2018 15th International ISC (Iranian Society of Cryptology) Conference on Information Security and Cryptology (ISCISC) August 28-29, 2018; Shahid Rajaee Teacher Training University - Tehran, Iran

Compressed Video Watermarking for Authentication and Reconstruction of the Audio part

Zahra Esmaeilbeig
Department of Electrical Engineering
Sharif University of Technology
Tehran, Iran
Email: esmaeilbayg_zahra@ee.sharif.edu

Shahrokh Ghaemmaghami
Department of Electrical Engineering
and Electronics Research Institute (ERI)
Sharif University of Technology
Tehran, Iran
Email: ghaemmag@sharif.edu

***Abstract*—This paper is concerned with designing a digital video watermarking system capable of authenticating and reconstructing the audio part of the video after possible attacks. As recompression is the most common attack on videos, we attempt to improve robustness of one of the recently presented and successful compressed video watermarking schemes against recompression. A comprehensive set of experiments are conducted to show that our watermarking scheme is robust against recompression attack and enables reconstruction of audio part with an acceptable quality based on PESQ score. To the best of our knowledge, this is the first work that uses the visual part of a video as a watermarking cover signal for tampering detection and reconstruction of the audio part. We specifically address videos with important audio content, e.g. news, reports, etc.**

***Index Terms*—Digital watermarking, video compression, robust watermarking, audio authentication, video forensics**

I. Introduction

Defined as the art of embedding a signal called a "watermark" within another signal called a "cover" or "host" signal, watermarking was initially introduced for copyright protection and later deployed as an effective technique for authentication and tampering detection in multimedia content. In recent years, watermarking, which can take various multimedia as cover, has been developed for different applications. For instance, detecting malicious manipulations in video content using watermarking has been addressed in [1], [2]. Although detecting visual tampering in video has been widely addressed in literature, audio part of the video has not been considered for protection against tampering. This is while the attacker who manipulates the visual content of video, needs to modify the audio part as well in order to maintain the compatibility between the audio and visual contents. Furthermore, in some special applications of Digital Video Broadcasting (DVB) like news videos, the audio part of the video can contain more important information in comparison with the visual part. Therefore, it is crucial to develop self-embedding systems that are capable of authenticating and reconstructing the audio part. It is worth noting that such self-embedding systems also have the potential to be used for secure audio transmission.

In order to fulfill the first aim of this paper which is authentication, hash bits generated from the audio part of a video, which we assume to be a speech signal, is embedded into the video. Next, a compressed version of the speech signal generated by a speech codec is embedded into the video for the aim of reconstruction. Therefore, the visual part of the video is the cover signal and the hash bits or compressed speech forms the watermark. Although the audio signal can act as a cover signal, in this paper we have chosen the visual part as the cover so that we achieve a higher embedding capacity. Since video signals are usually stored and distributed in a compressed format, embedding the watermark in uncompressed or raw video can easily cause the watermark to be destroyed during compression. An alternative is to embed the watermark in the compressed domain. Two of the latest video compression standards are H.264, also known as MPEG-4 Part 10, and H.265, also known as MPEG-H Part 2. H.265 promises doubling the data compression ratio at the same level of video quality, or substantially improved video quality at the same bit rate compared to H.264. However, the improved quality and reduced bandwidth of H.265 come at a cost. In fact, encoding and decoding the same video sequence using H.265 standard requires much more computing power compared to that of H.264.

Both codecs H.264 and H.265 use temporal and spatial redundancies to compress each frame of the video. Based on the type of the blocks of each frame to be compressed, temporal (inter-) prediction or spatial (intra-) prediction is used to reduce the redundancies. Afterwards, the difference between the original block and predicted

978-1-5386-7582-3/18/$31.00

one, which is called the residual signal, is DCT transformed and quantized to be passed to the entropy encoder. Entropy encoder reduces the statistical redundancies of data and prepares the compressed video bit stream for transmission [3]. In H.264 standard, each 16×16-pixel region of frames, known as a macroblock, is coded as a unit. However, H.265 expands coding units to sizes up to 64×64, called Coding Tree Units (CTU).

