057	41.2803	34.1987
	Level1	32.8377	39.0989	41.2714	34.1957
	Level2	32.8255	39.1008	41.2683	34.1845
	Level3	32.8137	39.0959	41.2685	34.1732
	Level4	32.8123	39.0957	41.2633	34.1719
Cactus	Original	34.5816	38.0711	40.1220	35.6088
	Level1	34.5759	38.0684	40.1246	35.6040
	Level2	34.5647	38.0630	40.1178	35.5937
	Level3	34.5345	38.0585	40.1056	35.5674
	Level4	34.5229	38.0599	40.1000	35.5567
Kimono1	Original	36.7353	40.3652	41.8822	37.7441
	Level1	36.7279	40.3645	41.8705	37.7373
	Level2	36.7406	40.3681	41.8780	37.7482
	Level3	36.7059	40.3417	41.8587	37.7147
	Level4	36.6960	40.3411	41.8558	37.7064
ParkScene	Original	34.7026	38.8134	40.2434	35.7782
	Level1	34.6941	38.8073	40.2402	35.7697
	Level2	34.6747	38.8049	40.2379	35.7525
	Level3	34.6574	38.7911	40.2311	35.7364
	Level4	34.6444	38.7944	40.2321	35.7254
Tennis	Original	36.3729	41.4713	42.2200	37.5419
	Level1	36.3678	41.4631	42.2213	37.5372
	Level2	36.3668	41.4277	42.2086	37.5331
	Level3	36.3036	41.4272	42.1686	37.4755
	Level4	36.2884	41.4102	42.1722	37.4616

Table 1 (continued)

Sequence	Schemes	PSNR			
		Y	U	V	Average
ChiaSpeed	Original	35.2486	40.4881	40.4395	36.3940
	Level1	35.1369	40.3750	40.3049	36.2807
	Level2	35.0809	40.3385	40.2782	36.2290
	Level3	34.9999	40.3737	40.2539	36.1590
Keiba	Original	33.6712	40.0169	41.1260	34.9948
	Level1	33.6347	39.9572	41.0996	34.9583
	Level2	33.5957	39.9713	41.0733	34.9223
	Level3	33.5647	39.9475	41.0580	34.8912
PartyScene	Original	30.7015	36.4316	36.6897	31.7929
	Level1	30.6283	36.3947	36.6461	31.7165
	Level2	30.6003	36.3891	36.6029	31.6858
	Level3	30.5928	36.3926	36.6110	31.6801
RaceHorses	Original	31.8367	36.9379	38.1586	33.0287
	Level1	31.8145	36.8851	38.1350	33.0054
	Level2	31.7508	36.8606	38.0380	32.9427
	Level3	31.7288	36.8258	38.0369	32.9218

reconstructed video images with or without hiding information, so it has no visual distortion after embedding secret information.

4.3 Objective Performance Evaluation

In order to measure the algorithm objectively, four common evaluation methods — the peak signal-to-noise ratio(PSNR), the increase of bit rate, the embedding capacity and the computation complexity are used to estimate the performance of the proposed algorithm.

For objective evaluation in visual quality, Table 1 shows the PSNR values of original and multilevel information hidden video frames under different resolutions. From Table 1, we can see that, compared with the frames without embedding, the PSNR values at different levels are almost unchanged. Because we hide information during the encoding process, the corresponding parts such as motion residuals will be calculated based on the modified PU partition modes, which means no new error will be introduced. Additionally, according to the experimental results, the data distribution of PSNR with other QPs are almost same with the QP = 32. Thus, the proposed algorithm has little influence on the visual quality.

