

Reversible Data Hiding Scheme Based On H.264/AVC without Distortion Drift

YunXia Liu^{1,2}

¹Zhoukou Normal University, Zhoukou, China

²Department of Computer Science and Technology, Huazhong University of Science and Technology, Wuhan, China
liuyunxia0110@mail.hust.edu.cn

Zhitang Li² and Xiaojing Ma²

²Department of Computer Science and Technology, Huazhong University of Science and Technology, Wuhan, China
leeying@mail.hust.edu.cn

Abstract—There is greater possibility that reversible video data hiding scheme causes more distortions than reversible image data hiding. In H.264/AVC video streams, intra-frame distortion drift is a big problem in data hiding. This paper proposes a novel readable reversible data hiding scheme in H.264/AVC which can avert the distortion drift. We embed data into the quantized discrete cosine transform (DCT) coefficients of I frames which meet the directions of intra-frame prediction, and the directions of intra-frame prediction are utilized to avert the distortion drift. It is proved analytically and shown experimentally that the proposed algorithm can achieve high embedding capacity and low visual distortion, and the original compressed video can be recovered exactly after the hidden data, in addition, the proposed system is fast and simple. Performance comparisons with other existing schemes are provided to demonstrate the superiority of the proposed scheme.

Index Terms—reversible data hiding, H.264/AVC, intra-frame distortion drift

I. INTRODUCTION

Data hiding is a technique that embeds data into cover media contents such as audios, images and videos. There exists a drawback in many data hiding schemes that the cover media is permanently distorted because of irreversible operations such as, quantization, round off or truncation replacement, which makes the application of data hiding prohibited in some fields, e.g. in the fields of law enforcement, medical systems and perceptual transparency. It is desired to reverse the marked media back to the original cover media without any distortions after the hidden data are retrieved.

Reversible data hiding is a novel category of data hiding schemes. The earliest idea of reversible data hiding is presented by Barton [3]. After that, many reversible data hiding schemes are proposed. In [4], the

authors present a reversible data hiding scheme to validate the JPEG images. Recently, some reversible data hiding methods to embed secret data into the quantization DCT coefficients of JPEG images [5]-[8]. For compressed videos, the authors extend the scheme of [4] to MPEG-2 video [9]. Another reversible watermarking algorithm for MPEG-4 video, which is able to authenticate video contents and detect manipulation locations, is proposed in [10]. In [11] a reversible drift compensation scheme is proposed to restrain the distortion for video data hiding. The existing H.264 video reversible data hiding schemes are studied by few scholars. A thorough investigation of this problem shows only in [12][13][14]. Reversible data hiding scheme is proposed for H.264/AVC video by using error resilience technique.

In practical multimedia storage and distribution systems, images and videos are often stored and transmitted in compressed format. H.264/AVC is the latest standard for video compression with high compression efficiency. It is well-adapted for network transmission which is poised to replace the existing video coding standards. Among many new techniques, the intra prediction technique is considered as one of the most important features in the success of H.264/AVC [2]. This technique increases the dependence of the neighbouring blocks. When one frame is changed by data hiding, the reconstructed pixels of related frames will be influenced, i.e., distortion drift happens. The distortion drift problem is first discussed in data hiding for compressed videos and a drift compensation method to counteract this problem is provided by Hartung and Girod [11]. After that many data hiding studies for compressed videos have employed H&G's method to restrain distortion in stego-video [1, 3, 6, 17.]. However, in those schemes, original video cannot be obtained, thus those schemes are not applicable to reversible data hiding. We find that a common problem exists in the video data hiding schemes because one frame is changed by data hiding, and the reconstructed pixels of related frames will be influenced, i.e., distortion drift happens [10, 12,16]. Only in [15], in order to reduce the distortion of stego-video, the authors

This work is supported by the National High Technology Research and Development Program of China (863 Program) under Grant No 2007AA01Z420, and by the Key project sponsored by Natural Science Foundation of Hubei Province under Grant No. 2008CDA021)

compute the prediction error blocks and then slightly modify the error values through shifting the prediction errors based on H.264/AVC intra prediction.

Till now, there is no specific algorithm to avert distortion drift for reversible H.264/AVC data hiding. To deal with this problem, a reversible data hiding based on H.264/AVC without intra-frame distortion drift is proposed in this paper to avert intra-frame distortion drift. In order to avert the distortion of stego-video, we use the directions of intra-frame prediction to choose 4×4 blocks which meet our conditions to embed data into the quantized discrete cosine transform (DCT) coefficients of I frames.