For implementation of the watermarking scheme presented in this paper, we have chosen H.264 because of its rich literature in various applications. However, the presented results will be similar in other modern DCT-based video codecs including H.265.

Inspired by the success of [1] and [4], in the context of H.264 watermarking, we propose a robust watermarking scheme which forms the foundation of our self-embedding system. In the first place, we apply the proposed spatial analysis in [4] on 4×4 blocks of quantized DCT coefficients to address the issue of robustness against recompression. Next, we use this spatial analysis to propose a robust watermarking method against recompression. Our method embeds one bit in each 4×4 block selected through the spatial analysis. An embedding function based on LSB matching steganography is used to modulate the LSB of the last nonzero level (LNZ) in selected blocks. Our watermarking scheme is very similar to the one proposed in [1], but, has two basic modifications made to improve robustness:

1) 4×4 blocks are discriminated based on a spatial analysis. Whilst in [1], for the sake of transparency, blocks having LNZ levels in higher positions are selected for embedding, our scheme selects blocks that are more robust against recompression.
2) The scheme presented in [1] computes the sum of all levels within a block and modifies the LSB of the last nonzero (LNZ) level based on the summation and the watermark bit. However, in our scheme embedding and detecting watermark from each 4×4 block is independent of the summation of all levels within the block. This modification contributes in improving robustness since summation of the levels is an unstable parameter and can change easily even when the LNZ level carrying the watermark is unchanged.

Subsequently, the watermarking scheme is exploited for tampering detection and reconstruction of the audio part of the video. The former is realized with the aid of the hash data generated from the audio part. To accomplish the latter we have exploited DPCM speech codec to form a compressed version of the audio part which is the watermark payload. The idea of embedding the hash information for tampering detection and source coded speech for recovery has been presented in [5] for speech signals.

The rest of this paper is organized as follows. In section II, we propose a watermarking scheme which is robust against recompression of video. Section III presents a method for detecting tampered frames of speech and section IV is devoted to the presented audio part reconstruction scheme. Experimental results are given in section V, which is followed by conclusions and future work addressed in section VI.

II. Compressed video watermarking

Compressed domain video watermarking refers to the methods in which the process of watermarking and compression are performed jointly. A generic block diagram of these methods is drawn in Fig. 1. Watermark is embedded whilst encoding the video and extracted at the time of decoding the video.

Compressed video watermarking methods presented in literature can be classified into two main categories : 1) Embedding in the quantized DCT coefficients; 2) Embedding in the Motion vectors. References [1], [6] and [7] embed a watermark in quantized AC coefficients of I frames. These methods are generally more robust against common video processing operations such as recompression, brightness increase and therefore are able to distinguish malicious tamperings from non-malicious ones. Methods presented in [8] and [9] embed a fragile watermark in motion vectors that can be easily removed by manipulation. Fragility of these methods is due to the inability of the watermarking method to preserve optimality of motion vectors. The video encoder searches for the optimal motion vector for encoding the B and P frames. Watermarking methods of this category modify the optimal motion vector chosen for a block to embed the watermark. During a one time recompression of the video, the motion vector is reverted to the optimal state and the watermark is removed. Thus these watermarking techniques are mainly fragile and suitable only for authentication applications.

The compressed video carrying the watermark encounters many possible tampering attacks. Attacker decompresses and exerts the manipulation on video and then recompresses the video. Therefore, recompression is the integral part in any video tampering process. This fact reveals the importance of robustness against recompression in a watermarking scheme.

Quantization and other lossy processes executed during recompression cause the recompressed version of a video sequence to be different from the original one. Hence, the quantized DCT coefficients of the residuals carrying the watermark payload are altered during recompression and

