Computers, Materials & Continua

Tech Science Press

DOI: 10.32604/cmc.2024.051364

ARTICLE

A Dual Domain Robust Reversible Watermarking Algorithm for Frame Grouping Videos Using Scene Smoothness

Yucheng Liang[1,2,*], Ke Niu[1,2,*], Yingnan Zhang[1,2] and Yifei Meng[1,2]

[1]College of Cryptographic Engineering, Engineering University of the Chinese People's Armed Police Force, Xi'an, 710086, China

[2]Key Laboratory of Information Security of the Chinese People's Armed Police Force, Engineering University of the Chinese People's Armed Police Force, Xi'an, 710086, China

*Corresponding Authors: Yucheng Liang. Email: liangyucheng2000@163.com; Ke Niu. Email: niuke@163.com

Received: 04 March 2024 Accepted: 14 May 2024 Published: 20 June 2024

ABSTRACT

The proposed robust reversible watermarking algorithm addresses the compatibility challenges between robustness and reversibility in existing video watermarking techniques by leveraging scene smoothness for frame grouping videos. Grounded in the H.264 video coding standard, the algorithm first employs traditional robust watermark stitching technology to embed watermark information in the low-frequency coefficient domain of the U channel. Subsequently, it utilizes histogram migration techniques in the high-frequency coefficient domain of the U channel to embed auxiliary information, enabling successful watermark extraction and lossless recovery of the original video content. Experimental results demonstrate the algorithm's strong imperceptibility, with each embedded frame in the experimental videos achieving a mean peak signal-to-noise ratio of 49.3830 dB and a mean structural similarity of 0.9996. Compared with the three comparison algorithms, the performance of the two experimental indexes is improved by 7.59% and 0.4% on average. At the same time, the proposed algorithm has strong robustness to both offline and online attacks: In the face of offline attacks, the average normalized correlation coefficient between the extracted watermark and the original watermark is 0.9989, and the average bit error rate is 0.0089. In the face of online attacks, the normalized correlation coefficient between the extracted watermark and the original watermark is 0.8840, and the mean bit error rate is 0.2269. Compared with the three comparison algorithms, the performance of the two experimental indexes is improved by 1.27% and 18.16% on average, highlighting the algorithm's robustness. Furthermore, the algorithm exhibits low computational complexity, with the mean encoding and the mean decoding time differentials during experimental video processing being 3.934 and 2.273 s, respectively, underscoring its practical utility.

KEYWORDS

Robust reversible watermarking; scene smoothness; dual-domain; U channel; H.264 encoding standard

1 Introduction

With the rapid advancement of computer technology, the Internet is witnessing an escalating demand for multimedia watermarking solutions, particularly in safeguarding the ownership and

copyright of the vast repository of freely available images and videos online [1]. To address these challenges, the field of information hiding has emerged as a pivotal area of research.

1.1 Major Technology

Within the realm of information security, Cryptography [2], Steganography [3], and Watermarking [4] stand out as significant research directions. Cryptography focuses on ensuring the confidentiality and integrity of data by employing encryption algorithms and robust key management practices to prevent unauthorized access, thereby emphasizing data security. In contrast, Steganography is a method that conceals confidential information within a cover medium to maintain its secrecy.

Watermarking technology, on the other hand, involves embedding imperceptible watermark information within digital media to safeguard intellectual property rights and uphold content integrity. Cryptography secures data through encryption techniques like Advanced Encryption Standard (AES) and Rivest-Shamir-Adleman (RSA), safeguarding information from illicit access [5]. Steganography conceals secret data within a carrier medium using techniques such as Least Significant Bit (LSB) replacement and frequency domain manipulation, commonly applied in image, audio, and video domains [6]. Watermarking, meanwhile, utilizes digital watermark embedding in media content to authenticate and preserve intellectual property rights [7].

The collective research efforts in these domains offer diverse solutions for information concealment, catering to varied security requisites and application contexts. Ongoing research endeavors are poised to further enhance these technologies, fortifying information security and fortifying intellectual property rights.

1.2 Research Status

Distinct from cryptographic methods, Digital watermarking technology serves as a vital tool for protecting multimedia data copyrights, traceability, and integrity authentication of digital media, like image [8–10] and video [11–13], by embedding watermark information within digital carriers. Nevertheless, traditional digital watermarking techniques, whether robust or fragile/semi-fragile, often lead to permanent distortion of the embedded image. In some specific applications such as medical and military image processing, even this loss of quality is unacceptable. While embedding information into digital carriers can lead to irreversible distortion, reversible watermarking (RW) techniques [14–17] have been developed to enable non-destructive carrier recovery post-watermark extraction. In algorithm design, the primary objectives typically involve minimizing embedding distortion, ensuring reversibility after information extraction, and maintaining a specified embedding capacity [16]. However, robustness considerations are often secondary, and even minor distortions can result in erroneous extraction of watermark information [17–19]. To enhance this aspect, the concept of robust reversible watermarking (RRW) was introduced [20,21]. In scenarios where there is no attack, the decoder is responsible for both watermark extraction and carrier recovery. However, if the carrier undergoes an attack, the reversibility feature may be compromised, but the robust watermark can still be accurately extracted, indicating potential tampering with the carrier [22–25].

Several RRW algorithms for digital images are introduced in the paper: Coltuc et al. [20,21] introduced the foundational framework for RRW, achieving both robustness and reversibility by sequentially embedding watermark information and auxiliary data. Building upon this work, Wang et al. [22] proposed a two-stage image RRW framework based on the concept of Independent Embedding Domains (IED), an extension of Coltuc's watermarking algorithm. This innovative approach involves decomposing the original carrier into high and low frequency components using the Haar wavelet

transform, enabling independent embedding processes based on two distinct IEDs. By employing this methodology, the framework successfully circumvents issues associated with robust and reversible embedding operations within the same domain, thereby preventing interference between the two types of watermarks. At present, video-oriented RRW algorithms are rare. Most of the video watermarking algorithms are robust watermarking algorithms for digital video: Hazim et al. [26] employed a two-dimensional wavelet transform to partition the image into four sub-bands. Chen et al. [27] organized frames into groups, segmented embedded frames into non-overlapping blocks, decomposed these blocks using singular value analysis, computed Zernike moments, and embedded the watermark accordingly. Fan et al. [28] integrated non-zero quantization coefficients and energy factors to select suitable chrominance subblocks, proposing an optimized modulation technique for embedding watermarks in DCT quantization coefficients to minimize subblock modifications. Takale et al. [29] enhanced watermarking flexibility by combining discrete wavelet transform (DWT) with principal component analysis (PCA). Farri et al. [30] utilized a combination of contourlet transform (CT) and singular value decomposition (SVD) to embed watermarks in low-frequency video sub-bands. The above robust watermarking algorithms for digital video do not have the ability to restore the original video. Although the RRW framework is effective, current applications are focused on protecting digital images.

1.3 Background and Structure Erection

The rapid growth of short video culture has elevated its status as a crucial medium for information dissemination, underscoring the pressing need for effective copyright protection and anti-tampering measures [31,32]. It is of great practical value to develop a video-oriented digital watermarking algorithm with both robustness and reversibility in this context. The H.264/Advanced Video Coding (AVC) video coding standard stands as the predominant video compression standard [33], renowned for its compression efficiency, compatibility, and widespread adoption in short video data transmission [34]. Notably, the coding standard for "TikTok" short videos aligns with AVC (H.264). Leveraging the multiple independent redundant domains inherent in this coding standard, this study refines the RRW framework proposed in [22] and introduces a novel video dual-domain watermarking algorithm centered on hidden frame selection. To facilitate this, video frames are organized into groups using a scene smoothness grouping mechanism, with robust watermarks embedded into the low-frequency quantization Discrete Cosine Transform (QDCT) coefficient domain, i.e., the Direct Current (DC) coefficient domain, through the selection of hidden frames. Concurrently, the high-frequency QDCT coefficient domain, i.e., the Alternating Current (AC) coefficient domain, serves as the embedding domain for reversible watermarking, ensuring both robustness of the watermark and lossless recovery of the original video content. The contributions of the proposed algorithm can be listed as follows:

1. Introducing a scene smoothness grouping mechanism and hidden frame selector for enhanced video fidelity across various sizes with low computational complexity and easy application to existing videos.

2. Optimization of the image-oriented RRW framework for effective copyright protection of MP4 videos, enabling original video restoration while preserving MP4 video copyrights.

3. Integration of robust watermark splicing and Reed-Solomon error correction coding technologies in the algorithm to ensure robust security against malicious attacks, with controlled embedding strength for a balance between invisibility and robustness. Leveraging Histogram Shifting watermark embedding technology allows for lossless extraction of reversible watermarks, facilitating seamless original video recovery.