The rest of this paper is organized as follows. Section II introduces the intra-frame prediction of H.264/AVC. Section III briefly reviews the intra-frame distortion drift and explains how to avoid it according to our paper [1]. Section IV describes our reversible data hiding algorithm in detail. Section V gives the experimental results of our algorithm. We also compare it with other current algorithms. In Section VI, we draw the conclusion.

II. BACKGROUNDS

A. Framework of Embedding Procedure Analysis

The information is embedded into the quantized luminance DCT coefficients. The algorithm only embeds data into 4 × 4 luminance blocks, because the human eye and brain (Human Visual System) are less sensitive to the brightness and 16 × 16 luminance block often varies smoothly.

The integer discrete cosine transform which is developed from the DCT is used in H.264/AVC standard. The transform based on 4 × 4 blocks is shown in (1). $Y_{4 \times 4}$ is the matrix of unscaled DCT coefficients corresponding to the residual block $R_{4 \times 4}$, \tilde{Y} is the matrix of scaled DCT coefficients. The transform between Y and \tilde{Y} can be expressed in a matrix form (1):

$$\tilde{Y} = C_f Y C_f^T \otimes E_f \quad (1)$$

Where

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

$$E_f = \begin{pmatrix} a^2 & \frac{ab}{2} & a^2 & \frac{ab}{2} \\ \frac{ab}{2} & \frac{b^2}{4} & \frac{ab}{2} & -\frac{b^2}{4} \\ a^2 & \frac{ab}{2} & -a^2 & \frac{ab}{2} \\ \frac{ab}{2} & \frac{b^2}{4} & \frac{ab}{2} & \frac{b^2}{4} \end{pmatrix}$$

The “core” part of the transform, $C_f R C_f^T$, can be carried out with integer arithmetic using only additions,

subtractions and shifts. The dynamic range of the transform operations is such that 16-bit arithmetic may be used throughout (except in the case of certain anomalous input patterns) since the inputs are in the range ± 255 . The post-scaling operation $\otimes E_f$ requires one multiplication for every coefficient which can be “absorbed” into the quantisation process. In the context of the H.264 CODEC, the approximate transform has almost identical compression performance to the DCT and has a number of important advantages.

The inverse transform is given by Equation (2). The H.264 standard [18] defines this transform explicitly as a sequence of arithmetic operations:

$$Y = C_i^T (\tilde{Y} \otimes E_i) C_i \quad (2)$$

Where

$$C_i = \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & \frac{1}{2} & -\frac{1}{2} & -1 \\ 1 & -1 & -1 & 1 \\ \frac{1}{2} & -2 & 1 & -1 \end{pmatrix}$$

$$E_f = \begin{pmatrix} a^2 & ab & a^2 & ab \\ ab & b^2 & ab & b^2 \\ a^2 & ab & a^2 & ab \\ ab & b^2 & ab & b^2 \end{pmatrix}$$

H.264 assumes a scalar quantiser. The mechanisms of the forward and inverse quantisers are complicated by the requirements to (a) avoid division and/or floating point arithmetic and (b) incorporate the post- and pre-scaling matrices E_f and E_i described above.

The basic forward quantiser operation is:

$$\tilde{Y}_{ij} = W_{ij} \text{round}\left(\frac{PF}{Q_{step}}\right) \quad (3)$$

Where

$$W = C_f R C_f^T$$

PF is $a/2$, $ab/2$ or $b^2/4$ depending on the position (i, j).

In order to simplify the arithmetic, the factor (PF/Qstep) is implemented in the reference model software as a multiplication by a factor MF and a right-shift, avoiding any division operations:

$$\tilde{Y}_{ij} = \text{round}\left(W_{ij} \frac{MF}{(2^{15+f \text{loor}(\frac{QP}{6})} Q_{step})}\right) \quad (4)$$