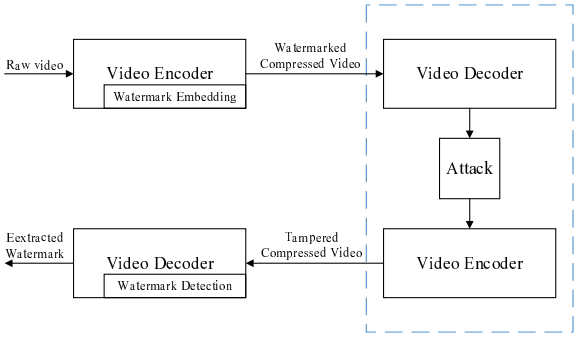


Fig. 1: Block Diagram of watermark embedding, attack, and watermark detection

watermark bits will be extracted with error. Since in this work the watermark is embedded in 4×4 blocks of I frames, three phenomena can alter the residual signal: 1) macroblock type change from I4 to I16 and vice versa, 2) prediction mode changes between nine I4 modes, and 3) prediction mode changes between four I16 modes. Needless to mention that the second and third one refer to the case when the macroblock type has retained the previous state and only the prediction mode is changed. According to [4], in blocks with higher spatial activity, these three phenomena are less likely to happen. Number of nonzero levels (NNZ) in a DCT block of residuals is a good measure for spatial activity of that block. We have conducted an experiment on 30 I frames of three standard video sequences Foreman, Mobile and Stefan from [10]. In this experiment, video sequences are compressed and recompressed with $QP = 24$ using the H.264 codec. The occurrence probability of each of the aforesaid phenomena given the NNZ value of the block is measured. As depicted in Fig. 2, the probability of macroblock type change and intramode change decrease when the NNZ value increases. Selection of blocks with NNZ values, higher than a threshold, yields lower rate of macroblock type changes and intramode changes, and therefore enhances robustness against recompression. This spatial analysis lays the foundation of our watermarking scheme.

A. Watermark Embedding

The watermark payload is embedded in the luminance component of the I frames. Blocks are typically zero after prediction, transformation, and quantization. Moreover, higher coding efficiency of video encoders while predicting B and P frames makes their residuals more sparse in comparison with I frames. In Fig. 3 residuals of one frame from Foreman sequence encoded as I, B, and p frames are illustrated. Most papers have suggested embedding the watermark merely in I frames, because watermark embedding in zero coefficients turns them to a nonzero value and can be perceptually visible [11], [12].

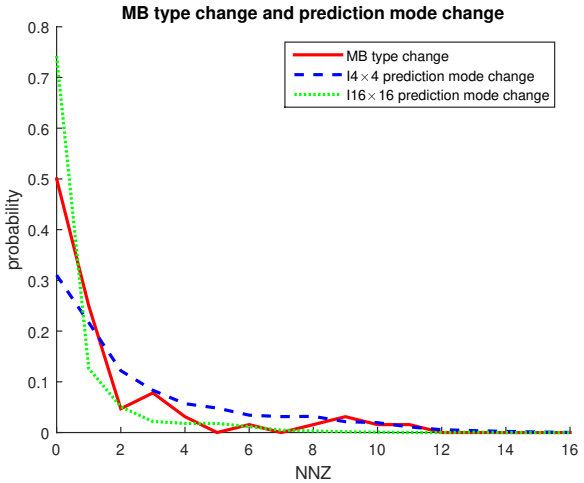
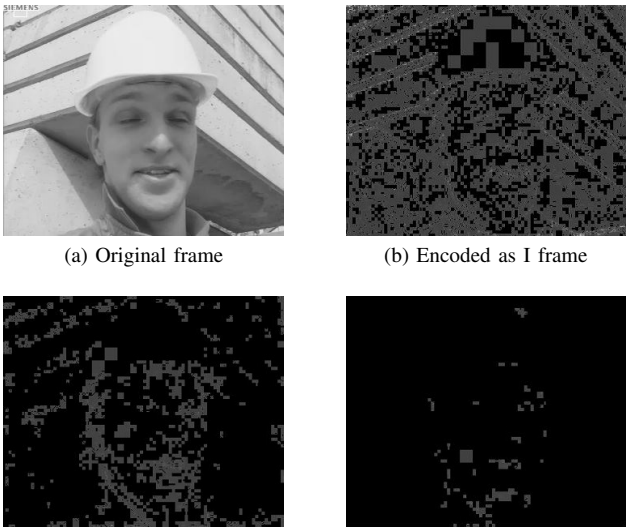


Fig. 2: Average rate of macroblock type change and intramode change given the NNZ value after recompressing 30 I frames of the Forememan, Mobile and Stefan sequences.



