A Privacy Protection Scheme in H.264/AVC by Data Hiding

Po-Chyi Su*, Wei-Yu Chen*, Shao-Yu Shiau*, Ching-Yu Wu* and Addison Y.S. Su†
* Dept. of Computer Science and Information Engineering
National Central University, Taiwan
E-mail: pochyisu@csie.ncu.edu.tw Tel: +886-3-4227151(ext.35314)
† Research Center for Advanced Science and Technology
National Central University, Taiwan
E-mail: ncuaddison@gmail.com Tel: +886-3-4263354(ext.322)

***Abstract*—In this research, a privacy protection mechanism in H.264/AVC videos is proposed. The sensitive or private visual information in frames, which should not be viewable by the general public or regular users, will be scrambled by directly modifying or removing the related data in H.264/AVC compressed bitstreams. In order to allow the authorized users to recover the partially scrambled video frames, the methodology of information hiding is employed; that is, the correct information is embedded and transmitted along with the video bitstream. After retrieving the data, the authorized users can descramble the protected areas in frames. Experimental results show that the partial scrambling can be achieved effectively and the size of the resulting video is kept under good control.**

I. Introduction

Digital videos are ubiquitous these days. People can now watch, create and even share videos as they like to enjoy the convenience brought by the digital technology. Many digital videos can thus be widely distributed through various media. However, some video clips may contain certain sensitive information that the owners or creators may not want to show to the public. Such information may include faces of people, license plates of vehicles and trademarks, etc. How to retain the privacy in videos becomes an important issue. Scrambling the content in these "privacy" areas is the most commonly used approach. Since the privacy area may not cover the entire picture, it is expected that a partial scrambling mechanism should be used; that is, only the privacy area is scrambled while other parts of video frames, *i.e.*, "non-privacy" regions, can be viewed normally. Compared with scrambling the entire video frame, this type of partial scrambling in digital videos can protect the sensitive content but the video can still convey the necessary visual information to the public or ordinary users. However, since the scrambling is usually irreversible, the authorized user, who is allowed to watch the complete video content, may have to store or ask for an additional copy of original video. A partial scrambling scheme may thus be made more feasibility if it can enable the authorized user to directly restore the privacy regions from the partially scrambled video. In addition, videos are usually compressed to facilitate transmission and archiving. A practical content scrambling method should be coupled with common codecs to achieve the above-mentioned functions.

Some methods have been proposed for partial scrambling in digital images and videos. The basic idea is to scramble the important features of content. As the Discrete Cosine Transform (DCT) is used in almost all the image and video codecs, the modification of DCT coefficients is a commonly used method [1], [2]. For H.264/AVC videos, J. Wang *i.e.* [3] proposed a partial scrambling scheme by scrambling such parameters as motion vector difference (MVD) and trailing ones of CAVLC (Context-adaptive variable-length coding). The signs of parameters in the encoding process are also changed according to a key stream. H.J. Lee *et al.* [4] manipulated the transform coefficients and the CABAC (Context-adaptive binary arithmetic coding) initialization table, to secure the visual content of an H.264/AVC video. The algorithm also supports different levels of security or scrambling strength. More different scrambling methods for H.264/AVC video [5], [6] are also proposed. F. Dufaux *et al.* [7] proposed to use two scrambling steps for H.264/AVC, with the first one randomly inverting the signs of AC transform coefficients of blocks in the protected areas, and the second one applying the permutation on these coefficients. To discriminate between scrambled and unscrambled regions, they exploit the Flexible Macroblock Ordering (FMO) mechanism of H.264/AVC to define two slice groups composed of macroblocks corresponding to the foreground and background respectively. L. Tong *et al.* [8] proposed a restricted prediction scheme to prevent the errors caused by the scrambling in the privacy areas of H.264/AVC video frames. The mode restricted intra prediction (MRIP) and search window restricted motion estimation (SWRME) methods are also proposed, which modify the original intra-prediction and motion estimation in H.264/AVC to avoid drift errors. [9] further discussed the issues of partial scrambling with the modifications of transform coefficients, Intra Prediction Mode (IPM) and Motion Vectors (MV). In addition to scrambling the features of image/video content, the other methodology resorts to the techniques of information hiding. P. Korus *et al.* [10] proposed a privacy protection mechanism for still images, which allows a user to blur the selected areas of an image by removing the corresponding important wavelet coefficients. These details necessary for reconstruction are retained via the approach of Quantization Index Modulation.