Where

$$\frac{MF}{2^{15+f \text{loor}(QP/6)}} = \frac{PF}{Q_{step}}$$

The pre-scaling factor for the inverse transform have a constant scaling factor of 64 to avoid rounding errors:

$$W'_{ij} = \tilde{Y}_{ij} \cdot Q_{step} \cdot PF \cdot 64 \quad (5)$$

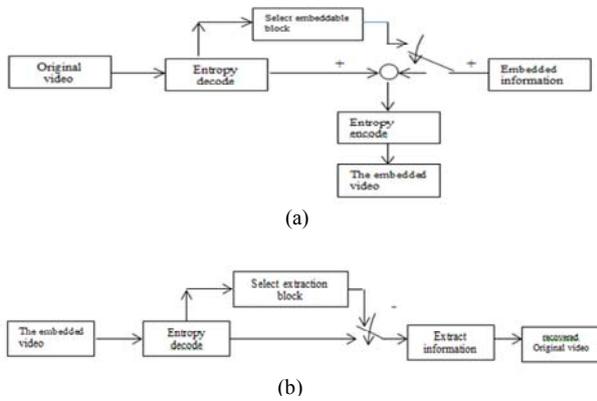


Fig.1 Proposed data hiding scheme diagram: (a) Embedding. (b) Extraction

The values at the output of the inverse transform are divided by 64 to remove the scaling factor, and then Post-scaling can be expressed in (6):

$$Y = \text{round}(C_i^T W C_i / 64) \quad (6)$$

R can be expressed in (7):

$$R = \text{round}(C_i^T W C_i / 64) \quad (7)$$

In data hiding algorithms, the information is embedded into the quantized luminance DCT coefficients as in (8),

$$\tilde{Y}' = \tilde{Y} + \Delta \quad (8)$$

where $\Delta 4 \times 4$ is the error matrix added to the quantized DCT coefficient matrix $\tilde{Y}_{4 \times 4}$ by data hiding. $\Delta = (a_{ij})_{4 \times 4}$

After the re-scaling step as depicted in (9), the inverse ICT and the post-scaling step as described in (10) of the decoder, we can get R' the residual block after embedding Δ .

$$\begin{aligned} Y' &= \tilde{Y}'_{ij} \cdot Q_{step} \cdot PF \cdot 64 \\ &= \tilde{Y}_{ij} \cdot Q_{step} \cdot PF \cdot 64 \\ &\quad + \Delta \cdot Q_{step} \cdot PF \cdot 64 \end{aligned} \quad (9)$$

$$\begin{aligned} R' &= \text{round}(C_i^T \tilde{Y}'_{ij} \cdot Q_{step} \cdot PF) \cdot C_i \\ &\quad + C_i^T \cdot (\Delta \cdot Q_{step} \cdot PF) \cdot C_i \end{aligned} \quad (10)$$

The deviation of the pixel luminance value between the original block and the one after embedding is $E_{4 \times 4}$, where $E_{4 \times 4} = (e_{ij})_{4 \times 4}$, which can be calculated according to (11).

$$\begin{aligned} E &= R' - R \\ &= \text{round}(C_i^T (\tilde{Y}_{ij} \cdot Q_{step} \cdot PF) \cdot C_i \\ &\quad + C_i^T \cdot (\Delta \cdot Q_{step} \cdot PF) \cdot C_i) \\ &\quad - \text{round}(C_i^T (\tilde{Y}_{ij} \cdot Q_{step} \cdot PF) \cdot C_i) \end{aligned} \quad (11)$$

Using B to express $C_i^T \cdot (\Delta \cdot Q_{step} \cdot PF) \cdot C_i$ we have $e_{i,j} = \text{round}(b_{i,j})$ or $\text{round}(b_{i,j}) \mp 1$.

Fig.1 depicts the scheme of our proposed algorithm.

B. H.264/AVC Intra prediction

In H.264/AVC intra prediction method, a prediction block is formed based on previously encoded adjacent blocks [2]. The sixteen elements in the 4×4 block (labeled from a to p in Fig.2) are predicted by using the boundary pixels of the upper and left blocks which are previously obtained (labeled from A to M) using a prediction formula corresponding to the selected optimal prediction mode.

There are nine prediction modes for each 4×4 block, four modes for each 16×16 luminance block. Fig.3 and Fig.4 describe these prediction modes [2]. For each 4×4 block, one of nine prediction modes can be selected by the encoder. The rate distortion optimization (RDO) technique is used to take full advantage of the mode selection regarding maximizing coding quality and minimizing data bits[17]. The RDO is applied to all of the 4×4 block intra-prediction modes to find the best one which can achieve the optimal prediction mode decision, there is more focus on developing the 4×4 intra-prediction mode decision techniques because of lower computational complexity.

M	A	B	C	D	E	F	G	H
I	a	b	c	d				
J	e	f	g	h				
K	i	j	k	l				
L	m	n	o	p				

Fig.2 Labeling of prediction samples.

