Postprint: A Compressed-Domain Digital Video Watermark Embedding and Extraction Method Combining Spatio-Temporal Feature Analysis with Random Keys

Authors: Yicheng Qiu, Xue Feng, Tang Jinglei

Date: 2018-07-09T00:00:00+00:00

Abstract

To address the issue that existing watermark embedding and extraction algorithms are relatively sensitive to filtering, compression, and noise conditions, this paper proposes a robust digital video watermark embedding and extraction method in the compressed domain based on spatio-temporal features. The proposed framework comprises a public key and a private key to thwart self-collusion attacks. The algorithm conducts spatio-temporal analysis of the video and extracts the public key from the spatio-temporal features of the compressed video, which is intrinsically robust. First, a random key is employed to select candidate blocks from a pre-selected block set, thereby ensuring the security of the watermarking framework. Subsequently, suitable 4$×$4 sub-blocks for watermark embedding are selected based on the spatio-temporal features of the compressed video. Finally, watermark bits are embedded utilizing non-zero quantized coefficients. This watermarking framework permits a limited increase in video bit rate while reducing computational overhead. Experimental results indicate that, compared with several other benchmark methods, the proposed method exhibits strong robustness and security.

Full Text

Preamble

Compression Domain Digital Video Watermark Embedding and Extraction Based on Spatiotemporal Features and Random Key

Qiu Yicheng[1], *Xue Feng*[1], *Tang Jinglei*[2]
(1. College of Information & Business, Zhongyuan Institute of Technology, Zhengzhou, Henan 451191, China;

2. School of Information Engineering, Northwest A&F University, Yangling, Shaanxi 712100, China)

Abstract

Existing watermark embedding and extraction algorithms exhibit sensitivity to filtering, compression, and noise conditions. To address these limitations, this paper proposes a robust compression-domain digital video watermarking method based on spatiotemporal features. The proposed framework employs a public key and a private key to prevent self-collusion attacks. The algorithm performs spatiotemporal analysis of video content and extracts the public key from the spatiotemporal features of compressed video, which provides inherent robustness. First, a random key is used to select candidate blocks from a preselected block set to ensure the security of the watermarking framework. Then, suitable 4$×$4 sub-blocks for watermark embedding are selected based on the spatiotemporal features of the compressed video. Finally, watermark bits are embedded using non-zero quantized coefficients. This watermarking framework allows only a limited increase in video bit rate while reducing computational overhead. Experimental results demonstrate that the proposed method achieves stronger robustness and security compared to several alternative approaches.

Keywords: digital watermarking; spatiotemporal features; robustness; compressed video; random key; non-zero quantized coefficients

0 Introduction

Digital watermarking serves as an effective means for multimedia copyright protection, and its efficiency and security have attracted widespread attention. Since digital videos are stored and transmitted in compressed formats, researchers have increasingly focused on performing watermarking operations directly in the compression domain [1-4]. Watermarking in the compressed domain avoids the need for complete decoding and re-encoding of video content.

High-Efficiency Video Coding (HEVC) represents the current state-of-the-art video compression standard, serving as an improved successor to H.264/AVC. HEVC focuses on supporting higher video resolutions through parallel processing architectures, enabling efficient compression while supporting customized 4K content and multi-view coding. However, watermarking techniques for HEVC-compressed video remain in the early stages of research [5,6].