In this research, we propose an efficient privacy protection mechanism, which is built under the framework of H.264/AVC coding process. The scheme allows the users to select the area with private information for scrambling while to maintain the visual quality of the non-privacy regions. The authorized users are able to restore the privacy regions. The approach of restricted encoding is used to avoid the scrambled areas from affecting the coding of non-privacy regions. A suitable way of modifying data is adopted to avoid significantly increasing the overall bit-rate. The methodology of data hiding is employed to recover the scrambled area. The authorized recipients can extract the embedded information to restore the privacy regions in frames, and be able to watch the entire video. The remainder of this paper is organized as follows. Sec. 2 will briefly introduce the parts of H.264/AVC coding standard that affects our design. The proposed privacy protection scheme will be presented in Sec. 3. The experimental results are shown in Sec. 4. The conclusive remarks will be given in Sec. 5.

II. The Intra/Inter Predictions of H.264/AVC

H.264/AVC provides a much better compression performance than the existing video coding standards due to its various encoding tools. As the previous video coding standards, H.264/AVC is the hybrid coding standard, which is based on the block-wise motion compensated, DCT-like transform coding. Each frame is compressed by partitioning it as one or more slices; each slice consists of macroblocks, which are blocks of 16×16 luma samples with the corresponding 8×8 chroma samples. Each macroblock will also be divided into sub-macroblock partitions for motion prediction. The prediction partitions can have seven different sizes $16 \times 16, 16 \times 8, 8 \times 16, 8 \times 8, 8 \times 4, 4 \times 8, 4 \times 4$ in inter-coded macroblocks. The large variety of partition shapes and the quarter sample compensation provide the enhanced prediction accuracy. In intra-coded slices, the 4×4 or 16×16 intra prediction will be applied based on neighboring reconstructed pixels. The 4×4 integer transform, which is an approximate DCT, is applied on the residual data. The point by point multiplication in the transform step will be combined with the quantization step and implemented by simple shifting operations to achieve the efficiency. CAVLC or CABAC will be used for the lossless coding. Next, we will discuss the parts of H.264/AVC that will affect our design.

The intra prediction in H.264/AVC is quite important for reducing the coding redundancy since a coding block is usually related to its neighbors. Four 16×16 or nine 4×4 intra prediction modes can be applied on the luma while four 8×8 prediction modes are for the chroma. Figure 1(a) shows the 4×4 intra prediction. Since the samples above and to the left (labeled as A to M) of the current block have been encoded/reconstructed previously and are available to both the encoder and decoder, nine prediction modes, including eight directions and one DC prediction can thus be calculated. The samples a, b, c, ..., p of the prediction block are calculated based on the samples A to M. There are eight prediction directions and one DC prediction mode for the 4×4 intra prediction as shown in Figure 1(b). It should be noted that if the neighboring upper or left block of the current block is not available, the number of available modes is reduced. The four 16×16 intra prediction modes are shown in Figure 2. For Mode 0 (Vertical), Mode 1 (Horizontal) and Mode 2 (DC), the samples are extrapolated as the 4×4 intra prediction. Mode 4 (Plane) makes use of the upper and left-hand samples and this mode works well in areas of smoothly-varying luminance.

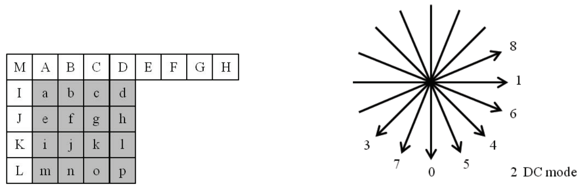

Fig. 1. (a) The labeling of prediction samples and (b) the directions of 4×4 intra prediction modes.