Generally, the change in the PU partition modes will cause the change in the bit rate, so it is necessary to analyze the increase of bit rate. In this paper, we will use the BRI [16] to measure the bit-rate increase. BRI is defined as:

$$BRI = (BR_h - BR_o)/BR_o \tag{1}$$

Where BR_o is the video bit rate without embedding, and BR_h is the video bit rate after embedding. Experimental results of different videos with different QPs are shown in Fig. 10. In Fig. 10, we can see that the maximum BRI is no more than 0.06, so the bit rate increase of the proposed algorithm is acceptable. Moreover, for each QP, the lowest bit rate increase is Level1 and as the level increases, the bit rate also rises. Since in Level1 only PU partition

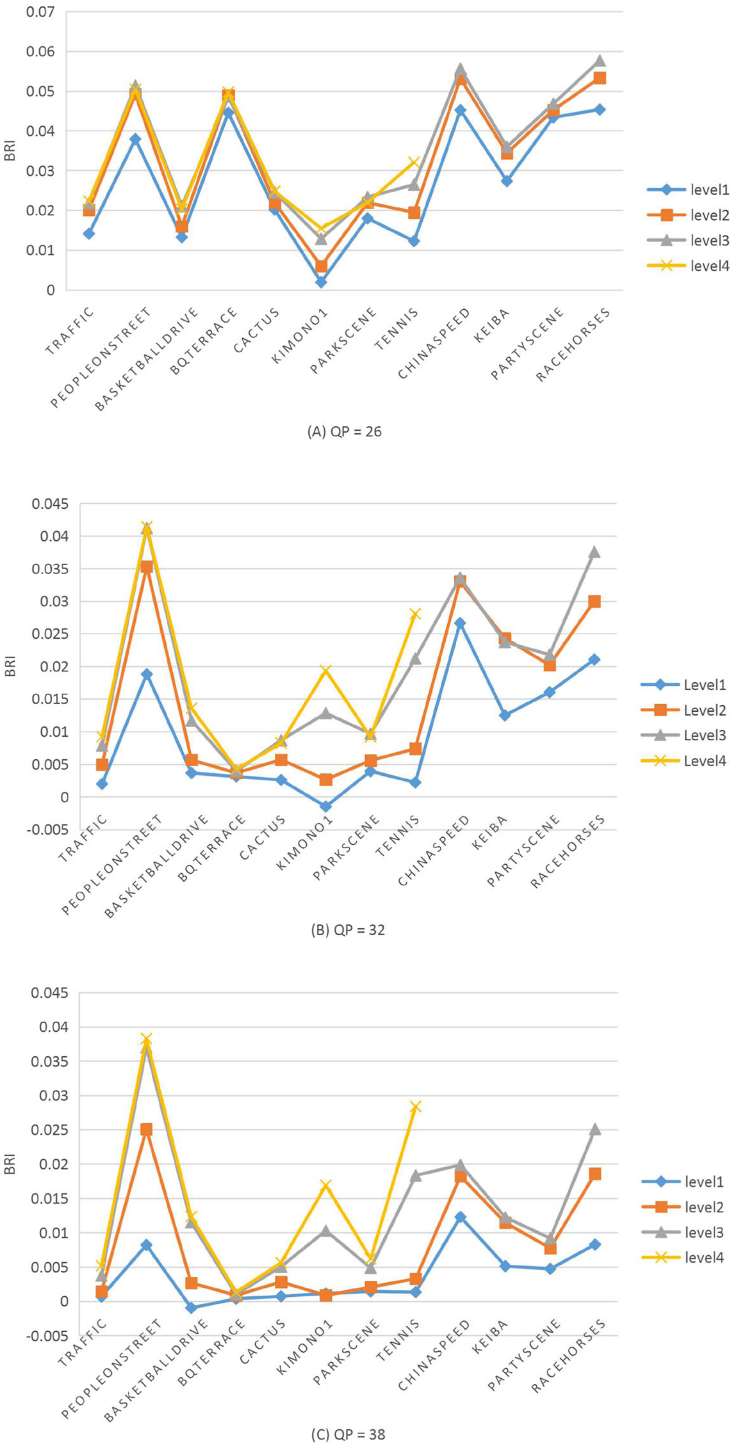


Fig. 10 The BRI of different decoded video sequences with multilevel information hiding with different QPs

Table 2 The embedding capacity with multilevel information hiding with QP = 26

	Level1	Level2	Level3	Level4
Traffic	869	1915	2270	2327
PeopleOnStreet	4861	7507	7894	8002
BasketballDrive	246	452	610	658
BQTerrace	1855	2746	2949	3058
Cactus	555	1083	1347	1389
Kimono1	46	286	660	709
ParkScene	740	1313	1493	1560
Tennis	268	561	938	967
ChiaSpeed	592	853	906	-
Keiba	245	383	436	-
PartyScene	821	993	1014	-
RaceHorses	585	860	951	-

modes in 8×8 CU are modified, while with the increase of level, more PU partition modes in other CU sizes are modified which leads to the further increase of bit rate. Besides, with the increase of QP, BRI declines on the contrary. Because QP has an influence on the number of PU and lager QP means less number of PU in small CU which leads to more poor visual quality. So when there are not so many PUs, the bit rate would increase less with higher QP.

