J. Vis. Commun. Image R. 25 (2014) 239–253

Contents lists available at ScienceDirect

J. Vis. Commun. Image R.

journal homepage: www.elsevier.com/locate/jvci

A DCT/DST-based error propagation-free data hiding algorithm for HEVC intra-coded frames ☆

Po-Chun Chang [a], Kuo-Liang Chung [a], Jiann-Jone Chen [b], Chien-Hsiung Lin [a,*], Tseng-Jung Lin [a]

[a] *Department of Computer Science and Information Engineering, National Taiwan University of Science and Technology, No. 43, Section 4, Keelung Road, Taipei 10672, Taiwan, ROC*
[b] *Department of Electrical Engineering, National Taiwan University of Science and Technology, No. 43, Section 4, Keelung Road, Taipei 10672, Taiwan, ROC*

ARTICLE INFO

Article history:
Received 26 June 2013
Accepted 24 October 2013
Available online 5 November 2013

Keywords:
Data hiding
DCT/DST
H.264/AVC
HEVC
Intra-coded frames
Intra-frame error propagation

ABSTRACT

Currently, two error propagation-free discrete cosine transform (DCT)-based data hiding algorithms, one by Ma et al. and the other by Lin et al., were presented for H.264/AVC intra-coded frames. However, the state-of-the-art video codec, high efficiency video coding (HEVC), adopts both integer DCT and discrete sine transform (DST) such that the previous DCT-based algorithms cannot fully utilize available capacity for data hiding in HEVC. This paper presents the first DCT/DST-based data hiding algorithm for HEVC intra-coded frames where the block DCT and DST coefficient characteristics are investigated to locate the transformed coefficients that can be perturbed without propagating errors to neighboring blocks. Experimental results confirm the merits of the proposed algorithm in providing the intra-frame error propagation-free advantage, the quality improvement for marked images, the compression power inherited from HEVC, and the superiority of embedding capacity for low bitrate coding when compared with the previous two algorithms for H.264/AVC.

1. Introduction

With the advance of digital multimedia communication techniques, data hiding becomes important in the authentication, identification, annotation, and copyright protection of digital media items [18,3,10,15], among which digital video is most widely used. However, due to the massive information amount of digital video signals, it needs highly efficient video coding to make video communication feasible. Recently, several transform domain based data hiding algorithms [16,21,14,22,5,8,6,17] have been developed in H.264/AVC [4], where the quantized discrete cosine transform coefficients (QDCTs) of the 4 × 4 luma blocks in intra-coded frames (IFs) are perturbed so as to carry the hidden data. Unfortunately, these algorithms have the intra-frame error propagation problem due to the improper perturbation strategies for the QDCTs, leading to severe quality degradation for the reconstructed video signals.

To solve this problem, Ma et al. [11] proposed an effective perturbation strategy for the QDCTs of the 4 × 4 luma blocks based on both the directions of intra prediction modes and the deviation in the pixel values resulted from changing the QDCTs values. Lin et al. [9] proposed to increase the embedding capacity for the 4 × 4 luma blocks achieved by Ma et al. and to additionally utilize the other 4 × 4 luma blocks for embedding one bit into each of them. Although the intra-frame error propagation has been tackled in the algorithms [11,9], the inter-frame error propagation is another challenging problem for these transform domain based data hiding algorithms [13,23].

As the application demands higher video signal resolution, the H.264/AVC standard can no longer achieve satisfied compression performance. A state-of-the-art video coding standard, high efficiency video coding (HEVC) [2,19], is developed to improve the compression efficiency for high resolution video signals. When considering data hiding in HEVC video sequences, one intuitive way is to adapt the previous data hiding algorithms used for H.264/AVC to the HEVC standard. However, since the HEVC 4 × 4 block does not utilize integer DCT for transform coding, the transform domain data hiding algorithms developed based on 4 × 4 DCT blocks of H.264/AVC cannot be applied to HEVC. Hence, developing a data hiding algorithm that can support the transform coding adopted by HEVC while eliminating the intra-frame error propagation and alleviating the inter-frame error propagation is required, which motivates this research.

In this paper, we propose an error propagation-free data hiding algorithm for HEVC IFs. The main idea of the proposed algorithm is to categorize blocks so as to impose specific quantized coefficient perturbation patterns for intra-frame error propagation-free data hiding under the HEVC framework. For HEVC IFs, we propose a DCT and discrete sine transform (DST)-based coefficient

☆ The work of K.L. Chung and C.H. Lin was supported by the National Science Council of ROC under the contracts NSC99-2221-E-011-078-MY3, NSC101-2221-E-011-139-MY3, and NSC102-2221-E-011-055-MY3. The work of J.J. Chen was supported by the National Science Council of ROC under the contract NSC101-2221-E-011-137.

[*] Corresponding author.
E-mail address: d9409301@mail.ntust.edu.tw (C.-H. Lin).

1047-3203/$ - see front matter © 2013 Elsevier Inc. All rights reserved.
http://dx.doi.org/10.1016/j.jvcir.2013.10.007

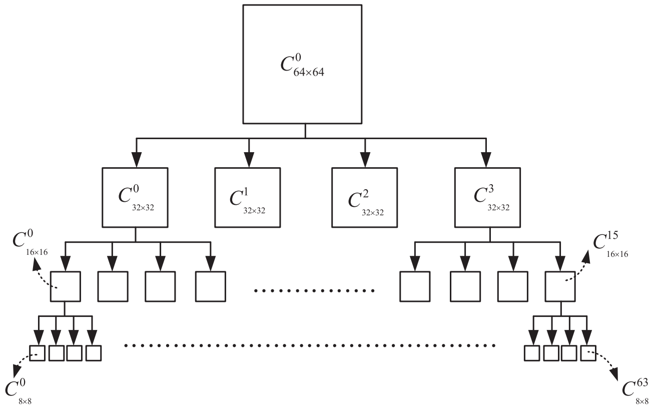


Fig. 1. The quadtree decompositions for one 64 × 64 coding unit.

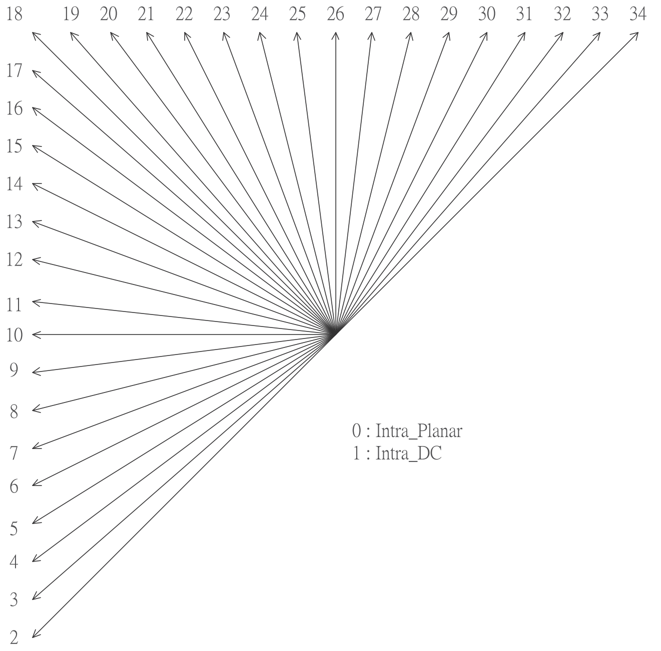


Fig. 2. The 35 intra prediction modes supported by HEVC.

perturbation scheme for embedding hidden bits. The error propagation patterns corresponding to different HEVC intra prediction modes are first classified for imposing different data hiding patterns. These data hiding patterns, which correspond to specific QDCT and QDST coefficient perturbations, are used to embed hidden bits into HEVC IFs without inducing intra-frame error propagation. In addition, one quality improvement scheme is proposed for the reconstructed IFs and meanwhile alleviates the quality degradation due to inter-frame error propagation. Experimental results on ten test videos confirm the merits of the proposed algorithm in providing the intra-frame error propagation-free advantage, the quality improvement for marked images, the compression power inherited from HEVC and the superiority of embedding capacity for low bitrate coding, although the embedding capacity is less than that of the previous two algorithms for H.264/AVC IFs for middle and high bitrate coding. To the best of our knowledge, this is the first time that such a DCT/DST-based data hiding algorithm is presented for HEVC.

The rest of this paper is organized as follows. In Section 2, the HEVC intra coding scheme and related block transform coding operations are reviewed. The proposed DCT/DST-based data hiding algorithm in HEVC and its theoretical analysis are presented in Section 3. Section 4 demonstrates experimental results and performance evaluation. Some concluding remarks are addressed in Section 5.

2. Intra coding scheme and the related transform coding in HEVC

The HEVC intra coding scheme, including block transform coding operations, will be first reviewed, with which the proposed data hiding algorithm can be clearly described.

2.1. Intra coding scheme in HEVC

When compressing video sequences by the HEVC encoder, coding unit (CU) is the basic unit to perform coding for each video frame. One video frame is first split into 64 × 64 non-overlapping CUs, each of which is further split into smaller ones with a recursive quadtree decomposition, as shown in Fig. 1, where $C^0_{N\times N}, C^1_{N\times N}, \ldots, C^{(64/N)^2-1}_{N\times N}$ denote the $(64/N)^2$ CUs of size $N \times N$ for $N = 64$, 32, 16, 8.

To exploit spatial redundancy for compression, the intra prediction is performed among adjacent blocks for predicting the current block in HEVC. For one $N \times N$ CU with $N \in \{64, 32, 16\}$, there exists only one prediction unit (PU) of size $N \times N$, whereas two PU sizes, 8×8 and 4×4, are available when $N = 8$. In addition to different PU sizes, HEVC supports 35 intra prediction modes, as shown in Fig. 2, while only nine modes are specified in H.264/AVC.

Among 35 intra prediction modes, the planar and DC modes, i.e., modes 0 and 1 in Fig. 2, are, respectively, appropriate for PUs where there exist gradual changes and little variation on the luma values. In addition to DC and planar modes, the rest of prediction modes considering 33 different directions are often used to capture the characteristics of edges in the PU.

For one $N \times N$ PU, the HEVC encoder generates 35 prediction blocks from corresponding neighboring pixels and intra prediction modes. These reference pixels for intra prediction are encoded pixels from upper-right, upper, upper-left, left and lower-left neighboring blocks, as shown in Fig. 3(a). Note that in Fig. 3(b), the reference pixels $\{s_{N+i,0}\}_{i=1,\ldots,N}$ from the lower-left block are unavailable since they are not yet encoded by HEVC at the time when encoding the current PU. Under this condition, these unavailable pixels $\{s_{N+i,0}\}_{i=1,\ldots,N}$ will be set to equal to the nearest encoded reference pixel $s_{N,0}$ to act as reference pixels. For each intra prediction mode, to calculate the RD cost on the corresponding residual signal, coding operations such as transform, quantization, and entropy coding have to be carried out. The intra prediction mode that yields the minimum RD cost is selected as the optimal one to predict and encode the current PU.

Once the optimal intra prediction mode for each PU is determined, HEVC updates the RD cost associated with each CU in Fig. 1. The hierarchical structure corresponding to the CU, $C^0_{64\times64}$, in Fig. 1 will be pruned and merged according to the RD optimization procedure and hence the optimal intra coding partition corresponding to $C^0_{64\times64}$ can be obtained.

2.2. Transform coding related to the luma intra prediction

To obtain the RD cost on the residual signal, HEVC specifies the transform unit (TU) for transform and quantization coding of the prediction residual. In the HEVC intra coding, before determining the optimal intra mode for one PU, to obtain the associated RD costs, for one $N \times N$ PU with $N \in \{32, 16, 8, 4\}$, there exists only one TU of size $N \times N$, whereas four 32×32 TUs are used when $N = 64$. In transform coding of the luma prediction residual, HEVC

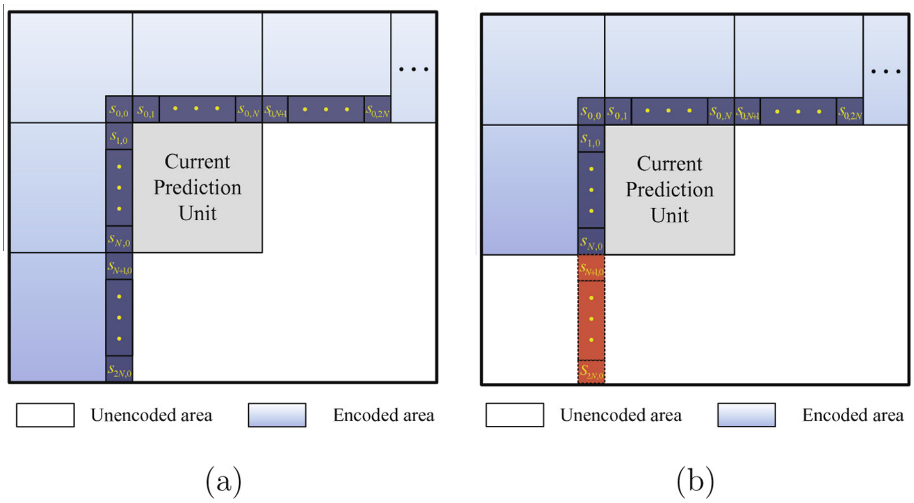


Fig. 3. Reference pixel locations of the HEVC intra prediction: (a) All reference pixels are available; (b) All but partial reference pixels are available.