The relevant contents of this paper and the arrangement of subsequent chapters are as follows: Section 1 outlines the theoretical underpinnings of the algorithm, Section 2 presents the algorithm framework, Section 3 details watermark embedding, extraction, and carrier recovery processes, Section 4 offers a comparative analysis of experimental results against other advanced robust watermarking algorithms, and finally, Section 5 provides a summary of the thesis.

2 Theoretical Basis

2.1 RRW Framework

The fundamental concept underpinning the RRW framework, as introduced by Wang et al. [22], revolves around the concept of Independent Embedding Domains (IED). In essence, the original carrier undergoes a transformation into two distinct IEDs, one designated for robust embedding and the other for reversible embedding. Through this independent transformation process, the quality of the intermediate IED resulting from the initial stage remains uncompromised by reversible embedding, thereby preserving the robustness of the watermark.

The original carrier is transformed into two embedding domains (EDs) X^{ro} and X^{re} through a frequency domain transformation denoted as F, i.e., $F = (X^{ro}, X^{re})$. These two embedding domains are independent of each other and serve the purposes of embedding robust watermarking and reversible watermarking, respectively. It is important to note that the frequency domain transformation F is invertible, ensuring the integrity of the transformation process.

The embedding process consists of two stages, as illustrated in the Fig. 1. Initially, Y^{ro} is generated by embedding the watermark W into X^{ro} using a robust embedding method. In this scenario, X^{re} remains constant, leading to Y^{re} being set equal to X^{re}. Subsequently, in the second stage, the scalar information resulting from the robust watermark embedding is combined to create auxiliary information M. The reversible embedding method is then applied to embed M into Y^{re}, resulting in the generation of Z^{re}. Given that Y^{ro} remains unchanged in this process, Z^{ro} is set to be equal to Y^{ro}. Finally, the label carrier Z is obtained by applying the inverse transformation of F, denoted as $Z = F^{-1}(Z^{ro}, Z^{re})$.

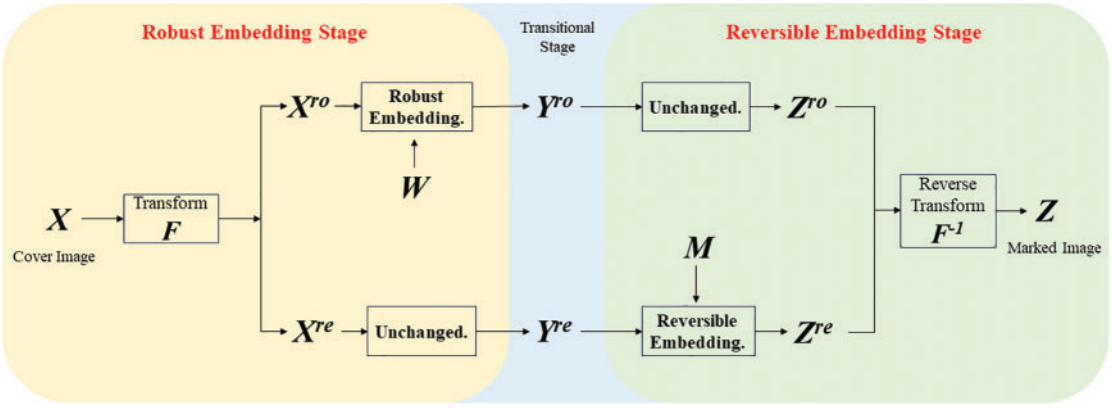


Figure 1: WANG's RRW framework

In the decoding process, the carrier Z can be retrieved, allowing for the extraction of watermark information without distortion. Initially, the labeled carrier Z is separated into two EDs by F, denoted as Z^{ro} and Z^{re}. Subsequently, M and Y^{re} are obtained through the process of reversible watermark

extraction, enabling the restoration of the original carrier. During this phase, Y^{ro} is equivalent to Z^{ro}, facilitating the extraction of the robust watermark W from Y^{ro}. The restoration of Y^{ro} to X^{ro} is achieved through M. Finally, the original carrier reverts to X using the inverse transformation of F, expressed as $X = F^{-1}(X^{ro}, X^{re})$.

If the encoded carrier Z undergoes distortion, the robust watermark W can still be directly extracted from the distorted Z. This paper is grounded on the concept of IED within this framework and applies it to safeguarding the copyright of compressed video content. Detailed explanations of these applications will be provided in Sections 2 and 3.

2.2 Watermark Error Correction Coding Techniques

Reed-Solomon code (RS) is a class of error-correcting codes proposed by Reed et al. in 1960 [35]. RS code is a linear error correction code known for its powerful error correction ability, which can correct multiple random errors and burst errors at the same time. In order to improve the fault-tolerance of watermarking, the methods outlined in this study include applying lossless RS coding (Reed-Solomon Code) to robust watermarking and reversible watermarking before embedding the watermark. The coding principle of RS code is shown in Eqs. (1) to (4).

$$m(x) = m_{k-1}x^{n-1} + \ldots + m_1 x^{2t+1} + m_0 x^{2t} + 0 \cdot x^{2t-1} + 0 \cdot x + 0 \tag{1}$$

$$g(x) = \prod_{j=0}^{2t-1} (x - a^j) \tag{2}$$

$$m(x) \,\%g(x) = p_{2t-1}x^{2t-1} + p_1 x + p \cdot 0 \tag{3}$$

$$c(x) = m(x) + m(x) \,\%g(x) \tag{4}$$

In the given scenario, $m(x)$ denotes the watermark information polynomial with 8-bit symbols, while $g(x)$ represents the primitive generation polynomial. The encoding of watermark information polynomial is denoted by $c(x)$, with $2t$ representing the error correction bit length, indicating the error correction capability of the code. Here, k signifies the number of symbols in the watermark information, n denotes the symbols in the encoding sequence, and a = 2. The number of bit errors in a symbol is considered independent of the total number of bit errors. All RS codes are categorized as symbol errors, exhibiting a high fault tolerance rate, and proving effective in addressing burst errors. In order to balance the comprehensive error correction ability and computational efficiency, the algorithm in this paper adopts 64-bit RS coding, in which 52 bits of valid watermarking bits are allowed, and 12 bits of error correction code are adopted, namely RS (52,40). The experimental results show that this setting can balance the error correction performance and calculation efficiency well.

3 Development and Evaluation of Algorithm Framework

In this section, we present the algorithm framework and introduce a scene smoothness grouping mechanism along with a frame selector. Frame selection plays a crucial role in the algorithm, especially considering the sensitivity of the watermark information. Embedding the watermark in every frame may expose the information to unauthorized users and significantly increase the video's bit rate. Therefore, it is essential to carefully choose frames with hidden embedding positions. Leveraging the IED framework and recognizing that human eyes are more responsive to brightness than chroma [36], the algorithm groups video frames and selects smooth frames within each group as the hidden frames, determining the embedding positions of the watermark.

3.1 Algorithm Framework

The algorithm in this paper is structured into three main stages, as illustrated in Fig. 2. The Hidden Frames Selecting Stage involves hidden frame selection, where frames from the cover video are grouped using a scene detection mechanism. Frames meeting the hidden frame selection criteria form the hidden frame set X, while those not meeting the criteria are temporarily set aside.

