An Overview of Information Hiding in H.264/AVC Compressed Video

Yiqi Tew, *Student Member, IEEE,* and KokSheik Wong, *Member, IEEE*

***Abstract*—Information hiding refers to the process of inserting information into a host to serve specific purpose(s). In this paper, information hiding methods in the H.264/AVC compressed video domain are surveyed. First, the general framework of information hiding is conceptualized by relating the state of an entity to a meaning (i.e., sequences of bits). This concept is illustrated by using various data representation schemes such as bit plane replacement, spread spectrum, histogram manipulation, divisibility, mapping rules, and matrix encoding. Venues at which information hiding takes place are then identified, including prediction process, transformation, quantization, and entropy coding. Related information hiding methods at each venue are briefly reviewed, along with the presentation of the targeted applications, appropriate diagrams, and references. A timeline diagram is constructed to chronologically summarize the invention of information hiding methods in the compressed still image and video domains since 1992. A comparison among the considered information hiding methods is also conducted in terms of venue, payload, bitstream size overhead, video quality, computational complexity, and video criteria. Further perspectives and recommendations are presented to provide a better understanding of the current trend of information hiding and to identify new opportunities for information hiding in compressed video.**

***Index Terms*—Compressed video, compression, data embedding, H.264, information hiding.**

I. Introduction

MOTION PICTURE, widely known as video, has become one of the most influential media in the entertainment industry. In the past, video was virtually a sequence of frames. As semiconductor technology advances, users become greedier and raise the bar of technological needs. We desire more features in video, including instant frame access, high resolution, high frame rate, fast forward, etc. Therefore, the Motion Picture Expert Group (MPEG) standard was established in 1993, and it enabled video compact disc (VCD) technology [1], followed by MPEG-2 in 1995, which enables the digital video disc (DVD) and satellite TV [2]. In the pursuit of higher efficiency in video coding, H.264 (H.264/advance video coding) is proposed by the Video Coding Experts Group [3] and it has become one of the most commonly practiced video compression formats since 2003. The design of H.264 provides an enhanced compression performance on video representation for various purposes, including video telephony, storage, broadcast, and streaming applications. H.264 achieves a significant improvement in rate distortion trade off by offering high video quality for relatively low bitrate as compared to the previous generations of video compression standard. As a result, various digital video technologies lay on the H.264 compression framework, such as Blu-ray video disc, video streaming (e.g., YouTube, Dailymotion), surveillance camera, handy video recorder, etc. A similar trend is expected for the recently finalized H.265 video compression standard [4].

Currently, with the existence of broadband Internet service and ubiquitous network coverage through a cellular data plan, video can be conveniently downloaded and broadcast through social networking services such as YouTube, Facebook, and Twitter. Therefore, there are various needs to manage and/or protect the vast number of videos including: 1) tracking illegal distribution of copyrighted video to secure business revenue; 2) hyperlinking related contents while ensuring the hyperlink information always stays intact with the video to enhance user experiences; and 3) monitoring video broadcasts and Internet distributions to generate reports regarding when, where, and how many times a video has been aired/streamed. Information hiding is one of the possible solutions to serve the aforementioned needs. It refers to the process of inserting information into a host to achieve certain features or to serve specific purposes. The components of information hiding are summarized in Fig. 1. In particular, we modify the watermarking framework in [5] to a general framework of information hiding to emphasize on the information insertion part. Here, the information is external to the content (e.g., ownership information and secret message) or deduced

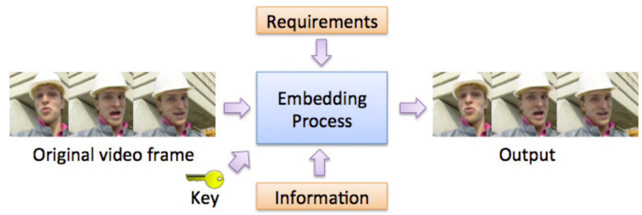


Fig. 1. General framework of information hiding.

Manuscript received March 11, 2013; revised May 30, 2013; accepted July 1, 2013. Date of publication August 6, 2013; date of current version February 4, 2014. This work was written under the Unified Scalable Information Hiding project and was supported by University of Malaya HIR under Grants UM.C/625/1/HIR/MOHE/FCSIT/01 and B000001-22001. This paper was recommended by Associate Editor R. C. Lancini.

The authors are with the Faculty of Computer Science and Information Technology, University of Malaya, Kuala Lumpur 50603, Malaysia (e-mail: yiqi@siswa.um.edu.my; koksheik@um.edu.my).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TCSVT.2013.2276710

from the content (e.g., checksum and hash value). The information is inserted into the host by means of modifying part(s) of the host based on the representation scheme in use and a key so that the output (i.e., content + inserted information as a single unit) satisfies the imposed properties and requirements. These properties include high perceptual quality, reversibility, secrecy of the inserted information, etc.

In the video domain, the application of information hiding (also referred to as data embedding) can be coarsely categorized as watermarking, steganography, error recovery (resilient), and general data embedding. In the case of watermarking, information is inserted into video to visibly (e.g., Fig. 2) or invisibly render the ownership information [6]–[8]. The inserted watermark information is retrieved from the content (i.e., when it is alleged to be an illegal copy) to prove ownership during a dispute, to detect any attempt in destroying the inserted watermark, or to detect act of tampering with the content [9]. On the other hand, steganography is the art and science of concealing the existence of the secret (external) information inserted into a cover such as image, video, audio, or text [10], [11]. The existence of the embedded information should be undetectable, that is, the modified content should be perceptually and statistically (with respect to certain features) similar to its original unaltered counterpart [12]. Furthermore, information hiding is also utilized to realize error recovery (also referred to as error concealment). Here, the important components of the compressed video (e.g., motion vectors, prediction modes) are embedded into the video itself [13], [14]. When an error occurs during transmission, the important components can be extracted to patch the lost or corrupted parts at reasonable perceptual quality. Lastly, general data embedding refers to the principle idea of inserting information into a host for specific purpose. For example, text description of a video is embedded into itself [15] for annotation purposes. Note that most of the current digital video formats enable the inclusion of text through metadata or using a separate file without requiring the use of data hiding. However, text embedding is viable because the information always stays intact with the video, but at the expense of possible video quality degradation due to information hiding. Similar arguments are applicable for embedding audio signal into video [16].

As of this writing, several surveys highlight the individual field of information hiding and its corresponding application ranging from 1998 to 2012. These surveys include watermarking, steganography, and related issues in digital data encryption. Su *et al.* [17] discussed the specific watermarking issues in text, image, and video, along with the problems in watermarking using a low-amplitude pseudo-noise carrier signal. Peticolas *et al.* [18] surveyed the limitation of information hiding techniques against basic, robustness, mosaic, and interpretational attacks. Cheddad *et al.* [11] advocated several fundamental requirements in the application of steganography, such as maintaining a balanced distribution of the embedding bits (i.e., "0 and 1"), avoiding smooth homogeneous areas, minimizing visual artifacts, and emphasizing the necessity to verify robustness against steganalysis. In addition, Kayarkar *et al.* [19] surveyed on recent information hiding

Fig. 2. Original video (left) and visible watermarked video (right).

techniques in still image, audio signal, network packets, and video sequences, along with comparative analysis of these techniques.

There are some survey papers on digital data encryption methods, in which part of them are closely related to information hiding. For example, Lin *et al.* [20] published a survey focusing on encryption and watermarking methods, along with the challenges and directions of digital right management (DRM) in video domain. Recently, Stutz *et al.* [21] published a critical overview of encryption methods in the H.264/SVC (scalable video coding) compressed video domain. They summarize the applications of video encryption in terms of packaging, streaming, transcoding, and watermarking.

All in all, we found that these papers are insufficient in providing a conspectus of information hiding in the compressed video domain. Hence, in this paper, we survey information hiding methods designed specifically for compressed video, illustrate possible hiding venues within the H.264 coding structure for information hiding, and review their applications. We considered H.264 (instead of the latest compression standard, i.e., H.265) because of its rich literatures in various applications. Here, we emphasize the techniques that manipulate the underlying coding structure of H.264 to realize data embedding and how each of the techniques affects the payload (i.e., the number of bits that can be inserted into the host video), bitstream size overhead, video quality, and computational complexity. Nevertheless, at times, information hiding methods designed for image are also reviewed since they can be readily applied to compressed video. The remainder of this article is organized as follows. Section II briefly reviews the H.264 video compression standard, emphasizing on the components that are commonly utilized to realize information hiding. Section III conceptualizes information hiding in a general setting and reviews the commonly utilized data representation schemes. Section IV discusses the venues in H.264 compressed video to realize information hiding. Sections V and VI summarize and analyze the current trend of information hiding in compressed video, respectively. Section VII presents the outlook and suggestions for further research directions, and Section VIII concludes this article.

II. Overview of H.264 Video Compression Standard

In comparison to the previous standards [2], H.264 incorporates various new features to further improve video compression efficiency. Notably, these features include intra-prediction in intra-frame, multiple frames reference capability,

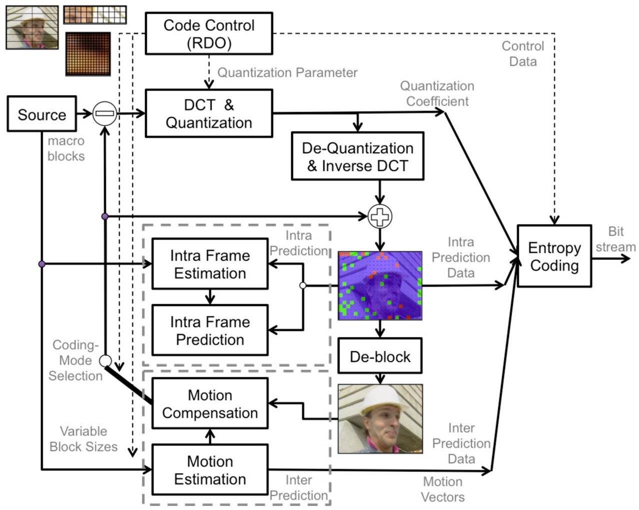


Fig. 3. H.264 hybrid video encoder.

quarter-pixel interpolation, deblocking filtering post-processing, and flexible macroblock ordering (FMO) [22]–[26]. In general, H.264 divides the sequence of frames (i.e., images) into several group of pictures (GOPs). These frames are labeled as I (intra), P (predicted), and B (bidirectionally predicted) frames, depending on the order in which they appear.

The hybrid encoding process of the H.264 video compression standard is shown in Fig. 3. At the source part, each frame is divided into nonoverlapping blocks of uniform size (i.e., 16×16 pixels) called macroblocks, and these macroblocks are handled uniquely depending on their types. Each macroblock can be further divided into smaller blocks as shown in Fig. 4, with 4×4 being the smallest possible block size. These macroblocks are subjected to discrete cosine transform (DCT), quantization, and entropy coding. First, the pixel values in a macroblock are used in the DCT and quantization process. The outputs of the DCT and quantization process, i.e., the quantized DCT coefficients, undergo the dequantization and inverse DCT process for prediction and motion estimation purposes. In particular, the intra- and inter-prediction processes utilize these reconstructed pixel values to execute pixel value estimation and to make decisions on coding-mode. Ordinarily, rate distortion optimization (RDO) is utilized to choose the best operational point between inter- or intra-mode for coding each macroblock. The code control block in Fig. 3 represents an optimizer that regulates the selection of coding modes and block sizes [27]. It also controls the quantization parameter to achieve the targeted video bitrate. Finally, the results of the DCT and quantization process, prediction data, motion vectors, control data from RDO are sent for entropy coding. The output of entropy coding is a series of compressed video contents in the binary stream preceded and/or inter-leaved with various predefined markers. The combined bitstream is then transmitted and/or stored in various mediums.