Table 1
PU sizes and TU sizes in RQT of HEVC.

Intra-coded PU	Possibly attempted TU (s)
64 × 64	32 × 32, 16 × 16
32 × 32	32 × 32, 16 × 16, 8 × 8
16 × 16	16 × 16, 8 × 8, 4 × 4
8 × 8	8 × 8, 4 × 4
4 × 4	4 × 4

provides two kinds of transforms: the integer DST and the integer DCT. The former is used for 4 × 4 TUs while the latter for $N \times N$ TUs with $N = 32, 16, 8$. For simplicity, only the integer DST to 4 × 4 TUs and the integer DCT to 8 × 8 TUs are presented.

Let $R^p_{N\times N}$ denote the luma prediction residual within the $N \times N$ TU for $N = 32, 16, 8, 4$. When performing the integer DST on $R^p_{4\times4}$, the QDST coefficient matrix of $R^p_{4\times4}$ can be expressed as

$$R^{\mathrm{QDST}}_{4\times4} = (S_f R^p_{4\times4} S_f^T) \times (1/Q) \overset{\mathrm{def}}{=} \begin{bmatrix} X_{00} & X_{01} & X_{02} & X_{03} \\ X_{10} & X_{11} & X_{12} & X_{13} \\ X_{20} & X_{21} & X_{22} & X_{23} \\ X_{30} & X_{31} & X_{32} & X_{33} \end{bmatrix}, \qquad (1)$$

where Q is the quantizer step size determined by a quantization parameter (QP) and

$$S_f = \begin{bmatrix} A & B & C & D \\ C & C & 0 & -C \\ D & -A & -C & B \\ B & -D & C & -A \end{bmatrix}.$$

At the decoding stage, after performing inverse QDST (IQDST) on $R^{\mathrm{QDST}}_{4\times4}$, the reconstructed residual can be represented as

$$R^r_{4\times4} = \mathrm{IQDST}\left(R^{\mathrm{QDST}}_{4\times4}\right) = S_f^{-1}\left(R^{\mathrm{QDST}}_{4\times4} \times Q\right)\left(S_f^T\right)^{-1}, \qquad (2)$$

where S_f^{-1} denotes the inverse matrix of S_f. The decoded luma PU can then be derived by adding the reconstructed residual to the corresponding prediction block.

Similarly, when performing the integer DCT on $R^p_{8\times8}$, the corresponding QDCT coefficient matrix $R^{\mathrm{QDCT}}_{8\times8}$ can be expressed as

$$R^{\mathrm{QDCT}}_{8\times8} = (C_f R^p_{8\times8} C_f^T) \times (1/Q) \qquad (3)$$

$$\overset{\mathrm{def}}{=} \begin{bmatrix} Y_{00} & Y_{01} & Y_{02} & Y_{03} & Y_{04} & Y_{05} & Y_{06} & Y_{07} \\ Y_{10} & Y_{11} & Y_{12} & Y_{13} & Y_{14} & Y_{15} & Y_{16} & Y_{17} \\ Y_{20} & Y_{21} & Y_{22} & Y_{23} & Y_{24} & Y_{25} & Y_{26} & Y_{27} \\ Y_{30} & Y_{31} & Y_{32} & Y_{33} & Y_{34} & Y_{35} & Y_{36} & Y_{37} \\ Y_{40} & Y_{41} & Y_{42} & Y_{43} & Y_{44} & Y_{45} & Y_{46} & Y_{47} \\ Y_{50} & Y_{51} & Y_{52} & Y_{53} & Y_{54} & Y_{55} & Y_{56} & Y_{57} \\ Y_{60} & Y_{61} & Y_{62} & Y_{63} & Y_{64} & Y_{65} & Y_{66} & Y_{67} \\ Y_{70} & Y_{71} & Y_{72} & Y_{73} & Y_{74} & Y_{75} & Y_{76} & Y_{77} \end{bmatrix},$$

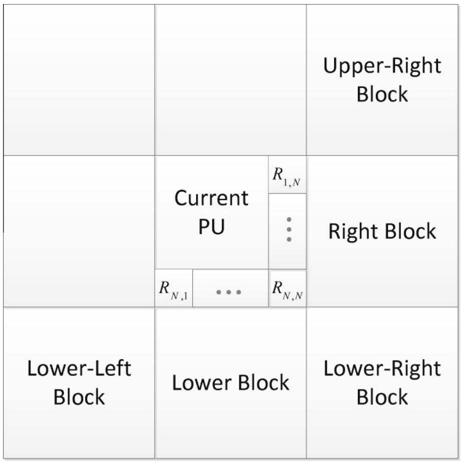


Fig. 4. The image pixels of the current PU used for intra prediction of adjacent unencoded blocks.

Table 2
Truth table of error propagation relationships in HEVC.

Case	Error propagation pattern			Protected pixel set
	Horizontal	Vertical	Diagonal	
1	True	False	True or false	$\{R_{i,N}\}_{i=1,2,\ldots,N}$
2	False	True	True or false	$\{R_{N,j}\}_{j=1,2,\ldots,N}$
3	False	False	True	$\{R_{N,N}\}$
4	False	False	False	None
5	True	True	True or false	$\{R_{i,N}\}_{i=1,2,\ldots,N} \cup \{R_{N,j}\}_{j=1,2,\ldots,N-1}$

where

$$C_f = \begin{bmatrix} a & a & a & a & a & a & a & a \\ b & d & e & g & -g & -e & -d & -b \\ c & f & -f & -c & -c & -f & f & c \\ d & -g & -b & -e & e & b & g & -d \\ a & -a & -a & a & a & -a & -a & a \\ e & -b & g & d & -d & -g & b & -e \\ f & -c & c & -f & -f & c & -c & f \\ g & -e & d & -b & b & -d & e & -g \end{bmatrix}.$$

At the decoder, with inverse QDCT (IQDCT) on $R^{\text{QDCT}}_{8\times8}$, the reconstructed residual can be represented as

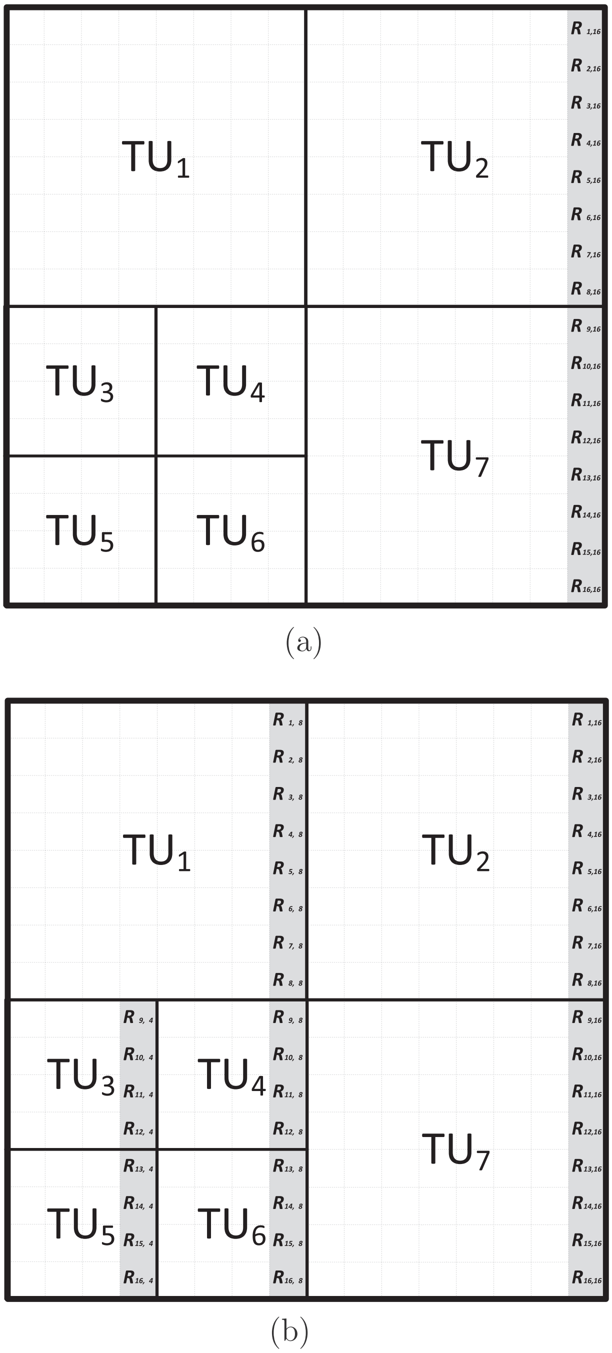


Fig. 5. An example of the proposed bit embedding procedure in dealing with the case that multiple TUs are within one 16 × 16 $Case_1$ PU.

$$R^r_{8\times8} = \text{IQDCT}\left(R^{\text{QDCT}}_{8\times8}\right) = C_f^{-1}\left(R^{\text{QDCT}}_{8\times8} \times Q\right)\left(C_f^T\right)^{-1} \tag{4}$$

and the decoded luma PU can be obtained in the same way.

Note that after the optimal sizes for all PUs and CUs are determined, HEVC attempts to further use smaller-sized TUs for transform coding of the prediction residual associated with the optimal mode, which is different from the H.264/AVC standard. This technique is called residual quadtree (RQT) [7], which aims to find the optimal residual partition such that the used TU (s) can efficiently adapt the characteristic of the prediction residual. For different-sized intra-coded PUs, Table 1 shows the possibly attempted sizes of TUs in RQT [20]. Similar to the quadtree corresponding to one 64 × 64 CU, only the RD optimized TUs in RQT will be preserved and practically applied to final transform and quantization coding.

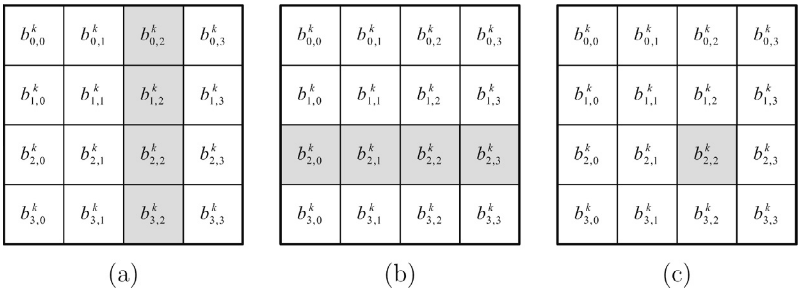
3. The proposed data hiding algorithm for HEVC IFs

3.1. Intra-frame error propagation patterns in HEVC

Since the HEVC intra prediction utilizes spatial correlation among image pixels for compression, the current $N \times N$ PU may refer to the neighboring pixels $\{s_{0,j}\}_{j=0,1,\ldots,2N}$ and $\{s_{i,0}\}_{i=1,\ldots,2N}$ collected from upper-right, upper, upper-left, left and lower-left blocks, as shown in Fig. 3(a), to perform intra prediction. In other words, errors on the pixels $\{R_{i,N}\}_{i=1,\ldots,N}$ and $\{R_{N,j}\}_{j=1,\ldots,N-1}$ of the current $N \times N$ PU, as shown in Fig. 4, may be propagated, via intra prediction, to neighboring upper-right, right, lower-right, lower, and lower-left blocks.

The directions of the intra-frame error propagation can be categorized into three patterns. The first pattern is defined as horizontal error propagation, which means that the errors on referred pixels $\{R_{i,N}\}_{i=1,\ldots,N}$ of the current PU would be propagated to the neighboring right and upper-right blocks. Under this condition, the intra prediction modes of the right and upper-right blocks will be within the mode numbers $\{0, 1-25\}$ and $\{0, 2-10\}$, respectively. The second one is vertical error propagation, where the errors on referred pixels $\{R_{N,j}\}_{j=1,\ldots,N}$ of the current PU would be propagated to the lower and the lower-left blocks. Under this condition, the intra prediction modes of the lower and lower-left blocks will be within the mode numbers $\{0, 1, 11-34\}$ and $\{0, 26-34\}$, respectively. The third one is diagonal error propagation, i.e., the error propagation of the referred pixel, $R_{N,N}$, of the current PU to the lower-right neighboring block, under which the prediction mode numbers will be within the set $\{0, 1, 11-25\}$.