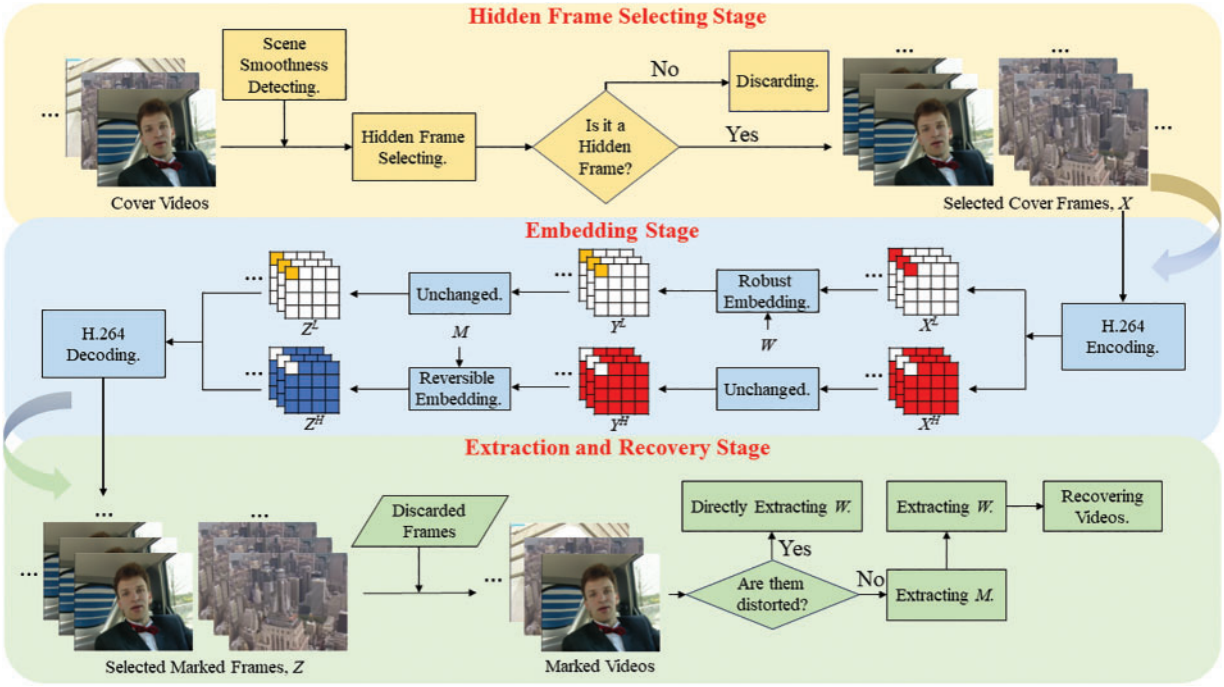


Figure 2: The proposed algorithmic framework overview

The Second Stage, known as the Embedding Stage, begins by obtaining two embedding domains in X through H.264 encoding: the low-frequency QDCT coefficient X^L and the high-frequency QDCT coefficient X^H of the U channel. These two domains are used independently to embed robust watermarking W and reversible watermarking M. The Embedding Stage consists of two sub-stages.

In the first sub-stage, Y^L is generated by embedding the robust watermark W into X^L using a robust splicing method, while X^H remains unchanged, with $Y^H = X^H$. In the second sub-stage, the reversible watermark M is used to restore Y^L to X^L, and then embedded into Y^H to produce Z^H. Here, Y^L remains unchanged, with $Z^L = Y^L$. The watermark mark frame Z is created through H.264 decoding.

The Extraction and Recovery Stage involves the extraction and recovery process, where Z is combined with the frames initially set aside in the first stage to produce the tagged video. If the marked video is undistorted, the reverse process of the second stage can be executed: encoding the marked video with H.264 to obtain two embedding domains Z^L and Z^H, extracting M and Y^H through reversible watermarking, and retrieving the robust watermark W from Y^L (Z^L) to restore Y^L to X^L.

If the marked video is distorted, the robust watermark W can be directly extracted from the distorted video. In this scenario, the robust watermark remains unaffected by distortion in the second

sub-embedding stage. The subsequent subsection will conduct a robustness analysis of the independent embedding domain framework proposed in this algorithm.

3.2 Analysis of Noise

In traditional video robust watermarking algorithms within the compressed domain, a common approach involves utilizing a single embedding domain [37–41]. In scenarios where the marked video undergoes distortion from an attack, the resulting distorted marked video can be denoted as $\tilde{Z} = Z + A_n$, where A_n represents a noisy signal. To recover the original video, it becomes imperative to address the distortion introduced by the robust watermark using reversible embedding auxiliary information. However, given the delicate nature of reversible embedding, the robust watermark W must be directly extracted from the distorted marked video.

During this process, the noise affecting the transitional video frame Y can be defined as follows: $\tilde{Z} - Y = Z - Y + A_n = A_{re} + A_n$, where $A_{re} = Z - Y$ signifies the embedding noise originating from reversibly embedded information within the video. It is reasonable to posit that these two distinct noise signals are independent of each other, thereby presenting the total noise as Eq. (5).

$$Var\,(A_{re} + A_n) = Var\,(A_{re}) + Var\,(A_n) \tag{5}$$

The variance function $Var()$ is utilized to analyze the noise affecting the robust watermark W. In the context of our algorithm's independent embedding domain framework, the distortion attack A_n introduces distortion effects on both IEDs, denoted as $\left(\widetilde{Z^L}, \widetilde{Z^H}\right) = 2D\_DCT\left(\tilde{Z}\right) = 2D\_DCT(Z+A_n)$. Consequently, the distortion noise impacting the robust watermark W is defined as ${A_n}^L = \widetilde{Z^L} - Z^L$. It is important to highlight that the IED framework employed in this algorithm ensures that the distortion noise affecting W is solely due to ${A_n}^L$, and is not influenced by the embedding noise from the reversible watermarking M. Therefore, the distortion of the robust watermark in this algorithm is characterized by $Var({A_n}^L)$, and the QDCT coefficient of the attacked U channel can be expressed as:

$$\widetilde{y^L} = \widetilde{z^L} = {X^L}_i + n \approx {X^L}_i + n^L \tag{6}$$

In the context of the noise affecting the embedding coefficient ${X^L}_i$, the application of $2D\_DCT()$ ensures that $n \approx n^L = {A_n}^L$. Consequently, the final distortion can be calculated as Eq. (7).

$$Var\left({A_n}^L\right) \approx a\,Var(A_n) \tag{7}$$

In the context of the H.264 encoding standard, where $a \in (0, 1)$, the specific value of a is dependent on this standard. Upon comparison, it becomes apparent that $Var\left({A_n}^L\right) \leq Var\,(A_{re}) + Var\,(A_n)$. Consequently, the robustness of the compressed video watermarking algorithm utilizing a single embedded domain will inevitably experience a reduction due to the presence of double noise. However, it is important to note that this reduction does not compromise the robust extraction of the watermark.

3.3 Design of Scene Smoothness Grouping Mechanism and Hidden Frame Selector

Utilizing the Scene Smoothness Grouping Mechanism (SSGM), a Hidden Frame Selector (HFS) is devised to effectively group frames and select appropriate frames for embedding. The distinction between frames plays a crucial role in determining whether they belong to the same group or different groups, as illustrated in Fig. 3.

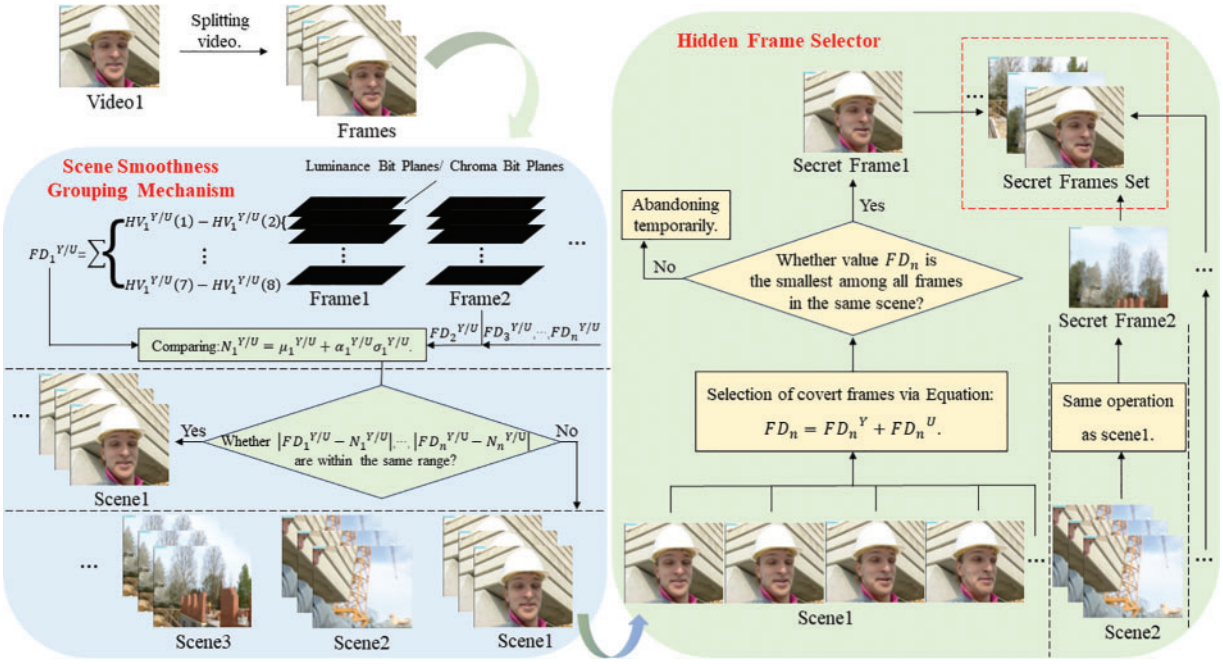


Figure 3: Development of scene smoothness grouping mechanism and hidden frame selector algorithm

Taking the video "Foreman" as a case study, the initial step involves dividing the video into individual frames. Subsequently, the differences between adjacent bits of the eight-bit effective bit plane of the Y channel and U channel within each frame are computed. Simultaneously, a threshold is determined, and the comparison between the difference and the threshold aids in grouping the video "Foreman" based on scene smoothness criteria. These frame variances can be represented in the form of histogram differences. The equation for histogram differences is presented in Eq. (8).