The embedding capacity is also an important aspect to measure the quality of the information hiding algorithm. The embedding capacity with multilevel information hiding and different QPs per P-frame are given in Tables 2, 3 and 4. From Table 3, we can see that in high resolution videos, the average capacity of Level1 is 266, the average capacity of Level2 is 660, the average capacity of Level3 is 918, and the average capacity of Level4 is 1525. In low resolution videos, the average capacity of Level1 is 209, the average capacity of Level2 is 346, and the average capacity of Level3 is 399. Therefore, the embedding capacity of the proposed algorithm at any level is high. Moreover, with the increase of level or decrease of QP, the embedding capacity increases, and for each QP most of high resolution videos have

Table 3 The embedding capacity with multilevel information hiding with QP = 32

	Level1	Level2	Level3	Level4
Traffic	126	395	718	776
PeopleOnStreet	1579	3679	4329	8753
BasketballDrive	52	116	213	271
BQTerrace	107	225	327	387
Cactus	82	274	500	589
Kimono1	10	130	419	484
ParkScene	126	273	428	444
Tennis	42	191	412	496
ChiaSpeed	233	410	450	-
Keiba	71	169	215	-
PartyScene	332	451	490	-
RaceHorses	198	353	439	-

Table 4 The embedding capacity with multilevel information hiding with QP = 38

	Level1	Level2	Level3	Level4
Traffic	15	80	205	265
PeopleOnStreet	436	1639	2379	2454
BasketballDrive	9	40	102	129
BQTerrace	9	21	35	59
Cactus	8	54	184	238
Kimono1	1	31	170	258
ParkScene	10	45	100	142
Tennis	6	48	226	317
ChiaSpeed	100	179	220	-
Keiba	9	48	91	-
PartyScene	67	137	177	-
RaceHorses	44	129	192	-

higher embedding capacity than low resolution videos. The number of modified PU partition modes decides the embedding capacity, so high resolution videos, higher levels and lower QP have higher embedding capacity for their larger number of PU partition modes.

The computation complexity has a significant effect on the real-time performance of encoding, so it is necessary to analyze the computation complexity of the algorithm proposed in this paper. From Table 5, we can see that the encoding time is increasing with the increase of level and resolution because more PU partition modes are needed in higher level and higher resolution videos which cost more time to encode. Overall, the fluctuations of the encoding time at any level with any QP are in an acceptable range. Hence the algorithm proposed in this paper has a low computation complexity and little effect on the real-time video encoding.

4.4 Anti-steganalysis analysis

With the development of information hiding algorithm, the steganography detection methods also make great progress. In recent years, a lot of image and video steganalysis methods [5, 8, 13, 19] have been proposed. So a good information hiding algorithm should resist steganalysis to a certain extent. In this subsection, we use the steganography detection method in [8] to evaluate the performance of our information hiding algorithm on anti-steganalysis. In [8], Sheng et al. proposed a steganography detection method featuring the change rate of the amount of PU different sizes with before and after re-compression. Since the optimal prediction mode in the original video is the prediction mode that makes the current cost minimum, once modified the optimal prediction mode, the cost would increase. So the prediction mode in the information hiding video after re-compression tends to return the optimal prediction mode in the original video before re-compression. According to this, we select several videos to calculate the average number and the average proportion of different PU partition modes in original and information hiding videos before and after re-compression. The experimental results are shown in Fig. 11. It can be seen that the change of different PU partition modes in information hiding video before and after re-compression are almost the same with that in original video, either from the amount or from the proportion. In addition, the standard deviations of difference between the change of PU proportion