C. Intra-frame Distortion Drift

In order to describe the intra-frame distortion drift, we assume that current prediction block is $B_{i,j}$, in intra-frame prediction, and each sample of $B_{i,j}$ is the sum of the predicted value and the residual value. Because the predicted value is calculated by the samples which are gray in Fig.5, the embedding in the samples induced error in blocks $B_{i-1,j-1}$, $B_{i,j-1}$, $B_{i-1,j}$, and $B_{i-1,j+1}$ would propagate to $B_{i,j}$, as illustrated in Fig.5. In brief, the embedding induced distortion of a block (especially of the samples at the bottom line and on the rightmost row) would propagate to other blocks by intra-frame prediction. This visual distortion would accumulate from the upper left to the lower right is defined as intra-frame distortion drift. For convenience, we give several definitions, the 4×4 block on the right of the current block is defined as right-block; the 4×4 block under the current block is defined as under-block; the 4×4 block on the left of the under-block is defined as under-left-block; the 4×4 block on the right of the under-block is defined as under-right-block.

D. Drift Prevention

The 4×4 block on the right of the current block is defined as right-block. The 4×4 block under the current block is defined as under-block, the 4×4 block on the left of the under-block is defined as under-left-block. The 4×4 block on the right of the under-block is defined as under-right-block [11]

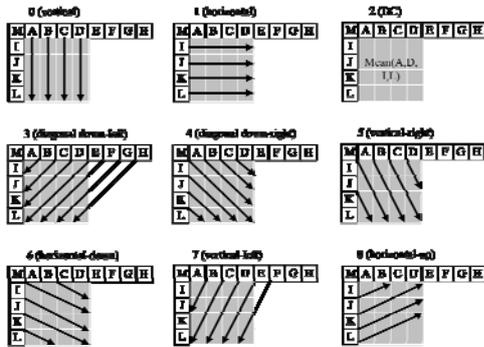


Fig.3 4 × 4 luma block prediction modes.

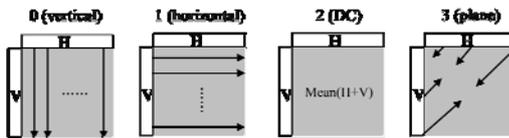


Fig.4 4 × 4 and 16x16 luma block prediction modes.

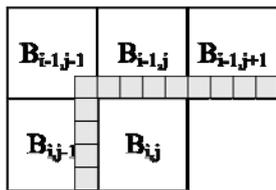


Fig.5 The prediction block $B_{i,j}$ and the adjacent encoded blocks

Condition 1: $\text{Right-mode} \in (0, 3, 7)_{4 \times 4} \cup \{0\}_{16 \times 16}$. The right-block has the prediction modes 0, 3 or 7, or it is the sub-block of the 16×16 block with prediction modes 0 as shown in Fig.6. If a macro block meets Condition 1, the pixel values of its samples on the rightmost column will not be used in the intra-frame prediction of its right-block. That is, the embedding induced distortion will not propagate to its right-block.

Condition 2: $\text{under-left-mode} \in \{0, 1, 2, 4, 5, 6, 8\}_{4 \times 4} \cup \{0, 1, 2, 3\}_{16 \times 16}$ AND $\text{under-mode} \in \{1, 8\}_{4 \times 4} \cup \{1\}_{16 \times 16}$. The under-left-block has the prediction modes 0, 1, 2, 4, 5, 6 or 8, or the under-left-block is the sub-block of the 16×16 block; and the under-block has the prediction modes 1 or 8, or the under-block is the sub-block of the 16×16 block with prediction modes 1 as shown in Fig.7. If a macro block meets Condition 2, the pixel values of its samples on the lowest row will not be used in the intra-frame prediction of its under-block and under-left-block. That is, the embedding induced distortion will not propagate to its under-block and under-left-block.

Condition 3: $\text{under-right-mode} \in \{0, 1, 2, 3, 7, 8\}_{4 \times 4} \cup \{0, 1, 2, 3\}_{16 \times 16}$. The under-right-block has the prediction modes 0, 1, 2, 3, 7 or 8, or the under-right-block is the sub-

block of the 16 × 16 block as shown in Fig.8. If a macro block meets Condition 3, the sample in the bottom right hand corner will not be used in the intra-frame prediction of its under-right-block. That is, the embedding induced distortion will not propagate to its under-right-block.