$$FD_n^{\,Y} = \sum_{l=1}^{L} HV_n^{\,Y}(l) - HV_n^{\,Y}(l+1)$$

$$FD_n^{\,U} = \sum_{l=1}^{L} HV_n^{\,U}(l) - HV_n^{\,U}(l+1) \tag{8}$$

where $FD_n^{\,Y}$ represents the histogram difference of the Y channel of the Nth frame, $HV_n^{\,Y}$ denotes the histogram value of the Y channel of the L-layer of the Nth frame, $FD_n^{\,U}$ signifies the histogram difference of the U channel of the L-layer of the Nth frame, and $HV_n^{\,U}$ indicates the histogram value of the U channel of the L-layer of the Nth frame, with l set to 7. The SSGM operates by grouping frames based on scene smoothness and analyzing the difference in the two-channel histograms. This is achieved by establishing a decision threshold to quantify abrupt changes in frame texture. The equation for the decision threshold is presented in Eq. (9).

$$N_n^{\,Y} = \mu_n^{\,Y} + \alpha_n^{\,Y}\sigma_n^{\,Y}$$

$$N_n^{\,U} = \mu_n^{\,U} + \alpha_n^{\,U}\sigma_n^{\,U} \tag{9}$$

where $N_n^{\,Y}$ represents the threshold of the Y channel of the Nth frame, $N_n^{\,U}$ denotes the threshold of the U channel of the Nth frame, $\mu_n^{\,Y}$ and $\sigma_n^{\,Y}$ stand for the mean and standard deviation of the Y channel component of the Nth frame, and $\mu_n^{\,U}$ and $\sigma_n^{\,U}$ represent the mean and standard deviation of the U channel component of the Nth frame. The parameter α is determined based

on the characteristics of the video scene. The comparison between the frame difference and the decision threshold can indicate the texture change within the scene. Moreover, the frame difference itself can serve as an indicator of whether the frame is suitable for embedding as a hidden frame to some extent. This is because the difference between bit planes reflects the extent of pixel value changes in the frame, with a smaller bit plane difference indicating minimal changes between adjacent pixels. Frames with smoother and more consistent details enhance the concealment of embedded watermark information. The criteria for grouping frames are based on FD_n and N_n, If the internal values of both $\left\{\left|FD_n{}^Y - N_n{}^Y\right|, \left|FD_{n+1}{}^Y - N_n{}^Y\right|, \left|FD_{n+2}{}^Y - N_n{}^Y\right|, \ldots, \left|FD_{n+t}{}^Y - N_n{}^Y\right|\right\}$ and $\left\{\left|FD_n{}^U - N_n{}^U\right|, \left|FD_{n+1}{}^U - N_n{}^U\right|, \left|FD_{n+2}{}^U - N_n{}^U\right|, \ldots, \left|FD_{n+t}{}^U - N_n{}^U\right|\right\}$ fall within a similar range, it indicates that the Nth frame to the $(N+t)$th frame belong to the same scene, with similar texture and slow changes, thus forming the same group. In the event of a fault occurring at $(N+t)$th frame, a new scene commences at $(N+t)$th frame, and the thresholds are updated to $N_{n+t}{}^Y$ and $N_{n+t}{}^U$. The HFS chooses the frame with the smallest difference between the Y channel and U channel within the same scene as the hidden frame for embedding, denoted as $FD_n = FD_n{}^Y + FD_n{}^U$.

It is postulated that the Nth frame is the frame within a group exhibiting the smallest difference, with FD_n representing the cumulative difference between the Y channel histogram and the U channel histogram of the Nth frame. The total histogram difference of each frame is compared across the grouped frames, and the frame displaying the minimal histogram difference is designated as the embedded frame X for the robust watermark W and the reversible watermark M.

3.4 Principles Analysis Underlying HFS

In each frame group, the HFS chooses the frame with the smallest difference between adjacent bit planes in the Y channel and U channel as the output. The underlying principle guiding HFS will be elucidated below. To begin, considering the Y channel, the Y channel of a frame is depicted as a matrix, where each element of the matrix corresponds to the Y channel value of a pixel, as illustrated below:

$$I(i,j) = b_7(i,j)\, b_6(i,j)\, b_5(i,j)\, b_4(i,j)\, b_3(i,j)\, b_2(i,j)\, b_1(i,j)\, b_0(i,j) \tag{10}$$

where b_7 to b_0 represent bits 7 to 0, respectively. To compute the frame difference $FD_n{}^Y$, a difference matrix D is defined. For each pixel, the Y channel difference can be calculated as:

$$\begin{aligned} D(i,j) = {} & |b_7(i,j) - b_6(i,j)| + |b_6(i,j) - b_5(i,j)| + |b_5(i,j) - b_4(i,j)| + |b_4(i,j) - b_3(i,j)| \\ & + |b_3(i,j) - b_2(i,j)| + |b_2(i,j) - b_1(i,j)| + |b_1(i,j) - b_0(i,j)| \end{aligned} \tag{11}$$

where $D(i,j)$ represents the difference of pixels in row i and column j of the difference matrix. Following the definition of geometric interval [42], the difference matrix D is interpreted as the geometric distance and denoted as the normalized value D', representing the distance from the sample point in the feature space to the hyperplane. The normalization equation is expressed as follows:

$$D' = \frac{D}{\|D\|} \tag{12}$$

where $\|D\|$ represents the norm of D. The normalized value D' is denoted as follows:

$$D' = w^T X + \frac{B}{\|w\|} \tag{13}$$

where w is the weight vector, X is the eigenvector, and B is a bias term. In HFS, pixels with $D' > 0$ are classified into positive classes, while pixels with $D' < 0$ are classified into negative classes. In practical

classification scenarios, the model's prediction accuracy is crucial, with closer proximity to the actual result indicating better classification. This proximity is reflected in the accuracy of classifying positive and negative classes and the shorter distance of each sample point to the hyperplane [42]. A smaller sum of differences between adjacent bit planes in the Y and U channels signifies a closer prediction to the actual result, indicating a more concentrated Y channel value distribution and smoother image appearance. The same principle applies to the U channel. In conclusion, frames with smaller sums of differences between adjacent bit planes in the Y and U channels within the same group result in smoother frames and are selected by the HFS mechanism.

4 Developing Two-Domain RRW Algorithm for Frame Grouping in Compressed Videos via Scene Smoothness Analysis

In this section, we propose a novel approach for embedding and extracting robust and reversible watermarks based on the algorithm framework presented in this paper, as well as the subsequent restoration of the original video. Considering the perceptual impact on the human eye, which is more sensitive to changes in brightness rather than chromaticity, our algorithm leverages the H.264 video encoding framework to efficiently utilize a large number of QDCT coefficients. Through the application of traditional robust watermark splicing techniques (such as those proposed by WANG), the robust watermark is embedded in the low-frequency QDCT domain X^L of the U channel, while the reversible watermark is embedded in the high-frequency QDCT domain Y^H of the U channel.

4.1 Robust Watermark Embedding: Enhancing Security and Integrity

At this stage, the robust watermark W is embedded in the low-frequency DC QDCT domain X^L of the U channel within the HFS outcome.

In Fig. 4, exemplified by the video "Foreman," the hidden frame is divided into m non-overlapping macroblocks following H.264 encoding.

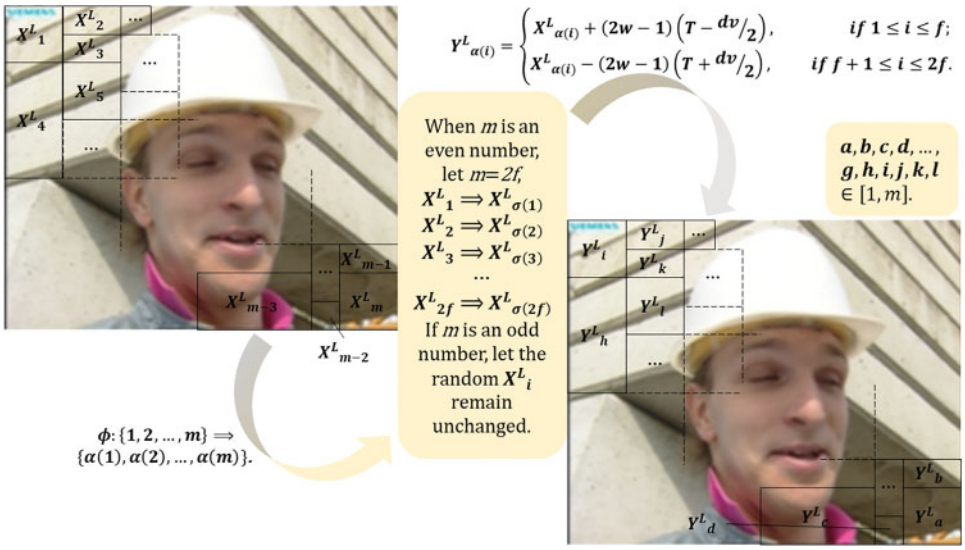


Figure 4: Robust watermark embedding in foreman video using traditional splicing technology