The occurrence of the three error propagation patterns for each $N \times N$ PU in IFs with $N = 64, 32, 16, 8, 4$ can be demonstrated by a truth table, as shown in Table 2, where the true and false values represent, respectively, whether the neighboring intra-coded blocks adopt the specific intra prediction modes or not. For example, a true value for the horizontal error propagation pattern means that the intra prediction mode adopted by the neighboring right block is within $\{0, 1-25\}$ or that adopted by the upper-right block is within $\{0, 2-10\}$. For each case of the current PU with different error propagation patterns, we propose to protect a specific pixel set against errors such that the intra-frame error propagation can be eliminated. These specific pixel sets are called the protected pixel sets here and demonstrated in the last column of Table 2. For example, when the current PU only has the horizontal error propagation pattern, i.e., $Case_1$ in Table 2, once there is no error on pixels $\{R_{i,N}\}_{i=1,\ldots,N}$ of the current PU, the intra-frame error propagation along the horizontal direction no longer happens to the neighboring right and upper-right blocks.



$b^k_{0,0}$	$b^k_{0,1}$	$b^k_{0,2}$	$b^k_{0,3}$
$b^k_{1,0}$	$b^k_{1,1}$	$b^k_{1,2}$	$b^k_{1,3}$
$b^k_{2,0}$	$b^k_{2,1}$	$b^k_{2,2}$	$b^k_{2,3}$
$b^k_{3,0}$	$b^k_{3,1}$	$b^k_{3,2}$	$b^k_{3,3}$

Fig. 6. The kth 4 × 4 candidate reconstructed block with $k = 1, 2, \ldots, K$ and the pixel (s) suffered from the embedding errors corresponding to (a) $Case_1$, (b) $Case_2$ and $Case_3$, and (c) $Case_5$.

Fig. 7. Test videos: (a) *PeopleOnStreet* (2560 × 1600), (b) *Traffic* (2560 × 1600), (c) *BasketballDrive* (1920 × 1080), (d) *ParkScene* (1920 × 1080), (e) *FourPeople* (1280 × 720), (f) *Johnny* (1280 × 720), (g) *PartyScene* (832 × 480), (h) *RaceHorses* (832 × 480), (i) *BasketballPass* (416 × 240), (j) *BlowingBubbles* (416 × 240).

Table 3
Configurations of HM 10.0 and JM 16.0.

HM 10.0		JM 16.0	
Configuration name	Value	Configuration name	Value
Frame rate	30	Frame rate	30
Frames to be encoded	96	Frames to be encoded	96
Intra period	32	Intra period	32
Quadtree TU max depth intra	3	Profile IDC	77
GOP size	32	Symbol mode	1
RDOQTS	1	Level IDC	50
Transform skip	1	RD optimization	1

3.2. The DCT/DST perturbation pattern analysis

For compressed video sequences, embedding data in the spatial domain is more time-consuming, when compared with that in the frequency domain, since it requires extra inverse integer DCT/DST and de-quantization operations. The proposed data hiding algorithm aims to embed hidden bits in the transform domain, in which the QDCTs and QDSTs of intra prediction residuals are perturbed to carry the hidden bits. According to different classified cases and protected pixel sets associated with the current PU, as shown in Table 2, different bit embedding patterns are presented for perturbing quantized coefficients without inducing intra-frame error propagation.

In HEVC, the residual of one RD optimized intra-coded PU is encoded by either the same-sized TU or smaller-sized TUs, which is determined by the RD optimization in RQT. For the former, we only discuss the cases that the PUs are of sizes 4 × 4 and 8 × 8 here for simplicity and the complete results for the proposed bit embedding patterns are given in [24]. For the latter, we present an example of multiple RD optimized TUs within one 16 × 16 intra-coded PU for demonstration.

For one 4 × 4 PU, denote the corresponding residual block and QDST coeeficient matrix as $R^p_{4\times4}$ and $R^{\text{QDST}}_{4\times4}$, respectively. When the current 4 × 4 PU is categorized into $Case_1$, we select coefficients from $R^{\text{QDST}}_{4\times4}$ to yield one set of four triples, i.e., $S_H = \{(X_{00}, X_{02}, X_{03}), (X_{10}, X_{12}, X_{13}), (X_{20}, X_{22}, X_{23}), (X_{30}, X_{32}, X_{33})\}$, for embedding four hidden bits without inducing error propagation to its neighboring blocks. The subscript of S_H stands for horizontal. Each triple in set S_H can be used to embed one bit by perturbing the three corresponding QDSTs and then all these perturbations on QDSTs associated with the four embedded bits would keep the leftmost and rightmost column pixel values of the reconstructed block unchanged. On the other hand, performing bit embedding on S_H will not induce intra-frame error propagation along the horizontal direction.

The following coefficient perturbing example is provided to justify the bit embedding procedure without intra-frame error propagation. To embed one hidden bit in the triple (X_{00}, X_{02}, X_{03}), the triple (X_{00}, X_{02}, X_{03}) is perturbed to $(X_{00} + t, X_{02} - t, X_{03} + t)$ with $t \in \mathscr{Z}$ to yield a perturbed QDST coefficient matrix $R^{\text{QDST}'}_{4\times4}$. The difference between $R^{\text{QDST}'}_{4\times4}$ and $R^{\text{QDST}}_{4\times4}$ can be represented as

$$\Delta R^{\text{QDST}}_{4\times4} = R^{\text{QDST}'}_{4\times4} - R^{\text{QDST}}_{4\times4} = \begin{bmatrix} t & 0 & -t & t \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}.$$

Table 4
Accuracy performance (%) for the proposed quality improvement scheme in different cases, transform schemes and QPs.

		Metric	QP = 17	QP = 22	QP = 27	QP = 32	QP = 37	QP = 42
Case 1	4 × 4 DST	TP	40	43	42	40	36	34
		TN	13	17	20	22	24	25
		FP	5	5	6	8	10	7
		FN	42	35	32	30	30	34
Case 2	4 × 4 DST	TP	36	42	43	41	37	32
		TN	13	16	19	22	24	24
		FP	5	5	5	6	7	4
		FN	46	37	33	31	32	40
Case 3	4 × 4 DST	TP	21	33	30	36	37	42
		TN	11	15	18	20	23	26
		FP	5	5	6	4	2	2
		FN	63	47	46	40	38	30
Case 5	4 × 4 DST	TP	45	48	49	50	50	50
		TN	47	48	49	49	50	50
		FP	3	2	1	1	0	0
		FN	5	2	1	0	0	0
	8 × 8 DCT	TP	45	47	48	50	50	50
		TN	42	46	49	49	50	50
		FP	8	4	2	1	0	0
		FN	5	3	1	0	0	0
	16 × 16 DCT	TP	45	49	49	50	49	50
		TN	44	47	50	49	51	50
		FP	6	2	0	1	0	0
		FN	5	2	1	0	0	0
	32 × 32 DCT	TP	34	47	50	50	50	51
		TN	50	50	49	50	51	49
		FP	2	1	0	0	0	0
		FN	14	2	1	0	0	0

Table 5
The PSNR (DB) gain from applying the EMBED-W over the EMBED-O under different QPS.

QP	PSNR gain
17	0.70
22	0.95
27	1.13
32	1.23
37	1.08
42	0.80
Average	0.98

The reconstructed perturbed residual block, denoted as $R^{r'}_{4\times4}$, can be obtained by performing IQDST defined in (2) on $R^{\mathrm{QDST}'}_{4\times4}$. The difference between $R^{r'}_{4\times4}$ and $R^{r}_{4\times4}$ can be calculated by

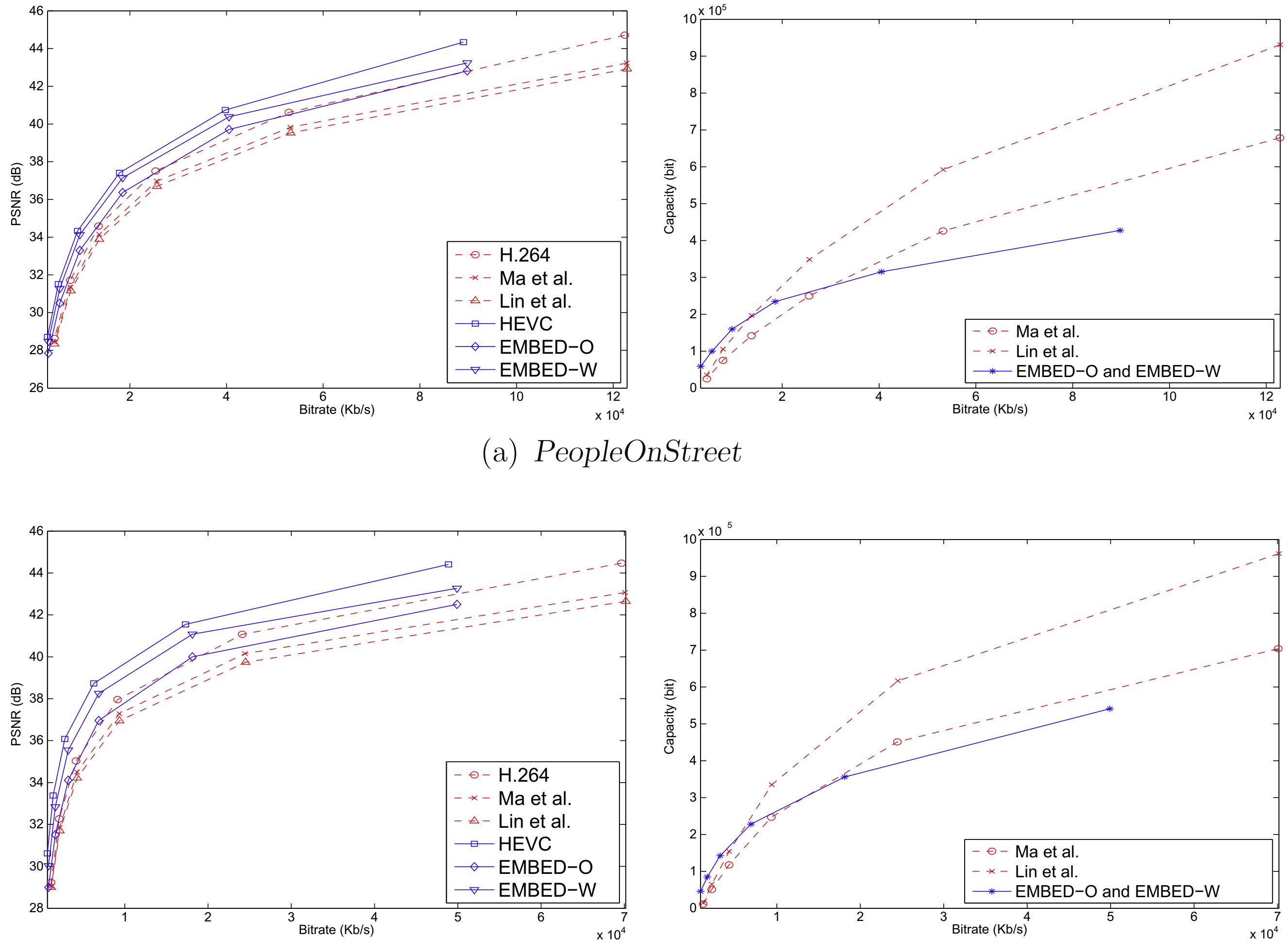
$$\Delta R^{r'}_{4\times4} = R^{r'}_{4\times4} - R^{r}_{4\times4} = S_f^{-1}\left(\Delta R^{\mathrm{QDST}}_{4\times4} \times Q\right)\left(S_f^{T}\right)^{-1}$$
$$= Q \times t \times \begin{bmatrix} 0 & 0 & 3AC & 0 \\ 0 & 0 & 3BC & 0 \\ 0 & 0 & 3C^2 & 0 \\ 0 & 0 & 3CD & 0 \end{bmatrix}.$$

As shown, the rightmost column of $\Delta R^{r}_{4\times4}$ is a zero vector, which means that imposing the perturbation pattern $(+t, -t, +t)$ on (X_{00}, X_{02}, X_{03}) will not induce errors on the rightmost column vector of the reconstructed residual block $R^{r'}_{4\times4}$. It implies that the perturbed PU will not propagate errors to its neighboring blocks which refer to the *Case*$_1$ protected pixels $\{R_{1,4}, \ldots, R_{4,4}\}$ for intra prediction. The perturbation patterns for the remaining three triples, (X_{10}, X_{12}, X_{13}), (X_{20}, X_{22}, X_{23}) and (X_{30}, X_{32}, X_{33}), can be carried out in the same way, and preserve the error propagation-free characteristic. In other words, we can embed four hidden bits in one 4 × 4 PU categorized into *Case*$_1$ without inducing intra-frame error propagation.