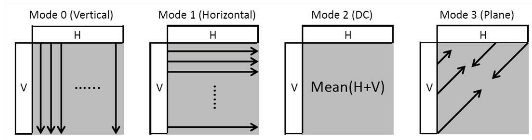

Fig. 2. Four prediction modes for 16×16 intra prediction.

The coding of the 4×4 intra prediction mode is as follows. There are nine directions of the intra 4×4 prediction so at least four bits are required to encode the selected mode correctly. Since the modes of adjacent blocks are related, the so-called "Most Probable Mode" (MPM) is determined by the intra coding modes of the upper and left blocks as shown in Figure 3, *i.e.*,

$$MPM = \min\{IPM_A, IPM_B\}, \qquad (1)$$

where $IPM_{A/B}$ represents the mode of a neighboring decoded block. If the mode matches with MPM, only one flag bit is asserted and sent. Otherwise, this flag bit will be set as "0" and three extra bits will be sent to signal which of the remaining eight modes is used.

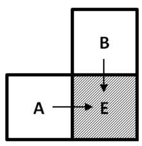

Fig. 3. The prediction of MPM.

The inter prediction provides a reference from one or more previously encoded video frames for effective encoding. H.264/AVC standard provides the variable block-size motion compensation with block sizes as large as 16×16 for flat areas, and as small as 4×4, which enables precise segmentation of moving regions. Choosing a large partition size (16×16, 16×8, 8×16) indicates that a small number of bits are required to inform the choice of motion vector and the type of partition. However, the motion compensated residuals may contain a significant amount of energy in the areas with more details. Choosing a small partition size (8×8, 8×4, 4×8, 4×4) may result in lower-energy residuals after the motion compensation but require a larger number of bits. In order to acquire the precise description of the displacements of moving areas, H.264 standard adopts the quarter-pixel precision for the motion compensation.

Because the motion vectors of neighboring partitions are usually related, it is efficient to determine the motion vector prediction (MVP) from the motion vectors of nearby/previously coded partitions by

$$MVP = \text{median}\{MV_A, MV_B, MV_C\}, \quad (2)$$

where $MV_{A/B/C}$ represents the neighboring motion vector as shown in Figure 4. The motion vector difference MVD between the current motion vector MV and MVP is calculated, *i.e.*,

$$MVD = MV - MVP. \quad (3)$$

At the decoder, MVP is generated in the same way as the encoder and added to the decoded MVD to form MV.

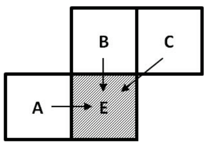

Fig. 4. The motion vector prediction.

III. The Proposed Method

The basic strategy of the proposed scheme is to remove certain data from the H.264/AVC bitstream such that the associated areas will not be decoded correctly to achieve the function of scrambling. In fact, the intra prediction modes and the information of motion vectors will be modified. Besides, since the scrambling process will usually affect the coding performance, the proposed scheme ensures that such modifications should reduce the size of data. In addition, due to the spatial and temporal data dependency in videos, the partial scrambling in H.264 video coding will cause the drift errors, which affect the quality of non-privacy regions. In order to prevent the drift errors by the partial scrambling, we adopt a restricted H.264/AVC video coding by limiting the intra prediction and inter prediction during H.264/AVC video encoding. Finally, a data hiding scheme, F4, which operates on the nonzero quantization indices in H.264/AVC video encoding, will be employed to carry the "removed" data for the recovery by the authorized users. The block diagram of the proposed privacy protection scheme for H.264/AVC video is shown in Figure 5.