In I-frame, the pixel values in a block are either coded directly by using coefficients in the transformed domain or predicted (i.e., intra-prediction) using neighboring blocks in the same frame to exploit the spatial redundancies within a frame. In P-frame, motion estimation (i.e., inter-prediction) between two frames can be implemented to take advantage of the temporal redundancies. For that, the previously encoded frame, which itself could be a motion compensated frame, is decoded and its prediction errors, if any, are decoded and added to the decoded frame for motion estimation purposes. In the case of B-frame, up to two frames (past and/or future) can be considered for motion estimation purposes.

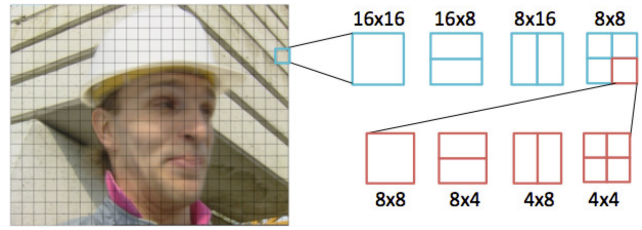


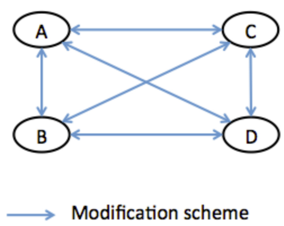
Fig. 4. Block size type selection for intra- and inter-macroblock to embed information.

Outputs from the aforementioned processes, including coefficient values, prediction errors, motion vectors, etc., are further entropy coded. There are two entropy coding methods in the H.264 compression standard to encode the quantized transform coefficients, namely, context-adaptive variable length coding (CAVLC) [28] and context-adaptive binary arithmetic coding (CABAC) [29]. CAVLC processes a macroblock in the form of run-level pairs, whereas CABAC binarizes all the entities for further processing. Both of them choose the best table or probability model depending on the local context to encode syntax including quantized transform coefficients, motion vector information, etc. CABAC always offers higher compression gain because it allows the assignment of a noninteger number of bits to each symbol of an alphabet, and permits the adaptation to statistics of nonstationary symbol. However, CABAC is of higher computational complexity when compared to CAVLC. Output of the entropy coder is then preceded by and/or inter-leaved with various predefined markers to form the H.264 format compliant video for transmission and storage purposes.

III. Data Representation Schemes

Conceptually, an information hiding method can be illustrated by the relationship among state, entity, and the meaning of each state. Here, meaning refers to part (e.g., the 16th 4-bit segment) of the information and it is represented (i.e., encoded) by a particular state of the entity in which case an entity can be a pixel, coefficient, coding mode, etc. The information determines the state in which the entity should be in during the embedding process, and the modification scheme changes the entity from one state to another to encode the desired information. For the example shown in Fig. 5, the message “dog hits cat" can be represented by three coefficient values, namely, $E_1 = 12$, $E_2 = 28$, and $E_3 = 3$, in the chrominance channel.

Next, we detail the commonly considered data representation schemes for information hiding, focusing on those that are applicable to video domain under the H.264 compression



State (condition)	Meaning
A (E_i < 10)	cat
B (10 ≤ E_i < 15)	dog
C (15 ≤ E_i < 20)	chase
D (20 ≤ E_i)	hit

Example:
Meaning : Dog hits cat
State : B, D, A
Entity : E_1 = 12, E_2 =28, E_3 =3

Fig. 5. State of the entity and its meaning for information hiding.

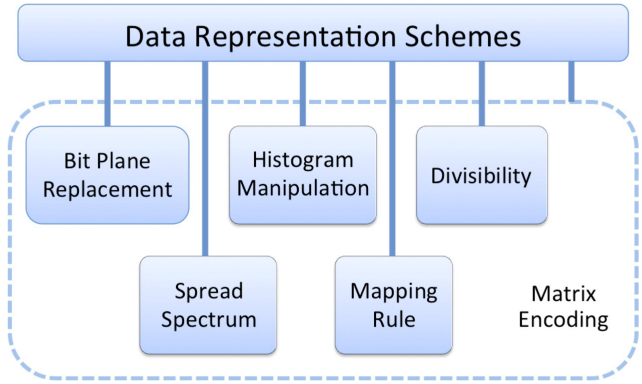


Fig. 6. Classification of data representation schemes for information hiding in H.264 compressed video.

standard. Fig. 6 shows the classification of data representation schemes in H.264 and the following subsections describe them in greater detail. Here, the dotted enclosing rectangle emphasizes that matrix encoding is applied on one or more of the data representation schemes (i.e., rectangles in solid line) in Fig. 6. We acknowledge that this list is nonexhaustive, but they are directly applicable to a compressed video.

A. Bit Plane Replacement

The general idea of bit plane replacement is to embed information in a particular bit plane (i.e., location) agreed upon by both the sender and receiver [30], [31]. In this approach, one bit can be inserted into a digital host without causing significant perceptual impact on the host content. This technique is commonly applied to entities such as raw pixel value, audio sample, motion vector information. It achieves high payload, low distortion, and its implementation is relatively straightforward.

Bit plane replacement is widely referred to as least significant bit (LSB), which manipulates the right most bit of an entity. The idea of LSB embedding is generalized by [32] and [33] in still image. They propose an optimal bit plane substitution method to hide information into an image. A genetic algorithm is deployed to reduce the search space (i.e., combination of k bits from the pixel value) to achieve the highest possible image quality based on a given embedding rate. In the video domain, Kapotas *et al.* [34] applied this technique to coefficients in the luminance and chrominance channels. Their results show that by modifying the last three bit planes of an 8-bit coefficient, the distortion is imperceptible. However, the aforementioned methods are not optimal because LSB (of the pixels, coefficients, motion vectors, etc.) is adjusted locally. Yu *et al.* [35] proposed an information hiding method based on the prediction process in image compression. In particular, the LSB of the prediction errors are modulated to embed information.

Generally, the LSB scheme is the most straightforward way to encode information, with insignificant perceptual quality degradation when applied on the raw pixel values. However, a direct LSB scheme is irreversible, and it will cause noticeable degradation when applied on syntax (e.g., sign bit) or index selection (e.g., prediction type) in H.264 encoding process.

B. Spread Spectrum

In a still image, spread spectrum-based information hiding technique is widely utilized for watermarking purposes. In this class of data representation scheme, hidden information is embedded within a range of acceptable changes and it is not perceptible to the human visual system. Cox *et al.* [36] introduced three watermarking techniques based on spread spectrum. These techniques can be formulated as

$$v'_i = v_i + aw_i \quad (1)$$
$$v'_i = v_i(1 + aw_i) \quad (2)$$
$$v'_i = v_i(e^{aw_i}) \quad (3)$$

where $w_i \in \{0, 1\}$ is the information, v_i and v'_i are the input and output values, respectively, and a is a scaling parameter that determines the extent to which w alters v_i. Equation (1) suggests that the output is generated by adding information to the host, which is invertible. Outputs of (2) and (3) are generated based on multiplication and exponential operations, and they are invertible only if $v_i \neq 0$. Marvel *et al.* [37] proposed a system that combines spread spectrum communication, error control coding, and image processing to hide information in a still image without causing bitstream size overhead. Budhia *et al.* [38] then proposed a steganographic method based on the spread spectrum technique and the exploitation of temporal information. This technique is able to withstand frame average attack. Similarly, Vinod *et al.* [39] proposed to hide information in the motion trajectories of blocks using the spread spectrum embedding scheme. On the other hand, Valizadeh *et al.* [40] proposed a correlation-aware improved spread spectrum embedding scheme. Their technique shows significant reduction in the visual quality distortion of the host signal and improved the hidden message extraction performance.

The spread spectrum scheme allows information to be hidden in (i.e., spread throughout) the entire video content, where the hidden information can hardly be decoded from only part of the content. Yet, it is of higher computational complexity than that of the LSB scheme because the video needs to be processed entirely.

C. Histogram Manipulation

Fig. 7 shows the histogram of pixels in a compressed still image (equivalent to one frame in video). The pixels values are usually associated with another value to embed information.

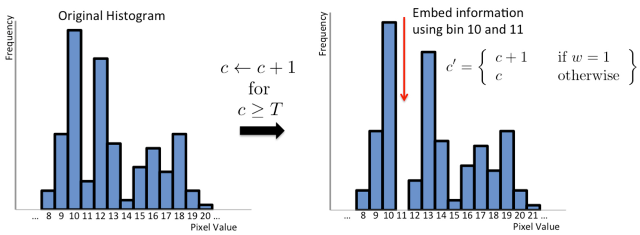


Fig. 7. Hiding information using histogram.

State	Entity (Block Size)	Meaning (Information)
A	16 x 16	0 0
B	16 x 8	0 1
C	8 x 16	1 0
D	8 x 8	1 1

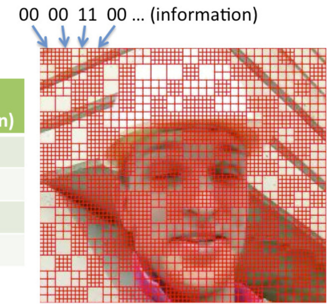


Fig. 8. Mapping rules for macroblock size to embed information.

In particular, Ni *et al.* [41] utilized the zero and peak points of the histogram and slightly modify the (grayscale) pixel values to embed information. The original histogram is preprocessed to zero out the Tth bin (i.e., the bin next to the peak point) for data embedding purposes by increasing the value c to $c+1$ for all $c \geq T$. Ni *et al.* achieved reversible data embedding with high payload while maintaining high perceptual quality (i.e., >48 dB). Here, reversibility refers to the ability to perfectly reconstruct the original host. Similarly, Wang *et al.* [42] divided the image into nonoverlapping blocks and manipulated the peak point (i.e., most frequently occurred pixel value) to embed information. A location map is employed to mark the position of the peak point for extraction and reversibility purposes. Unlike Ni *et al.*'s [41] method, each peak point can encode 3 bits. This technique can be applied recursively for multilayer data embedding while maintaining image quality.

In Vleeschouwer *et al.*'s method [43], the image is also divided into nonoverlapping blocks and each block is further divided into two groups based on their locations. The values in each group are mapped to a circle using a circular interpretation of bijective transformation to achieve lossless watermarking. Specifically, this proposal manipulates the relative angle between two vectors, each elongated from the center of the circle to the direction of the center of mass, to convey information. In a different work, Zhao *et al.* [44] embed information by forming histogram of pixel value differences and apply multilevel histogram shifting technique to achieve high payload. It is also possible to associate existing values (i.e., those that actually occur in the image) to nonexisting ones in histogram to encode the information. Ong *et al.* [45] divided an image into nonoverlapping blocks and manipulate the histogram of each block. Specifically, each value is paired with a mirror value to embed information and degrade image quality.