When the current PU is categorized into *Case*$_2$, for avoiding the intra-frame error propagation along the vertical direction, we define a set of four triples selected from $R^{\mathrm{QDST}'}_{4\times4}$, i.e. $S_V = \{(X_{00}, X_{20}, X_{30}), (X_{01}, X_{21}, X_{31}), (X_{02}, X_{22}, X_{32}), (X_{03}, X_{23}, X_{33})\}$. For each triple $(X_{0j}, X_{2j}, X_{3j}), j = 0, 1, 2, 3$ in set S_V, we can embed one hidden bit by perturbing the triple (X_{0j}, X_{2j}, X_{3j}) with $(+t, -t, +t)$ such that four bits can be embedded in total for one 4 × 4 PU in the *Case*$_2$ category. Derivations, which are similar to that for *Case*$_1$ PUs, revealed that the matrix $\Delta R^{r}_{4\times4}$ for one *Case*$_2$ PU is exactly the transpose of that for *Case*$_1$ PUs. Since the bottom row of $\Delta R^{r}_{4\times4}$ is a zero vector, the perturbed PU block will not propagate errors to its neighboring blocks that utilize the *Case*$_2$ protected pixels $\{R_{4,1}, \ldots, R_{4,4}\}$ for intra prediction. For one *Case*$_3$ PU, since only the lower-right corner pixel $R_{4,4}$ needs to be protected, it can be handled in the same way as that for either *Case*$_1$ or *Case*$_2$ PUs. For simplicity, the *Case*$_3$ PU is handled as that for *Case*$_2$ PUs and hence four bits can be embedded. For *Case*$_4$ PUs, since no reference pixels need to be protected, all the sixteen quantized coefficients can be used for embedding hidden bits.

For *Case*$_5$ PUs, we propose to select nine coefficients for embedding one hidden bit, i.e., $(X_{00}, X_{02}, X_{03}, X_{20}, X_{22}, X_{23}, X_{30}, X_{32}, X_{33})$. The perturbation pattern for embedding one bit can be represented by the following equation that specifies the relation between $R^{\mathrm{QDST}'}_{4\times4}$ and $R^{\mathrm{QDST}}_{4\times4}$, i.e.,

$$\Delta R^{\mathrm{QDST}}_{4\times4} = R^{\mathrm{QDST}'}_{4\times4} - R^{\mathrm{QDST}}_{4\times4} = \begin{bmatrix} t & 0 & -t & t \\ 0 & 0 & 0 & 0 \\ -t & 0 & t & -t \\ t & 0 & -t & t \end{bmatrix}.$$



(a) *PeopleOnStreet*

(b) *Traffic*

Fig. 8. Objective data hiding performance evaluation with PSNR vs. bitrate and capacity vs. bitrate plots for concerned algorithms on test videos of size 2560 × 1600.

The difference between $R^{r'}_{4\times4}$ and $R^{r}_{4\times4}$ can be computed by

$$\Delta R^{r}_{4\times4} = R^{r'}_{4\times4} - R^{r}_{4\times4} = S_f^{-1}\left(\Delta R^{\mathrm{QDST}}_{4\times4}\times Q\right)\left(S_f^{T}\right)^{-1}$$
$$= Q\times t\times\begin{bmatrix}0&0&0&0\\0&0&0&0\\0&0&9C^2&0\\0&0&0&0\end{bmatrix}.$$

As shown above, the rightmost column and the lowest row of the $\Delta R^{r}_{4\times4}$ are zero vectors such that, after embedding one bit, the protected pixels of $Case_5$ would be intact, from which the neighboring blocks can perform intra prediction without error propagation.

We then describe the embedding procedure and perturbation patterns for the 8 × 8 intra-coded PU. Denote the corresponding residual block and QDCT coefficient matrix as $R^{p}_{8\times8}$ and $R^{\mathrm{QDCT}}_{8\times8}$, respectively. When the current PU is categorized into $Case_1$, the set of eight coefficient pairs selected from $R^{\mathrm{QDCT}}_{8\times8}$ for avoiding the intra-frame error propagation along the horizontal direction is denoted as $S_H = \{(Y_{00}, Y_{04}), (Y_{10}, Y_{14}), \ldots, (Y_{70}, Y_{74})\}$. Each pair in S_H can be used to embed one bit by perturbing (Y_{i0}, Y_{i4}) with $(+t, -t)$ for $i = 0, 1, \ldots, 7$ and eight bits can be embedded in one $Case_1$ PU without intra-frame error propagation.

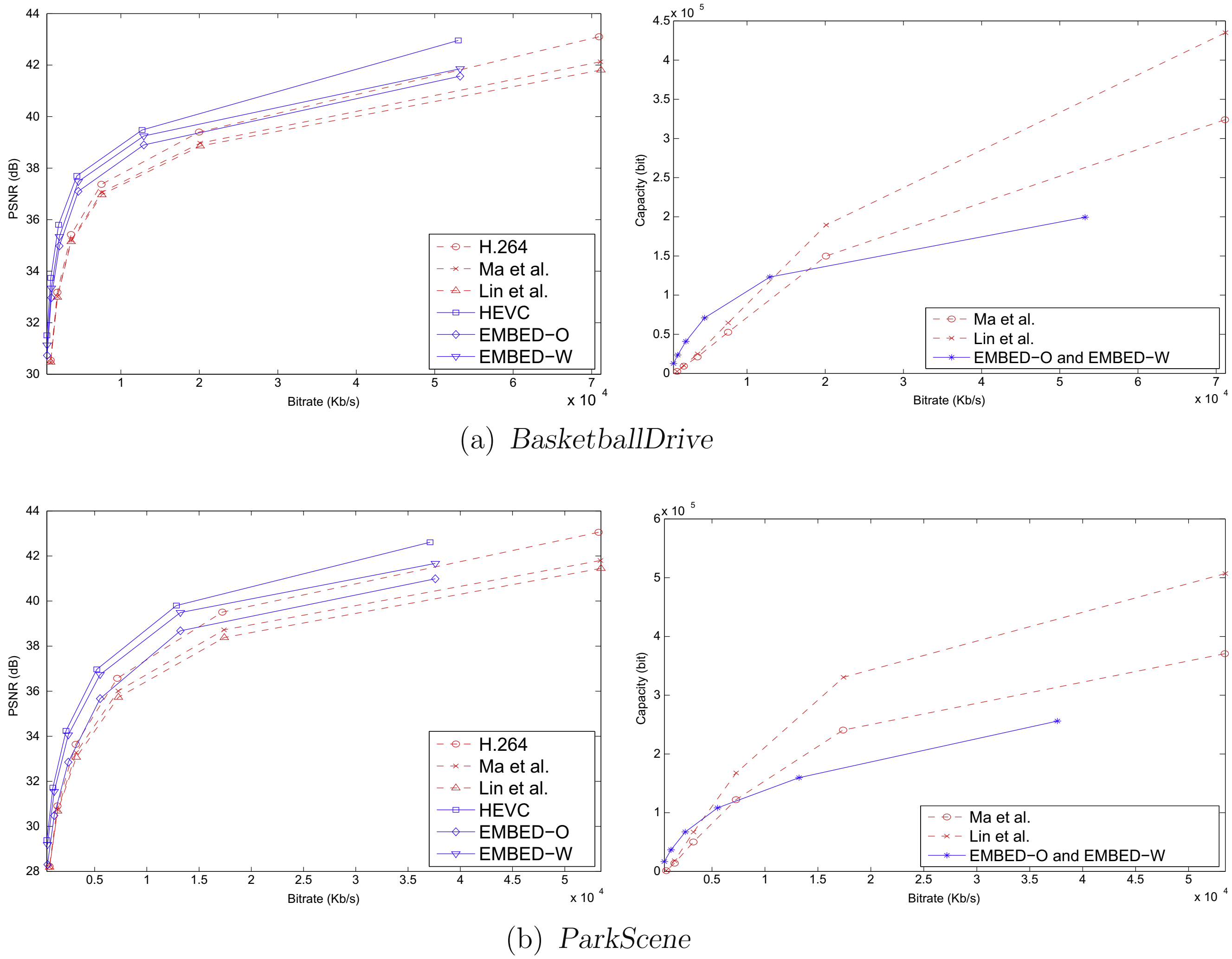
The following example is provided to justify the intra-frame error propagation-free characteristic of the imposed perturbation pattern for $Case_1$ PUs. By perturbing (Y_{00}, Y_{04}) with $(+t, -t)$, the difference between the perturbed and the original matrixes, i.e., $R^{\mathrm{QDCT}'}_{8\times8}$ and $R^{\mathrm{QDCT}}_{8\times8}$, can be expressed as

$$\Delta R^{\mathrm{QDCT}}_{8\times8} = R^{\mathrm{QDCT}'}_{8\times8} - R^{\mathrm{QDCT}}_{8\times8} = \begin{bmatrix}t&0&0&0&-t&0&0&0\\0&0&0&0&0&0&0&0\\0&0&0&0&0&0&0&0\\0&0&0&0&0&0&0&0\\0&0&0&0&0&0&0&0\\0&0&0&0&0&0&0&0\\0&0&0&0&0&0&0&0\\0&0&0&0&0&0&0&0\end{bmatrix}.$$

By performing inverse IQDCT on $\Delta R^{\mathrm{QDCT}}_{8\times8}$, the difference between $R^{r'}_{8\times8}$ and $R^{r}_{8\times8}$, $\Delta R^{r}_{8\times8}$, can be represented as

$$\Delta R^{r}_{8\times8} = R^{r'}_{8\times8} - R^{r}_{8\times8} = C_f^{-1}\left(\Delta R^{\mathrm{QDCT}}_{8\times8}\times Q\right)\left(C_f^{T}\right)^{-1}$$
$$= Q\times t\times\begin{bmatrix}0&2a^2&2a^2&0&0&2a^2&2a^2&0\\0&2a^2&2a^2&0&0&2a^2&2a^2&0\\0&2a^2&2a^2&0&0&2a^2&2a^2&0\\0&2a^2&2a^2&0&0&2a^2&2a^2&0\\0&2a^2&2a^2&0&0&2a^2&2a^2&0\\0&2a^2&2a^2&0&0&2a^2&2a^2&0\\0&2a^2&2a^2&0&0&2a^2&2a^2&0\\0&2a^2&2a^2&0&0&2a^2&2a^2&0\end{bmatrix}.$$

The rightmost column of $\Delta R^{r}_{8\times8}$ is a zero vector such that there is no error propagation to neighboring blocks along the intra prediction



(a) *BasketballDrive*

(b) *ParkScene*

Fig. 9. Objective data hiding performance evaluation with PSNR vs. bitrate and capacity vs. bitrate plots for concerned algorithms on test videos of size 1920 × 1080.

directions specified in $Case_1$. For 8×8 PUs categorized into $Case_2$, $Case_3$, or $Case_4$, since the operations for preserving the intra-frame error propagation-free characteristic are similar to those for 4×4 PUs categorized into $Case_2$, $Case_3$, or $Case_4$, only the set of coefficient pairs in the embedding procedure of 8×8 $Case_2$ PUs for avoiding the error propagation along the vertical direction is provided here, i.e., $S_V = \{(Y_{0j}, Y_{4j})\}_{j=0,1,\ldots,7}$.