In the proposed scheme, the intra prediction modes are modified to achieve the effect of scrambling in the privacy regions. We choose to modify the 4×4 intra prediction

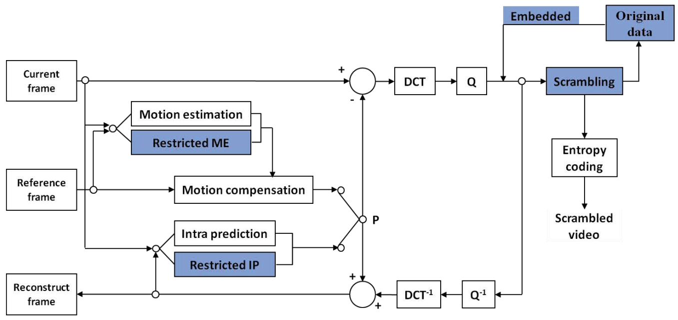

Fig. 5. The overview of the proposed scheme.

modes only. The main reason is that the blocks coded with 4×4 intra prediction modes usually contain details or textures. Scrambling these blocks will have better effects. In addition, maintaining the size of video data is important. Modifying too many data will affect the coding efficiency considerably. The proposed method is simple; all of the 4×4 intra prediction modes will be changed to MPM so only one flag bit is needed for these blocks. The objective of such a strategy is to decrease the size of the scrambled video. The resultant image of using such strategy is shown in Figure 6, in which the privacy area cannot be viewed clearly. The original values will serve as the confidential information and be embedded in the bit-stream by the information hiding.

Fig. 6. The scrambled frame by modifying the intra modes.

MVD values of Eq. (3) associated to the privacy area will also be modified to zero such that the motion vector will be decoded to the wrong value at the privacy area in the decoder. In the proposed scheme, only the blocks with larger partition, *i.e.*, 16×16, 16×8 and 8×16, are modified. Again, the motivation here is to reduce the data to be modified since the correct values of these data will be transmitted via the data hiding. Besides, modifying the motion vectors of larger blocks can scramble the content better than using the motion vectors of smaller blocks. The resultant frames of modifying all the motion vectors and only the motion vectors of larger partition are shown in Figure 7 and Figure 8 respectively. We can observe that the PSNR values of privacy areas are both very low. In addition, to reduce the data volume of confidential information, we only select one component of MVD and

Fig. 7. The scrambled frame by modifying all MVD and the average PSNR of privacy region is 15.24 dB.

Fig. 8. The scrambled frame by modifying the MVD in 16×16, 16×8 and 8×16 partitions. The average PSNR of privacy region is 16.23 dB.

modify its value to zero. That is,

$$\max\{MVD_x, MVD_y\} = 0, \tag{4}$$

where MVD_x/MVD_y is the component of MVD. This way helps to further reduce the data to be embedded and the privacy area cannot be viewed clearly either. Figure 9 shows an example of the resultant frame. It should be noted that the received MVD_x and MVD_y may both be zero. In such a case, the encoder has to inform which one is modified. In addition, if the original value is zero, then $MVD = 0$ will be sent again.

Because of the spatial and temporal dependency in digital videos, if we just change the data for the partial scrambling, the neighboring region will also be affected by the drift errors. The basic strategy of preventing the drift errors for the intra prediction is to avoid referring the data of the scrambled blocks when encoding the non-privacy region. The area around the privacy region is divided into the parts as shown in Figure 10 and we restrict the use of certain intra modes for different parts. Table I shows the available cases of intra modes in different parts for the 4×4 intra prediction and the 16×16 intra prediction. For example, the modes using the upper-right blocks are forbidden in L and the modes using the upper or upper-right blocks are forbidden in B.

Fig. 9. The scrambled frame by disturbing one component of MVD in 16×16, 16×8 and 8×16 block partitions. The average PSNR of privacy region is 17.06 dB.

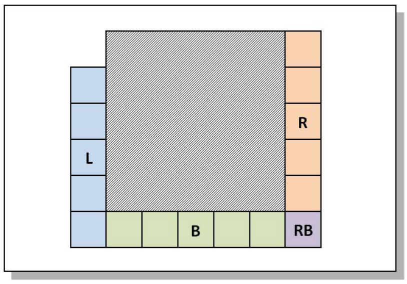

Fig. 10. The partition of the area around the privacy region.

TABLE I
THE AVAILABLE CASES OF INTRA MODES IN DIFFERENT PARTS.