The histogram manipulation scheme is applicable to both the spatial and frequency domains. It is a common way to achieve reversible information hiding. However, it requires expensive preprocessing (e.g., vacating a bin for data embedding) and suffers from the under/overflow problems.

D. Mapping Rules

Mapping rules rely on a set of codewords for information embedding and extraction purposes. It depends on the structure or syntax of the content and requires a common mapping rule agreed by the sender and receiver. Basically codewords are associated with predefined meaning (e.g., "00," "01," "10," or "11") and they are chosen based on the information to be embedded. As such, an algorithm is forced to operate in a mode determined by the information to be embedded instead of the original decision. Fig. 8 illustrates a possible mapping rule by utilizing the macroblock size to represent information. During the encoding process, each macroblock in a video frame is decomposed into different block sizes according to the information to be embedded.

Kapotas *et al.* [46] implemented this approach to embed information during the prediction process. It forces each macroblock to assume certain block-combinations of various sizes for embedding information. This method offers high payload but suffers from the severe bitstream size increment. On the other hand, Wong *et al.* [47] embedded information by considering the combinations of cardinalities (i.e., number of nonzero coefficients) in a group of 8 × 8 blocks (τ of them in a group). In particular, these cardinalities form a t-tuple $(\tau_1, \tau_2, \cdots, \tau_t)$ where each is associated to an unique meaning. Hence, the information to be embedded decides the number of zero run level pairs (i.e., nonzero coefficients) to be in each 8 × 8 block in the group. This approach offers scalable payload without bitstream size overhead but it is computationally expensive.

Commonly, mapping rules are flexible to be applied in different components of the H.264 framework (i.e., block size, prediction type). However, it has limited payload when it is applied to the hiding venue with limited number of operational modes (e.g., only eight intra-prediction modes for an intra-macroblock).

E. Divisibility

The divisibility of a value by a specific divisor can be exploited as an essential property for reversible information hiding [48], [49]. For example, in [50] Wong *et al.* scaled the magnitude of each coefficient in the macroblock by a prime number when "1" is to be embedded, or leave them as they are (i.e., no change) to embed "0". In order for this method to be viable, divisibility of all coefficients by the chosen prime number needs to be checked to avoid error during extraction. A more sophisticated method considers a pair of neighboring pixels x and y, and the value n [51]. These values are then transformed to obey a simple equation as

$$y_i + nx_{i+1} \equiv 0 \bmod (n+1). \tag{4}$$

The transformed pixels can be modified by adding an integer in the range of $[1, n]$ to dissatisfy (4). On the other hand, the

Information (ω_i)	Modification (μ_i)						
	μ_1	μ_2	μ_3	μ_4	μ_5	μ_6	μ_7
ω_1	✓		✓		✓		✓
ω_2		✓	✓			✓	✓
ω_3				✓	✓	✓	✓

Fig. 9. Example of a $(1, 3, 7)$ matrix encoding [52].

untransformed pixels can be modified to satisfy the divisibility equation. Here, let x_i be an untransformed pixel and let q_i be $(x_i + nx_{i+1}) \mod (n+1)$ for $q_i \in \{0, n-1\}$. By simply subtracting q_i from x_i, the modified pixel fulfills the divisibility condition. Then, correction codes (codewords) are collected and a map containing the location of transformed pixels is created. Finally, the information together with the correction data are encoded as a sequence of integers in the range of $[1, n]$ and they are embedded into the transformed pixels according to the location map.

Traditionally, the divisibility scheme is invented for reversible information hiding. This approach maintains high perceptual quality of the embedded video. However, it is of high computational complexity and often requires a location map for decoding and reconstruction purposes.

F. Matrix Encoding (ME)

ME is a general principle that can be applied on top of the aforementioned data representation schemes to improve their embedding efficiencies, i.e., the number of modifications per embedded bit. Fig. 9 shows an example of the $(1, k, 2^k - 1) = (1, 3, 7)$ matrix encoding scheme and the dependencies of the information μ_i on the entities ω_i. Here, at most one modification is required to embed $k = 3$ bits using $2^k - 1 = 2^3 - 1 = 7$ entities. To embed information, μ_i is compared against the parity (denoted as p_i) of the selected μ_i's. The scheme then decides which μ_i to modify when necessary. For example, if $p_1 \neq \omega_1$, $p_2 \neq \omega_2$ but $p_3 = \omega_3$, then μ_3 will be modified based on the dependency stipulated in Fig. 9. ME is proposed by Crandall *et al.* [53] and further improved by Fridrick *et al.* [54]. In Fridrick *et al.*'s paper, ME is improved based on two approaches: simplex codes and random linear codes. These approaches aim to achieve high payload. F5 is the first data embedding method that utilizes ME to reduce the number of modifications [55]. Fan *et al.* [56] further improved the payload of ME while reducing image quality degradation by considering quantization error.

Choi *et al.*'s [57] method embeds a secret image (encoded using Huffman) in selected bit planes of all color channels of a host image using ME. The output image (i.e., manipulated host) assumes less distortion when compared to the ordinary non-ME-based data representation schemes. In [58], a hybrid data representation scheme is proposed by applying ME recursively on selected blocks (those that are identified to be carrying edge information) and modifying only large coefficients (in terms of magnitude) when necessary. Payload is improved while trading off with embedding efficiency.

The matrix encoding scheme offers high embedding efficiency by encoding k-bits using $2^k - 1$ hiding elements

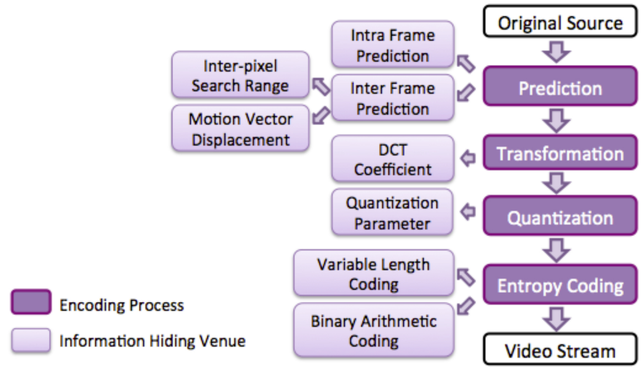


Fig. 10. Venues for information hiding in H.264.

with at most one modification. The efficiency is adjustable by trading payload for perceptual quality. Nevertheless, the high embedding efficiency is only achievable when inserting a small amount of information.

IV. Opportunities for Information Hiding in H.264

H.264 is a hybrid video encoding standard that consists of several crucial processes, including prediction, transformation (i.e., DCT), quantization, and entropy coding, as shown in Fig. 10. Over the years, various information hiding methods are proposed and realized using various components, referred to as venues [59], within each process. Essentially, information hiding can take place prior to the encoding process, i.e., conventional methods (robust against compression) can be applied directly to each frame. However, researchers refrain from taking this approach due to the inevitable lost of information caused by compression that leads to inaccurate extraction of the embedded information. Instead, it is more promising to directly manipulate entities of a H.264 compressed video to realize information hiding. In the following subsections, we review the function(s) of the aforementioned processes in H.264 and detail the representative information hiding methods related to each process. To compare the performance of the hiding venues in H.264, the pros and cons of each venue with respect to payload, bitstream size overhead, video quality, and computational complexity are presented in Table I.

A. Prediction Process

Several researchers have manipulated the block prediction process in vector quantization-based image compression to embed information. Different coding methods are applied on dedicated blocks, such as truncate coding [60], and side-match vector quantization [61]. In the compressed video domain, similar approaches are taken by exploiting mode, block size, entities, etc., that are related to the prediction process.

1) *Intra-Frame Prediction:* If a macroblock is encoded in intra-mode, the prediction is carried out by utilizing one of the 14 prediction modes (i.e., nine for 4×4 blocks, four for 16×16 blocks, and the skip mode) while referring to the previously encoded and reconstructed blocks, where they themselves could be macroblocks predicted using the intra-prediction mode. To exploit mode selection for information

Size (State)	Bits represented (Meaning)
16×16	00
16×8 or 8×16	01
8×8 or 4×4	10
8×4 or 4×8	11

	CAR	Information
Example :	434152	ASCII (Hex)
	0100 0011 0100 0001 0101 0010	ASCII (Binary)

Separated in 2 bits per group

01	00	00	11	01	00	...

Select block type according to the mapping rule

8×16	16×16	16×16	8×4	8×16	16×16	...

Fig. 11. Mapping rules for prediction block type to embed information.

hiding, mapping rules are usually considered to improve the payload without causing significant bitrate overhead [62], [63]. These methods categorize the selected prediction modes for I4MB (i.e., intra 4×4) into two groups so that the first group denotes "0" and the other denotes "1." The prediction process is forced to assume the best mode among those belonging to the group that represents the information to be embedded. The embedded message can be readily decoded by referring to flags such as *pre_intra_4×4_pred_mode*. Kim *et al.* [64] also exploited the intra-prediction mode (in combination with coefficients) to realize blind (i.e., the extraction process can be performed without referring to the original frame) and semiblind watermarks. A similar approach is proposed by Xu *et al.* [65] where macroblocks are selectively chosen based on a chaotic sequence and the most probable prediction mode is manipulated to embed information.

Yang *et al.* [23] restricted information hiding to 4×4 blocks in I-frame using matrix encoding. 4×4 blocks are chosen because they contain a high number of nonzero DCT coefficients and modifying their prediction modes (for hiding information purposes) seldom leads to visible artifacts as compared to the case of 16×16 blocks. Two bits of information are encoded by three blocks through matrix encoding. Experimental results on several test sequences demonstrate that this technique can achieve blind extraction in real-time.

2) *Inter-Frame Prediction:* In order to increase the coding efficiency in inter-prediction mode, the H.264 standard has adopted seven different block sizes (namely, 16×16, 16×8, 8×16, 8×8, 8×4, 4×8, and 4×4) and the motion estimation algorithm is invoked for each block size. The block type that results in the minimum number of bits will be selected. Kapotas *et al.* [66] proposed to force the encoder to choose a particular block type according to the information to be embedded. In this technique, each block type is assigned to represent two bits. Then, the information is divided into segments (i.e., each of length two bits) and each segment is encoded using block size as shown in Fig. 11. These macroblocks are then motion estimated using the forced block size. This technique only affects the visual quality of the video insignificantly. The payload is high and it is proportional to the size of host video.

3) *Motion Vector Displacement:* Information hiding can be achieved by using the motion vector attributes, including phase angle, horizontal, and vertical magnitudes. Jordan *et al.* [67] initiated this technique for the video watermarking purpose. Then, Zhang *et al.* [68] and Dai *et al.* [69] proposed enhanced versions of Jordan *et al.*'s technique by restricting information hiding to specific types of inter-frame. In particular, frames consisting of motion vectors with a large magnitude and small in phase angle are considered. These three methods are studied by Su *et al.* [70] and a steganalysis method is proposed. Similarly, Guo *et al.* [71] proposed a method to embed secret information in the motion vectors between two P frames. In particular, horizontal and vertical offsets (i.e., odd or even) in motion vectors are modified to embed information. Experimental results show that this technique meets the requirement for real-time application in stream switching application.