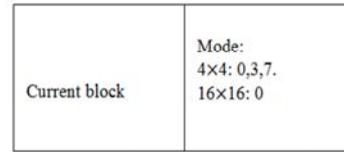


Fig.6 The current block meets condition 1

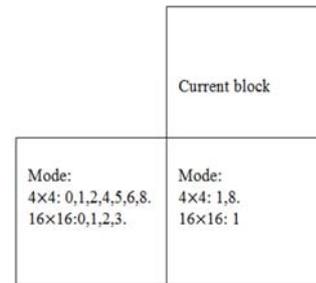


Fig.7 The current block meets condition 2

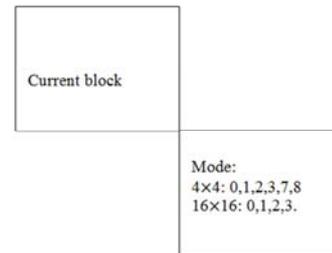


Fig.8 The current block meets condition 3

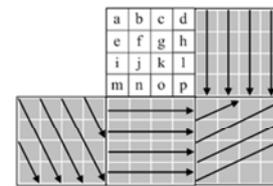


Fig.9 Example of the adjacent blocks of the chosen block meets our Conditions.

To prevent the distortion drift, the current block must meet Condition 1, Condition 2 and Condition 3 as depicted in Fig.9. Under such conditions, all the pixel values of current block will not be used in the following intra-frame prediction. Thus, the embedding distortion would not propagate to other blocks.

II. PROPOSED SCHEME

Embedding

The original video is firstly entropy decoded to get the intra-frame prediction modes and quantized DCT coefficients, then the appropriate DCT coefficients of 4 × 4 luminance DCT blocks which meets three Conditions are selected to embed data, all the quantized DCT

coefficients are then entropy encoded to get the target embedded video.

To make the embedding procedure more clear, select a positive integer N and a coefficient \tilde{Y}_{ij} ($i,j=0,1,2,3$) from each of chosen block which is exploited for data hiding as an example.

If the current block meets Condition 1, Condition 2 and Condition 3, the 4×4 luminance DCT block for embedding is selected.

If $|\tilde{Y}_{ij}|=N+1$ or $|\tilde{Y}_{ij}| \neq N$, then \tilde{Y}_{ij} is modified according to the formula(1-1);

$$\tilde{Y}_{ij} = \begin{cases} \tilde{Y}_{ij} + 1, & \text{if } \tilde{Y}_{ij} \geq 0 \text{ and } |\tilde{Y}_{ij}| > N, \\ \tilde{Y}_{ij} - 1, & \text{if } \tilde{Y}_{ij} < 0 \text{ and } |\tilde{Y}_{ij}| > N, \\ \tilde{Y}_{ij}, & \text{if } |\tilde{Y}_{ij}| < N. \end{cases} \quad (1-1)$$

$$\tilde{Y}_{ij} = \begin{cases} \tilde{Y}_{ij} + 1, & \text{if } \tilde{Y}_{ij} \geq 0, \\ \tilde{Y}_{ij} - 1, & \text{if } \tilde{Y}_{ij} < 0. \end{cases} \quad (1-2)$$

$$\tilde{Y}_{ij} = \begin{cases} \tilde{Y}_{ij} - 1, & \text{if } \tilde{Y}_{ij} \geq 0 \text{ and } |\tilde{Y}_{ij}| > N + 1, \\ \tilde{Y}_{ij} + 1, & \text{if } \tilde{Y}_{ij} < 0 \text{ and } |\tilde{Y}_{ij}| > N + 1, \\ \tilde{Y}_{ij}, & \text{if } |\tilde{Y}_{ij}| < N. \end{cases} \quad (2-1)$$

$$\tilde{Y}_{ij} = \begin{cases} \tilde{Y}_{ij} - 1, & \text{if } \tilde{Y}_{ij} \geq 0, \\ \tilde{Y}_{ij} + 1, & \text{if } \tilde{Y}_{ij} < 0. \end{cases} \quad (2-2)$$

If the embedded bit is 1 and $|\tilde{Y}_{ij}| = N$, then \tilde{Y}_{ij} is modified according to the formula (1-2);

If the embedded bit is 0 and $|\tilde{Y}_{ij}| = N$, then \tilde{Y}_{ij} is not modified.

Hidden Data Extraction

The retrieval of hidden information is simple and fast. After entropy decoding of the H.264/AVC video, we choose the embeddable blocks and extract the hidden data as follows:

If the current block meets Condition 1, Condition 2 and Condition 3, the 4×4 luminance DCT block for embedding is selected.