When the current 8×8 PU is categorized into $Case_5$, the suggested four coefficients for embedding one hidden bit are $(Y_{00}, Y_{04}, Y_{40}, Y_{44})$. They are perturbed in a way such that the difference between $R^{\mathrm{QDCT}'}_{8\times8}$ and $R^{\mathrm{QDCT}}_{8\times8}$ would be

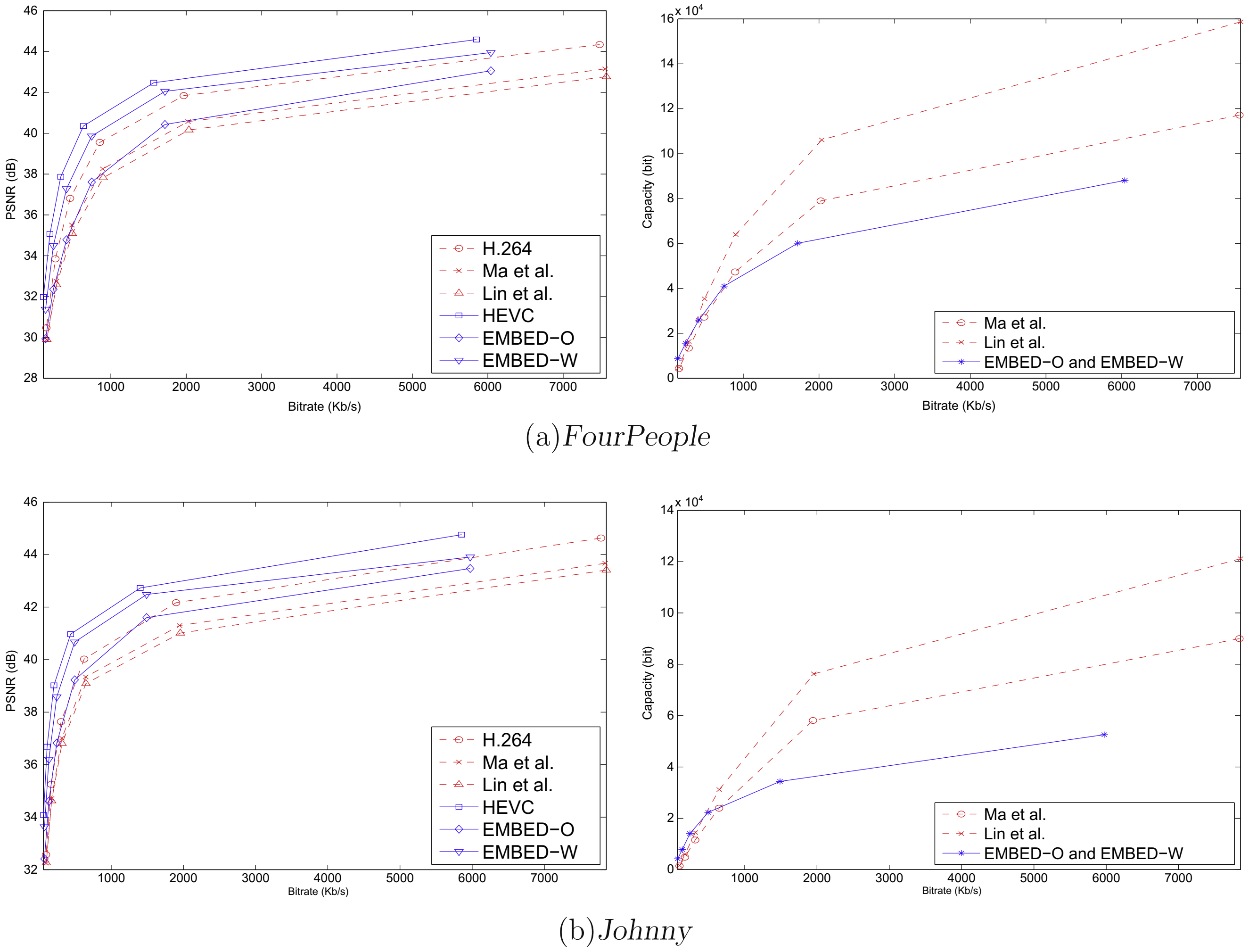
$$\Delta R^{\mathrm{QDCT}}_{8\times8} = R^{\mathrm{QDCT}'}_{8\times8} - R^{\mathrm{QDCT}}_{8\times8} = \begin{bmatrix} t & 0 & 0 & 0 & -t & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -t & 0 & 0 & 0 & t & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}.$$

The difference between $R^{r'}_{8\times8}$ and $R^{r}_{8\times8}$ can be represented as

$$\Delta R^{r}_{8\times8} = R^{r'}_{8\times8} - R^{r}_{8\times8} = C_f^{-1}(\Delta R^{\mathrm{QDCT}}_{8\times8} \times Q)(C_f^T)^{-1}$$
$$= Q \times t \times \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 4a^2 & 4a^2 & 0 & 0 & 4a^2 & 4a^2 & 0 \\ 0 & 4a^2 & 4a^2 & 0 & 0 & 4a^2 & 4a^2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 4a^2 & 4a^2 & 0 & 0 & 4a^2 & 4a^2 & 0 \\ 0 & 4a^2 & 4a^2 & 0 & 0 & 4a^2 & 4a^2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}.$$

The rightmost column and the bottom low of $\Delta R^{r}_{8\times8}$ are zero vectors such that neighboring blocks performing intra prediction along the directions specified in $Case_5$ would not suffer intra-frame error propagation.

Note that when the residual of one intra-coded PU is split into smaller blocks for transform coding, these smaller-sized TUs share the same intra prediction information with the corresponding PU. According to which case the corresponding PU is categorized into, the aforementioned QDCT/QDST perturbation patterns can also be used for embedding hidden bits into these split residual blocks without inducing the intra-frame error propagation. To better understand the proposed bit embedding procedure in dealing with the PU with smaller-sized TUs, we present an example for demon-



(a) *FourPeople*

(b) *Johnny*

Fig. 10. Objective data hiding performance evaluation with PSNR vs. bitrate and capacity vs. bitrate plots for concerned algorithms on test videos of size 1280 × 720.

stration, as shown in Fig. 5, where seven RD optimized TUs are within one 16 × 16 intra-coded PU categorized into $Case_1$. Since the corresponding PU is categorized into $Case_1$, according to the defined protected pixel set for $Case_1$ PUs in Table 2, pixels at locations marked as gray in Fig. 5(a), $\{R_{1,16}, R_{2,16}, \ldots, R_{16,16}\}$, have to be protected for avoiding the error propagation to neighboring blocks along the horizontal direction. By perturbing the quantized coefficients of 8 × 8 QDCT and 4 × 4 QDST in these smaller-sized TUs, respectively, with the designed patterns for $Case_1$ PUs, the pixels at locations marked as gray in Fig. 5(b) will be kept intact since all these perturbation patterns do not change the rightmost column pixel values for each reconstructed sub-block. The pixels that are protected, as shown in Fig. 5(b), indeed contain the pixels that have to be protected, as shown in Fig. 5(a). In addition, since these split smaller-sized TUs are coded independently, there is no error propagation between these TUs within the PU. This example demonstrates that the proposed bit embedding procedure can also eliminate intra-frame error propagation for intra-coded PUs with smaller-sized TUs.

3.3. The proposed DCT/DST based data hiding algorithm

In the proposed data hiding algorithm, the embedding procedure is carried out in a way such that the value of the first coefficient in each perturbation pattern of QDSTs or QDCTs would be modified to make its odevity consistent with that of the hidden bit, i.e., 1 or 0, for embedding one-bit hidden data, while the extraction procedure identifies the hidden bit by checking the odevity of the first coefficient in each perturbation pattern. Since the procedures of embedding and extraction for the quantized coefficient matrixes within different-sized TUs are basically the same, only the procedure for 4 × 4 TUs is presented here.

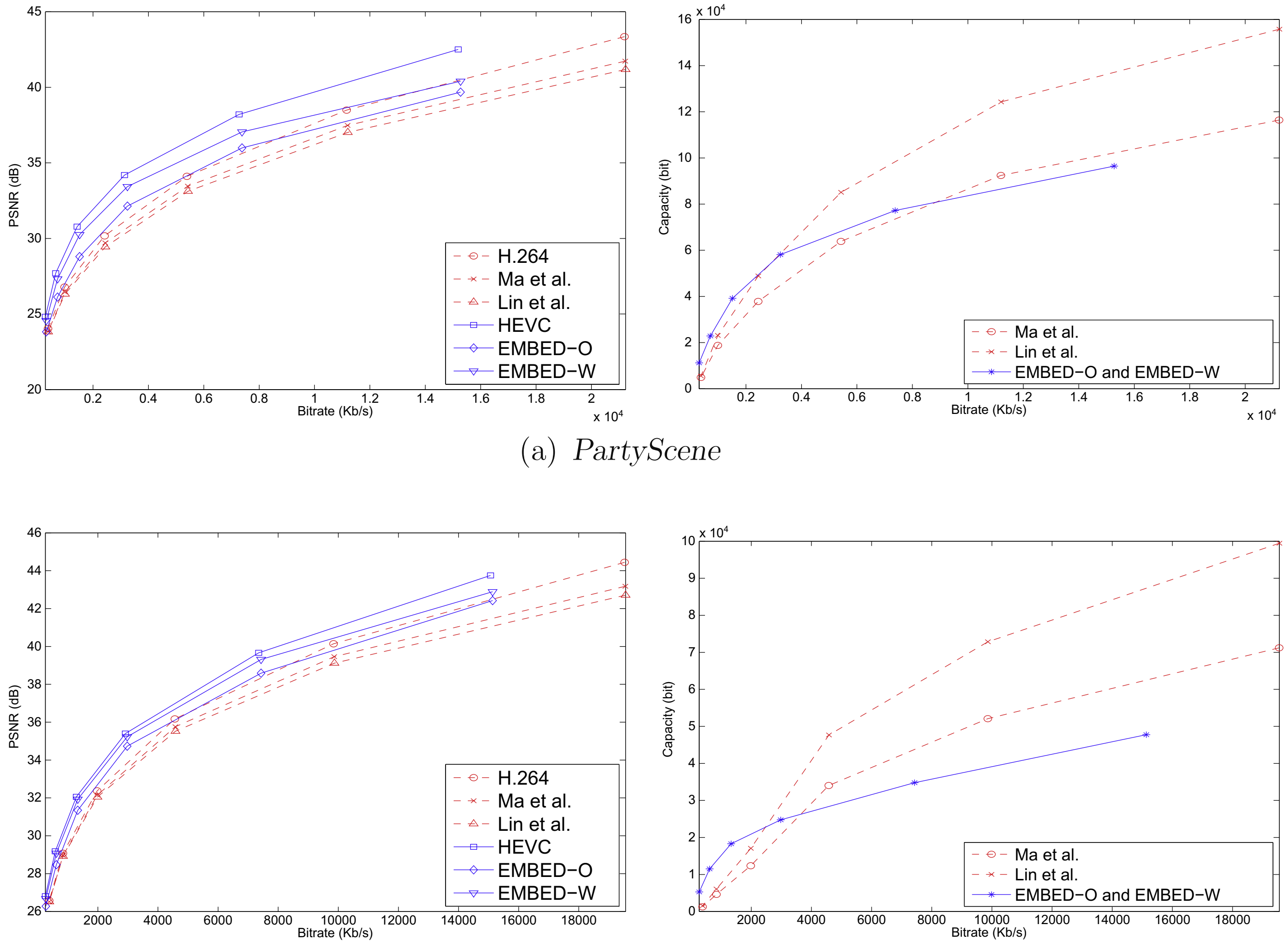
At the encoder side, for the quantized coefficient matrix within one 4 × 4 TU corresponding to an intra-coded PU categorized into $Case_1$, each triple in S_H, expressed as (X_{ij}, X_{mn}, X_{pq}), will be checked first. For each triple in S_H, only when its first coefficient, X_{ij}, is nonzero, the triple (X_{ij}, X_{mn}, X_{pq}) is perturbed to $(X'_{ij}, X'_{mn}, X'_{pq})$ for carrying one hidden bit h; that is,

$$(X'_{ij}, X'_{mn}, X'_{pq}) = (X_{ij} + t, X_{mn} - t, X_{pq} + t),$$

where

$$t = \begin{cases} 1, & \text{if } (X_{ij} > 0, h = 1, \text{ and } X_{ij} \text{ is even}), \\ & \text{or } (X_{ij} > 0, h = 0, \text{ and } X_{ij} \text{ is odd}), \\ -1, & \text{if } (X_{ij} < 0, h = 1, \text{ and } X_{ij} \text{ is even}), \\ & \text{or } (X_{ij} < 0, h = 0, \text{ and } X_{ij} \text{ is odd}), \\ 0, & \text{if } (X_{ij} \neq 0, h = 1, \text{ and } X_{ij} \text{ is odd}), \\ & \text{or } (X_{ij} \neq 0, h = 0, \text{ and } X_{ij} \text{ is even}). \end{cases} \quad (5)$$

When the corresponding PU is categorized into $Case_2$ or $Case_3$, each triple in S_V can be treated as that in S_H for carrying one hidden bit. For $Case_4$, since no pixel in the current PU has to be protected,



(a) *PartyScene*

(b) *RaceHorses*

Fig. 11. Objective data hiding performance evaluation with PSNR vs. bitrate and capacity vs. bitrate plots for concerned algorithms on test videos of size 832 × 480.

each of the sixteen QDSTs, $\{X_{ij}\}_{i,j=0,1,2,3}$, is perturbed to X'_{ij} by (6) for carrying one hidden bit h if X_{ij} is non-zero.

$$X'_{ij} = X_{ij} + t, \tag{6}$$

where t is defined like (5). For $Case_5$, the nine coefficients suggested in the previous subsection, $(X_{00}, X_{02}, X_{03}, X_{20}, X_{22}, X_{23}, X_{30}, X_{32}, X_{33})$, are perturbed to $(X'_{00}, X'_{02}, X'_{03}, X'_{20}, X'_{22}, X'_{23}, X'_{30}, X'_{32}, X'_{33})$, as shown in (7), for embedding h if X_{00} is non-zero.

$$\begin{aligned}(X'_{00}, X'_{02}, X'_{03}, X'_{20}, X'_{22}, X'_{23}, X'_{30}, X'_{32}, X'_{33}) \\ = (X_{00} + t, X_{02} - t, X_{03} + t, X_{20} - t, X_{22} + t, X_{23} - t, X_{30} \\ + t, X_{32} - t, X_{33} + t),\end{aligned} \tag{7}$$

where t has the same definition as (5) but replaces X_{ij} with X_{00} totally. Note that embedding hidden bits into a block with all zero coefficients will lead to degraded perception quality [11,12]. This is the reason why we embed hidden bits on non-zero coefficients.