Later, Xu *et al.* [72] considered to embed information using DCT coefficients in I-frame and magnitude of motion vectors in P-frame to achieve higher payload. Aly extends Xu *et al.*'s [73] technique by proposing a different information hiding approach aiming to achieve a minimum prediction error and bitstream size overhead. Instead of using magnitude and phase angle, Aly's technique exploits the prediction errors caused by the associated motion vector displacement to determine its suitability for information hiding. In particular, the prediction error is compared to an adaptive threshold. This technique causes low distortion in the video and suppresses bitstream size increment. Recently, Cao *et al.* [74] designed an adaptive and reversible data embedding technique based on motion vectors. Cao *et al.* implemented calibration techniques to recover the inter-macroblocks whose motion vectors are modified for embedding purposes. Deng *et al.* compared the methods proposed by Su *et al.* and Cao *et al.*, and proposed an improved technique for higher detection accuracy [75].

4) *Motion Vector Search Range:* Hierarchical-based motion estimation is adopted in the H.264 standard to support a range of block sizes and quarter-pixel precision for achieving high compression efficiency. For each macroblock, the motion estimation process starts by searching for the best macroblock in the integer-pixel level, then proceeds to the subpixel level around the best integer-pixel position, and finally continues searching at quarter-pixel level around the selected subpixel position to find the best matching point. The information can be embedded by modulating the search points of the motion estimation process according to the mapping rule. In particular, this technique utilizes two nonoverlapping sets of search points (i.e., $\mathbb{M}$ and $\mathbb{N}$) to embed information. A possible arrangement is shown in Fig. 12, where w denotes the bit to be embedded. Experimental results from Zhu *et al.* [76] indicated that no obvious change is observed in terms of bitrate as well as quality of the video. Nonetheless, the change in direction of motion vector inevitably introduce a larger prediction error. However, this error will be handled automatically (i.e., absorbed into the residual signal) and its effect to bitrate is insignificant.

B. Transform Process

Similar to information hiding in still image, luminance DCT coefficients are commonly utilized as the

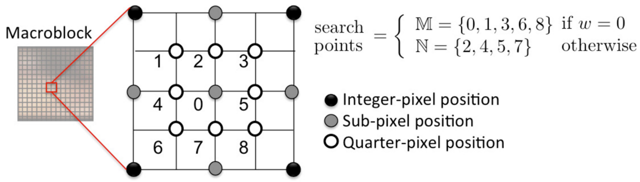


Fig. 12. Quarter pixel search point position for information hiding.

venue to hide information by using bit plane replacement (i.e., odd–even) embedding technique. Ma *et al.* [77] proposed to embed information into the quantized DCT coefficients (luminance) in I-frame. Based on the analysis of the relationship between the DCT coefficients and the distortion incurred in pixel values, several coefficients are paired for data embedding and distortion adjustment purposes. Results show that this method is able to eliminate I-frame distortion drift, achieves higher payload, and causes lower visual distortion. As an extension of Ma's work, Lin *et al.* [78] proposed to embed two bits in the luminance channel of the selected macroblocks. Prediction mode (i.e., I4MB) and selected pixels in this macroblock are defined in their proposed mapping rule to achieve higher payload while maintaining video quality.

Earlier, Huang *et al.* [79] present an algorithm based on communication theory. It embeds a message in the DC coefficient, followed by low-frequency AC coefficients. A similar technique is proposed by Barni *et al.* [80], who define the video content as a video object plane in the video object layer. Barni *et al.*'s technique computes the frequency mask to select a pair of DCT coefficients and divide them into two parts. For the nonzero DCT coefficients part, information is inserted into coefficients of magnitude greater than a predefined threshold level. For the zero DCT coefficients part, the quantization parameter is manipulated to represent a hidden message. Chung *et al.*'s [14] technique applies histogram shifting on DCT coefficients in I-frame and manipulated motion vectors in neighboring macroblocks in P/B frames to realize error concealment. Similarly, Shahid *et al.* [81] proposed to manipulate nonzero DCT coefficients in intra- and inter-frame with different quantization parameters to embed information. In [82], Chen *et al.* [83] exert Watson's visual mask construction and Lin *et al.*'s [78] payload estimation method to realize information hiding using the selected DCT coefficients in I-frame.

Thiesse *et al.* [84]–[86] hide motion vector competition index (MVComp) in the chroma and luma DCT coefficients to reduce the total bitrate in the H.264 video stream. A mapping rule is introduced based on the sum of DCT coefficients S_i to control the bitrate change and minimize the distortion caused by motion prediction at reduced precision. The parity of S_i (coefficient sum) is utilized to represent MVComp by adding h_i to S_i (when necessary) to denote the predictor index $I_i \in \{0, 1\}$ as

$$S_i' = \begin{cases} S_i & \text{if } |S_i| \bmod 2 = I_i, \\ S_i + h_i & \text{Otherwise.} \end{cases} \tag{5}$$

The results show good compromise among bit saving, prediction error propagation in luma texture, and visual quality in chroma aspect.

Meuel *et al.* [87] worked on a similar technique to hide region of interest (ROI) information into the quantized DCT coefficients. ROI information is utilized to represent significant object in still image and it is constructed based on skin pixel (boundary of object in still image)

$$\sqrt{(P_u - \widetilde{P_u})^2 + (P_v - \widetilde{P_v})^2} < d \tag{6}$$

where P_u and P_v are the Cb and Cr (chrominance) components, respectively, $\widetilde{P_u}$ and $\widetilde{P_v}$ are the reference components, and d is the threshold determining if the current pixel is marked as a skin pixel. Its position, width, and height values are embedded into two LSBs of the nonzero DCT coefficients of the current frame. This technique achieves lossless reconstruction, but the results indicate that the frame payload is insufficient to host the entire ROI information.

Similarly, Yin *et al.* [88] proposed to hide information in edge pixels by using edge detection and multidirectional interpolation techniques on residual information. This technique is designed for error concealment application at the decoder in still image. Along the same direction, Yilmaz *et al.* [89] proposed to hide quantized edge information (deduced from neighboring macroblocks) for error concealment purposes. Based on [88] and [89], Kang *et al.* [90] embed the important information of macroblocks, including coding mode(s), reference frame(s), motion vector(s), into the next frame using odd–even embedding method in DCT coefficients. Li *et al.* [91] embed information in DWT coefficients for video watermarking purposes. The scaling coefficients in DWT are utilized to embed low resolution video frame while the watermark information is embedded using wavelet coefficients. Besides that, Wu *et al.* [92], [93] proposed information hiding architecture, design and implementation in still image and video domains. They recursively embed information in each video frame by using modulation and multiplexing techniques selectively in different regions for handling uneven payload.

Instead of modifying nonzero DCT coefficients, Nakajima *et al.* [94] exploit the (zero) run component of nonzero coefficients to embed information in a compressed video. For each block, the position of the last nonzero coefficient (with respect to the zigzag scanning order), denoted by l, is computed. The value $\gamma = \log_2(64 - l)$ then determines the number of bits that can be embedded in the current block. Information is embedded by introducing a nonzero coefficient θ at position $l + \lambda_{10}$, where λ_{10} is the decimal representation of γ bits from the information to be embedded. The sign and magnitude of θ can also be exploited for information hiding purposes.

C. Quantization Process

In Wong *et al.*'s technique, quantization scale of each macroblock (if it is coded) is utilized for information hiding. This method is able to preserve the video bitstream size with low embedding complexity [95]. In another paper, Wong *et al.* [50] maintained quality of the modified video exactly to that of the original host even after data embedding. If “0” is to be embedded, the macroblock is left as it is. Otherwise, the macroblock is manipulated by dividing the quantization scale by a prime number and multiplying each nonzero DCT coefficient by the same prime number.

Shanableh [52] utilize matrix encoding technique to hide information in coded quantization scales and motion vectors of H.264/SVC compressed video. A video transcoding process is applied to allow information embedding in motion vectors using a noniterative procedure regardless of the availability of the original raw video. Matrix encoding is utilized to minimize the number of modifications on quantization scale. Here, the coding structure of H.264/SVC is exploited to increase payload. In particular, quantization scales in both the base and enhancement layer(s) are utilized to embed information. In another article by Shanableh [26], the FMO feature and quantization scale are modulated to embed up to three bits of information per macroblock.

Su *et al.* [96] embeded information in the nonzero DCT coefficients that are representing the prediction residuals. This technique manipulates the selected DCT coefficients by using quantization step based on the just noticeable difference (JND) to determine the amount of information that is allowed to be embedded into each coefficient. Su *et al.* [83] adopted Watson’s perceptual model and implement this technique as a video watermarking scheme.

D. Entropy Coding

Two entropy coding methods, namely CAVLC and CABAC, are available in the H.264 compression standard, and they are also utilized for data embedding purposes. In CAVLC, run-level coding is utilized to compactly represent strings of zeros by referring to the T1s (trailing ones) table to mark the last three ± 1 coefficients [97]. Liao *et al.* utilized the T1s codeword (0–3) to carry information based on the following mapping rule

$$\widetilde{T1s} = \begin{cases} 2, & \text{if } w = 0 \text{ and T1s} \geq 3, \\ 1, & \text{if } w = 1 \text{ and T1s} = 2 \text{ or} \\ & \text{if } w = 1 \text{ and T1s} = 0, \\ 0, & \text{if } w = 0 \text{ and T1s} = 1, \\ \text{unchanged}, & \text{otherwise.} \end{cases} \tag{7}$$

$\widetilde{T1s}$ is the modified T1s codeword and w is the information bit to be embedded. This method is of low complexity and quality degradation caused by data embedding that is imperceptible in the resulting video. At the same time, this technique results in less variation in bit length (i.e., bitstream size) and it is able to execute in real time. A similar approach is taken by Kim *et al.* [98] where sign of the nonzero DCT coefficients and the number of nonzero DCT coefficients in I4MB are modified to embed information. Lu *et al.* [99] considered the run-level pairs in macroblock for video watermarking purpose. In particular, the difference of average value of levels (from run-level pairs in each macroblock) from the original and filtered frames are utilized to encode the watermark information. On the other hand, Mobasseri *et al.* [100] utilized the codeword of unused run-level pairs (i.e., those that never occurred in the video) in CAVLC to embed information. They associate selected run-level pairs with unused ones to represent “0” and “1,” respectively. This algorithm forces the selected pairs in intra-coded macroblock to be the associated pairs depending on the information to be embedded. However, side information is required to mark the originally unused codewords in the VLC table for detecting the embedded information.

Seo *et al.* [101] apply LSB insertion on significant coefficient sig_ctx in context mapping during CABAC process. The LSB of each sig_ctx (absolute value) is manipulated by ± 1 to indicate the embedded bit. In 2011, Wang *et al.* [102] embedded information in LSB of syntax elements (represented by values) during the binarization process in CABAC, which is a process to concatenate all the syntax elements in binary format (i.e., unary binarization) with delimiters. Xu *et al.* [103] manipulated the Kth exponential Golomb code in the binarization scheme to embed information based on code mapping. Both researchers manipulate CABAC for watermarking purposes.