If $|\tilde{Y}_{ij}|=N+2$ or $|\tilde{Y}_{ij}| \neq N+2$ or $|\tilde{Y}_{ij}| \neq N+1$, then \tilde{Y}_{ij} is modified according to the formula (2-1);

If $|\tilde{Y}_{ij}|=N+1$, then the extracted bit is 1 and \tilde{Y}_{ij} is modified according to the formula (2-2).

If $|\tilde{Y}_{ij}|=N$, then the extracted bit is 0 and \tilde{Y}_{ij} is not modified.

All the quantized DCT coefficients are then entropy encoded again to get Original video.

III. EXPERIMENTAL RESULTS

The four video sequences Container, News, Mobile and Coastguard are used as test samples. The proposed algorithm has been implemented in the H.264/AVC reference software version JM16.0. The H.264 encoder uses a fixed quantization step parameter of 28 for all I-frames. The standard video sequences are encoded into 20 frames at 30frame/s. The PSNR (Peak Signal-to-Noise

Ratio) value in this paper is calculated compared to the original frames of the corresponding YUV files, and the PSNR values are the average of all the I frames of a video sequence. The embedding capacity of a video sequence is the number of bits embedded in the 20 I-frames of that sequence. In addition, the conditions of the chosen block is $|\tilde{Y}_{ij}| \geq 2$. In these experiments, the extraction of the embedded video is completely restored to the original video, and the embedded message can be accurately retrieved.

Fig.10. shows the original and embedded frames of Coastguard (frame 6) and News (frame 15). It can be seen, the reversible information hiding method can obtain a better visual concealment.

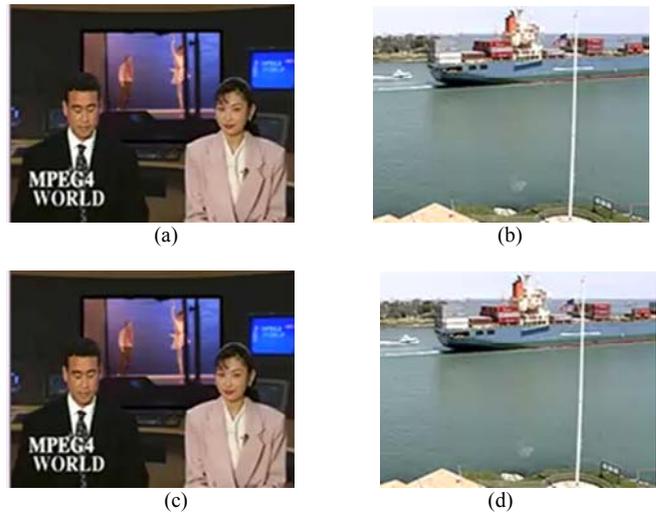


Fig.10 The original and embedded frames of News and Container (a) News and (b) Container without data hiding. (c) News and (d) Container with data hiding.

Fig.11. shows the PSNR variations of the video sequences News with different coefficient when N value changes. As shown in the figure, when the N value is the largest one, the performance of the proposed scheme is increased, whereas the embedding capacity is decreased. Data hiding into the high-frequency coefficients will get better visual quality than data hiding into the low-frequency coefficients, and the more number of modifications to the DCT coefficients of our scheme.

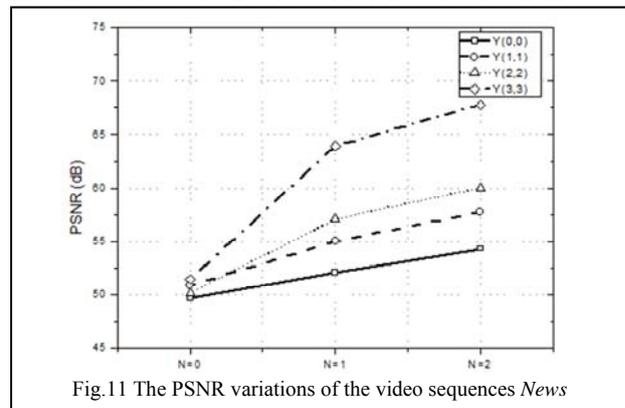


Fig.11 The PSNR variations of the video sequences News

In Table I, we compared the experimental result of original video and the embedded video ($N=0$) with