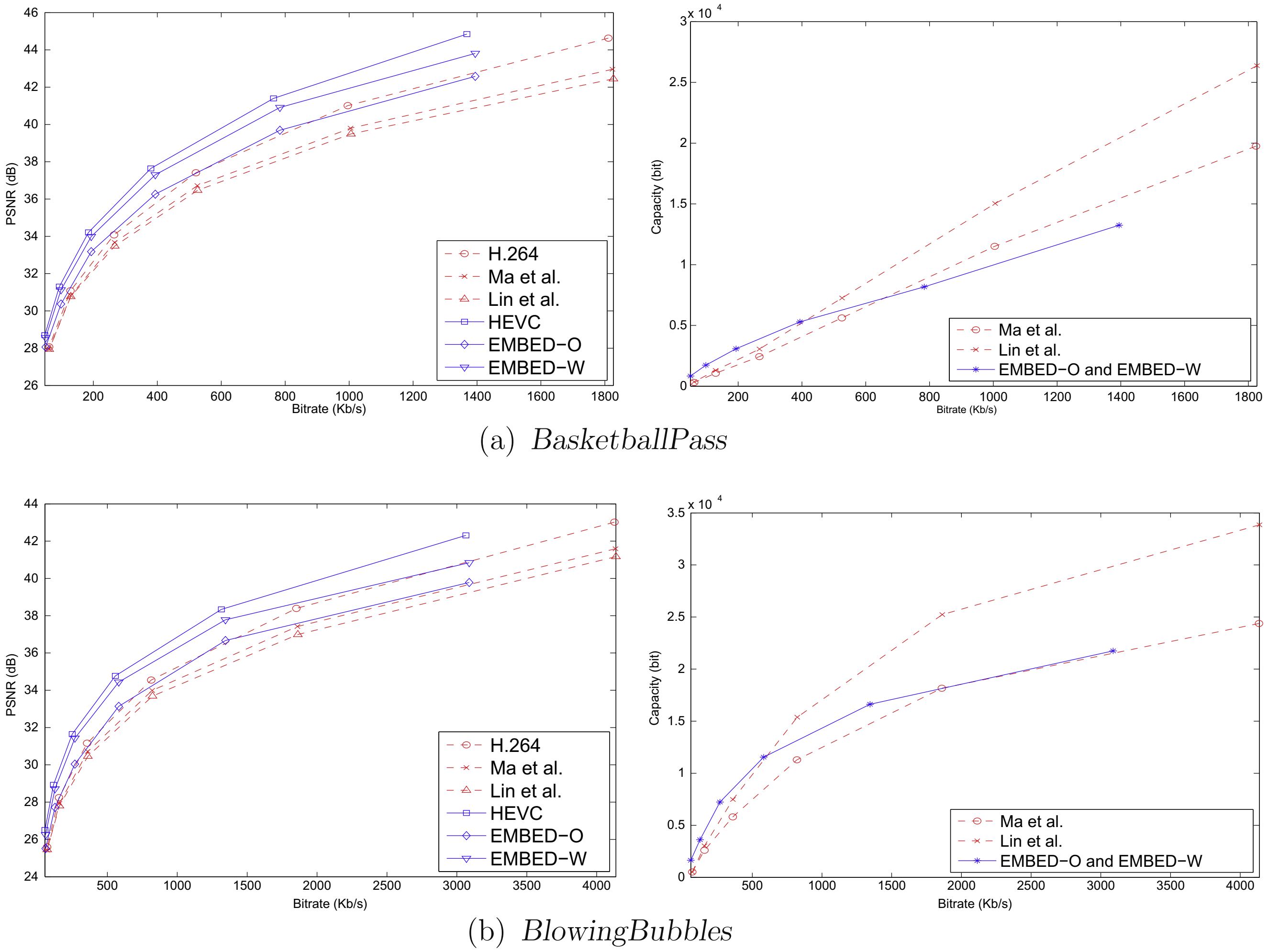
At the decoder side, the extraction procedure is carried out after the HEVC compressed video sequence is entropy decoded. Given the de-quantized coefficient matirx within one 4 × 4 TU, we first determine which case the corresponding PU is categorized into according to the intra prediction modes of neighboring blocks. When the corresponding PU is categorized into $Case_1$, $Case_2$, or $Case_3$, the first coefficient X'_{ij} of each triple $(X'_{ij}, X'_{mn}, X'_{pq})$ in S_H or S_V will be checked, respectively. If X'_{ij} is non-zero, the hidden bit can be extracted by

$$h = \begin{cases} 1, & \text{if } X'_{ij} \text{ is odd}, \\ 0, & \text{otherwise}. \end{cases} \tag{8}$$

For $Case_4$, the hidden bits can be extracted from non-zero coefficients among the sixteen quantized coefficient $\{X'_{ij}\}_{i,j=0,1,2,3}$ by (8). For $Case_5$, if the first coefficient X'_{00} of the nine suggested coefficients is non-zero, the hidden bit can be extracted by (8) with replacing X'_{ij} with X'_{00}.

3.4. Quality improvement for IFs reconstruction

In this subsection, we describe how to improve the quality of IFs in the reconstructed video sequence based on spatial domain pixel correlations at the decoder. In general, spatial inter-pixel correlation is stronger in the unperturbed block than that in the perturbed one. To improve the quality of the reconstructed IFs, we propose a scheme to investigate the inter-pixel correlation such that after extracting the hidden bit (s), the original unperturbed block can be found from all possible candidate blocks derived from the current decoded PU. By replacing with found unperturbed block, the errors in the current block caused by embedding can be eliminated and meanwhile the problem of the inter-frame error propagation is also alleviated.



(a) *BasketballPass*

(b) *BlowingBubbles*

Fig. 12. Objective data hiding performance evaluation with PSNR vs. bitrate and capacity vs. bitrate plots for concerned algorithms on test videos of size 416 × 240.

Table 6
Percentages (%) of adopted blocks on different QPS under H.264/AVC and HEVC frameworks.

	QP = 17	QP = 22	QP = 27	QP = 32	QP = 37	QP = 42
H.264/AVC						
4 × 4 block	84	78	66	57	49	38
Non 4 × 4 Block	16	22	34	43	51	62
HEVC						
4 × 4 block	73	73	68	59	48	32
Non 4 × 4 Block	27	27	32	41	52	68

The hidden bit (s) is first extracted from the current decoded PU as 1 or 0, respectively, according to the coefficient, odd or even. To prevent the current decoded PU from reconstruction with embedding errors, the decoder is designed to select, from several candidate unperturbed blocks, the one that yields strongest inter-pixel correlation. Since this quality improvement procedure can be carried out in the same way for different-sized TUs, we present the procedure that deals with one 4 × 4 TU as an example for simplicity.

Given the de-quantized coefficient matrix within one 4 × 4 TU whose corresponding PU is categorized into $Case_1$, we first check whether the first coefficient $X\prime_{ij}$ of each triple, $(X'_{ij},X'_{mn},X'_{pq})$, in S_H is zero or not. If X'_{ij} is zero, it denotes that no hidden bit is embedded in the triple, i.e. $(X'_{ij},X'_{mn},X'_{pq}) = (X_{ij},X_{mn},X_{pq})$. For a non-zero X'_{ij}, in addition to extracting the hidden bit from X'_{ij}, we can infer the relation between $(X'_{ij},X'_{mn},X'_{pq})$ and (X_{ij},X_{mn},X_{pq}) from (5) as follows:

$$(X_{ij},X_{mn},X_{pq}) = \begin{cases} (X'_{ij},X'_{mn},X'_{pq}) \text{ or } (X'_{ij}-1,X'_{mn}+1,X'_{pq}-1), \\ \text{if } X'_{ij}>0, \\ (X'_{ij},X'_{mn},X'_{pq}) \text{ or } (X'_{ij}+1,X'_{mn}-1,X'_{pq}+1), \\ \text{otherwise}. \end{cases}$$

Depending on the value of X'_{ij}, there are at most two trials for each triple $(X'_{ij},X'_{mn},X'_{pq})$ to yield original one (X_{ij},X_{mn},X_{pq}). By combining all possible trials for the four triples in S_H, there are at most $K = 2^4$ candidate original blocks. Denote the k-th candidate block as $B_k, k = 1,2,\ldots,K$, which can be reconstructed by adding the decoded residual block to the prediction block. To evaluate the confidence of each candidate block B_k, a cost function is designed to measure the inter-pixel correlation for each B_k. The candidate that yields the minimum cost function value is selected as the original unperturbed block. For $Case_1$ PUs, only the third column pixels of the reconstructed block would suffer from embedding errors, as shown in Fig. 6(a), where $b^k_{i,j}$ denotes the luma value of the pixel at position (i,j) for the k-th candidate block. The cost function is designed to reflect the spatial correlation of the candidate original block based on inter-pixel differences between perturbed and

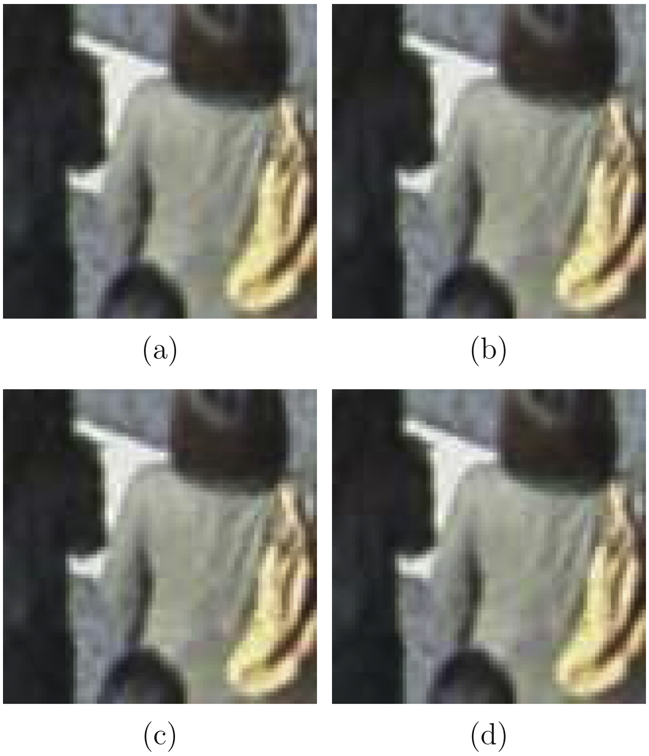


Fig. 13. Subjective performance evaluation of the reconstructed IFs of the first frame of the *PeopleOnStreet* sequence by: (a) Original, (b) Ma et al.'s algorithm, (c) Lin et al.'s algorithm, (d) proposed EMBED-W with QP = 22.

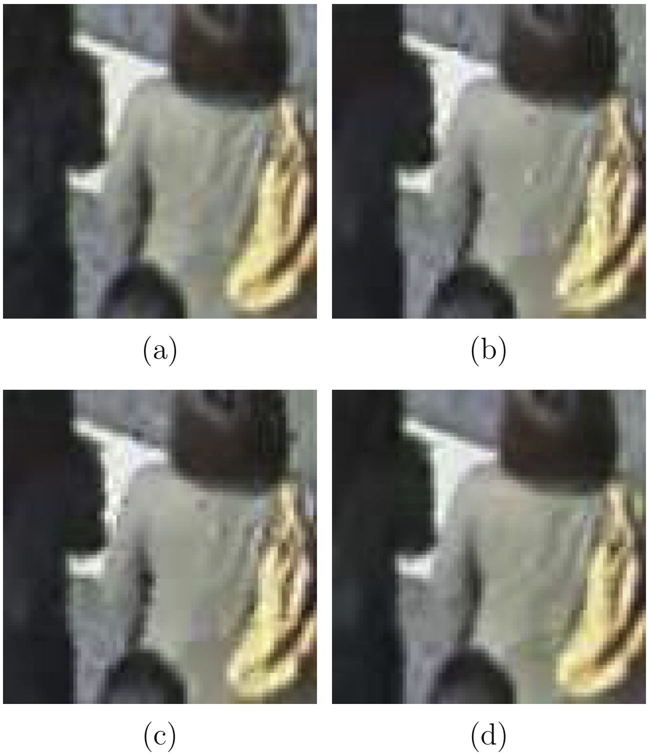


Fig. 14. Subjective performance evaluation of the reconstructed IFs of the first frame of the *PeopleOnStreet* sequence by: (a) Original, (b) Ma et al.'s algorithm, (c) Lin et al.'s algorithm, (d) proposed EMBED-W with QP = 32.