V. General Trends

Information hiding in video is an extension of information hiding in still image due to the similarity in their coding structures and processing frameworks. Fig. 13 shows the timeline for standardization of image and video compression formats. The year at which a component (i.e., entity in a format) is first exploited for information hiding purposes is captured to the best of our knowledge. Note that the list of components is nonexhaustive and it is restricted to the components in compressed video, as well as those in compressed image that are readily applicable to compressed video.

Since the introduction of JPEG image standard in year 1992, numerous researches on information hiding take place. Manipulation on transform coefficients in still image by using LSB is first reported by Cox *et al.* [104] in 1995. This idea is proposed in NEC technical report, followed by a publication on spread spectrum watermarking in DCT domain [36], and a patent is filed in 1999 [105]. DCT coefficients are also considered in JSteg [106], OutGuess [107], and F5 [55] for steganographic purposes from 1999 to 2001. Although quantization table is part of the JPEG standard, it is not exploited for information hiding by Chang *et al.* [108] until 2002. Similarly, the VLC codeword remains unexploited to information hiding until Mobasseri *et al.* [100] proposed the code mapping method in 2005.

Meanwhile, MPEG1, MPEG2, and MPEG4 are introduced in 1993, 1995, and 1998, respectively, which lead to various opportunities for information hiding in the MPEG standards. In 1997, Jordan *et al.* [67] proposed a watermarking technique using motion vectors in the ISO/IEC Joint Technical Committee. Similar to Cox *et al.*’s proposal, Hartung *et al.* [109], and Chae *et al.* [110] hide information into DCT coefficients for video watermarking purposes in 1998 and 1999, respectively. Prior to H.264, there are numerous information hiding methods proposed for MPEG1 or MPEG2 compressed videos, but the hiding venues are restricted to DCT coefficients and motion vectors. A possible

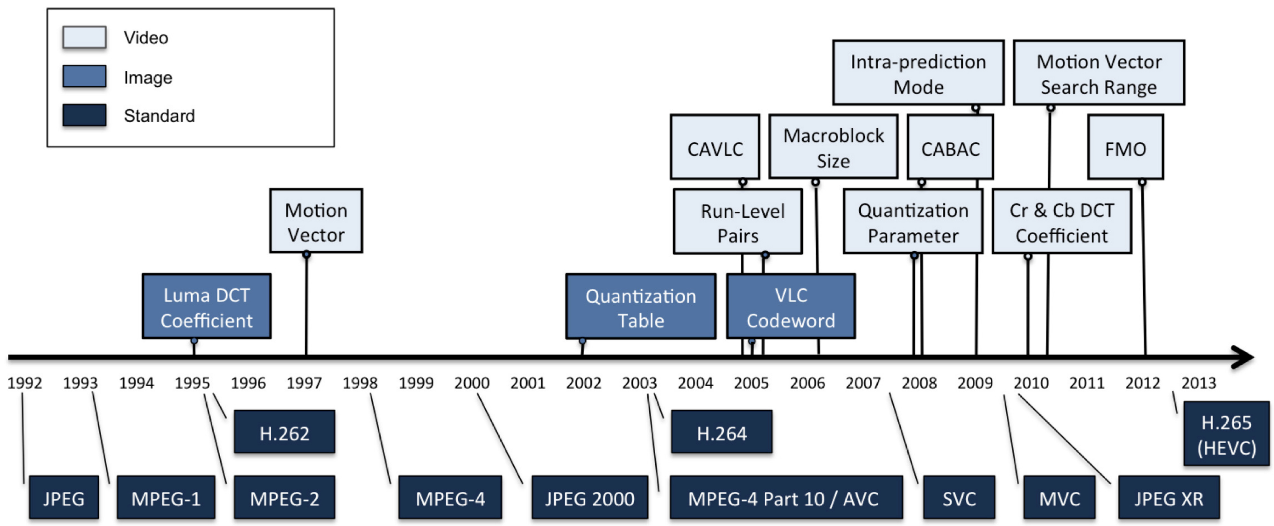


Fig. 13. Timeline of compression standards and first use of their components to realize information hiding.

TABLE I
COMPARISON OF VENUES FOR INFORMATION HIDING IN H.264

Hiding Component (Venue)	Hiding Scheme	Related Process	Found In	Payload	Size Overhead	Impact on Video Quality	Computational Complexity
Chroma & luma DCT coefficient	Spread Spectrum/ Divisibility	Transform	I/P/B Frame, Intra prediction	High	Low	Low	Low
Motion estimated residual value	Bit Plane Replacement	Prediction/ Transform	P/B Frame, Inter prediction	High	Low	Low	Low
Motion vector	Mapping Rule	Prediction	P/B Frame, Inter prediction	Moderate	Moderate	High	Moderate
Quantization parameter	Spread Spectrum/ Divisibility	Quantization	I/P/B Frame, Intra/Inter prediction	Low	High	High	High
Block size selection	Mapping Rule	Prediction	I/P/B Frame, Intra/Inter prediction	Low	Moderate	Low	High
Intra-prediction type selection	Mapping Rule	Prediction	I/P/B Frame, Intra prediction	Low	Moderate	High	Moderate
CAVLC codeword *	Mapping Rule	Entropy Coding	I/P/B Frame, Intra/Inter prediction	High	High	High	Low
CABAC binary syntax element *	Bit Plane Replacement	Entropy Coding	I/P/B Frame, Intra/Inter prediction	High	Low	High	Low

* Quality of the embedded video is severely distorted.

reason is that the main differences between MPEG1 and MPEG2 (e.g., nonlinear quantization scale, interlaced coding mode for a frame) are not suitable for information hiding purposes. The situation changes in 2003 when the MPEG4 Part 10 (H.264/AVC) standard is introduced. Information hiding in MPEG compression standard has then received much attention due to the introduction of various new coding methods/structures in H.264/AVC, which lead to opportunities for information hiding. Between 2003 and 2012, various venues in video compression standards, including macroblock size [62], [63], prediction mode [66], quantization scale [95], zero-run pairs with subqueues decomposition [111], FMO [26], etc., are exploited for information hiding purposes.

We observe from the diagram that the trend of information hiding is related to the release of new image/video compression standard. In the 1990s, the number of publications related to information hiding in JPEG does not increase until some years later after its standardization. Researchers probably need a few years to understand, experiment, design, and implement new information hiding techniques after a new standard is released. However, in the 21st century, the number of publications is increasing rapidly and follows closely to the release of the new image/video standard. The invention of new developing tools (e.g., open source software development kit, compiler, assembly coder) and the ease of communication (e.g., forums, Internet sourcing) could be the possible reasons behind this trend. Researchers are now more equipped with advanced computers and sophisticated devices to conduct experiments efficiently and they are more keen to share their results globally. At the same time, the conducted researches have become more specialized, which can be observed through the trend in survey papers themselves [19], [21]. As such, researches related to information hiding in commonly deployed standards such as JPEG image and H.264 video might be saturated, but we believe that new opportunities will surface for information hiding purposes in the recently finalized High Efficiency Video Coding (HEVC) standard [4].

TABLE II
COMPARISON OF VENUES BASED ON VIDEO CRITERIA AND APPLICATIONS

Hiding Component (Venue)	Video Criteria			Possible Application Scenario
	Alleviation	GOP	Bitrate	
Chroma & luma DCT coefficient	NA	< 10	Moderate	Scenarios requiring high payload (e.g., video compression [86], error concealment [14])
Motion estimated residual value	Moderate	> 10	Moderate	Scenarios requiring high payload (e.g., video compression, error concealment)
Motion vector	Low	> 10	Moderate	Joint application of perceptual encryption and information hiding for administrative purposes [46], [48]
Quantization parameter	NA	NA	Moderate	Scenarios requiring real-time extraction (e.g., authentication [97], content annotation)
Block size selection	Low	> 10	Moderate	Scenarios requiring real-time extraction (e.g., watermarking [35])
Intra-prediction type selection	NA	< 10	High	Scenarios requiring real-time extraction (e.g., watermarking [24])
CAVLC codeword	NA	NA	High	Joint application of perceptual encryption and high payload information hiding for administrative purposes [46], [48]
CABAC binary syntax element	NA	NA	High	Joint application of perceptual encryption and high payload information hiding for administrative purposes [46], [48]

GOP (Group of pictures - number of I-frame interleaves) NA - Not applicable

VI. Discussions and Analysis

Intuitively, among all the information hiding techniques, block size, and intra-prediction type selections offer simple ways to encode information by associating the indices (i.e., states) with groups of bits (i.e., meaning). These techniques maintain coding efficiency with insignificant fluctuation in bitrate, but cause video bitstream size increment in general. On the other hand, hiding information in block size selection provides minimal impact on video quality and bitrate. The current approaches merely divide the block size selection into two groups and offer minimal payload. Hence, a straightforward improvement is to extend the selection groups (to four or eight groups) to encode more information.

The availability of motion vector in large number leads to high payload for information hiding in P and B frames. However, techniques involving motion vector increase the complexity of the video encoding process. This class of techniques requires precise computation to avoid inaccurate motion compensation during the frame reconstruction process in which case the errors may propagate until the next GOP is encountered. On the other hand, transform coefficients can provide arguably the highest payload for information hiding purposes. However, this approach may lead to a noticeable degradation in video quality and a significant bitstream size increment when embedding at high rate.

Modulating the quantization parameter may cause significant degradation in visual quality. Therefore, matrix encoding technique is usually applied to reduce the number of modifications required at the expense of lower payload and higher complexity. On the other hand, realizing information hiding in the entropy coding stage allows information to be embedded with low embed/extract time overhead, but at the expense of possible visual artifacts. Nevertheless, the conventional techniques exploiting CAVLC can be further extended to the latest CABAC to achieve higher payload and minimize the change in video bitrate.

Table I summarizes the performance of each venue to realize information hiding in the H.264 video compression standard. The labels of high, moderate, and low are context dependent. In the case of payload, it is determined by the maximum number of available hiding elements in a macroblock. For example, in each 4:2:0 chroma subsampled macroblock, there are at most 384 DCT coefficients (16×16 in luminance, two 8×8 in chrominance), 32 motion vectors, 1 quantization parameter, 16 decisions on intra-prediction mode, etc. Thus, 384 is the maximum achievable number within a macroblock and it is chosen as the reference value for payload. Consequently, chroma and luma DCT coefficient, motion vector, and quantization parameter are labeled as high, moderate, and low, respectively. Here, we consider a venue (e.g., motion vector) is of moderate payload if its value is approximately $10 \sim 15\%$ of the maximum achievable payload (i.e., 384) among the venues considered. On the other hand, for size overhead, the labels are given based on the bitstream size overhead for a fixed quality measurement (e.g., PSNR = 40 dB) per embedded bit. For example, the quantization parameter has the highest bitstream size overhead because it is part of the rate controller and it affects all the encoding operations in a macroblock while DCT coefficient has negligible impact on the bitstream size. Similarly, for video quality, the labels are provided based on the degradation introduced per embedded bit over a fixed bitrate scenario (e.g., 500 kb/s). Finally, complexity refers to the computational time required to reach the embedding venue, embed the information, and the processes that follow after the modification is made.