(a) Original frame (b) Encoded as I frame

(c) Encoded as P frame (d) Encoded as B frame

Fig. 3: Residuals of a frame encoded as I,P and B frames

Owing to the fact that quantization is a lossy operation, it is desirable to embed the watermark after quantization. We propose to embed watermark payload in the quantized DCT coefficients of residuals, computed inside video encoder after the prediction process, in order to avoid possible erasure of the watermark during compression. Watermark embedding is performed through the following steps:

STEP 1: λ percent of 4×4 blocks of quantized DCT residuals having NNZ value more than a threshold τ satisfying $S_{NNZ}(\tau) = \lambda$ are candidates for embedding. $S_{NNZ}(\tau)$ is the complementary cumulative distribution function defined by the following statement:

$$S_{NNZ}(\tau) = P(NNZ > \tau) = \sum_{NNZ > \tau} p(NNZ) \quad (1)$$

STEP 2: In each candidate 4×4 block, the LNZ level is modulated to carry the watermark bit w, as follows:

$$L' = \begin{cases} L+1 & \text{if } L \text{ is odd}, L \neq -1 \text{ and } w = 0 \\ L-1 & \text{if } L \text{ is odd}, L = -1 \text{ and } w = 0 \\ L & \text{if } L \text{ is odd}, w = 1 \\ L & \text{if } L \text{ is even}, w = 0 \\ L+1 & \text{if } L \text{ is even}, L \neq -1 \text{ and } w = 1 \\ L-1 & \text{if } L \text{ is even}, L = -1 \text{ and } w = 1 \end{cases} \quad (2)$$

where L is the original LNZ level and L' is the modulated LNZ level. Since the LNZ level is required to extract the watermark in receiver, no LNZ level should be converted to zero while embedding. Thus we have seperated the $L = -1$ and $L \neq -1$ cases to avoid changing a non zero level to zero.

B. Watermark detection

Because watermark payload is embedded in quantized DCT residuals, inverse quantization and inverse DCT are not necessary to perform for watermark extraction. To extract the watermark payload while decompressing

video, the following steps are taken on compressed video bit-stream following the entropy decoding.

STEP 1: In each macroblock the watermarked 4×4 blocks are determined according to (1).

STEP 2: One watermark bit is extracted from the LNZ level, L', in each of the above selected blocks, as follows:

$$w' = \begin{cases} 0 & \text{if } L' \text{ is even} \\ 1 & \text{if } L' \text{ is odd} \end{cases} \quad (3)$$

where w' is the extracted bit.

III. Authentication of the audio part

The original audio part, which we assume is 8-KHz sampled speech signal, is divided into frames of l seconds. Hence, each speech frame consists of $8000 \times l$ samples. b_h hash bits are generated from each frame of speech using a hash generation algorithm [5]. The payload of the watermark consists of the hash bits generated from the original audio part as illustrated in Fig. 4.

At the receiver, procedure of authenticating the audio part, drawn in Fig. 5, is composed of three parts and determines whether or not the audio frames have been manipulated. First, watermark payload is extracted from the cover signal which, in our scheme, is the visual part of the video. Then, the same hash generation algorithm, as that in the transmitter, is applied to each frame of speech of length l seconds. Finally, hash bits of each frame are compared to their corresponding bits in extracted watermark payload and speech frames are marked as healthy, if they match, and tampered otherwise. we expect the probability of marking a tampered frame as a healthy one to be 2^{-b_h}, which asymptotically equals to zero if b_h is large enough.

This hash based authentication method, however, may lead to confusions due to false tampering detection at the receiver, even if the video undergoes a permissible modification, e.g. recompression. Accordingly, the proposed scheme is inapplicable to cases that such modifications to the video are allowed.