TABLE I
EMBEDDING PERFORMANCES OF OUR SCHEME WHEN N=0

sequences /DCT coefficient		$\tilde{Y}_{3,3}$	$\tilde{Y}_{2,2}$	$\tilde{Y}_{1,1}$	$\tilde{Y}_{0,0}$	More coefficients $\tilde{Y}_{3,3}, \tilde{Y}_{2,3}, \tilde{Y}_{1,3}$
Container	PSNR (dB)	51.07	48.73	49.02	48.53	45.85
	Capacity(bits)	1765	1508	1186	907	5049
News	PSNR (dB)	51.43	50.15	50.86	49.7	46.4
	Capacity(bits)	1704	1451	1261	766	4905
Mobile	PSNR (dB)	49.75	48.11	48.00	47.46	44.61
	Capacity(bits)	2306	1533	1132	754	6192
Coastguard	PSNR (dB)	49.50	47.70	46.93	44.56	44.92
	Capacity(bits)	1848	1728	1530	1089	5510

TABLE II
EMBEDDING PERFORMANCES OF OUR SCHEME USING DIFFERENT N VALUES

	N	Capacity(bits (bits))	PSNR (dB)
$\tilde{Y}_{0,0}$	0	785	52.49
	1	476	55.58
	2	201	58.13
$\tilde{Y}_{1,1}$	0	1278	51.65
	1	250	58.15
	2	113	59.29
$\tilde{Y}_{2,2}$	0	1473	50.67
	1	213	59.33
	2	50	66.58
$\tilde{Y}_{3,3}$	0	1739	52.2
	1	81	66.48
	2	1	73.5

TABLE III
EMBEDDING PERFORMANCE COMPARISON BETWEEN OUR SCHEME AND [14]

video sequences	Our scheme ($\tilde{Y}_{3,3}, \tilde{Y}_{2,3}, \tilde{Y}_{1,3}$)			Algorithm ^[14]		
	PS NR (dB)	Capacity(bits (Bits/frame))	Bit-rate increase(%)	PS NR (dB)	Capacity(bits (Bits/frame))	Bit-rate increase (%)
Container	45.85	5049	0.035	39.42	2927	0.021
News	46.4	4905	0.036	40.87	3522	0.024
Mobile	44.61	6192	0.016	38.15	6077	0.016
Coastguard	44.93	5510	0.041	40.56	2288	0.019

Data hiding into $\tilde{Y}_{3,3}, \tilde{Y}_{2,2}, \tilde{Y}_{1,1}, \tilde{Y}_{0,0}$ respectively when N = 0, the embedding capacity according to $\tilde{Y}_{3,3}, \tilde{Y}_{2,2}, \tilde{Y}_{1,1}$ and $\tilde{Y}_{0,0}$, order decreases, since these probability coefficient value of zero chance to follow the same sequence decreases. Furthermore, the PSNR values of each video sequence are less than 44dB.

Table II shows the embedding capacity variations of the video sequences News with different coefficient when N value changes. Corresponding to four different embedding DCT coefficients, embedding capacity decreases in the order of N = 0 to N = 2. As H.264/AVC excellent compression performance, mostly quantized DCT coefficient values are 0. The performance of the proposed scheme is the best when the N value is the smallest one (N = 0). The smaller the N value, the larger the amount of the embedding information. Because the probability of high frequency DCT coefficient equal to 0 is greater than the low.

Table III gives the embedding performance comparison between the proposed scheme and the algorithms in [14]. The results proved that our method often performs better than those in [14] in both embedding capacity and PSNR. But the bit-rate increase of our scheme are more significant than the algorithm in [14]. It is caused by high embedding capacity and the more number of modifications to the DCT coefficients of our scheme. And after video processing operations for an I frame the embedded bits survival rates of the sequence Container, News, Mobile, Coastguard are 88%,93.3%,94%,92.3% respectively.

IV. CONCLUSION

This paper presents a novel high-capacity reversible data hiding algorithm without Distortion Drift which is based on the directions of intra-frame prediction. The proposed data hiding procedure includes entropy decoding, data embedding and entropy encoding, and the embedded data retrieval procedure includes entropy decoding and data extraction. Both procedures are simple and fast. decoding and data extraction. Both procedures are simple and fast.