Table 5 The encoding time (s) per frame with multilevel information hiding with different QPs

	QP	original	Level1	Level2	Level3	Level4
Traffic	26	692.219	715.640	733.953	747.287	749.124
	32	380.289	385.918	394.304	402.965	836.624
	38	528.753	517.354	520.954	529.131	532.806
PeopleOnStreet	26	1122.486	1192.729	1248.809	1225.963	1289.980
	32	614.272	643.551	690.810	655.683	660.234
	38	703.824	735.261	779.354	803.951	807.810
BasketballDrive	26	456.812	457.869	452.615	470.312	498.382
	32	4713.618	403.228	1096.474	461.210	449.702
	38	306.927	311.446	308.361	312.688	315.084
BQTerrace	26	411.833	452.440	479.702	482.223	486.314
	32	200.954	205.451	200.890	210.624	212.543
	38	272.293	271.125	266.603	270.682	262.799
Cactus	26	381.642	870.836	454.410	459.798	1097.845
	32	210.620	209.409	213.137	219.163	335.482
	38	276.707	273.880	277.812	283.552	283.459
Kimono1	26	868.305	537.237	757.726	548.506	542.009
	32	416.481	401.906	401.740	419.313	424.686
	38	307.185	303.271	309.407	318.001	323.668
ParkScene	26	398.080	502.381	410.931	415.548	409.643
	32	1563.998	354.568	346.395	368.836	346.295
	38	266.639	263.658	603.008	292.961	290.358
Tennis	26	476.182	508.009	506.228	527.321	534.887
	32	2007.832	521.391	435.453	444.616	441.093
	38	364.860	346.680	365.228	561.540	394.045
ChiaSpeed	26	202.365	215.230	215.274	216.811	-
	32	159.845	394.349	214.641	188.786	-
	38	123.221	135.104	148.539	162.102	-
Keiba	26	92.744	102.459	107.286	121.207	-
	32	72.997	75.109	77.653	80.429	-
	38	61.771	63.351	60.255	61.798	-
PartyScene	26	107.206	122.449	127.279	125.532	-
	32	79.273	84.432	85.646	86.952	-
	38	64.787	68.491	70.491	71.818	-
RaceHorses	26	124.331	127.641	130.304	131.053	-
	32	89.083	97.005	100.290	107.508	-
	38	69.641	72.614	77.198	78.430	-

in original video and that in information hiding video are 0.008265, 0.008692, 0.009612 and 0.007854 individually, from which we can know that the difference in the amount and proportion of PU partition modes between original video and information hiding video is too small to distinguish. Therefore, it can be supposed that the algorithm proposed in this paper has a good performance on anti-steganalysis.

Fig. 11 The change of PU amount and proportion in original and information hiding videos. **a** is the change of PU amount and PU proportion in original video before and after re-compression. **b** is the change of PU amount and PU proportion in information hiding video in level1 before and after re-compression. **c** is the change of PU amount and PU proportion in information hiding video in level2 before and after re-compression. **d** is the change of PU amount and PU proportion in information hiding video in level3 before and after re-compression. **e** is the change of PU amount and PU proportion in information hiding video in level4 before and after re-compression

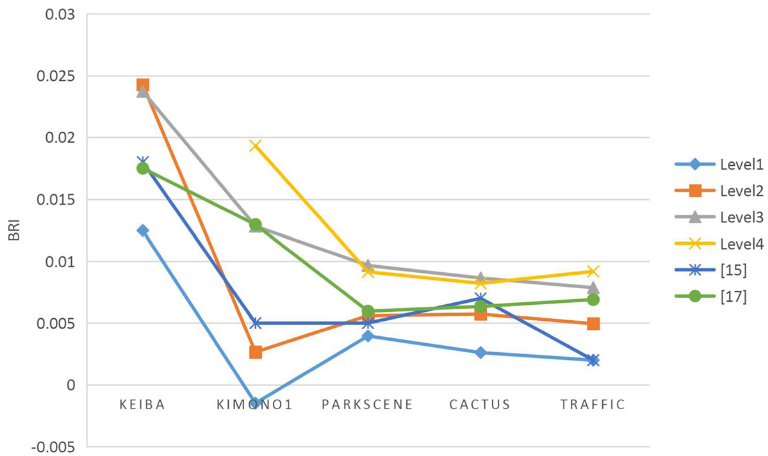


Fig. 12 The BRI comparison with Yang and Li's and Xie et al.'s of different decoded video sequences