Several approaches have been proposed for compressed-domain watermarking. A blind extraction algorithm was presented in [7] that embeds watermarks in compressed video frames, selecting appropriate blocks based on prediction mode results and the number of non-zero quantized coefficients (NNZ) in 4$×$4 blocks, where watermark bits are hidden by converting non-zero coefficients to zero. Reference [8] designed a data hiding framework utilizing the concept of forbidden zones for data embedding region selection. In [9], watermark embedding in I-frames required modifying the parity of coefficient signs and the magnitude of mid-frequency coefficients. Reference [10] embedded watermarks into non-zero coefficients of I-frames in the compressed domain to maintain good visual quality while limiting bit rate increase. A non-blind H.264/CAVLC structure-preserving substitution watermarking algorithm was proposed in [11], which performed bit substitution in motion vectors of non-reference frames. Reference [12] introduced a non-blind watermarking algorithm for embedding in both I-frames and P-frames. Meanwhile, [13] utilized spread spectrum and Watson’ s visual model for coefficient selection. However, these methods fail to maintain high quality and robustness under filtering, compression, and noise conditions.

To address these challenges, this paper proposes a novel video watermarking method for HEVC that selects watermark embedding blocks based on the spatiotemporal features of compressed video, thereby minimizing synchronization errors. The watermark embedding and extraction algorithms are inherently robust. The proposed framework comprises a public key and a private key to prevent self-collusion attacks, with the public key extracted from the spatiotemporal features of compressed video. A random key selects candidate blocks from a pre-selected block set to ensure framework security, where the block set is extracted based on spatiotemporal features. By embedding watermark bits using only non-zero quantized coefficients, the framework permits only limited video bit rate increases. The compression-domain feature extraction for public key generation, block selection, and watermark embedding reduces computational overhead. Experimental results using the JM17.2 reference software validate the effectiveness of the proposed method.

---

1 Spatiotemporal Feature Analysis and Key Extraction

This section presents a robust watermarking framework for HEVC-encoded video. Candidate blocks are first selected in I-frames for watermark embedding in HEVC-compressed video, with these selected regions referred to as embedding areas. To improve robustness against content-preserving operations while ensuring sensitivity to malicious tampering, the proposed method performs spatiotemporal feature-based analysis of H.264 stream semantics to identify suitable regions for watermark embedding. Subsequently, the watermark embedding and extraction algorithms are presented, along with procedures for public key extraction and embedding threshold evaluation.

1.1 Spatiotemporal Feature Analysis

Due to the characteristics of human visual perception, the human eye is more sensitive to noise in flat motion regions than in complex motion regions and video edge areas. Consequently, regions with complex texture are more suitable for watermark embedding. The efficiency of a watermarking framework depends on the selection of embedding regions. To identify appropriate embedding areas, the proposed method first selects suitable I-frame video images for watermark embedding. Since I-frames contain more extensive information than P and B frames, processing is performed on I-frames. Any manipulation of I-frames by an attacker would result in significant degradation of video visual quality [14].

In HEVC-encoded video I-frames, a block is termed a smooth mode block if it possesses prediction modes of size 32$×32,32×16,16×32,or16×$16. Blocks with these prediction modes cannot be used for watermark embedding as they correspond to smooth regions in a frame, where embedding watermarks would cause visible artifacts. Additionally, due to embedding distortion, blocks in smooth regions tend to transform into different prediction mode blocks, increasing synchronization failures during watermark extraction at the decoder. Therefore, the proposed framework does not employ smooth mode blocks for watermark embedding.

Statistical analysis of the spatial motion characteristics of specific blocks can be performed using the number of non-zero quantized residuals (NNZ) and the number of non-zero quantized AC coefficients in H.264-encoded video. Statistical results demonstrate that 4$×4sub − blockscontainingmorenon − zerocoefficientsexhibitstrongerspatialmotionfeatures.Toverifythisresult, severalwatermark−freevideosequenceswerere−encodedtoobtaintheintra−predictionmodechangeratesin16×16and4×4sub−blockswithdifferentNNZvalues.Figure1[Figure1 : seeoriginalpaper]illustratesthechangesinnon−zeroquantizedresidualsafterre − encodingseveralsequenceswithQP = 18andQP = 28.Thefigureindicatesthat4×$4 prediction modes with fewer NNZ values are more susceptible to change.