It is obvious that there is no perfect solution to achieve high payload, low complexity, high video quality (i.e., low distortion), and low video bitstream size overhead simultaneously. In the future, it is recommended to implement the aforementioned techniques adaptively to offset the shortcoming among each other and to achieve interesting properties, such as complete video quality preservation in the manipulated video [50]. Also, the RDO process can be utilized to determine a suitable technique for each block, macroblock, or frame. Certainly, the computation complexity will be higher than the case of applying only a particular technique. Nonetheless, the increment in computational complexity can be justified by the ever improving computational capabilities and the common trend of including extended instruction sets to handle video encoding/decoding in new generations of CPU.

Although many of the existing methods maintain the H.264 format and hence the manipulated videos can be readily decodable by the standard decoder, visual artifacts and noticeable distortions are inevitable in some approaches, including those that manipulate the binary syntax element on CABAC, etc. Therefore, to ensure video playback at certain quality and extraction of the embedded information, the decoder is modified to partially or losslessly (i.e., reversibly) recover the original video. On the other hand, there are techniques that merely cause imperceptible distortion and hence the modified video is readily decodable by standard decoders with acceptable perceptual quality without the corrective processes.

VII. Recommendation and Further Research Direction

We recommend the following choices of venue for information hiding purposes based on the video criteria recorded in Table II. In particular, for video with lower motion alleviations between frames (e.g., objects are only slightly displaced in two consecutive frames), the encoder tends to select inter-prediction mode (over intra-mode) during the encoding process. Therefore, block size selection and motion vector are two recommended approaches for information hiding in this type of video. On the other hand, since the size of GOP (as well as M-factor, which is the distance between I/P frames) determines the ratio between I and P/B frames, it can also be considered to select a suitable information hiding technique. For size (GOP) > 10 frames, block size selection, motion vector and manipulation on motion estimated residual value are recommended. These schemes provide high payload because the number of inter-frame (P/B) is high in each GOP.

Meanwhile, the video bitrate can also be considered as a criterion in determining the hiding component to be utilized. In particular, for video of higher bitrate, intra-prediction mode selection is recommended because most of the macroblocks are likely to be intra-coded, which leads to higher payload and low distortion. In addition, CAVLC codeword (CABAC binary syntax element) selection is also recommended due to the availability of codewords (bits) in large number. For the completion of the discussion, the possible applications are provided in Table II.

For future work, one may consider to achieve a different objective in information hiding. In particular, most of the existing information hiding methods aim at producing output (i.e., video in this case) with high perceptual quality. However, one can distort the perceptual quality of the video through information hiding to render the output video in an unintelligible form, i.e., perceptual encryption [112]. For a still image, Ong *et al.* [45] and Wong *et al.* [47] proposed quality degradative information hiding approaches in the spatial and frequency domains, respectively. Since there are numerous venues to realize information hiding in a compressed video, various combinations can be considered to achieve quality degradative information hiding. On the other hand, the unification of encryption and information hiding [113], [114] and compression of encrypted content [115], [116] have also received much attention in recent years. However, the ideas remained unexplored in the compressed video domain, except in [112] for unification of encryption and information hiding in compressed video. Therefore, we foresee the development of joint coding framework for compression, encryption, and information hiding in the compressed video domain as a direct extension of the conventional single-objective approaches.

In addition, video compression and motion tracking are two possible applications achievable by using information hiding techniques. There are successful cases such as embedding chrominance information in luminance channel in JPEG compressed image [117], [118], which motivates the same approach in the compressed video domain for achieving higher compression efficiency. On the other hand, since the motion vectors suggested by the compliant encoder do not precisely reflect to the real motion experienced by a macroblock, the actual displacement information is embedded for motion tracking purpose. With the embedded motion information, the actual displacement of each macroblock is readily available for precise and efficient motion tracking.

VIII. Conclusion

In this paper, we surveyed the conventional information hiding methods in the compressed video domain, focusing on the H.264 video compression standard. Commonly considered data representation schemes and the hiding venues were summarized. The general trend of information hiding in the compressed video domain were presented. Then, we categorized the existing information hiding methods based on the venues at which they operate and highlighted their strengths and weaknesses. Video criteria such as motion alleviation, GOP size and bitrate were recommended as guidelines to select appropriate technique for information hiding, and future research directions were suggested.

This survey is limited to the techniques that manipulate the underlying coding structure of H.264 to realize data embedding. The decoding process (e.g., in multibit watermark application [119]) and the detection process (e.g., in zero-bit watermark application [120]) as well as the security issues involved [121], [122] will be investigated as our future work. In addition, we aim at proposing new information hiding methods or consolidating the existing ones for actual application purposes such as video compression, motion tracking, etc. We also aim for exploring new information hiding opportunities in the latest H.265 video compression standard.

References

[1] ISO, *Information Technology—Coding of Moving Pictures and Associated Audio for Digital Storage Media at up to About 1.5 Mbit/s—Part 1: System*, ISO/IEC 11172-1:1993, International Organization for Standardization, Geneva, Switzerland, 1993.

[2] ISO, *Information Technology—Generic Coding of Moving Pictures and Associated Audio Information: Video*, ISO/IEC 13818-2:2000, International Organization for Standardization, Geneva, Switzerland, 2000.

[3] T. Wiegand, G. J. Sullivan, G. Bjontegaard, and A. Luthra, "Overview of the H.264/AVC video coding standard," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 13, no. 7, pp. 560–576, Jul. 2003.

[4] G. J. Sullivan, J. Ohm, W.-J. Han, and T. Wiegand, "Overview of the High Efficiency Video Coding (HEVC) standard," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 22, no. 12, pp. 1649–1668, Dec. 2012.

[5] I. Cox, M. Miller, J. Bloom, J. Fridrich, and T. Kalker, *Digital Watermarking and Steganography*, 2nd ed. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2008, ch. 3.

[6] A. Khan, S. A. Malik, A. Ali, R. Chamlawi, M. Hussain, M. T. Mahmood, and I. Usman, "Intelligent reversible watermarking and authentication: Hiding depth map information for 3D cameras," *Inform. Sci.*, vol. 216, pp. 155–175, Dec. 2012.

[7] S. Bhattacharya, T. Chattopadhyay, and A. Pal, "A survey on different video watermarking techniques and comparative analysis with reference to H.264/AVC," in *Proc. IEEE 10th Int. Symp. Consum. Electron.*, Jun. 2006, pp. 1–6.

[8] A. Alattar, E. Lin, and M. Celik, "Digital watermarking of low bit-rate advanced simple profile MPEG-4 compressed video," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 13, no. 8, pp. 787–800, Aug. 2003.

[9] M. Stamm, W. Lin, and K. Liu, "Temporal forensics and anti-forensics for motion compensated video," *IEEE Trans. Inform. Forensics Security*, vol. 7, no. 4, pp. 1315–1329, Aug. 2012.

[10] O. Cetin and A. T. Ozcerit, "A new steganography algorithm based on color histograms for data embedding into raw video streams," *Comput. Security*, vol. 28, no. 7, pp. 670–682, Oct. 2009.

[11] A. Cheddad, J. Condell, K. Curran, and P. M. Kevitt, "Digital image steganography: Survey and analysis of current methods," *Signal Process.*, vol. 90, no. 3, pp. 727–752, Mar. 2010.

[12] I. Cox, M. Miller, J. Bloom, J. Fridrich, and T. Kalker, *Digital Watermarking and Steganography*, 2nd ed. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2008.

[13] C. Adsumilli, S. Mitra, T. Oh, and Y. Kim, "Error concealment in video commun. by informed watermarking," in *Proc. Adv. Image Video Technol.*, vol. 4319. 2006, pp. 1094–1102.

[14] K.-L. Chung, Y.-H. Huang, P.-C. Chang, and H.-Y. Liao, "Reversible data hiding-based approach for intra-frame error concealment in H.264/AVC," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 20, no. 11, pp. 1643–1647, Nov. 2010.

[15] D. Stanescu, M. Stratulat, B. Ciubotaru, D. Chiciudean, R. Cioarga, and M. Micea, "Embedding data in video stream using steganography," in *Proc. 4th Int. Symp. Appl. Comput. Intell. Inform.*, May 2007, pp. 241–244.

[16] D. Mukherjee, J. J. Chae, and S. Mitra, "A source and channel coding approach to data hiding with application to hiding speech in video," in *Proc. Int. Conf. Image Process.*, vol. 1. Oct. 1998, pp. 348–352.

[17] J. K. Su, F. Hartung, and B. Girod, "Digital watermarking of text, image, and video documents," *Comput. Graph.*, vol. 22, no. 6, pp. 687–695, Dec. 1998.

[18] F. Petitcolas, R. Anderson, and M. Kuhn, "Information hiding–a survey," *Proc. IEEE*, vol. 87, no. 7, pp. 1062–1078, Jul. 1999.

[19] H. Kayarkar and S. Sanyal, "A survey on various data hiding techniques and their comparative analysis," *ACTA Tech. Corviniensis Cryptography Security*, vol. 5, pp. 35–40, Jul.-Sep. 2012.

[20] E. Lin, A. Eskicioglu, R. Lagendijk, and E. Delp, "Advances in digital video content protection," *Proc. IEEE*, vol. 93, no. 1, pp. 171–183, Jan. 2005.

[21] T. Stutz and A. Uhl, "A survey of H.264 AVC/SVC encryption," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 22, no. 3, pp. 325–339, Mar. 2012.

[22] ISO, *Information Technology—Coding of Audio–Visual Objects—Part 1: Systems*, ISO/IEC 14496-1:2010, International Organization for Standardization, Geneva, Switzerland, 2010.

[23] G. Yang, J. Li, Y. He, and Z. Kang, "An information hiding algorithm based on intra-prediction modes and matrix coding for H.264/AVC video stream," *AEU Int. J. Electron. Commun.*, vol. 65, no. 4, pp. 331–337, Apr. 2011.

[24] T. Wedi, "Adaptive interpolation filter for motion compensated prediction," in *Proc. Int. Conf. Image Process.*, vol. 2. 2002, pp. 509–512.

[25] P. List, A. Joch, J. Lainema, G. Bjontegaard, and M. Karczewicz, "Adaptive deblocking filter," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 13, no. 7, pp. 614–619, Jul. 2003.

[26] T. Shanableh, "Data hiding in MPEG video files using multivariate regression and flexible macroblock ordering," *IEEE Trans. Inform. Forensics Security*, vol. 7, no. 2, pp. 455–464, Apr. 2012.

[27] G. J. Sullivan, T. Wiegand, and P. Corporation, "Rate-distortion optimization for video compression," *IEEE Signal Process. Mag.*, vol. 15, no. 6, pp. 74–90, Nov. 1998.

[28] G. Bjontegaard and K. Lillevold, *Context-Adaptive VLC (CAVLC) Coding of Coefficients*, JVT-C028, 3rd Meeting, Fairfax, Virginia, USA, May 2002.

[29] D. Marpe, H. Schwarz, and T. Wiegand, "Context-based adaptive binary arithmetic coding in the H.264/AVC video compression standard," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 13, no. 7, pp. 620–636, Jul. 2003.