Set	4×4 block	16×16 block
L	Vertical, horizontal, DC, diagonal down-right, vertical-right, horizontal-down, horizontal-up	Vertical, horizontal, DC, plane
B	Horizontal, horizontal-up	Horizontal
R	Vertical, diagonal down-left, vertical-left	Vertical
RB	Vertical, horizontal, DC, diagonal down-left, vertical-left, horizontal-up	Vertical, horizontal, DC, plane

In the inter coding, we limit the search window of motion estimation so that the range of non-privacy block cannot overlap the privacy region. This way avoids the current block from producing a motion vector pointing to the area in the privacy block. It is worth noting that the inter coding may use the skip mode, in which the motion vector is predicted from the neighboring blocks' motion vectors by using the motion vector prediction. Because the motion vector of skip mode is predicted from the neighboring blocks, we allow the use of skip mode in the region L, B and R, only when the motion vector of skip mode does not point to the scrambled block.

It should be noted that both the authorized and regular users should be able to decode the non-privacy regions correctly. However, the use of MPM in the intra coding and MVD in the inter coding may introduce the additional drift errors since they may refer to the scrambled areas. In the proposed scheme, the subsequent blocks (after the scrambled portion) will use the scrambled modes for predicting MPM and MVD, instead of the correct values. This strategy indicates that the authorized user should record the modes of scrambled content, although he or she can descramble the region, for determining the correct MPM and MVD for the subsequent blocks.

IV. The Data Hiding Scheme

The methodology of data hiding is employed to transmit the information to the authorized decoder for restoring the scrambled video. The confidential data include the correct 4×4 intra prediction mode for intra-coded blocks and the motion vector differences of larger partitions for inter-coded blocks. In addition, the coordinate values of privacy regions have to be embedded in the video too. Because the privacy region of each frame may be different, the coordinates of the entire video are first embedded at the beginning of the video. The embedding of the confidential information starts after the encoding of the associated privacy region is finished. The authorized decoder will extract the coordinates of privacy regions for all the frames first and then proceed to extract the data to restore the scrambled regions.

The confidential information is embedded into the quantized coefficients during the video encoding. Existing methods of information hiding in the residuals include JStego, F4 and F5 algorithm [11]. We adopt F4 in our scheme. Algorithm 1 shows the pseudo code of F4. The first loop takes all of the AC values in a block as shown in line 1 of Algorithm 1. For each non-zero AC coefficients coe, if it is a positive number and its LSB (Least Significant Bit) is not equal with $nextBitToEmbed$ (the next bit to be embedded), its absolute value is decreased, as shown in line 3. On the other hand, for a negative coefficient and its LSB is equal to $BitToEmbed$, the modification has to be applied and the coefficient is added by 1. The LSB of the negative number is thus equal to the inverted target bit. After the embedding operation, it is required to check whether the index becomes 0. If yes, this bit will be skipped by the decoder so it has to be embedded once again.

Algorithm 1 : F4 Algorithm

Input: $nextBitToEmbed \in \{0, 1\}$
1: **for all** Quantized AC coefficients coe in a block **do**
2: **if** $coe > 0 \wedge LSB(coe) \neq nextBitToEmbed$ **then** /* positive number */
3: $coe \leftarrow coe - 1$
4: **else if** $coe < 0 \wedge LSB(coe) = nextBitToEmbed$ **then** /* negative */
5: $coe \leftarrow coe + 1$
6: **else** /* skip zero value */
7: continue;
8: **end if**
9: **if** $coe \neq 0$ **then** /* successfully embedded */
10: /* get next bit to embed here */
11: $nextBitToEmbed$ = embeddedData.readBit();
12: **end if**
13: **end for**

The reason of choosing F4 is as follows. It has been reported that we can reveal the existence of hidden information by checking the statistics of samples in JStego and F3 since they may change the histogram of coefficients after the information embedding. Besides, the original/natural message induced by unchanged carrier media may have more steganographic ones than zeros due to the appearance of ± 1 so we have to keep this situation. In F4, if we require a higher degree of safety, some coefficients may be skipped for embedding given that both the embedder and detector know the rule. The information hiding by F4 has a positive side-effect in video coding as the magnitudes of the resultant coefficients tend to become smaller. When using a fixed QP to encode a video, the video size may even be reduced after the information embedding, and this may offset the negative effects from the information hiding on the intra and inter predictions.