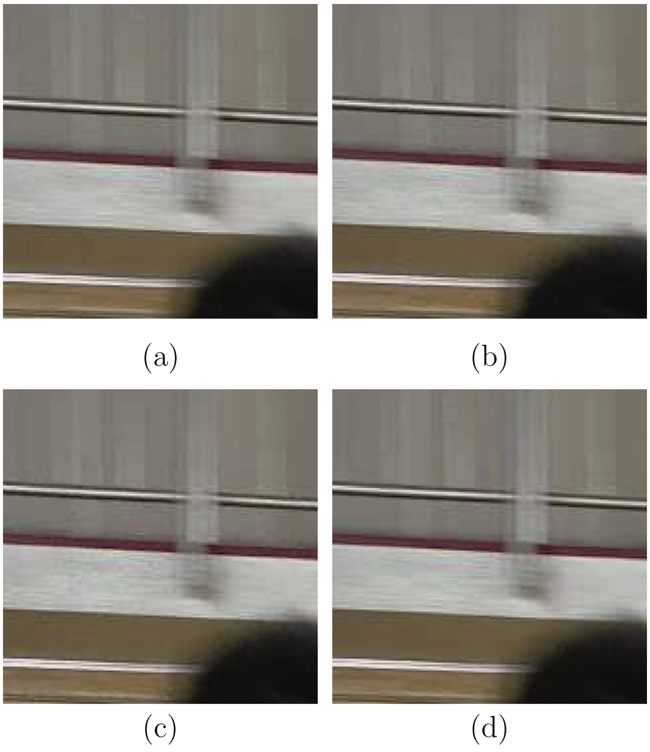


Fig. 15. Subjective performance evaluation of the reconstructed IFs of the first frame of the *BasketballDrives* sequence by: (a) Original, (b) Ma et al.'s algorithm, (c) Lin et al.'s algorithm, (d) proposed EMBED-W with QP = 22.

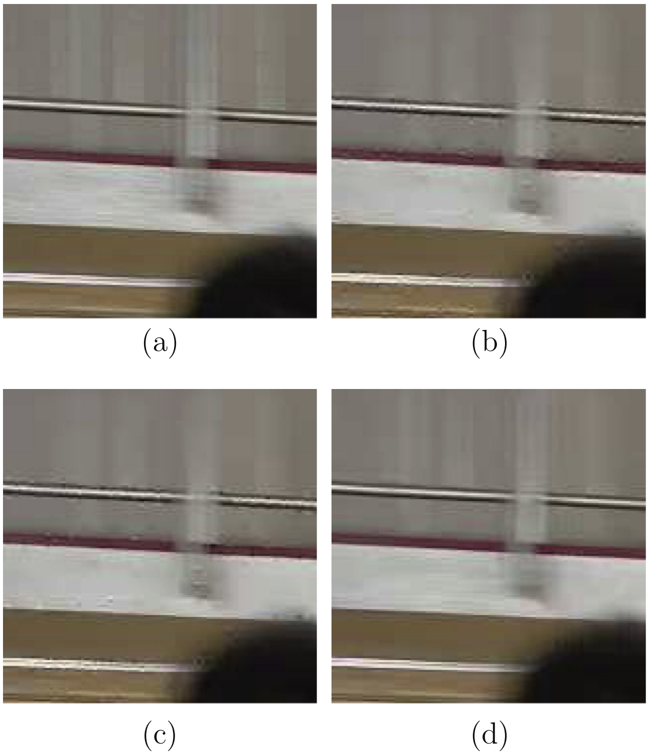


Fig. 16. Subjective performance evaluation of the reconstructed IFs of the first frame of the *BasketballDrives* sequence by: (a) Original, (b) Ma et al.'s algorithm, (c) Lin et al.'s algorithm, (d) proposed EMBED-W with QP = 32.

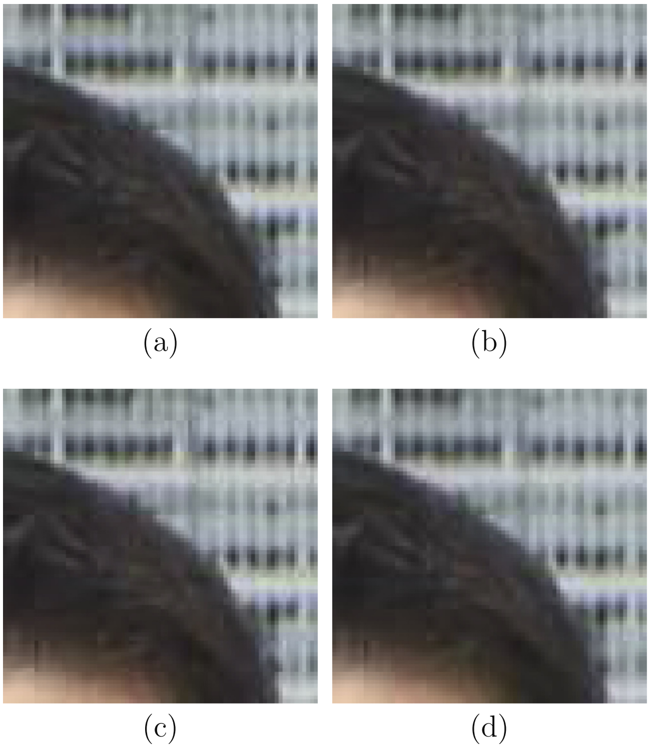


Fig. 17. Subjective performance evaluation of the reconstructed IFs of the first frame of the *Johnny* sequence by: (a) original, (b) Ma et al.'s algorithm, (c) Lin et al.'s algorithm, (d) proposed EMBED-W with QP = 22.

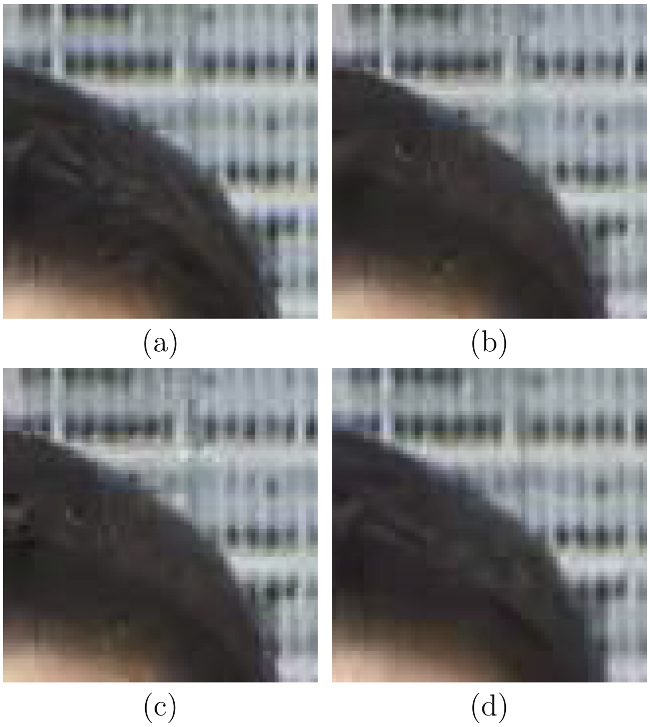


Fig. 18. Subjective performance evaluation of the reconstructed IFs of the first frame of the *Johnny* sequence by: (a) original, (b) Ma et al.'s algorithm, (c) Lin et al.'s algorithm, (d) proposed EMBED-W with QP = 32.

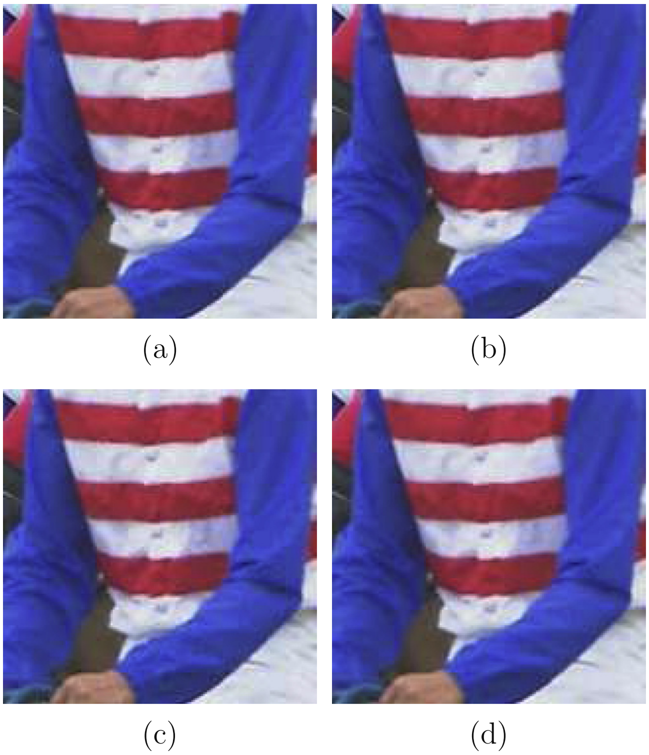


Fig. 19. Subjective performance evaluation of the reconstructed IFs of the first frame of the *RaceHorses* sequence by: (a) original, (b) Ma et al.'s algorithm, (c) Lin et al.'s algorithm, (d) proposed EMBED-W with QP = 22.

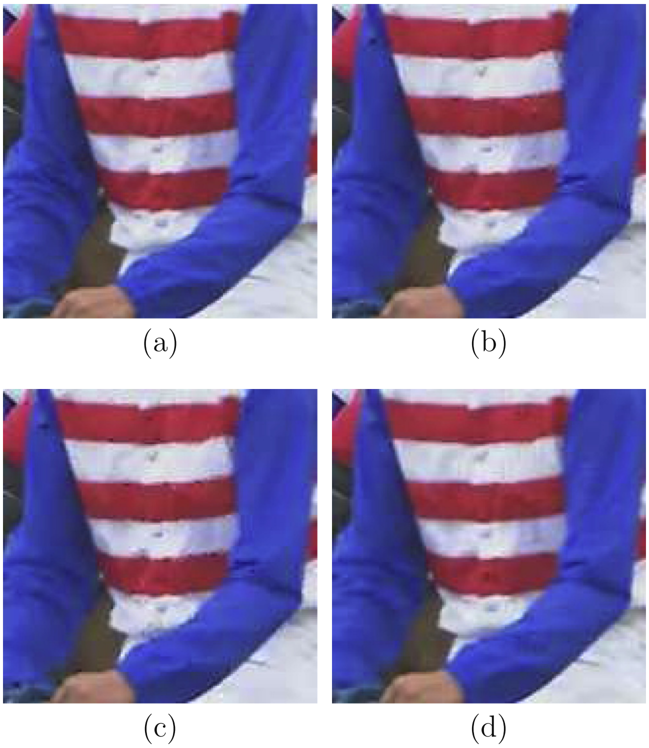


Fig. 20. Subjective performance evaluation of the reconstructed IFs of the first frame of the *RaceHorses* sequence by: (a) original, (b) Ma et al.'s algorithm, (c) Lin et al.'s algorithm, (d) proposed EMBED-W with QP = 32.

neighboring unperturbed pixels. Consequently, the cost function for $Case_1$ PUs is defined by

$$C_1(k) = \sum_{i=0}^{3} |b^k_{i,2} - b^k_{i,3}| + |b^k_{i,2} - b^k_{i,1}|.$$

The index k^* of the optimal candidate for the original unperturbed block can be determined by

$$k^* = \arg \min_{1 \leqslant k \leqslant K} C_1(k).$$

By replacing with found unperturbed block B^*_k, the embedding errors can be effectively eliminated to yield better quality of the reconstructed IFs and inter-coded frames.

For $Case_2$, $Case_3$, and $Case_5$ PUs, the quality improvement procedure can be carried out in the same way under different error patterns specified in Fig. 6 for different cases. For $Case_2$ and $Case_3$ PUs, only the third row pixels would suffer from embedding errors, as shown in Fig. 6(b), the cost function is defined as

$$C_{2,3}(k) = \sum_{j=0}^{3} |b^k_{2,j} - b^k_{3,j}| + |b^k_{2,j} - b^k_{1,j}|.$$

For $Case_5$ PUs, since only the pixel $b_{2,2}$ is subject to embedding errors, the cost function is defined as

$$C_5(k) = |b^k_{2,2} - b^k_{1,2}| + |b^k_{2,2} - b^k_{2,3}| + |b^k_{2,2} - b^k_{3,2}| + |b^k_{2,2} - b^k_{2,1}|.$$

Note that $Case_4$ PUs are excluded in the proposed quality improvement procedure. For $Case_4$ PUs, since all the sixteen quantized coefficients can be used for embedding bits, all pixels in the reconstructed block may suffer from the embedding errors. Under this condition, since no unperturbed pixels can be referenced to estimate spatial correlation and many combinational trials from the sixteen coefficients are time consuming, it is infeasible for the proposed quality improvement procedure to adopt $Case_4$ PUs.