IV. Reconstruction of the audio part

The block diagram in Fig. 6 demonstrates our proposed video watermarking system capable of reconstructing the audio part. Due to capacity constraints, the watermark should contain a compressed version of the audio part that is assumed to be a speech signal. In speech compression literature, several speech codecs have been developed among which we have chosen the G.726 which is an ITU-T ADPCM speech codec standard covering the transmission of voice at rates of 16, 24, 32, and 40 kbps [13]. Quality of the reconstructed speech is determined by the applied speech codec. As a rule of thumb, codecs of lower compression rates provide lower quality of speech. Therefore, choice of the speech codec is made by taking into account the desired quality of reconstructed speech and the available watermarking capacity in the cover signal.

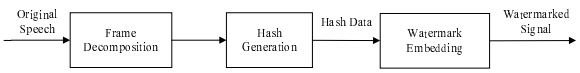


Fig. 4: Block diagram of hash data generation and embedding

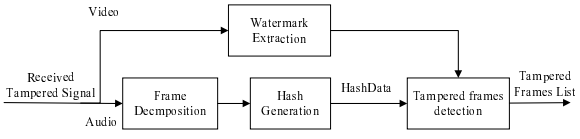


Fig. 5: Block diagram of the proposed system for tampering detection

TABLE I: Configuration parameters of the JM software

Parameter	Value
FramesToBeEncoded	100
ProfileIDC	77 (main)
IntraPeriod	1
QPISlice	24
SymbolMode	1(CABAC)
RDOptimization	0 (Low complexity mode)

V. Experimental Evaluation

In order to verify the performance of our proposed watermarking system for tampering detection and reconstruction, we have utilized the H.264 reference software JM19.0. In this software most of the configuration parameters have retained their default values, except for those given in Table I.

In this section, we first present the results of robustness test of the proposed watermarking scheme in V-A. Four standard test video sequences, Foreman, Stefan, Mobile, and Football, in QCIF format (176×144 pixels) are chosen to evaluate the performance of our watermarking method [10]. Afterwards we demonstrate the effectiveness of our method in tampering detection and in reconstruction of the audio part of the video in V-B. For this part, we have gathered a video dataset consisting of twenty test video sequences of different resolutions. Characteristics of these video sequences are summarized in Table II. All raw videos are sampled at 4:2:0 (Y:C_b:C_r) Color space and their audio part mainly contains voice.

The results for five sample sequences of our dataset plus the average result is..

A. Watermarking

To examine the robustness of our watermarking scheme we have watermarked and compressed four video sequences with QP=24 and recompressed them with the same QP. The condition under which the attacker changes the codec parameters such as GOP length and QP is

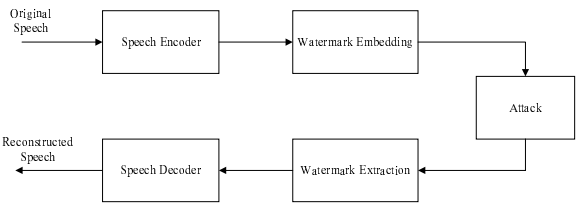


Fig. 6: Block diagram of the proposed system for reconstruction of the audio part

TABLE II: Characteristics of our video dataset

Sequence	Resolution	Frame Rate (fps)	Length (s)
Video 01	1280×536	24	2
Video 02	608×256	24	5
Video 03	528×224	25	8
Video 04	480×352	25	5
Video 05	352×288	25	2

TABLE III: Comparison between recompression bit error rate of our method and the method presented in [1]

Sequence	λ	τ	BER(proposed)	BER([1])
Foreman	11.00	8	0.01	0.28
Stefan	47.7	5	0.08	0.20
Mobile	61.8	6	0.02	0.12
Akiyo	10.65	5	0.07	0.36
Football	29.92	7	0.08	0.06
Average	33.21	-	0.05	0.20

beyond the scope of this paper and can be considered for further studies. We suppose the codec parameters retain their initial values during recompression. Bit error rate (BER) between the extracted and the original watermark is measured for the video sequences in our dataset and compared with the method in [1]. Result of five sample video sequences and the average results are shown in Table III. While implementing our watermarking scheme, we have changed and optimized the threshold value τ for each sequence to get the best results. Also to implement the method presented by Fallahpour et al. in [1], $k = 8$ watermark bits are embedded in each macroblock and the encoder is configured with QP=24. As the average row illustrate, the average BER in our scheme is 0.05, while it is 0.2 using the method in [1]. Since we have chosen the highest frequency levels in each block for embedding, the watermarked video is guaranteed to be transparent as investigated thoroughly in [1].