The above spatial characteristic analysis reveals that NNZ values vary across different video sequences, and 4$×$4 prediction modes with fewer NNZ values are more prone to variation. Therefore, an adaptive threshold ω must be established when selecting suitable 4$×$4 sub-blocks. To determine the ω value, the distribution function $\mathrm{F}(\omega)$ is calculated as follows:

$$\mathrm{F}(\omega) = \mathrm{P}(\mathrm{NNZ} \leq \omega)$$

where $\mathrm{P}(\mathrm{NNZ} \leq \omega)$ represents the probability that NNZ is less than or equal to ω. Blocks with NNZ values less than ω are selected for watermark embedding, ensuring sensitivity to malicious content tampering while maintaining visual quality.

1.2 Embedding Region

Embedding uncorrelated watermarks in consecutive frames typically causes temporal flickering and visual artifacts. Motion information is essential for avoiding such temporal flickering. Since any disturbance in a low-motion-intensity region has less impact than disturbances in high-motion-intensity or static regions, the quality of watermarked video can degrade. The motion vector magnitude in I-frames is zero because these frames are intra-coded. To represent motion vectors for I-frames, motion vectors are selected from the P-frame nearest to the I-frame in the previous Group of Pictures (GOP), referred to as pseudo-motion vectors. The specific procedure is as follows:

1. Partition the I-frame into non-overlapping blocks.
2. Calculate motion vectors for all blocks in P-frames of the previous GOP.
3. Smooth all motion vectors using a 3$×$3 median filter.
4. Since motion vectors for intra-coded blocks in I-frames contain zero values, evaluate adjacent intra-coded blocks within their respective Largest Coding Units (LCUs).
5. Assign pseudo-motion vectors to each block in the I-frame by inserting motion vectors from collocated blocks in the nearest P-frame.

Finally, pseudo-motion vectors are obtained for each block in the I-frame. Blocks with pseudo-motion vector values greater than zero can be selected for watermark embedding. Figure 2 [Figure 2: see original paper] illustrates the relationship between prediction mode change rates and pseudo-motion vectors, demonstrating that embedding watermarks in low-motion-intensity regions provides robustness against synchronization errors.

An appropriate threshold tM is selected for watermark embedding, where $0 <$ $\text{tM} < 1$. All selected blocks for watermark embedding have non-zero pseudo-motion vector values less than threshold tM. The threshold value depends on the relationship between prediction mode change rates and pseudo-motion vectors. Based on the results in Figure 2, tM is set to 20. Although error bars are not visible, results were obtained with 80% confidence levels, and Figure 2 is based on characteristics of CIF and QCIF videos.

The proposed embedding region selection procedure incorporates a visual threshold tV to maintain acceptable visual quality in watermarked video. DCT coefficients in a 4$×*4blockaredividedintoDCandACcoefficients.SinceACcoefficientsarelesssensitivetoembeddin*
*zeroACcoefficientsinaDCT−transformedblock.IftheabsolutedifferencebetweenAC*1*andAC*2*exceedsthres*
intra-prediction blocks for watermark embedding.

Program 1: Block Selection

```
Input: Unwatermarked video
Output: Suitable blocks for watermark embedding
For each I-frame in the unwatermarked video:
  For each block in the I-frame:
    If the prediction mode is smooth (zero-degree):
      Continue without modification
    Calculate NNZ in the block
    If NNZ > N:
      Compute pseudo-motion vector from the previous GOP' s P-frame at the same location
      If pseudo-motion vector > tM:
```

```
    If at least two non-zero AC coefficients exist:
      If |AC1 - AC2| > tV:
        This block can be used for watermark embedding
```

1.3 Key Extraction

In each I-frame, a random key selects candidate blocks from those chosen by Program 1. Watermarks are actually embedded in these candidate blocks, and using a pseudo-random key to select them enhances framework security. Different frames in the video will have different keys, generated using a pseudo-random number generator to prevent self-collusion attacks [16]. Program 2 ensures algorithm security by randomly selecting candidate blocks from the set chosen by Program 1.