V. Experimental Results

The performance of the proposed scheme in terms of intelligibility and coding efficiency is evaluated. Standard test sequences including Foreman, Hall Monitor, News, Coastguard and Silent, all in CIF format, are utilized to verify the advantages of the proposed method. The scheme is implemented in JM 18.0.

A. Intelligibility

We select different privacy regions for different tested videos to implement the partial scrambling. Table II shows the size of selected privacy area. We scramble the eyes of Foreman, the face of anchorman in News, the screen of News, the hull of Coastguard, the eyes of person in Silent, and the person of Hall Monitor. Figure 11 shows the experimental results of test sequences and the comparison with [9]. From these figures, we can see that our method can effectively scramble the privacy regions. Table III shows the comparison of the proposed scheme and [9] by checking the PSNR of Foreman after the partial scrambling. Although the PSNR of privacy area in the proposed scheme is slightly higher, the intelligibility for human is concealed as shown in Figure 11. The PSNR after the information embedding in the proposed scheme decreases a bit in the non-privacy regions. Table IV shows the PSNR values of the proposed scheme on the tested video sequences. It should be noted that the amount of private information is the major factor to affect the quality of frames.

TABLE II
The sizes of privacy regions in the tested video sequences.

Sequence	Privacy Area
Foreman	160×64
News1	64×80
News2	80×80
Coastguard	176×64
Silent	80×48
Hall Monitor	64×128

Table VI shows the comparison of PSNR values for the recovered video Foreman with the original H.264/AVC video coding and [9]. The results demonstrate that the proposed scheme can restore the privacy regions well. The PSNR of recovered data is almost equal to that of the original H.264/AVC