B. Tampering detection and reconstruction of the audio part

To evaluate the performance of our authentication scheme, we exploited the MD5 hash algorithm with $b_h = 128$ to generate hash bits of speech frames of length $l = 20ms$. The 16-byte result of the MD5 for each group of 160 speech samples is embedded as watermark payload in visual part of the video. The original signal of length $t = 2$ seconds shown in Fig. 7(a) is tampered by substituting 30.28% of the samples with zero which take up three out of eight words of the speaker's sentence. The tampered signal illustrated in Fig. 7(b) is delivered at the receiver. The result of detecting the healthy and tampered regions of audio part using our proposed scheme is demonstrated in Fig. 7(c).

According to the proposed design detailed in section IV, the number of watermark bits must be less than or equal to the total available watermarking capacity. Hence, to implement our presented watermarking scheme for reconstruction of the audio part, first, we have to determine all the parameters involved in watermarking capacity. In addition to λ, which denotes the percentage

TABLE IV: Determination of Intraperiod and speech compression rate

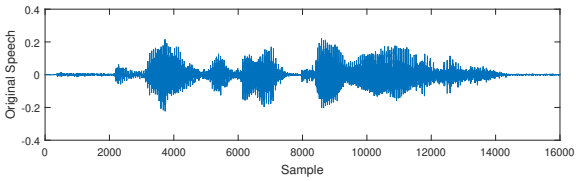
sequence	λ	Intraperiod	speech compression rate(kbps)
Video 01	37.5	12	32
Video 02	61.67	6	24
Video 03	64.93	5	24
Video 04	30.30	5	16
Video 05	50.50	5	16

of suitable blocks for embedding, the number of I frames in a video sequence is determinative, because we tend to merely embed the I frames and leave the B and P ones intact. Intraperiod or GOP length is the configuration parameter of JM19.0 which determines the number of I frames in each GOP. Assuming r, as the compression rate of the speech codec in bit per seconds (bps), and λ, as percent of 4×4 blocks in each frame, are selected for embedding according to the spatial analysis, the following inequality must be satisfied so that the required watermarking capacity is available:

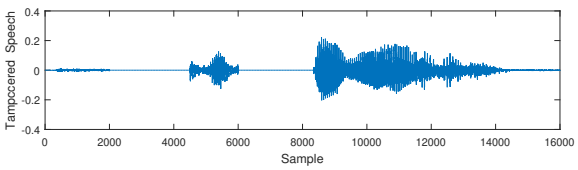
$$\frac{\lambda.m.n}{16} \times \frac{\text{Frame Rate}}{\text{Intraperiod}} \geq r \tag{4}$$

where resolution of video sequence is assumed to be $m \times n$ pixels. The left side of this inequality is the total watermarking capacity in one second of video and has to be more than r bits which is the length of the watermark payload generated from one second of the audio part.

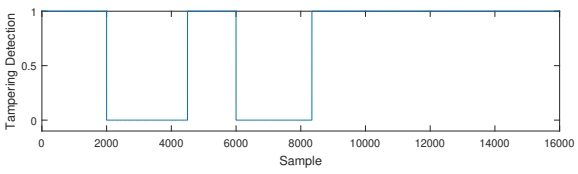
Performance of self-embedding watermarking schemes is usually described by quality of the recovered signal. For assessing quality of reconstructed speech, we employed the Perceptual Evaluation of Speech Quality (PESQ) method presented in the ITU-T P.862 Recommendation [14], which compares an original signal with a degraded signal and outputs a PESQ score in the range between -0.5 to 4.5 as a prediction of the perceived



(a) Original Speech Signal



(b) Tampered Speech Signal



(c) Tampered and healthy frames of speech detected at receiver

Fig. 7: Performance of the proposed method for authenticating

TABLE V: Results of the PESQ quality evaluation of the original and reconstructed audio part after recompression attack

sequence	BER	PESQ	
		Original	Reconstructed
Video01	0.19	4.267	3.883
Video02	0.15	3.661	3.308
Video03	0.25	3.197	2.743
Video04	0.08	2.145	1.928
Video05	0.14	2.486	2.148
Average	0.16	3.151	2.802

quality. In Table V results of PESQ measurement of the original audio part, compressed at the rates given in Table IV, and also the PESQ of reconstructed speech after recompression, are shown.