Program 2: Candidate Block Selection

```
Input: Unwatermarked video
Output: Candidate blocks for watermark embedding
For each I-frame in the unwatermarked video:
  For each block selected by Program 1:
    Generate a key using a pseudo-random number generator
    Select candidate blocks using the random key
    Store the random key for decoder use when palette codes are unavailable
```

If watermark embedding occurs at the same positions in correlated I-frames, the embedding algorithm becomes vulnerable to self-collusion attacks [16]. This would require a very long video sequence, but transmitting such a long sequence is impractical. This problem can be solved by generating sequences through a combination of a public key and a private key. The public key is extracted from features of LCUs, which the copyright owner processes along with a private key. A 256-bit public key is extracted from each LCU and transferred to an encryption system with the private key.

Compression-domain features from HEVC decoder information can be used to generate public keys without requiring further decoding. The public key is extracted from certain LCU features that cannot be altered by attackers without degrading video visual quality. To enhance robustness, features sensitive to human perception are employed. One such feature is the DC coefficient of the LCU. If the public key were extracted using DC coefficients, attackers could modify them uniformly, making watermark detection impossible. In I-frames, changing non-zero DC coefficients to zero or zero-valued DC coefficients to non-zero values would degrade watermarked video quality or cause blocking artifacts. Another feature is the chroma prediction mode of the LCU.

A 256-bit public key is extracted from each LCU based on DC components and chroma prediction modes. The first 64 bits represent DC coefficients of each 4$\times 4 block in the LCU, while the subsequent 192 bits represent chroma modes of each 4\times$4 block in the LCU, with each chroma mode represented by 3 bits.

If watermarks are not embedded in appropriate coefficients, the algorithm' s robustness degrades and may cause visual artifacts in watermarked video. Therefore, selecting suitable coefficients or features is critical. Candidate blocks contain one or more compression-domain features such as NNZ, quantized coefficient magnitude and sign, and motion vector magnitude and sign. The following criteria are used to select appropriate features:

1. **Magnitude of low-frequency non-zero quantized AC coefficients**: Low-frequency coefficients exhibit greater stability against synchronization errors [15]. Modifying these coefficients before embedding watermark bits can maintain quality degradation while increasing bit rate and synchronization error robustness.
2. **NNZ value**: In [3], converting non-zero coefficients to zero for watermark embedding reduces NNZ and increases synchronization errors. Blocks predicted from watermarked blocks match well with neighboring blocks at the decoder. However, converting non-zero coefficients to zero significantly degrades video quality. Similarly, converting zero coefficients to non-zero increases synchronization failures and substantially raises bit rate. Therefore, a block' s NNZ value should not be altered for watermark embedding.
3. **DC coefficients**: DC coefficients cannot be used for watermark embedding as this would significantly degrade video quality and cause blocking artifacts [16].
4. **Sign of AC coefficients**: In [9], changing AC coefficient signs for watermark embedding substantially reduces watermarked video quality. Therefore, coefficient signs should not be modified.
5. **High-frequency coefficients**: High-frequency AC coefficients are reset to zero due to synchronization errors and filter usage.

In summary, only low-frequency non-zero quantized AC coefficients in 4\$ $\times 4 blocksofI-framesaresuitableforwatermarkembedding.Toreducecomplexity, twonon-zeroACcoefficients, AC1andAC2, arrangedinzigzagorderina4\times$ \$4 block are modified to embed watermark bits. Embedding watermarks in such coefficients causes noticeable visual impact, and their values cannot be altered in cases of synchronization failure due to watermark noise.