[30] J. Mielikainen, "LSB matching revisited," *IEEE Signal Process. Lett.*, vol. 13, no. 5, pp. 285–287, May 2006.

[31] X. Li, B. Yang, D. Cheng, and T. Zeng, "A generalization of LSB matching," *IEEE Signal Process. Lett.*, vol. 16, no. 2, pp. 69–72, Feb. 2009.

[32] R.-Z. Wang, C.-F. Lin, and J.-C. Lin, "Image hiding by optimal LSB substitution and genetic algorithm," *Pattern Recognit.*, vol. 34, no. 3, pp. 671–683, Mar. 2001.

[33] C.-K. Chan and L. Cheng, "Hiding data in images by simple LSB substitution," *Pattern Recognit.*, vol. 37, no. 3, pp. 469–474, Mar. 2004.

[34] S. Kapotas and A. Skodras, "Real time data hiding by exploiting the IPCM macroblocks in H.264/AVC streams," *J. Real-Time Image Process.*, vol. 4, pp. 33–41, Oct. 2009.

[35] Y.-H. Yu, C.-C. Chang, and Y.-C. Hu, "Hiding secret data in images via predictive coding," *Pattern Recognit.*, vol. 38, no. 5, pp. 691–705, Sep. 2005.

[36] I. J. Cox, J. Kilian, F. T. Leighton, and T. Shamoon, "Secure spread spectrum watermarking for multimedia," *IEEE Trans. Image Process.*, vol. 6, no. 12, pp. 1673–1687, Dec. 1997.

[37] L. M. Marvel, C. G. Boncelet, Jr., and C. T. Retter, "Spread spectrum image steganography," *IEEE Trans. Image Process.*, vol. 8, no. 8, pp. 1075–1083, Aug. 1999.

[38] U. Budhia, D. Kundur, and T. Zourntos, "Digital video steganalysis exploiting statistical visibility in the temporal domain," *IEEE Trans. Inform. Forensics Security*, vol. 1, no. 4, pp. 502–516, Dec. 2006.

[39] V. P., G. Doerr, and P. K. Bora, "Assessing motion-coherency in video watermarking," in *Proc. 8th Workshop Multimedia Security*, 2006, pp. 114–119.

[40] A. Valizadeh and Z. Wang, "Correlation-and-bit-aware spread spectrum embedding for data hiding," *IEEE Trans. Inform. Forensics Security*, vol. 6, no. 2, pp. 267–282, Jun. 2011.

[41] Z. Ni, Y.-Q. Shi, N. Ansari, and W. Su, "Reversible data hiding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 16, no. 3, pp. 354–362, Mar. 2006.

[42] Z.-H. Wang, C.-F. Lee, and C.-Y. Chang, "Histogram-shifting-imitated reversible data hiding," *J. Syst. Software*, vol. 86, no. 2, pp. 315–323, Feb. 2013.

[43] C. De Vleeschouwer, J.-F. Delaigle, and B. Macq, "Circular interpretation of bijective transformations in lossless watermarking for media asset management," *IEEE Trans. Multimedia*, vol. 5, no. 1, pp. 97–105, Mar. 2003.

[44] Z. Zhao, H. Luo, Z.-M. Lu, and J.-S. Pan, "Reversible data hiding based on multilevel histogram modification and sequential recovery," *AEU Int. J. Electron. Commun.*, vol. 65, no. 10, pp. 814–826, Oct. 2011.

[45] S. Ong, K. Wong, and K. Tanaka, "Reversible data embedding using reflective blocks with scalable visual quality degradation," in *Proc. 8th Int. Conf. Intell. Inform. Hiding Multimedia Signal Process.*, Jul. 2012, pp. 363–366.

[46] S. Kapotas, E. Varsaki, and A. Skodras, "Data hiding in H.264 encoded video sequences," in *Proc. IEEE 9th Workshop Multimedia Signal Process.*, Oct. 2007, pp. 373–376.

[47] K. Wong and K. Tanaka, "DCT based scalable scrambling method with reversible data hiding functionality," in *Proc. 4th Int. Symp. Commun. Control Signal Process.*, Mar. 2010, pp. 1–4.

[48] D. Coltuc and J.-M. Chassery, "High capacity reversible watermarking," in *Proc. IEEE Int. Conf. Image Process.*, Oct. 2006, pp. 2565–2568.

[49] J. Fridrich, M. Goljan, and R. Du, "Invertible authentication watermark for JPEG images," in *Proc. Int. Conf. Inform. Technol. Coding Comput.*, Apr. 2001, pp. 223–227.

[50] K. S. Wong, K. Tanaka, K. Takagi, and Y. Nakajima, "Complete video quality—Preserving data hiding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 19, no. 10, pp. 1499–1512, Oct. 2009.

[51] D. Coltuc, "Improved capacity reversible watermarking," in *Proc. IEEE Int. Conf. Image Process.*, vol. 3. Oct. 2007, pp. 249–252.

[52] T. Shanableh, "Matrix encoding for data hiding using multilayer video coding and transcoding solutions," *Signal Process. Image Commun.*, vol. 27, pp. 1025–1034, Oct. 2012.

[53] R. Crandall. (1998, Dec.) Some notes on steganography [Online]. Available: http://www.di.unisa.it/%7eads/corso-security/www/CORSO-0203/steganografia/LINKS%20LOCALI/matrix-encoding.pdf

[54] J. Fridrich and D. Soukal, "Matrix embedding for large payloads," *IEEE Trans. Inform. Forensics Security*, vol. 1, no. 3, pp. 390–395, Sep. 2006.

[55] A. Westfeld, "F5—A steganographic algorithm. High capacity despite better steganalysis," in *Proc. Inform. Hiding*, vol. 2137. 2001, pp. 289–302.

[56] L. Fan, T. Gao, Q. Yang, and Y. Cao, "An extended matrix encoding algorithm for steganography of high embedding efficiency," *Comput. Electr. Eng.*, vol. 37, no. 6, pp. 973–981, Nov. 2011.

[57] Y. S. Choi and H. J. Kim, "Improving the modified matrix encoding on steganography method," in *Proc. 5th Int. Conf. Inform. Assurance Security*, vol. 1. Aug. 2009, pp. 205–208.

[58] K. Wong and K. Tanaka, "Improvement of StegErmelc with hybrid recursive matrix encoding," in *Proc. Int. Symp. Intell. Signal Process. Commun. Syst.*, Feb. 2009, pp. 1–4.

[59] M. Karim and K. Wong, "Re-conceptualization of applications achieved by data hiding," in *Proc. Int. Symp. Intell. Signal Process. Commun. Syst. (ISPACS)*, Nov. 2012, pp. 447–451.

[60] J.-M. Guo and J.-J. Tsai, "Reversible data hiding in low complexity and high quality compression scheme," *Digital Signal Process.*, vol. 22, no. 5, pp. 776–785, Sep. 2012.

[61] M.-N. Wu, C.-C. Lin, and C.-C. Chang, "An embedding technique based upon block prediction," *J. Syst. Softw.*, vol. 81, no. 9, pp. 1505–1516, Sep. 2008.

[62] Y. Hu, C. Zhang, and Y. Su, "Information hiding based on intra prediction modes for H.264/AVC," in *Proc. IEEE Int. Conf. Multimedia Expo*, Jul. 2007, pp. 1231–1234.

[63] H. Zhu, R. Wang, D. Xu, and X. Zhou, "Information hiding algorithm for H.264 based on the predition difference of intra 4×4," in *Proc. 3rd Int. Congr. Image Signal Process.*, vol. 1. Oct. 2010, pp. 487–490.

[64] D.-W. Kim, Y.-G. Choi, H.-S. Kim, J.-S. Yoo, H.-J. Choi, and Y.-H. Seo, "The problems in digital watermarking into intra-frames of H.264/AVC," *Image Vision Comput.*, vol. 28, no. 8, pp. 1220–1228, Aug. 2010.

[65] D. Xu, R. Wang, and J. Wang, "Prediction mode modulated data-hiding algorithm for H.264/AVC," *J. Real-Time Image Process.*, vol. 5, no. 3, pp. 1–10, Aug. 2010.

[66] S. Kapotas and A. Skodras, "A new data hiding scheme for scene change detection in H.264 encoded video sequences," in *Proc. IEEE Int. Conf. Multimedia Expo*, Apr. 2008, pp. 277–280.

[67] F. Jordan, M. Kutter, and T. Ebrahimi. (1997, Jul.). Proposal of a watermarking technique for hiding/retrieving data in compressed and decompressed video [Online]. Available: http://www.alpvision.com/pdf/mvt.pdf

[68] J. Zhang, J. Li, and L. Zhang, "Video watermark technique in motion vector," in *Proc. XIV Brazilian Symp. Comput. Graph. Image Process.*, Oct. 2001, pp. 179–182.

[69] Y. Dai, L. Zhang, and Y. Yang, "A new method of MPEG video watermarking technology," in *Proc. Int. Conf. Commun. Technol.*, vol. 2. Apr. 2003, pp. 1845–1847.

[70] Y. Su, C. Zhang, and C. Zhang, "A video steganalytic algorithm against motion-vector-based steganography," *Signal Process.*, vol. 91, no. 8, pp. 1901–1909, Aug. 2011.

[71] Y. Guo and F. Pan, "Information hiding for H.264 in video stream switching application," in *Proc. IEEE Int. Conf. Inform. Theory Inform. Security*, Dec. 2010, pp. 419–421.

[72] C. Xu, X. Ping, and T. Zhang, "Steganography in compressed video stream," in *Proc. 1st Int. Conf. Innovative Comput. Inform. Control*, vol. 1. Sep. 2006, pp. 269–272.

[73] H. Aly, "Data hiding in motion vectors of compressed video based on their associated prediction error," *IEEE Trans. Inform. Forensics Security*, vol. 6, no. 1, pp. 14–18, Mar. 2011.

[74] Y. Cao, X. Zhao, and D. Feng, "Video steganalysis exploiting motion vector reversion-based features," *IEEE Signal Process. Lett.*, vol. 19, no. 1, pp. 35–38, Jan. 2012.

[75] Y. Deng, Y. Wu, H. Duan, and L. Zhou, "Digital video steganalysis based on motion vector statistical characteristics," *Optik Int. J. Light Electron Optics*, vol. 124, pp. 1705–1710, Jul. 2013.

[76] H. Zhu, R. Wang, and D. Xu, "Information hiding algorithm for H.264 based on the motion estimation of quarter-pixel," in *Proc. 2nd Int. Conf. Future Comput. Commun.*, vol. 1. May. 2010, pp. 423–427.

[77] X. Ma, Z. Li, H. Tu, and B. Zhang, "A data hiding algorithm for H.264/AVC video streams without intra-frame distortion drift," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 20, no. 10, pp. 1320–1330, Oct. 2010.

[78] T.-J. Lin, K.-L. Chung, P.-C. Chang, Y.-H. Huang, H.-Y. M. Liao, and C.-Y. Fang, "An improved DCT-based perturbation scheme for high capacity data hiding in H.264/AVC intra frames," *J. Syst. Software*, vol. 86, pp. 604–614, Mar. 2013.