For the macroblocks $\{X^L_1, X^L_2, \ldots, X^L_m\}$, a random map ϕ: $\{1, 2, \ldots, m\} \Longrightarrow \{\beta(1), \beta(2), \ldots, \beta(m)\}$ is defined to embed the watermark $w \in \{0, 1\}$ into the U channel low-frequency QDCT

coefficient X^L of a macroblock by means of the mapping ϕ. In the case where m is even, m is denoted as $2f$, as described below:

$$Y^L{}_{\beta(i)} = \begin{cases} X^L{}_{\beta(i)} + (2w-1)\left(T - \dfrac{dv}{2}\right), & if\ 1 \le i \le f \\ X^L{}_{\beta(i)} - (2w-1)\left(T + \dfrac{dv}{2}\right), & if\ f+1 \le i \le 2f \end{cases} \tag{14}$$

If m is an odd number, a macroblock $X^L{}_i$ is selected at random to serve as a stationary block, while the remaining even number of macroblocks participate in the mapping ϕ and the embedding of the watermark W. Where T represents the threshold for controlling robustness, and dv denotes the difference between the two sets of QDCT coefficients. With the increase of threshold T, the watermark embedding strength increases, and the watermark robustness of the proposed algorithm is enhanced. This is because the amount of modification increases with the increase of the threshold T, and the greater the amount of modification, the less vulnerable to external attacks. On the contrary, its robustness will weaken. However, in order to evaluate whether a watermarking algorithm is good enough and can be applied to practical work, it needs to be both invisible and robust. The algorithm in this paper controls the embedding intensity of robust watermarking by threshold T, and a lot of experiments are needed to find the most suitable threshold.

$$dv = \frac{\sum_{i=1}^{f} X^L{}_{\beta(i)} - \sum_{i=f+1}^{2f} X^L{}_{\beta(i)}}{f} \tag{15}$$

At the same time the difference dv is modified to:

$$dv^w = \frac{\sum_{i=1}^{f} Y^L{}_{\beta(i)} - \sum_{i=f+1}^{2f} Y^L{}_{\beta(i)}}{f} = \begin{cases} T, & if\ w = 1 \\ -T, & if\ w = 0 \end{cases} \tag{16}$$

It is important to highlight those adjustments to the threshold T may lead to an overflow or underflow issue, resulting in QDCT coefficient values exceeding 511 or falling below −512. In cases where abnormal macroblocks are encountered, maintaining the original value unchanged and recording the position based on the embedding order are essential to prevent errors during watermark extraction. Notably, a practical observation of the proposed algorithm reveals that certain video scenes exhibit slow and minimal transformations, leading to a reduced number of HFS outcomes and insufficient hidden frame embedding capacity for accommodating the watermark W. To address this challenge, the algorithm introduces a compensation mechanism: if the hidden frame's embedding capacity falls short of accommodating W, a sub-hidden frame is introduced to compensate for the capacity shortfall. The selection criterion for the sub-hidden frame is determined by the sum of the minimum differences between the Y channel and the U channel within the same scene.

4.2 Reversible Watermark Embedding: Techniques and Considerations

In the reversible embedding phase, the reversible watermark M intended for the inversion robust embedding phase is inserted into the U channel high-frequency AC QDCT domain Y^H of the HFS selection outcome. At the decoder end, leveraging the lossless attributes of the reversible watermarking algorithm allows for the lossless extraction of the reversible watermark, thereby eliminating any distortion introduced by the robust watermark embedding process, ultimately facilitating the recovery of X^L. The reversible watermark M is composed of the threshold T, the difference dv, and the macroblock position P from the robust watermark embedding phase. To mitigate the distortion arising

from the robust watermark embedding phase, the information is transmitted to the decoder for the restoration of the original video.

$$M = T \oplus DV \oplus P \tag{17}$$

where DV represents the original differences between two sets of QDCT coefficients, the reversible watermarking M is integrated into the high-frequency QDCT domain Y^H of the U channel using one-dimensional histogram translation technology (HS), as illustrated in the accompanying figure. The specific embedding procedure is outlined as follows:

$$z^H = \begin{cases} 2y^H + m, if\ y^H \in [\theta^L, \theta^R] \\ y^H + \theta^R + 1, if\ y^H > \theta^R \\ y^H + \theta^L, if\ y^H < \theta^L \end{cases} \tag{18}$$

where y^H and z^H represent the QDCT coefficients of Y^H and Z^H, respectively, and $m \in \{0, 1\}$ denotes a reversible watermark, θ^L and θ^R serve as thresholds governing the HS embedding capacity. Notably, when θ^L is specified, a larger separation between θ^R results in increased embedding capacity. Given the limited embedding capacity and emphasis on reversibility in this paper's reversible watermarking algorithm, only conventional HS technology is employed for embedding the reversible watermark. The challenge of overflow and underflow is addressed for θ^L and θ^R, with the algorithm utilizing location map technology to manage overflow concerns. In contrast to the macroblock position P generated in robust watermark embedding, the location map is losslessly compressed and subsequently embedded into the unused coefficient values using HS translation technology. Detailed information on the location mapping process can be referenced in traditional HS techniques [43] and is not reiterated in this article.

4.3 Extraction of Watermark and Carrier Recovery in Two Domains

The algorithm in this study delineates the extraction and recovery stages into two scenarios:

(1) In the event that Z is transmitted to the decoding endpoint without distortion, the initial step involves extracting the reversible watermark through the following operations:

$$m = mod\left(\frac{z^H}{2}\right), if\ z^H \in \left[2\theta^L, 2\theta^R + 1\right] \tag{19}$$

According to the inverse translation technique of HS, Y^H is lossless restored according to Eq. (20).

$$Y^H = \begin{cases} \left\lfloor \dfrac{z^H - m}{2} \right\rfloor, if\ z^H \in [2\theta^L, 2\theta^R + 1] \\ z^H - \theta^R - 1, if\ z^H > 2\theta^R + 1 \\ z^H - \theta^L, if\ z^H < 2\theta^L \end{cases} \tag{20}$$

According to Eq. (14), the robust watermark extraction operation is as follows:

$$w = \begin{cases} 1, if\ dv^w > 0 \\ 0, if\ dv^w \le 0 \end{cases} \tag{21}$$

The distortion introduced by robust watermarking is mitigated through the reversible watermark M, facilitating the restoration of the original X^L as follows:

$$X^{L}{}_{\beta(i)} = \begin{cases} Y^{L}{}_{\beta(i)} - (2w-1)\left(T - \dfrac{dv}{2}\right), & if\ 1 \le i \le f \\ Y^{L}{}_{\beta(i)} + (2w-1)\left(T + \dfrac{dv}{2}\right), if\ f+1 \le i \le 2f \end{cases} \tag{22}$$

When Z is transformed into $\tilde{Z}$, successful recovery of the original video becomes unfeasible. The distorted version $\widetilde{Z^L}$ is obtained directly through H.264 video encoding, following the Eq. (21). Subsequent to this, the robust watermarking W is extracted directly. The correct extraction of the robust watermark W is ensured as long as the threshold T provides a sufficiently robust level of resilience.

4.4 Algorithm Step

This subsection provides a detailed overview of the algorithmic steps employed in this study for watermark embedding and extraction. The watermark embedding process is delineated as follows:

Step 1 (Selection of Hidden Frames): The hidden frame set $X = \{F_1, F_2, \ldots, F_j\}$ is chosen for embedding in the original video using double-domain watermarking, based on the criteria of minimizing the difference between adjacent bit planes of the U channel.

Step 2 (H.264 Video Encoding): The hidden frame set X is encoded using H.264, and the embedding domains X^L and X^H within the hidden frame set F are determined.

Step 3 (Robust Watermark Embedding): The macroblocks in F are organized into a set $\{X^L{}_1, X^L{}_2, \ldots, X^L{}_{2f}\}$, and the error-corrected encoded W is embedded into the low-frequency QDCT coefficients of each U channel in the set, resulting in Y^L, as per Eq. (11). Positions of macroblocks causing overflows or underflows are recorded as P. During this phase, X^H remains unchanged, with $Y^H = X^H$.

Step 4 (Reversible Watermark Embedding): A reversible watermark M is generated using Eq. (17), and the watermark bit is embedded into the high-frequency QDCT coefficients of the U channel, resulting in Y^H and Z^H, following Eq. (18). Y^L remains unchanged, with $Z^L = Y^L$.

Step 5 (H.264 Video Decoding): Z^H and Z^L are combined through H.264 decoding to produce the video Z with a two-domain watermark.

The critical aspect of the video watermarking algorithm lies in ensuring efficient copyright protection, facilitating successful watermark extraction and video recovery. The process of watermark extraction and video recovery is outlined as follows:

Step 1 (H.264 Video Encoding): The two-domain watermarking video is encoded using H.264, resulting in Z^L and Z^H.