4.5 Comparative analysis

In this subsection, we will compare the proposed algorithm with the latest video information hiding algorithm [16] and [14]. In Yang and Li's work [16], there are different embedding strength, we choose the strongest embedding strength here. Since our algorithm has advantageous performance in capacity at little cost of bit rate, such a situation will appear – the embedding capacity in the proposed algorithm is much higher than that of [16], but its bit rate increases less than ours, so it is difficult to compare and evaluate when not controlling one of embedding capacity and bit rate. As for [14], since our work is based on [14] and Level3 includes the algorithm in [14], it is possible that we have a higher capacity by little cost of the bit rate increase. Therefore, in order to perform a fair comparison and evaluation, we choose five video sequences with the similar bit rate level to compare, they are Keiba, Kimono1, ParkScene, Cactus and Traffic.

As shown in Fig. 12, some of information hiding levels have higher bit rates than [16], while others have lower bit rates than [16], so we select Level1 and Level2 that have similar

Table 6 The embedding capacity comparison with Yang and Li's and Xie et al.'s

	Level1	Level2	Level3	[15]	[17]
Keiba	**71**	169	215	26	104
Kimono1	10	**130**	419	80	320
ParkScene	**126**	273	428	62	176
Cactus	82	**274**	500	170	307
Traffic	**126**	395	718	104	441

bit rate with [16] to compare the embedding capacity. Figure 12 also suggests that Level2 and Level3 have the similar bit rate with [14] so these two levels are used to compare the embedding capacity with [14]. For convenience of viewing, we have enlarged the comparison capacity compared with [16] and mark the comparison capacity red compared with [14] in Table 6. From Fig. 12 and Table 6, all the videos with a similar BRI with that of [16] and [14] have a much higher capacity than that of [16] and [14]. It can be concluded that the proposed algorithm significantly outperforms the work [16] and the work [14] in embedding capacity.

5 Conclusions

A high capacity and multilevel information hiding algorithm for HEVC based on PU partition modes is proposed in this paper. According to the hidden information, the PU partition modes are modified during the inter prediction process, so the visual quality of the video is almost undamaged. Multiple information hiding levels are provided to meet the demand of different embedding capacities and bit rate. For each qualified CU, an average of 1.5 bits can be embedded, since the number of PU partition modes in each P-frame is large, the embedding capacity for any level is high. Furthermore, the proposed information hiding algorithm is independent from most of the existing embedding methods neither in principle nor in implementation. Therefore, it is convenient to embed hidden bits in I-frames, and combine with the algorithm proposed by this paper to greatly increase the capacity without further degradation in visual quality. Moreover, the storage space occupied by the inter prediction is much less than that of the intra prediction. To reduce the transmission bandwidth of the video, there surely will be more P-frames to replace I-frames in HD videos. Thus, the information hiding in P-frames will have a great application in the future.

Acknowledgments This work was supported by the National Natural Science Foundation of China (No.61702034).

References

1. Chang P-C, Chung K-L, Chen J-J, Lin C-H, Lin T-J (2014) A dct/dst-based error propagation-free data hiding algorithm for HEVC intra-coded frames. J Vis Commun Image Represent 25(2):239–253
2. Chen Y, Yu Z, Wang X, Guo L (2017) Video information hiding algorithm based on diamond coding. Journal of Computer Applications
3. Fallahpour M, Shirmohammadi S, Ghanbari M (2015) A high capacity data hiding algorithm for h.264/AVC video. Secur Commun Netw 8(16):2947–2955
4. Lin T-J, Chung K-L, Chang P-C, Huang Y-H, Liao H-YM, Fang Cx-Y (2013) An improved dct-based perturbation scheme for high capacity data hiding in h.264/AVC intra frames. J Syst Softw 86(3):604–614
5. Ma Y, Luo X, Li X, Bao Z, Zhang Y (2018) Selection of rich model steganalysis features based on decision rough set α-positive region reduction. IEEE Transactions on Circuits and Systems for Video Technology
6. Po CC, Chung KL, Chen JJ, Lin CH, Lin TJ (2014) An error propagation free data hiding algorithm in hevc intra-coded frames. In: Signal and information processing association summit and conference, pp 1–9