VI. Conclusion

In this paper, a compressed video watermarking system for authentication and reconstruction of the audio part of a video is designed. For this sake, we proposed a robust watermarking approach that features low complexity and high quality of the watermarked video. The results exhibit robustness with average bit error rate of 0.05 after recompression. We have also proposed two different applications of video watermarking for localizing the tampering and reconstructing the original audio part. After recompression attack, we are able to reconstruct the audio part with average quality degradation of 0.349 based on PESQ score.

References

[1] Mehdi Fallahpour, Shervin Shirmohammadi, Mehdi Semsarzadeh, and Jiying Zhao, "Tampering detection in compressed digital video using watermarking," *IEEE Transactions on Instrumentation and Measurement*, vol. 63, no. 5, pp. 1057–1072, 2014.

[2] Vahideh Amanipour and Shahrokh Ghaemmaghami, "Video-tampering detection and content reconstruction via self-embedding," *IEEE Transactions on Instrumentation and Measurement*, 2017.

[3] Mohammed Ghanbari, *Standard Codecs: Image compression to advanced video coding*, Telecommunications. Institution of Engineering and Technology, 2011.

[4] Azadeh Mansouri, Ahmad Mahmoudi Aznaveh, Farah Torkamani-Azar, and Fatih Kurugollu, "A low complexity video watermarking in H.264 compressed domain," *IEEE Transactions on Information Forensics and Security*, vol. 5, no. 4, pp. 649–657, 2010.

[5] Saeed Sarreshtedari, Mohammad Ali Akhaee, and Aliazam Abbasfar, "A watermarking method for digital speech self-recovery," *IEEE/ACM Transactions on Audio, Speech and Language Processing (TASLP)*, vol. 23, no. 11, pp. 1917–1925, 2015.

[6] Lotfi Abdi, Faten Ben Abdallah, and Aref Meddeb, "Real-time watermarking algorithm of h. 264/avc video stream.," *International Arab Journal of Information Technology (IAJIT)*, vol. 14, no. 2, 2017.

[7] Dawen Xu, Rangding Wang, and Jicheng Wang, "A novel watermarking scheme for h. 264/avc video authentication," *Signal Processing: Image Communication*, vol. 26, no. 6, pp. 267–279, 2011.

[8] Ke Niu, Xiaoyuan Yang, and Yingnan Zhang, "A novel video reversible data hiding algorithm using motion vector for h. 264/avc," *Tsinghua Science and Technology*, vol. 22, no. 5, pp. 489–498, 2017.

[9] Hong Zhang, Yun Cao, and Xianfeng Zhao, "Motion vector-based video steganography with preserved local optimality," *Multimedia Tools and Applications*, vol. 75, no. 21, pp. 13503–13519, 2016.

[10] "Derf's test media collection," Available at https://media.xiph.org/video/derf/.

[11] W-M Chen, C-J Lai, H-C Wang, H-C Chao, and C-H Lo, "H.264 video watermarking with secret image sharing," *IET Image Processing*, vol. 5, no. 4, pp. 349–354, 2011.

[12] Dawen Xu, Rangding Wang, and Jicheng Wang, "A novel watermarking scheme for H.264/AVC video authentication," *Signal Processing: Image Communication*, vol. 26, no. 6, pp. 267–279, 2011.

[13] "ITU-T recommendation G.726 40 32 24 16 kbit/s adaptive differential pulse code modulation (ADPCM)," *ITU-T*, Aug. 1990.

[14] "ITU-T recommendation P.862 perceptual evaluation of speech quality (PESQ)," *ITU-T*, Feb. 2001.