Step 2 (Reversible Watermark Extraction and Recovery): The reversible watermark M is extracted based on the position indication within set F and Eq. (19), resulting in Z^H and Y^H. During this phase, Z^L remains unchanged, with $Y^L = Z^L$.

Step 3 (Robust Watermark Extraction and Recovery): The robust watermark W is extracted following Eq. (21), and the distortion caused by the robust watermark is restored using M as per Eq. (22), resulting in Y^L and X^L. YH remains unchanged, with $X^H = Y^H$.

Step 4 (Original Video Recovery): The original video is recovered by merging X^H with X^L through H.264 decoding. It is essential to note that Step 4 can only be executed if the two-domain watermarking video has not been compromised by an attack. In cases where the reversible embedding method fails

in Step 2, direct extraction of the robust watermark W from the distorted $\widetilde{Z^L}$ is necessary in Step 3 due to the fragile nature of the reversible embedding method.

5 Simulation Study and Experimental Findings in Video Watermarking

In this section, the performance evaluation of the proposed algorithm is conducted through a comparative analysis with three existing video watermarking algorithms [28,44,45] in terms of perceptual quality and robustness. Each of the three video watermarking algorithms employs distinct strategies for selecting embedding locations within video frames or blocks.

Fan et al. [28] utilize a combination of non-zero quantization coefficient and energy factor to identify suitable chrominance subblocks for watermark embedding. They further introduce an optimal modulation technique to embed the watermark within the QDCT coefficient of the selected subblocks.

Singh et al. [44] employ a histogram difference method to extract color moving frames and still frames from the original video. A first-level linear wavelet transform is applied to the chroma channel of moving frames, with the low-frequency sub-band LL chosen for watermark embedding.

Sharma et al. [45] propose a frame selection mechanism based on scene change detection. They group video frames according to scene transitions and introduce a hybrid approach involving graph-based transformation, singular value decomposition, and chaotic encryption for watermark embedding.

In the embedding capacity assessment conducted in this study, a 256-bit watermark sequence is utilized. Notably, the algorithm presented in this paper and Fan et al.'s method leverage H.264 video coding, enabling a watermark payload of 256 bits. On the other hand, Singh et al. and Sharma et al. focus on image watermarking. For these methods, the watermark bit string is transformed into a grayscale image following Zig-Zag scanning, serving as the input for watermark embedding.

The experimental setup involves the MatlabR2021a platform running on an i5 processor, connected to x.264 via the MEX interface for simulation execution. The experimental video dataset comprises three segments of 176 × 144 traditional YUV video sequences in QCIF format: "Foreman", "Carphone", "Mobile", "Tennis", "City", "Soccer" (https://media.xiph.org/video, accessed: Feb. 20, 2023.) and three short video sequences of 720 × 1080 resolution in MP4 format ("Tree", "Street", "Food"). The proposed algorithm's invisibility and robustness are assessed across these video sequences.

5.1 Evaluation of Invisibility in Video Watermarking Algorithms

In the H.264 video standard, the largest macroblock is 16 × 16 block. Since most of the scenes in the video is smooth scenes, 16 × 16 macroblocks are also the largest number of macroblocks in most videos. In the invisibility analysis, we block the stego-frame of each experimental video into 16 × 16 blocks. According to the embedding method of robust splicing technology, 1 robust watermark bit is embedded in the DC coefficient of each macroblock. Finally, for the experimental video with a resolution of 176 × 144, the robust embedding capacity of a single hidden frame remains at 99 robust watermark bits; for the experimental video with a resolution of 720 × 1080, except for the 16 × 16 block Except for pixels, the remaining pixels are not considered for embedding bits, and the robust embedding capacity of a single hidden frame remains at 3015 robust watermark bits. The size of the reversible watermark bits generated after embedding the robust watermark changes with different experimental videos.

At the subjective evaluation stage, the experimental videos were visually inspected before and after the watermark embedding process by human observers. Fig. 5 illustrates one of the original video frames of each experimental video alongside the corresponding frames post-embedding. Despite conducting comparisons across the experimental population, conclusive determinations regarding the presence of watermarks within the video frames could not be ascertained at the subjective level.

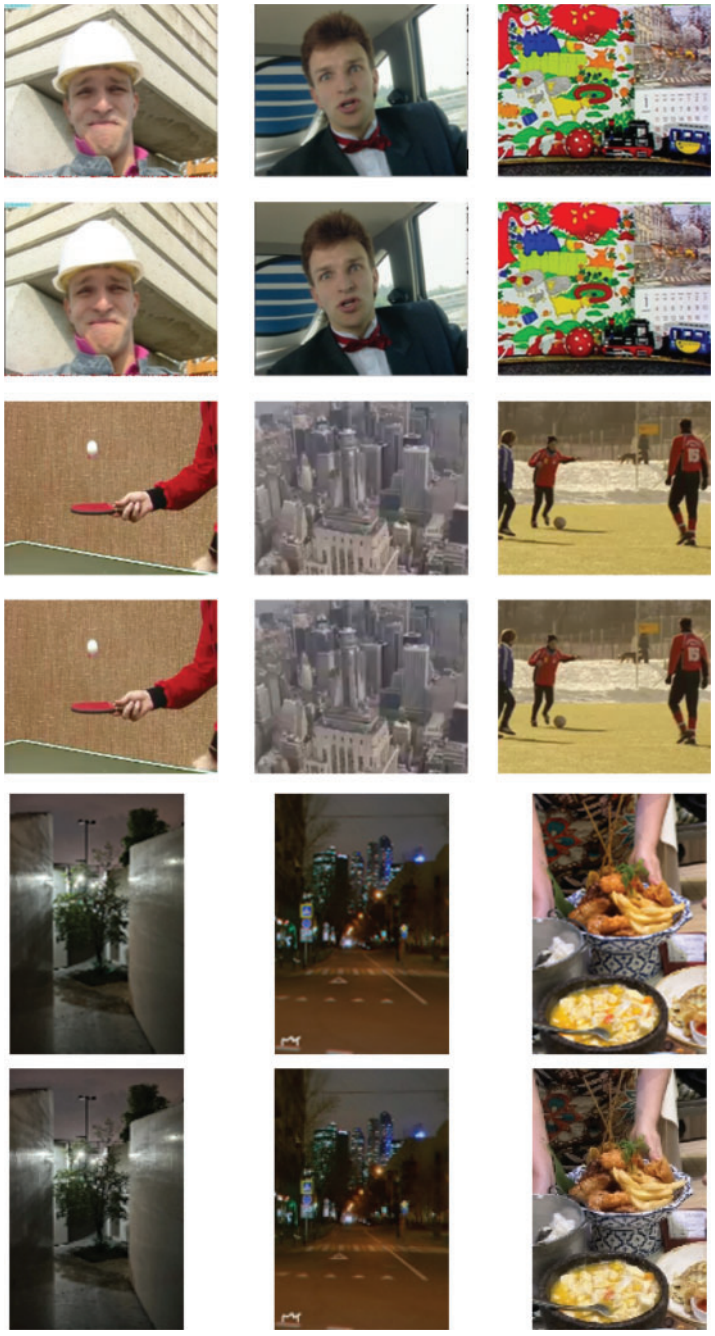


Figure 5: Comparison of original video frames with corresponding post-embedding frames in experimental videos

At the objective evaluation stage, the algorithm proposed in this study is assessed through the computation of the mean peak signal-to-noise ratio (MPSNR) and mean structural similarity (MSSIM) metrics between the original video frame and the concealed video frame, as defined in Eqs. (23) and (24).

$$MPSNR(F, F') = \frac{1}{N} \sum_{j=1}^{F} PSNR_j \tag{23}$$

$$MSSIM(F, F') = \frac{1}{N} \sum_{j=1}^{F} SSIM_j \tag{24}$$

where F represents the hidden frame, F' denotes the embedded hidden frame, and N signifies the total number of hidden frames. Figs. 6 and 7 present the Peak Signal-to-Noise Ratio (PSNR) and Structural Similarity Index (SSIM) values of the first 150 frames of traditional and short videos after embedding watermarks W and M, with the legend indicating the experimental video names. Additionally, Tables 1 and 2 display the Mean PSNR (MPSNR) and Mean SSIM (MSSIM) results following the embedding of watermarks and messages in the initial 150 frames of traditional and short videos, respectively.