4. Experimental results

Since the proposed algorithm is the first DCT/DST-based error propagation-free data hiding algorithm designed specifically for HEVC IFs, two previous DCT-based error propagation-free data hiding algorithms based on H.264/AVC IFs, the algorithms of Ma et al. [11] and Lin et al. [9], are performed for comparison. Performance comparisons among the proposed data hiding algorithm and the two previous ones are made in terms of embedding capacity, PSNR, and bitrate. The standard H.264/AVC and HEVC coding algorithms are implemented to evaluate the performance of embedding capacity and quality degradation for the three compared data hiding algorithms.

Ten sample videos [1] with different resolutions, as shown in Fig. 7, are used as test ones. The GOP size is 32 and the coding structure is IPP…P. The values of QP are set to be 17, 22, 27, 32, 37, and 42. The video codec platforms for H.264/AVC and HEVC are JM 16.0 and HM 10.0, respectively, whose configuration parameters are listed in Table 3. The data hiding algorithms and the two codec platforms are developed with Visual C++ 2012 on a 64-bit computer with Intel i7–3770 3.4 GHz CPU and 8 GB RAM. Detailed experimental results are available in [24].

4.1. Quality improvement scheme analysis

To evaluate the performance of the proposed quality improvement scheme mentioned in Section 3.4, our proposed data hiding algorithm is performed without or with the quality improvement scheme, in which the former is denoted as EMBED-O and the latter denoted as EMBED-W. Here we use four metrics, true positive (TP), true negative (TN), false positive (FP), and false negative (FN), as the measures of accuracy performance for the proposed quality improvement scheme and they are, respectively, defined by

$$\mathrm{TP} = \frac{N_{\mathrm{TP}}}{N_T} \times 100\%, \quad \mathrm{TN} = \frac{N_{\mathrm{TN}}}{N_T} \times 100\%,$$

$$\mathrm{FP} = \frac{N_{\mathrm{FP}}}{N_T} \times 100\%, \quad \mathrm{FN} = \frac{N_{\mathrm{FN}}}{N_T} \times 100\%,$$

where N_T denotes the number of total embedded blocks; N_{TP} denotes the number of the embedded blocks in each of which the quantized coefficients are perturbed in the proposed embedding procedure and then are correctly recovered into the original values by the proposed scheme; N_{TN} denotes the number of the embedded blocks in each of which the quantized coefficients are not perturbed in the proposed embedding procedure and then are not modified by the proposed scheme; N_{FP} denotes the number of the embedded block in each of which the quantized coefficients are not perturbed in the proposed embedding procedure and then are wrongly modified into other values by the proposed scheme; N_{FN} denotes the number of the embedded block in each of which the quantized coefficients are perturbed in the proposed embedding procedure and then are not correctly recovered into the original values by the proposed scheme.

The accuracy performance of the proposed quality improvement scheme obtained from experiments is shown in Table 4. Experiments revealed that since the distortion of $Case_5$ blocks is like impulse noise, as shown in Fig. 6(c), it can be easily detected and recovered by using the proposed quality improvement scheme that utilizes inter-pixel correlations. As shown in Table 4, the sums of TP and TN values for $Case_5$ blocks that are transform coded by different-sized TUs are from 84% to 100%, which are the highest among all Cases. Note that the $Case_1$, $Case_2$, and $Case_3$ blocks that are transform coded by non 4 × 4 TUs are not recovered in experiments since these blocks involve much high computational complexity in determining the best candidate for the original unperturbed block. As for the remaining $Case_1$, $Case_2$, and $Case_3$ blocks, their sums of TP and TN values are lower than those for $Case_5$ blocks and their FN values are quite high. The reason for that can be explained as follows. Since there are four bits embedded in one 4 × 4 DST block which is categorized into $Case_1$, $Case_2$, or $Case_3$, the distortion due to embedding is one of the total 2^4 possible embedded distortion patterns. In addition, the embedded distortion cannot be separated into individual ones for each hidden bit. Under this condition, it is difficult to find the original block for such one perturbed block, but the proposed quality improvement still achieved at least 32% of accuracy in terms of their sums of TP and TN values, when compared with $1/2^4 = 6.25\%$ of accuracy by a blind guess.

The PSNR gain contributed by the quality improvement procedure, or the improved PSNR by the EMBED-W over the EMBED-O, is shown in Table 5; the PSNR gain is from 0.70 dB to 1.23 dB, which justifies the capability of the proposed quality improvement scheme.

4.2. Performance evaluations of data hiding algorithms

In general, allocating higher bitrates for coding video often demonstrates higher PSNRs and embedding capacity. The overall data hiding performance can be evaluated objectively by plots of bitrates vs. distortion (RD) and bitrates vs. embedding capacity (RC). The RD and RC plots of concerned data hiding algorithms on the ten test videos are illustrated at the left and right, respectively, in Figs. 8–12. The proposed EMBED-W outperforms the algorithms of Ma et al. and Lin et al. in PSNR. It is expected that since more intra prediction modes and sizes of PUs and TUs are available in HEVC, higher compression performance can be achieved when compared with H.264/AVC. However, more prediction modes also imply stricter constraints on embedding hidden bits in HEVC intra-coded blocks. For example, one $N \times N$ TU with $N \in \{4, 8, 16, 32\}$ in HEVC can be embedded with N bits at most, while the same-sized residual one using 4×4 transform block (s) in H.264/AVC can be embedded with $4 \times (\frac{N}{4})^2$ bits at most.

For low bitrate video coding, the proposed EMBED-W demonstrates the best performances in both PSNRs and embedding capacity. In both HEVC and H.264/AVC, it is frequent to adopt larger blocks for IFs when QP value is larger, as shown in Table 6. The data hiding algorithms by Ma et al. and Lin et al. do not embed bits in larger blocks, such as 16×16 ones of H.264/AVC, but the proposed algorithm can embed bits in 8×8, 16×16, 32×32, and 64×64 blocks of HEVC. That is the reason why the latter can achieve higher embedding capacity than the former for low bitrate video coding.

For subjective performance evaluation, the reconstructed images with embedded bits from the videos, *PeopleOnStreet*, *BasketballDrives*, *Johnny* and *RaceHorses*, with QP = 22 and QP = 32 are demonstrated in Figs. 13–20. It is observed that the quality degradation of the marked images by Ma et al. and Lin et al. are severe and with poor preception quality. On the contrary, the marked images by the proposed EMBED-W demonstrate less visual artifact no matter what the QP value is. In summary, for low bitrate coding, the proposed EMBED-W algorithm can achieve better data hiding performance, i.e., higher embedding capacity and lower objective/subjective distortions, when compared with previous works.

5. Conclusions

To embed hidden bits in HEVC IFs without propagating errors to neighboring blocks and adjacent frames, we proposed to classify HEVC coding blocks, according to certain intra prediction mode combinations of neighboring blocks, to perform specific data hiding patterns. The signal characteristics of DCT/DST are analyzed to locate the QDCTs/QDSTs that can be perturbed without propagating errors to neighboring intra-coded blocks. In additional to the intra-frame error propagation-free data hiding algorithm, we also presented one quality improvement scheme for the reconstructed IFs to alleviate the artifact of the inter-frame error propagation. Experimental results confirmed the merits of the proposed algorithm in providing the intra-frame error propagation-free advantage, the quality improvement for marked images, the compression power inherited from HEVC, and the superiority of embedding capacity for low bitrate coding, although the embedding capacity is less than that of the previous two algorithms for H.264/AVC IFs for middle and high bitrate coding.

References

[1] F. Bossen, Common test conditions and software reference configurations, JCTVC-L1100 (2013).
[2] B. Bross, W.J. Han, J.R. Ohm, G.J. Sullivan, T. Wiegand, High efficiency video coding (HEVC) text specification draft 10, document JCTVC-L1003 (2013).
[3] H.T. Chang, C.C. Hsu, C.H. Yeh, D.F. Shen, Image authentication with tampering localization based on watermark embedding in Wavelet domain, Optical Engineering 48 (5) (2009). pp. 057002-1-057002-10.
[4] Draft ITU-T recommendation and final draft international standard of joint video specification, document ITU-T Rec. H.264/ISO/IEC 14496-10 AVC, JVT of ISO/IEC and ITU-T, 2003.
[5] X. Gong, H.M. Lu, Towards fast and robust watermarking scheme for H.264 video, in: Proceedings of IEEE International Symposium on Multimedia, 2008, pp. 649–653.
[6] W. Huo, Y. Zhu, H. Chen, A controllable error-drift elimination scheme for watermarking algorithm in H.264/AVC stream, IEEE Signal Processing Letters 18 (9) (2011) 535–538.
[7] I. Kim, J. Min, T. Lee, W. Han, J. Park, Block partitioning structure in the HEVC standard, IEEE Transactions on Circuits and Systems for Video Technology 22 (12) (2012) 1697–1706.
[8] Y. Li, H.X. Chen, Y. Zhao, A new method of data hiding based on H.264 encoded video sequences, in: Proceedings of IEEE International Conference on Signal Processing, 2010, pp. 1833–1836.
[9] T.J. Lin, K.L. Chung, P.C. Chang, Y.H. Huang, H.Y. Liao, C.Y. Fang, An improved DCT-based perturbation scheme for high capacity data hiding in H.264/AVC intra frames, Journal of Systems and Software 86 (3) (2013) 604–614.
[10] T.Y. Liu, W.H. Tsai, Quotation authentication: a new approach and efficient solutions by data hiding and cascaded hashing techniques, IEEE Transactions on Information Forensics and Security 5 (4) (2010) 945–954.
[11] X.J. Ma, Z.T. Li, H. Tu, B. Zhang, A data hiding algorithm for H.264/AVC video streams without intra-frame distortion drift, IEEE Transactions on Circuits and Systems for Video Technology 20 (10) (2010) 1320–1330.
[12] M. Noorkami, R. Mersereau, Compressed-domain video watermarking for H.264, in: Proceedings of IEEE International Conference on Image Processing, vol. 2, 2005, pp. 890–893.
[13] M. Noorkami, R.M. Mersereau, Towards robust compressed-domain video watermarking for H.264, in: Proceedings of SPIE—Security, Steganography, and Watermarking of Multimedia Contents VIII, vol. 6072, 2006, pp. 489–497.
[14] M. Noorkami, R. Mersereau, A framework for robust watermarking of H.264-encoded video with controllable detection performance, IEEE Transactions on Information Forensics and Security 2 (1) (2007) 14–23.
[15] X. Qi, X. Xin, A quantization-based semi-fragile watermarking scheme for image content authentication, Journal of Visual Communication and Image Representation 22 (2) (2011) 187–200.
[16] G. Qiu, P. Marziliano, A.T. Ho, D. He, Q. Sun, A Hybrid Watermarking Scheme for H.264/AVC Video, in: Proceedings of IEEE International Conference on Pattern Recognition, vol. 4, 2004, pp. 865–868.
[17] B. Samira, A. Hassina, H. Latifa, H.264/AVC data hiding algorithm with a limitation of the drift distortion, in: Proceedings of IEEE International Conference on Signal Processing, vol. 3, 2012, pp. 1682–1687.
[18] H.T. Sencar, M. Ramkumar, A.N. Akansu, Data Hiding Fundamentals and Applications, Elsevier Academic Press, London, UK, 2004.
[19] G.J. Sullivan, J.R. Ohm, W.J. Han, T. Wiegand, Overview of the high efficiency video coding (HEVC) standard, IEEE Transactions on Circuits and Systems for Video Technology 22 (12) (2012) 1649–1668.
[20] G. Tian, S. Goto, An optimization scheme for quadtree-structured prediction and residual encoding in HEVC, in: Proceedings of IEEE Asia Pacific Conference Circuits and Systems, 2012, pp. 547–550.
[21] K. Wong, K. Tanaka, X. Qi, Multiple messages embedding using DCT-based mod4 steganographic method, in: Proceedings of LNCS International Workshop Multimedia Content Representation, Classification, Security, 2006, pp. 57–65.
[22] J. Zhang, A.T.S. Ho, G. Qiu, Robust video watermarking of H.264/AVC, IEEE Transactions on Circuits and Systems II: Express Briefs 54 (2) (2007) 205–209.
[23] L. Zhang, Y. Zhu, L.M. Po, A novel watermarking scheme with compensation in bit-stream domain for H.264/AVC, in: Proceedings of IEEE International Conference on Acoustics Speech and Signal Processing, pp. 1758-1761, 2010.
[24] [Online]. http://140.118.175.164/PCChang/paper/HEVCDataHiding/.