2.1 Watermark Embedding

A dual watermark sequence W is embedded into candidate blocks of I-frames in HEVC-encoded video using compression-domain features. In the proposed framework, watermark bits are embedded by modifying two non-zero AC coefficients, AC1 and AC2. To enhance robustness, a robustness threshold tR is employed, where tR > 0. The embedding procedure is as follows:

If the watermark bit value is 1 and |AC1 - AC2| $\leq$ tR, the modified coefficients are represented as: AC1' = AC1 + sign(D) × (tR + 1) AC2' = AC2 - sign(D) × (tR + 1) where D = MagRound(AC1, AC2) and sign() represents the sign function.

Similarly, if the watermark bit value is 0 and $|AC1 - AC2| > tR$, the modified coefficients can be represented as: AC1' = AC1 - sign(D) × (tR + 1) AC2' = AC2 + sign(D) × (tR + 1)

If the watermark bit value is 1 and $|AC1 - AC2| > tR$, or the watermark bit value is 0 and $|AC1 - AC2| \leq tR$, no modifications are made. If the tR value is large, the absolute difference between AC1 and AC2 becomes large, enhancing algorithm robustness but significantly degrading visual quality. The general watermark embedding process is presented in Program 3.

Program 3: Watermark Embedding

```
Input: Unwatermarked video
Output: Watermarked video
For each I-frame in the unwatermarked video:
  For each candidate block satisfying Program 3:
    If (watermark bit = 1 and $|AC1 - AC2| \leq tR$) or (watermark bit = 0 and $|AC1 - AC2| > tR$):
      Modify AC1 and AC2 coefficients
```

2.2 Watermark Extraction

Watermark extraction is performed at the decoder after entropy decoding. When the public key is compromised, the private key is used to generate a synthetic sequence (palette code) that indicates embedding positions in candidate blocks. The palette code is a binary file sequence containing position structures of candidate blocks [3], used for watermark extraction at the decoder.

In the proposed framework, the decoder generates a palette code by combining the private and public keys. In most literature [3,5], palette codes are transmitted to the decoder through a secure channel. In this framework, only the compressed private key is sent to authorized users for watermark bit extraction. When palette codes are unavailable at the decoder, Program 1 is used to select candidate blocks and generate a random key. However, lacking the palette code reduces framework robustness due to synchronization errors, as block positions selected at the decoder may not match those at the encoder.

In an I-frame, let AC1' and AC2' represent the first two non-zero coefficients in the watermarked video' s candidate blocks. If $|AC1' - AC2' | > 0$, the extracted watermark bit will be 1; otherwise, the extracted watermark bit will be 0. The general watermark extraction process is presented in Program 4.

Program 4: Watermark Extraction and Verification

```
Decompress private and public keys
Encrypt private and public keys
Generate palette code through private and public key combination
```

```
Select candidate blocks for extraction
For each candidate block:
  Extract watermark bits using AC1' and AC2'
  If |AC1' - AC2' | > 0:
    Watermark bit = 1
    Else:
    Watermark bit = 0
```

3.1 Embedding Capacity

The number of blocks suitable for watermark embedding represents the video' s embedding capacity. Blocks are selected based on spatiotemporal features. For most frames, NNZ values in HEVC are relatively smaller than corresponding values in H.264/AVC. Consequently, the embedding capacity of H.264/AVC-encoded video sequences is greater than that of HEVC-encoded videos. Embedding capacity is inversely proportional to motion threshold tM and directly proportional to visual threshold tV. Therefore, embedding capacity is a function of λ, expressed as:

$EC = F(\lambda)$

where $\lambda = tM \times N$, and N represents the number of candidate blocks. Equation (1) provides the expression for EC. A video' s embedding capacity depends on the spatiotemporal features of each 4$×$4 block. Thus, videos with most blocks exhibiting high texture and motion will have higher embedding capacity.

3.2 Performance Evaluation

This section measures the visual quality of watermarked video using Peak Signal-to-Noise Ratio (PSNR) and Video Quality Metrics (VQM) across the effective payload range. Bit rate increase is measured using the Bit Increase Rate (BIR) metric, defined as the probability of bit increase per embedded bit [19]:

BIR = (Bit count in watermarked video - Bit count in original video) / (Payload × Bit count in original video) × 100

where payload represents the total number of watermark bits actually embedded in the video sequence. Video quality represents a formal or informal measure of perceived video degradation after transmission/processing (typically compression of the original video).