7. Sheng Q, Wang R, Pei A, Wang B (2016) An information hiding algorithm for hevc based on differences of intra prediction modes. In: International conference on cloud computing and security. Springer, pp 63–74
8. Sheng Q, Wang RD, Huang ML, Li Q, Xu DW (2017) A prediction mode steganalysis detection algorithm for hevc. J Optoelectron-laser 28(4):433–440
9. Tew Y, Wong K (2014) Information hiding in hevc standard using adaptive coding block size decision. In: 2014 IEEE international conference on image processing (ICIP). IEEE, pp 5502–5506
10. Wang J, Wang R, Li W, Xu D, Huang M (2014) A high-capacity information hiding algorithm for hevc based on intra prediction mode. J Comput Inform Syst 10(20):8933–8943
11. Wang JJ, Wang RD, Wei LI, Da-Wen XU, Di QY (2014) An information hiding algorithm for hevc based on intra prediction. J Optoelectron Laser 25(8):1578–1585
12. Wang JJ, Wang RD, Li W, Xu DW, Xu J (2015) Video information hiding in intra prediction and block codes for hevc. Guangdianzi Jiguang/J Optoelectron Laser 26(5):942–950
13. Wang J, Li T, Shi Y-Q, Lian S, Ye J (2017) Forensics feature analysis in quaternion wavelet domain for distinguishing photographic images and computer graphics. Multimed Tools Appl 76(22):23721–23737
14. Xie W-c, Yang Y-y, Li Z-h, Wang J-w, Zhang M (2018) An information hiding algorithm for hevc videos based on pu partitioning modes. In: International conference on cloud computing and security. Springer, pp 252–264
15. Xu D, Wang R, Wang J (2012) Prediction mode modulated data-hiding algorithm for H.264/AVC. J Real-Time Image Process 7(4):205–214
16. Yang J, Li S (2018) An efficient information hiding method based on motion vector space encoding for HEVC. Multimed Tools Appl 77(10):11979–12001
17. Yang G, Li J, He Y, Kang Z (2011) An information hiding algorithm based on intra-prediction modes and matrix coding for h.264/avc video stream. AEU - Int J Electron Commun 65(4):331–337
18. Yi C, Wang H, Wu H, Liu Y (2018) Reversible video data hiding using zero QDCT coefficient-pairs. CoRR, arXiv:1804.06628
19. Yi Z, Qin C, Zhang W, Liu F, Luo X (2018) On the fault-tolerant performance for a class of robust image steganography. Signal Process 146:99–111
20. Zhang J, Maitre H, Li J, Zhang L (2001) Embedding watermark in mpeg video sequence. In: 2001 IEEE fourth workshop on multimedia signal processing. IEEE, pp 535–540
21. Zhang H, Cao Y, Zhao X (2016) Motion vector-based video steganography with preserved local optimality. Multimed Tools Appl 75(21):13503–13519

Yiyuan Yang is now an undergraduate student in School of Computer Science and Engineering, Beihang University, China.

Zhaohong Li received the BS degrees in Communication Engineering from Southwest Jiaotong University in 2003, and the Ph.D. degree in Traffic Information Engineering and Control from Beijing Jiaotong University in 2008. Since 2008, she has been working in School of Electronic and Information Engineering, Beijing Jiaotong University, and she is now an associate professor. Her research interest includes video forensics, information hiding, watermarking and anti-forgery.

Wenchao Xie is now an master student in School of Electronic Science and Technology, Beijing Jiaotong University, China.

Zhenzhen Zhang received the B.S. degree in Communication Engineering from Zhengzhou University in 2009, and the Ph.D. degree in Circuits and Systems from Beijing Jiaotong University in 2017. After the completion of her Ph.D. program, she has been working as an Assistant Professor in Beijing Institute of Graphic Communication. Her research interests include video forensics, video watermarking, and information hiding.