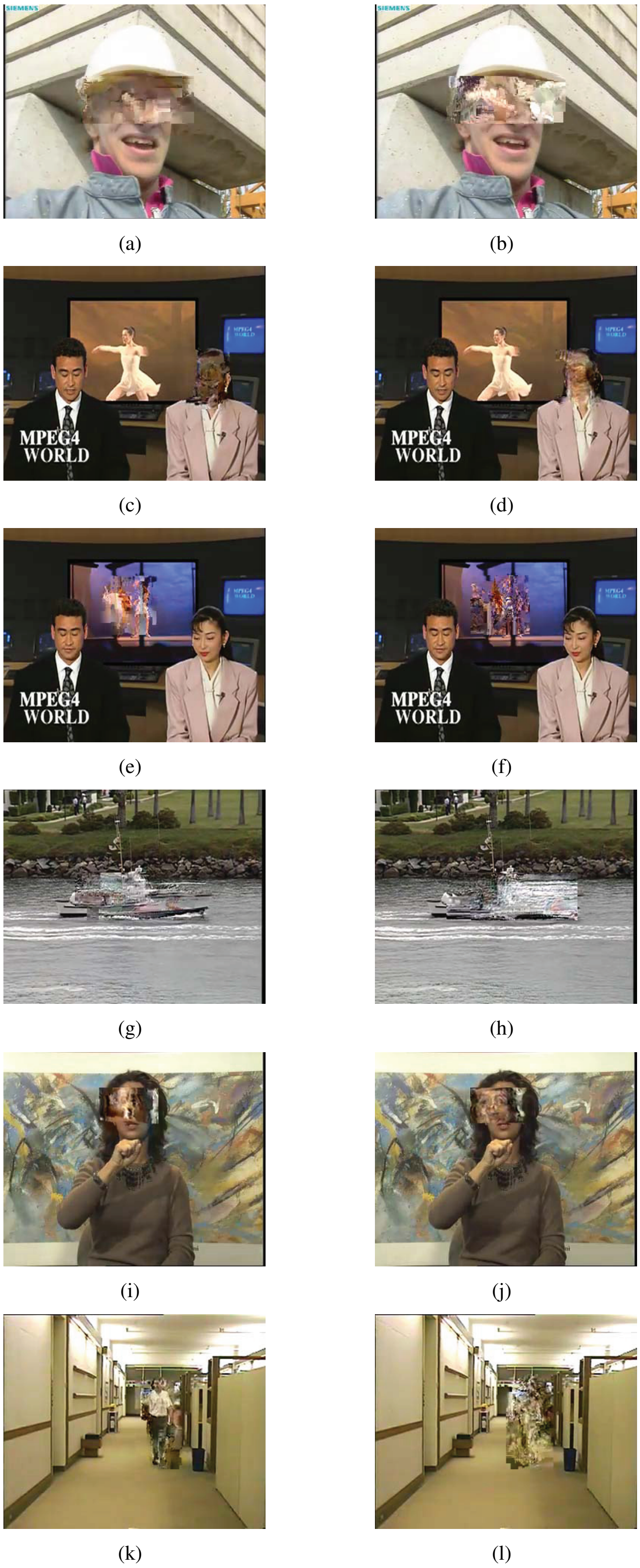

Fig. 11. (a)(c)(e)(g)(i)(k) are the results of partial scrambling in our scheme, and (b)(d)(f)(h)(j)(l) are those in [9].

TABLE III
THE PSNR(DB) OF SCRAMBLED FOREMAN IN THE PRIVACY AREA (PA) AND NON-PRIVACY AREA (NA).

Foreman	L. Tong [9]	Proposed (before embedding)	Proposed (after embedding)
PA	13.01	17.58	17.57
NA	36.42	36.47	35.18

TABLE IV
THE PSNR(DB) OF THE SCRAMBLED VIDEO SEQUENCES IN PRIVACY AREAS (PA) AND NON-PRIVACY AREAS (NA).

Sequence	L. Tong [9]		Proposed	
	PA	NA	PA	NA
Foreman	13.01	36.42	17.57	35.18
News1	15.39	38.31	13.81	37.75
News2	15.67	38.31	16.28	37.36
Coastguard	11.72	35.18	15.68	34.43
Silent	14.94	35.92	14.29	35.54
Hall Monitor	17.81	37.92	14.89	37.38
Average	**14.76**	**37.01**	**15.43**	**36.27**

encoding. The PSNR of non-privacy area is relatively lower by 1 dB due to the information embedding. Table VII shows the average PSNR of recovered videos which contain five different video sequences and six different privacy areas.

B. *Coding Efficiency*

In the proposed scheme, the data volume of compressed data is lower than those in other methods because we use the restricted prediction and modified references, instead of employing group slices for avoiding the drift errors. In addition, F4 algorithm further reduces the video size. Therefore, if there are more data to be embedded, the video size may become even smaller. Table VIII shows the compressed file size for tested video sequences. We can see that our scheme performs much better. The average increased size of the proposed scheme is 11.05%. After the embedding of the private information by F4 algorithm, the average size increase

TABLE V
THE NUMBER OF EMBEDDED BITS IN DIFFERENT VIDEO SEQUENCES.

Sequence	Number of embedded bits
Foreman	99454
News1	32695
News2	49810
Coastguard	79710
Silent	37177
Hall Monitor	34970

TABLE VI
THE PSNR(DB) OF RECOVERED FOREMAN, COMPARED WITH H.264/AVC AND [9].

Foreman	H.264/AVC	L. Tong [9]	Proposed
PA	37.61	37.61	37.60
NA	36.68	36.53	35.29

TABLE VIII
THE BIT RATES AND FILE SIZE INCREASE OF TESTED VIDEO SEQUENCES.