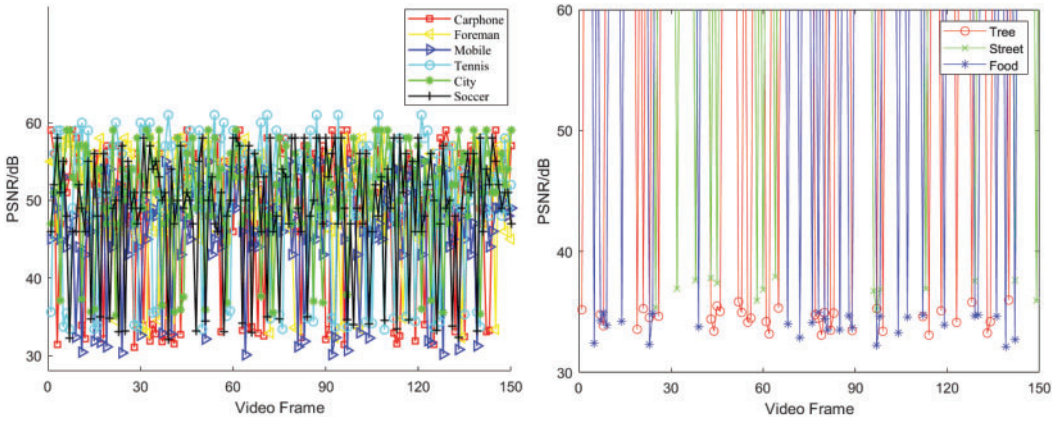


Figure 6: Comparative analysis of PSNR performance for the initial 150 frames in traditional and short videos embedded with watermark $\boldsymbol{W}$ and $\boldsymbol{M}$

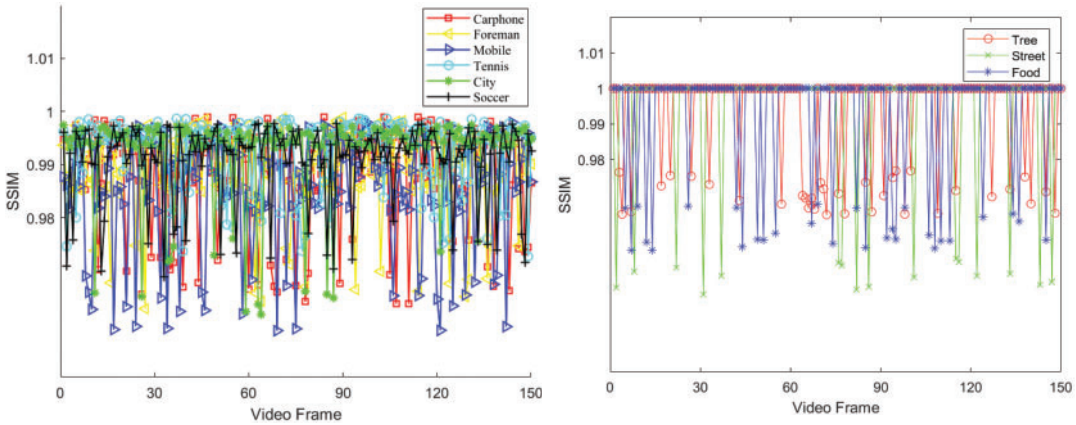


Figure 7: Comparative analysis of SSIM performance for the initial 150 frames in traditional and short videos embedded with watermark $\boldsymbol{W}$ and $\boldsymbol{M}$

Table 1: Mean PSNR and Mean SSIM results for watermark information $\boldsymbol{W}$ and auxiliary message $\boldsymbol{M}$ embedding in the initial 150 frames of traditional videos

Traditional video	Carphone	Foreman	Mobile	Tennis	City	Soccer
MPSNR/dB	49.3157	49.1410	48.5933	50.0257	49.8875	48.9504
MSSIM	0.9998	0.9995	0.9992	0.9999	0.9995	0.9993

Table 2: Mean PSNR and Mean SSIM results for watermark $\boldsymbol{W}$ and $\boldsymbol{M}$ embedding in the initial 150 frames of short videos

Short video	Tree	Street	Food
MPSNR/dB	50.0733	49.5533	48.9067
MSSIM	0.9999	0.9998	0.9993

Note that the hidden frame after embedding the watermark in Fig. 5, and all the data in Figs. 6, 7 and Tables 1 and 2 are completed at the same embedding strength. Specifically, we calculate the PSNR and SSIM for the first 150 frames of each experimental video at the threshold T = 1 (Eq. (11)), and the MPSNR and MSSIM corresponding to the two metrics. A large number of watermarking embedded video experiments show that the comprehensive invisibility of experimental video is the best at this time. The X value in Fig. 6 is the video frame number of the experimental video, and the Y value is the PSNR between each original video frame and the video frame embedded with the watermark. The X value in Fig. 7 is the same as that in Fig. 6, and the Y value is the SSIM between each original video frame and the video frame after embedding the watermark. Tables 1 and 2 correspond to the mean values of experimental indicators in Figs. 6 and 7, respectively. The same embedding strength will be used in subsequent residual invisibility comparative analysis experiments and robustness analysis experiments.

Based on the data from Figs. 6, 7, Tables 1, and 2, the Mean Peak PSNR (MPSNR) values for the six traditional videos in QCIF format after embedding watermarks W and M are 49.3157, 49.1410, 48.5933, 50.0257, 49.8875, and 48.9504 dB, with corresponding Mean SSIM (MSSIM) values of 0.9998, 0.9995, 0.9992, 0.9999, 0.9995, and 0.9993, respectively. For the three short videos in MP4 format after embedding W and M, the MPSNR values are 50.0733, 49.5533, and 48.9067 dB, with MSSIM values of 0.9999, 0.9998, and 0.9993, respectively. These results suggest that the proposed algorithm has minimal impact on the video quality both before and after information embedding.

It is important to note that the algorithm discussed in this study focuses on a watermarking technique specifically designed for H.264 compressed videos. Analysis of the experimental data reveals that the PSNR and SSIM of traditional QCIF videos post-information embedding fall within a specific range. As QCIF videos undergo transformation into MP4 format following compression by the H.264 standard, each frame experiences information loss compared to the original frame. Consequently, frames not designated as hidden frames may exhibit variations compared to the original uncompressed video frames. Nonetheless, the minimum PSNR of the hidden frame post-information embedding remains above 30 dB, with the lowest SSIM exceeding 0.95. Additionally, the PSNR and SSIM values for video frames not selected as hidden frames in short MP4 videos are infinite and 1, indicating the algorithm's independent embedding process, ensuring that the embedding of hidden frames does not impact other frames. The minimum PSNR and SSIM values for hidden frames

after information embedding are maintained above 32 dB and 0.94, respectively. In conclusion, the algorithm demonstrates good invisibility characteristics.

Table 3 presents the MPSNR and MSSIM values for each video in accordance with the comparison scheme. Analysis of the table reveals that the average MPSNR and MSSIM values for the proposed method are 49.3830 dB and 0.9996, respectively. These exceptional metrics serve as evidence of the superiority of the proposed technique over the compared scheme.

Table 3: Mean PSNR and Mean SSIM results for watermark $\boldsymbol{W}$ and $\boldsymbol{M}$ embedding in the initial 150 frames of short videos

Video	Fan et al. [28]		Singh et al. [44]		Sharma et al. [45]		Proposed method	
	MPSNR	MSSIM	MPSNR	MSSIM	MPSNR	MSSIM	MPSNR	MSSIM
Carphone	46.6419	0.9993	48.1972	0.9996	48.2718	0.9997	49.3157	0.9998
Foreman	45.5611	0.9992	48.0117	0.9993	48.1267	0.9994	49.1410	0.9995
Mobile	43.4761	0.9991	47.8913	0.9991	47.8992	0.9991	48.5933	0.9992
Tennis	46.7381	0.9993	48.3094	0.9997	48.8301	0.9997	50.0257	0.9999
City	46.0033	0.9993	48.2763	0.9995	48.6937	0.9995	49.8875	0.9995
Soccer	44.0009	0.9992	47.9996	0.9995	48.1006	0.9991	48.9504	0.9993
Tree	47.7559	0.9995	48.4357	0.9997	48.6913	0.9998	50.0733	0.9999
Street	46.8937	0.9992	48.0013	0.9993	48.0947	0.9994	49.5533	0.9998
Food	46.0094	0.9989	47.2787	0.9990	47.6418	0.9992	48.9067	0.9993

Given the limitations in interpreting invisibility solely based on PSNR and SSIM, we conducted additional experiments to assess the performance of the proposed algorithm in terms of bit-rate expansion. The results of these experiments are summarized in Table 4.