[79] J. Huang and Y. Q. Shi, "Reliable information bit hiding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 12, no. 10, pp. 916–920, Oct. 2002.

[80] M. Barni, F. Bartolini, and N. Checcacci, "Watermarking of MPEG-4 video objects," *IEEE Trans. Multimedia*, vol. 7, no. 1, pp. 23–32, Feb. 2005.

[81] Z. Shahid, M. Chaumont, and W. Puech, "Considering the reconstruction loop for data hiding of intra- and inter-frames of H.264/AVC," *Signal Image Video Process.*, vol. 7, no. 1, pp. 75–93, Jan. 2013.

[82] Q. Chen, H. Maitre, and Q. ping Deng, "Reliable information embedding for image/video in the presence of lossy compression," *Signal Process. Image Commun.*, vol. 27, no. 1, pp. 66–74, Jan. 2012.

[83] A. B. Watson, "DCT quantization matrices visually optimized for individual image," in *SPIE Human Vision, Visual Process. and Digital Display*, vol. 1913-14, Sep. 1993, pp. 202–216.

[84] J.-M. Thiesse, J. Jung, and M. Antonini, "Data hiding of intra prediction information in chroma samples for video compression," in *Proc. IEEE 17th Int. Conf. Image Process.*, Sep. 2010, pp. 2861–2864.

[85] J.-M. Thiesse, J. Jung, and M. Antonini, "Data hiding of motion information in chroma and luma samples for video compression," in *Proc. IEEE Int. Workshop Multimedia Signal Process.*, Oct. 2010, pp. 217–221.

[86] J.-M. Thiesse, J. Jung, and M. Antonini, "Rate distortion data hiding of motion vector competition information in chroma and luma samples for video compression," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 21, no. 6, pp. 729–741, Jun. 2011.

[87] P. Meuel, M. Chaumont, and W. Puech, "Data hiding in H.264 video for lossless reconstruction of region of interest," in *Proc. 15th Eur. Signal Process. Conf. (EUSIPCO)*, Sep. 2007, pp. 2301–2305.

[88] P. Yin, B. Liu, and H. Yu, "Error concealment using data hiding," in *Proc. IEEE Int. Conf. Acoust. Speech Signal Process.*, vol. 3. May 2001, pp. 1453–1456.

[89] A. Yilmaz and A. Alatan, "Error concealment of video sequences by data hiding," in *Proc. Int. Conf. Image Process.*, vol. 2. Sep. 2003, pp. 679–682.

[90] L.-W. Kang and J.-J. Leou, "An error resilient coding scheme for H.264/AVC video transmission based on data embedding," *J. Visual Commun. Image Representation*, vol. 16, no. 1, pp. 93–114, Feb. 2005.

[91] G. Li, Y. Ito, X. Yu, N. Nitta, and N. Babaguchi, "Recoverable privacy protection for video content distribution," *EURASIP J. on Inform. Security*, vol. 2009, pp. 4:1–4:11, Jan. 2010.

[92] M. Wu and B. Liu, "Data hiding in image and video I—Fundamental issues and solutions," *IEEE Trans. Image Process.*, vol. 12, no. 6, pp. 685–695, Jun. 2003.

[93] M. Wu, H. Yu, and B. Liu, "Data hiding in image and video II—Designs and applications," *IEEE Trans. Image Process.*, vol. 12, no. 6, pp. 696–705, Jun. 2003.

[94] K. Nakajima, K. Tanaka, T. Matsuoka, and Y. Nakajima, "Rewritable data embedding on MPEG coded data domain," in *Proc. IEEE Int. Conf. Multimedia Expo*, Jul. 2005, pp. 682–685.

[95] K. Wong and K. Tanaka, "A data hiding method using Mquant in MPEG domain," *The J. of the Inst. of Image Electron. Engineers of Japan*, vol. 37, no. 3, pp. 256–267, 2008.

[96] P.-C. Su, C.-S. Wu, I.-F. Chen, C.-Y. Wu, and Y.-C. Wu, "A practical design of digital video watermarking in H.264/AVC for content authentication," *Signal Process. Image Commun.*, vol. 26, pp. 413–426, Oct. 2011.

[97] K. Liao, S. Lian, Z. Guo, and J. Wang, "Efficient information hiding in H.264/AVC video coding," *Telecomm. Syst.*, vol. 49, no. 2, pp. 261–269, Jun. 2010.

[98] S. Kim, S. Kim, Y. Hong, and C. Won, "Data hiding on H.264/AVC compressed video," in *Proc. Image Anal. Recognit.*, vol. 4633. 2007, pp. 698–707.

[99] C.-S. Lu, J.-R. Chen, and K.-C. Fan, "Real-time frame-dependent video watermarking in VLC domain," *Signal Process. Image Commun.*, vol. 20, no. 7, pp. 624–642, Aug. 2005.

[100] B. G. Mobasseri and M. P. Marcinak, "Watermarking of MPEG-2 video in compressed domain using VLC mapping," in *Proc. 7th Workshop Multimedia Security*, May 2005, pp. 91–94.

[101] Y.-H. Seo, H.-J. Choi, C.-Y. Lee, and D.-W. Kim, "Low-complexity watermarking based on entropy coding in H.264/AVC," *IEICE Trans. Fundamentals Electron. Commun. Comput. Sci.*, vol. E91-A, no. 8, pp. 2130–2137, Aug. 2008.

[102] R. Wang, L. Hu, and D. Wu, "A watermarking algorithm based on the CABAC entropy coding for H.264/AVC," *J. Comput. Inform. Syst.*, vol. 7, no. 6, pp. 2132–2141, Jun. 2011.

[103] D. Xu and R. Wang, "Watermarking in H.264/AVC compressed domain using Exp-Golomb code words mapping," *Optical Eng.*, vol. 50, no. 9, pp. 097402-1–097402-11, Sep. 2011.

[104] I. J. Cox, J. Kilian, T. Leighton, and T. Shamoon, *Secure Spread Spectrum Watermarking for Multimedia*, NEC Research, Inst. Tech. Rep., 1995.

[105] L. N. Ingemar J. Cox, P. J. N. Joseph, J. Kilian, and P. N. Talal G. Shamoon, "Secure spread spectrum watermarking for multimedia data," U.S. Patent 5 930 369, Jul. 27, 1999.

[106] M. Wu, Z. Zhu, and S. Jin, "A new steganalytic algorithm for detecting JSteg." in *Proc. ICCNMC*, vol. 3619. Sep. 2005, pp. 1073–1082.

[107] N. Provos, "Defending against statistical steganalysis," in *Proc. 10th Conf. USENIX Security Symp.*, Feb. 2001, pp. 323–335.

[108] C.-C. Chang, T.-S. Chen, and L.-Z. Chung, "A steganographic method based upon JPEG and quantization table modification," *Inform. Sci.*, vol. 141, no. 1–2, pp. 123–138, Mar. 2002.

[109] F. Hartung and B. Girod, "Watermarking of uncompressed and compressed video," *Signal Process.*, vol. 66, no. 3, pp. 283–301, May 1998.

[110] J. J. Chae and B. S. Manjunath, "Data hiding in video," in *Proc. Int. Conf. Image Process.*, vol. 1. 1999, pp. 311–315.

[111] K. Wong, S. Ong, and K. Tanaka, "Improvement of carrier capacity for scalable scrambling method with reversible information insertion functionality," in *Proc. IEEE Int. Conf. Signal Image Process. Applicat.*, Nov. 2011, pp. 312–317.

[112] S. Lian, Z. Liu, Z. Ren, and H. Wang, "Commutative encryption and watermarking in video compression," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 17, no. 6, pp. 774–778, Jun. 2007.

[113] X. Zhang, "Separable reversible data hiding in encrypted image," *IEEE Trans. Inform. Forensics Security*, vol. 7, no. 2, pp. 826–832, Apr. 2012.

[114] M. Cancellaro, F. Battisti, M. Carli, G. Boato, F. G. B. De Natale, and A. Neri, "A joint digital watermarking and encryption method," *Proc. of Security, Forensics, Steganography and Watermarking of Multimedia Contents X*, vol. 6819, pp. 68191C-1–68191C-10, Jan. 2008.

[115] X. Zhang, "Lossy compression and iterative reconstruction for encrypted image," *IEEE Trans. Inform. Forensics Security*, vol. 6, no. 1, pp. 53–58, Mar. 2011.

[116] W. Liu, W. Zeng, L. Dong, and Q. Yao, "Efficient compression of encrypted grayscale images," *IEEE Trans. Image Process.*, vol. 19, no. 4, pp. 1097–1102, Apr. 2010.

[117] M. Chaumont and W. Puech, "A DCT-based data-hiding method to embed the color information in a jpeg grey level image," *Proc. Eur. Signal Process. Conf.*, Sep. 2006.

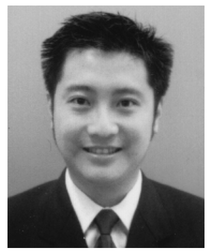
[118] P. Campisi, D. Kundur, D. Hatzinakos, and A. Neri, "Compressive data hiding: An unconventional approach for improved color image coding," *EURASIP J. Adv. Signal Process.*, vol. 2002, no. 2, pp. 152–163, Feb. 2002.

[119] T. Furon, "A constructive and unifying framework for zero-bit watermarking," *IEEE Trans. Inform. Forensics Security*, vol. 2, no. 2, pp. 149–163, Jun. 2007.

[120] A. Tefas, and I. Pitas, "Multi-bit image watermarking robust to geometric distortions," in *Proc. Int. Conf. Image Process.*, vol. 3. Sep. 2000, pp. 710–713.

[121] C. Cachin, "An information-theoretic model for steganography," *Inform. Comput.*, vol. 192, no. 1, pp. 41–56, 2004.

[122] F. Cayre, C. Fontaine, and T. Furon, "Watermarking security: Theory and practice," *IEEE Trans. Signal Process.*, vol. 53, no. 10, pp. 3976–3987, Oct. 2005.



Yiqi Tew (S'08) received the B.Eng. (Hons.) electronics degree in computers from Multimedia University, Cyberjaya, Malaysia, in 2008 and the master's degree in computer science from National University of Malaysia, Selangor, Malaysia, in 2011. He is pursuing the Ph.D. degree at the Faculty of Computer Science and Information Technology, University of Malaya, Kuala Lumpur, Malaysia, under the supervision of Dr. KokSheik Wong.

His research interests include information hiding in compressed video and their applications. He is focussing on the fundamental research related to advanced video coding in MPEG4/H.264 and HEVC standards.

KokSheik Wong (M'10) received the B.S. and M.S. degrees in both computer science and mathematics from Utah State University, Logan, UT, USA, in 2002 and 2005, respectively. In 2009, he received the Doctor of Engineering degree from Shinshu University, Shinshu, Japan, under the scholarship of Monbukagakusho.

In 2010, he joined the Faculty of Computer Science and Information Technology, University of Malaya, Kuala Lumpur, Malaysia, where he is a Senior Lecturer. He is also a member of the Centre of Image and Signal Processing. His research interests include information hiding, steganography, watermarking, multimedia perceptual encryption, multimedia signal processing, and their applications.