The proposed watermarking framework was implemented using H.265/HEVC reference software HM9.0 on a PC with Intel Core i3 @ 3.30 GHz, 4GB RAM, running Windows 10 64-bit. Advanced tools in HM9.0 have been thoroughly studied in JCT-VC, which uses common test conditions for simulations without optimization. All intra main configurations were used, with RDQ and RDQTS disabled. These conditions primarily reflect typical bitstreams, and this configuration was used to evaluate the proposed method' s coding efficiency. The configuration used main random access with closed GOP, setting intra refresh type to 2.

Common Intermediate Format (CIF) and Quarter CIF (QCIF) [18] video sequences with 100 frames average were used, with payload ranges of {100, 150, 200, 250, 300}. Videos were encoded using a binary high-delay hierarchical prediction structure with one GOP. Since GOP contains 8 and 4 active reference images, it includes an intra period. Input videos had diverse attributes: high/low texture, low/high motion, unchanged camera shots, and uniform/noisy backgrounds. The quantization parameter (QP) was set to 28.

The proposed watermarking framework was also run using H.264/AVC reference software JM17.2 and compared with current methods [3,4,5], which are H.264/AVC-based and use HEVC-like configurations. For each video sequence, payload was kept constant during encoding with both HEVC and H.264.

Figure 3 [Figure 3: see original paper] shows block selection rates from Program 1 for (a) Yacht video, (b) Ski video, and (c) Hot Air Balloon video. Figure 4 [Figure 4: see original paper] presents average PSNR and VQM results comparing the proposed framework with other methods [3,4,5], averaged across five payload values from the set {100, 150, 200, 250, 300}. HEVC-encoded video results indicate that quality degradation in watermarked video is imperceptible to human observers. The results also show that the proposed framework generally achieves better visual quality on H.264/AVC-encoded video compared to other algorithms.

The proposed watermark embedding framework and block selection algorithm are designed to minimize quality degradation in watermarked video. Figure 4(c) shows that video bit rate increase is minimal in terms of BIR. Compared with other algorithms [3,4,5] that convert zero coefficients to non-zero or modify motion vectors for embedding, the proposed framework only uses two non-zero coefficients per block for watermark embedding, leaving NNZ, motion vectors, and coefficient values unchanged. For HEVC-encoded video, bit rate increase is weakened because HEVC reduces video bit rate by nearly 30-40% compared to H.264 at the same quality level.

---

4 Conclusion

This paper proposes a novel compression-domain digital video watermarking framework that invisibly embeds a readable watermark into I-frames of HEVC-encoded video. Extensive experiments demonstrate the algorithm' s effectiveness in terms of visual quality and bit rate increment. The proposed method employs compression-domain features for block selection, embedding, and extraction, while public key extraction reduces computational complexity. Future research will extend this approach to develop a general compression-domain watermarking platform that can encode video using different codecs in a codec-independent manner, thereby resisting a wider variety of attacks.

References

[1] Xie Qingkun, Jiang Yanru, Zhang Wenfei, et al. Information Encryption Technology Based on Digital Watermarking and Iterative Algorithm [J]. *Acta Optica Sinica*, 2016, 33(6): 76-83.

[2] Liu Xuanxuan, Wang Xiaohong. Digital watermarking algorithm based on saliency map [J]. *Optical Technique*, 2016, 42(3): 260-263.

[3] Chen Mengying, Wang Zhanqing. Adaptive Digital Watermarking Algorithm Based on FRFT and HVS [J]. *Application Research of Computers*, 2017, 34(7): 2180-2183.