REFERENCES

- [1] M. Fallahpour and D. D. Megías, "Reversible data hiding based on h.264/avc intra prediction," in IWDW, ser. Lecture Notes in Computer Science, Y. Q. Shi, H.-J. Kim, and S. Katzenbeisser, Eds., vol. 5041. Springer Verlag, 2009, in press.
- [2] I.E.G. Richardson, H. 264 and MPEG-4 video compression: video coding for next-generation multimedia, Chichester, Wiley, 2003.
- [3] J. M. Barton, "Method and Apparatus for Embedding Authentication Information Within Digital Data," U.S. Patent 5 646 997, 1997.
- [4] Fridrich J, Goljan M, Du R (2001) Invertible authentication watermark for JPEG images. In: Proc. of the International Conference on Information Technology: Coding and Computing, pp 223 - 227
- [5] Chen CC, Kao DS (2007) DCT-based reversible image watermarking approach. In: Proc. of the third International Conference on Intelligent Information Hiding and Multimedia Signal Processing 2:489 - 492
- [6] Xuan GR, Shi YQ, Ni ZC, Chai PQ, Cui X, Tong XF (2007) Reversible data hiding for JPEG images based on histogram pairs. Lect Notes Comput Sci Image Anal Recognit 4633:715-727

- [7] .Chang CC, Lin CC, Tseng CS, Tai WL (2007) Reversible hiding in DCT-based compressed images. *Inf Sci* 177(13):2768–2786
- [8] Zeng X, Chen ZY, Chen M, Xiong Z (2010) Invertible image watermarking based on zero coefficient index. *J Comput Res Dev* 47(7) (in press)
- [9] Rui D, Fridrich J (2002) Lossless authentication of MPEG-2 video. In: *Proc. of the International Conference on Image Processing* 2:893 – 896
- [10] Chen H, Chen ZY, Zeng X, Fan W, Xiong Z (2008) A novel reversible semi-fragile watermarking algorithm of MPEG-4 video for content authentication. In: *Proc. of the Second International Symposium on Intelligent Information Technology Application* 3:37 – 41
- [11] Issues and solution on distortion drift in reversible video data hiding
- [12] Lin, S.D., Hsiang-Cheng Meng, and Yu-Lung Su. A novel error resilience using reversible data embedding in H.264/AVC. in *International Conference on Information, Communications and Signal Processing*. 2007. ICICS'07. Singapore. 2007:1 -5
- [13] A. Piva, R. Caldelli, and F. Filippini. Data hiding for error concealment in H.264/AVC. in *IEEE Workshop on Multimedia Signal Processing*. 2004. MMSP'04. Siena, Italy. 2004:199-202
- [14] W. Lie, T. C. Lin, D. Tsai et al. Error Resilient Coding Based on Reversible Data Embedding Technique for H. 264/AVC Video. in *IEEE International Conference on Multimedia and Expo*. 2005. ICME'05. Amsterdam, The Netherlands. 2005: 1174 -1177
- [15] Reversible Data Hiding Based On H.264/AVC Intra
- [16] Rui D, Fridrich J (2002) Lossless authentication of MPEG-2 video. In: *Proc. of the International Conference on Image Processing* 2:893 – 896
- [17] Sullivan, G. J. and Wiegand, T., “Rate-distortion optimization for video compression,” *IEEE Signal Process. Mag.*, vol. 15, pp. 74–90, Nov. 1998
- [18] ISO/IEC 14496-10 and ITU-T Rec. H.264, Advanced Video Coding, 2003.

YunXia Liu was born in Zhoukou, China in 1972, and received her B.E. degree from Henan Normal University, Xinxiang, China, 1996. She is currently a PhD student in the Department of Computer Science and Technology at Huazhong University of Science and Technology and an associate professor in the Mathematics and Information Science, Zhoukou Normal University. Her research interests include: cryptography, network security and multimedia security.

Zhitang Li was born in Jianli, China in 1951, and received his M.E. degree in Computer Architecture from Huazhong University of Science and Technology, Wuhan, China, 1987, and PhD degree in Computer Architecture from Huazhong University of Science and Technology, Wuhan, China, 1992. His research interests include computer architecture, network security and P2P networks.

He is currently the director of China Education and Research Network (CERNET) in Central China. He was a vice president of Department of Computer Science and Technology, Huazhong University of Science and Technology, China. He has published more than one hundred papers in the areas of computer architecture, network security and P2P networks.

Xiaojing Ma was born in Jinzhai, China in 1983, and received her B.E. degree from Wuhan University of Science and Technology, Wuhan, China, 2003. She is currently a PhD student in the Department of Computer Science and Technology at Huazhong University of Science and Technology. Her research interests include: cryptography, network security and multimedia security. She was a Software Engineer of 263 On-Line Communications Inc. She has published several papers in the area of multimedia security.