Sequence	H.264 /AVC	L. Tong [9]		Proposed (without F4)		Proposed (with F4)	
	Bit rate (kb/s)	Bit rate (kb/s)	Size (%)	Bit rate (kb/s)	Size (%)	Bit rate (kb/s)	Size (%)
Foreman	493.18	579.16	17.43%	540.49	9.59%	439.27	-10.93%
News1	214.52	236.42	10.21%	231.18	7.77%	212.44	-0.97%
News2	214.52	252.23	17.58%	245.01	14.21%	229.56	7.01%
Coastguard	1078.35	1221.17	13.24%	1188.37	10.20%	1146.73	6.34%
Silent	257.2	299.68	16.52%	289.63	12.61%	260.11	1.13%
Hall Monitor	296.76	345.9	16.56%	342.07	15.27%	308.64	4.00%
Average			**14.88%**		**11.05%**		**1.65%**

TABLE VII
THE AVERAGE PSNR(DB) OF RECOVERED FRAMES IN THE TESTED VIDEO SEQUENCES.

Sequence	H.264/AVC	L. Tong [9]	Proposed
Foreman	36.76	36.62	35.46
News1	38.43	38.17	37.70
News2	38.43	38.26	37.38
Coastguard	35.37	35.22	34.64
Silent	36.18	35.96	35.60
Hall Monitor	38.02	37.78	37.29
Average	**37.20**	**37.00**	**36.35**

is only 1.65%.

VI. CONCLUSION

We propose a selective privacy region protection mechanism by the partial scrambling and information hiding in the H.264/AVC video coding. The scrambled video can be restored to the original encoded video by the authorized users. This makes the scheme more flexible in the applications requiring privacy protection. The experimental results show that the proposed scheme can effectively scramble the privacy region and achieve good coding efficiency because the bit rate of the scrambled video is not affected significantly. Future work will be further enhancing the security of the scheme and allowing multiple privacy areas.

ACKNOWLEDGMENT

This research is supported by the National Science Council in Taiwan, under Grant NSC101-2221-E-008-121-.

REFERENCES

[1] S. Lian, J. Sun, and Z. Wang, "A novel image encryption scheme based-on JPEG encoding," in *Information Visualisation, 2004. IV 2004. Proceedings. Eighth International Conference on*. IEEE, 2004, pp. 217–220.

[2] L. Krikor, S. Bab, T. Ari, and Z. Shaaban, "Image encryption using DCT and stream cipher," *European Journal of Scientific Research*, vol. 32, no. 1, pp. 47–57, 2009.

[3] J. Wang, Y. Fan, T. Ikenaga, and S. Goto, "A partial scramble scheme for H.264 video," in *ASIC, 2007. ASICON'07. 7th International Conference on*. IEEE, 2007, pp. 802–805.

[4] H. Lee and J. Nam, "Low complexity controllable scrambler/descrambler for H.264/AVC in compressed domain," in *Proceedings of the 14th annual ACM international conference on Multimedia*. ACM, 2006, pp. 93–96.

[5] J. Ahn, H. Shim, B. Jeon, and I. Choi, "Digital video scrambling method using intra prediction mode," *Advances in Multimedia Information Processing-PCM 2004*, pp. 386–393, 2005.

[6] S. Lian, Z. Liu, Z. Ren, and H. Wang, "Secure advanced video coding based on selective encryption algorithms," *Consumer Electronics, IEEE Transactions on*, vol. 52, no. 2, pp. 621–629, 2006.

[7] F. Dufaux and T. Ebrahimi, "H.264/AVC video scrambling for privacy protection," in *Image Processing, 2008. ICIP 2008. 15th IEEE International Conference on*. IEEE, 2008, pp. 1688–1691.

[8] L. Tong, F. Dai, Y. Zhang, and J. Li, "Prediction restricted H.264/AVC video scrambling for privacy protection," *Electronics letters*, vol. 46, no. 1, pp. 47–49, 2010.

[9] ——, "Restricted h. 264/avc video coding for privacy region scrambling," in *Image Processing (ICIP), 2010 17th IEEE International Conference on*. IEEE, 2010, pp. 2089–2092.

[10] P. Korus, W. Szmuc, and A. Dziech, "A scheme for censorship of sensitive image content with high-quality reconstruction ability," in *Multimedia and Expo (ICME), 2010 IEEE International Conference on*. IEEE, 2010, pp. 1073–1078.

[11] A. Westfeld, "F5 - a steganographic algorithm," in *Information Hiding*. Springer, 2001, pp. 289–302.