[4] Liu Fan, Yang Hongyong, Su Qingtang. Color image blind watermarking algorithm based on Schur decomposition [J]. *Application Research of Computers*, 2017, 34(10): 3085-3089.

[5] Pastuszak G, Trochimiuk M. Architecture design of the high-throughput compensator and interpolator for the H.265/HEVC encoder [J]. *Journal of Real-Time Image Processing*, 2016, 11(4): 663-673.

[6] Zoran M. Miličević, Zoran S. Bojković. Performance of high efficiency video coding (HEVC) HM-16.6 encoder [C]// *Telecommunications Forum (TELFOR)*. IEEE, 2016: 712-715.

[7] Abdi L, Abdallah F B, Meddeb A. A robust video watermarking algorithm in H.264/AVC compressed domain [C]// *ACM Symposium on Applied Computing*. ACM, 2015: 1291-1293.

[8] Chen Qing, Weng Xufeng. Novel blind image watermarking based on pseudo Zernike moments [J]. *Application Research of Computers*, 2016, 33(9): 2810-2812.

[9] Mehmood N, Mushtaq M. A fragile watermarking scheme using prediction modes for H.264/AVC content authentication [C]// *Local Computer Networks Workshops*. IEEE, 2013: 1014-1021.

[10] Ling Chen, Urrehman Obaid, Zhang Wenjun. Semi-fragile Watermarking Scheme for H.264/AVC Video Content Authentication Based on Manifold Feature [J]. *KSII Transactions on Internet & Information Systems*, 2014, 8(12): 4568-4587.

[11] Stütz T, Autrusseau F, Uhl A. Non-Blind Structure-Preserving Substitution Watermarking of H.264/CAVLC Inter-Frames [J]. *IEEE Transactions on*

Multimedia, 2014, 16(5): 1337-1349.

[12] Kang Liangcheng, Li Chaofeng. JPEG compressed image quality assessment method based on digital watermarking in contourlet domain [J]. *Computer Engineering and Applications*, 2016, 52(21): 206-210.

[13] Dalmia N, Okade M. First quantization matrix estimation for double compressed JPEG images utilizing novel DCT histogram selection strategy [C]// *the Tenth Indian Conference*. 2016: 1-8.

[14] Gaj S, Patel A S, Sur A. Object based watermarking for H.264/AVC video resistant to RST attacks [J]. *Multimedia Tools & Applications*, 2016, 75(6): 3135-3156.

[15] Ma Zhaofeng, Huang Jianqing, Jiang Ming, et al. A Video Watermarking DRM Method Based on H.264 Compressed Domain with Low Bit-Rate Increasement [J]. *Chinese Journal of Electronics*, 2016, 25(4): 641-647.

[16] Hale K S, Stanney K M. *Handbook of Virtual Environments: Design, Implementation, and Applications* [M]. CRC Press, Inc. 2014.

[17] Wang Duyao, Huang Sujuan, Feng Guorui, et al. Perceptual differential energy watermarking for H.264/AVC [J]. *Multimedia Tools & Applications*, 2012, 60(3): 537-550.

[18] Ansari R, Devanalamath M, Hussain M, et al. Robust BPSO and Scene Change Based Digital Video Watermarking Algorithm in DWT-DFT-SVD Domain [M]// *Security in Computing and Communications*. Springer Berlin Heidelberg, 2013: 168-178.

[19] Dong Jiangtao, Li Jingping. A Robust Zero-Watermarking Algorithm for Encrypted Medical Images in the DWT-DFT Encrypted Domain [M]// *Innovation in Medicine and Healthcare 2016*. Springer International Publishing, 2016.

[20] Wang Gang, Li Chaofeng. Blind stereo image quality assessment based on disparity map and complex contourlet transform [J]. *Computer Engineering and Science*, 2017, 39(3): 512-518.

Note: Figure translations are in progress. See original paper for figures.

Source: ChinaXiv —Machine translation. Verify with original.