Table 4: Mean PSNR and Mean SSIM results for watermark $\boldsymbol{W}$ and $\boldsymbol{M}$ embedding in the initial 150 frames of short videos

Video	Fan et al. [28]		Singh et al. [44]		Sharma et al. [45]		Proposed method	
	Bitrate expansion (%)	Ratio	Bitrate expansion (%)	Ratio	Bitrate expansion (%)	Ratio	Bitrate expansion (%)	Ratio
Carphone	8.89	1.21	8.66	1.21	8.46	1.23	8.14	1.24
Foreman	12.61	1.14	12.41	1.15	12.33	1.15	11.74	1.17
Mobile	15.37	1.04	15.12	1.05	14.67	1.09	13.89	1.12
Tennis	9.01	1.19	8.83	1.21	8.51	1.21	8.14	1.25
City	10.73	1.20	10.44	1.19	10.10	1.23	9.77	1.17
Soccer	14.95	1.07	14.12	1.10	13.94	1.12	13.29	1.11
Tree	8.17	1.24	7.83	1.25	7.47	1.27	7.01	1.31
Street	11.47	1.18	11.28	1.19	11.09	1.19	10.25	1.20
Food	11.89	1.15	11.64	1.17	11.38	1.18	11.13	1.19

Bitrate expansion involves increasing the video file size to accommodate embedded information. Table 4 displays the bit-rate expansion and the ratio of embedded information to video increment for the proposed algorithm, Fan et al.'s method, Singh et al.'s method, and Sharma et al.'s method. The Ratio represents the amount of embedded information relative to the video size increase, with the embedded information comprising $W + M$. By comparing MPSNR and MSSIM values, it is evident that the proposed method minimally impacts the original carrier and exhibits superior invisibility performance.

5.2 Evaluation of Robustness in Video Watermarking Algorithms

In this subsection, we will evaluate the robustness of the proposed algorithm and compare it to the methods of Fan et al., Singh et al., and Sharma et al., under various video attacks. To comprehensively evaluate the robustness of the proposed algorithm, we will choose two types of attacks for experiments.

The first type of attack is offline, including Gaussian noise with an average of 0 and variances of 0.01, 0.05 and 0.1, salt and pepper noise with variances of 0.01, 0.05 and 0.1, Gaussian low-pass filtering with variances of 0.1, 0.05 and 0.1, and 3×3 filters with variances of 0.1, 0.2 and 0.3 (fuzzy attacks). Frame Dropping (random 10%∼35%), frame average (random 10%∼35%), Frame Swapping (random 10%∼35%), and then compression (quantization programming QP = 26∼28).

The second type of attack is online attack: collusion attack is a kind of attack that needs to be mainly guarded against in online attacks, which refers to the cooperation between the attacker and the information hider to obtain the hidden information or the original carrier. In the field of information hiding, there are two main categories:

Class I collusion: Embedding the same pseudo-watermark in multiple different carriers. This kind of collusion obtains the legitimate watermark by removing or estimating the legitimate watermark embedded in the carrier through a linear average.

Class II collusion: Embedding different pseudo-watermarks in different copies of the same carrier. This kind of collusion obtains an estimated version of the carrier without legitimate watermarks through a linear average, compares the estimated version with the current suspected carrier with legitimate watermarks, and obtains an original carrier without legitimate watermarks.

For video watermarking, attackers aim to illegally infringe the copyright security of videos in order to obtain videos without robust watermarking. Therefore, in the robustness analysis experiment of this paper, Class II collusion is used to attack experimental videos to evaluate the robustness of the proposed algorithm against online attacks. The frames of the same scene in the video are approximately the same. The specific practice of this experiment is to group each frame obtained according to SSGM as the video frame of the same scene, and randomly generate multiple bit strings with the same length as the robust watermark bit string after error correction coding as different pseudo-watermarks.

Finally, we extract watermarks from the hacked hidden frames and evaluate the robustness of the algorithm by calculating and comparing the mean normalized correlation number (MNC) and mean bit error rate (MBER) of the original watermarks. The definitions of MNC and MBER are shown in Eqs. (25) and (26).

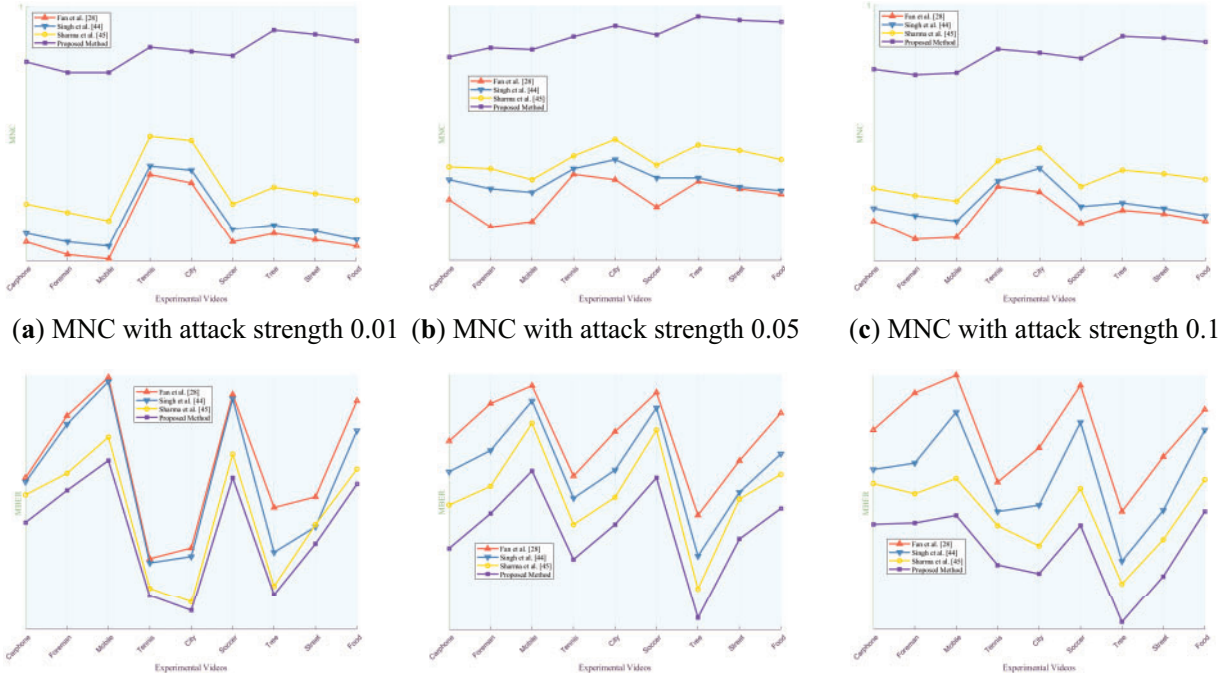
$$MNC = \frac{1}{N}\sum_{j=1}^{F}\frac{\sum[(W-\mu_W)(W'-\mu_{W'})]}{\sigma_{\mu_W}\sigma_{\mu_{W'}}} \tag{25}$$

$$MBER = \frac{1}{N}\sum_{j=1}^{F}\frac{L_E}{L_W} \tag{26}$$

where W and W' represent the original watermark and extracted watermark, respectively, μ_W and $\mu_{W'}$ denote the average values of the original watermark and extracted watermark, while σ_{μ_W} and $\sigma_{\mu_{W'}}$ indicate the standard deviations of the original watermark and extracted watermark, respectively, with a value range of MNC being $[-1, 1]$. L_E and L_W refer to the number of error watermark bits and the number of original watermark bits, N represents the total number of hidden frames, and F denotes the hidden frame.

5.2.1 Robustness Analysis of Offline Attacks

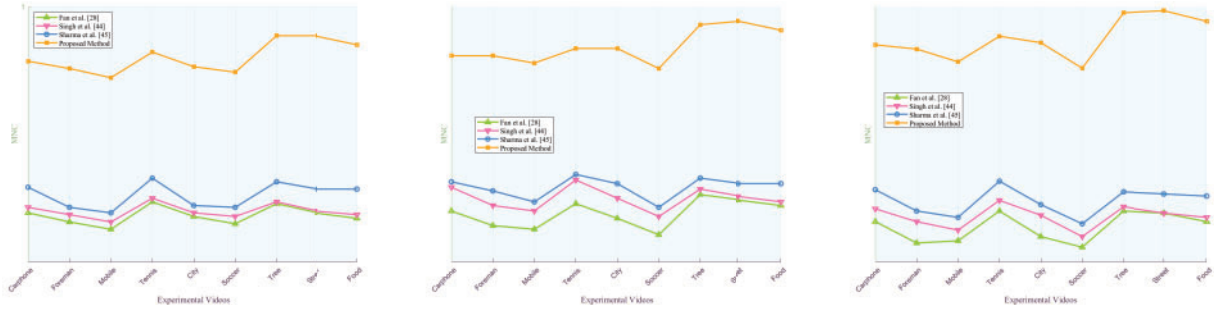
This subsubsection describes the robustness of the proposed algorithm against offline attacks by combining six traditional YUV experiment videos and three MP4 experiment videos. The robust watermark is extracted from the hidden frame after offline attack, and the NC value between the extracted robust watermark in each hidden frame and the corresponding original robust watermark is calculated, and the MNC value is taken to evaluate the robustness of the proposed algorithm. The robustness of the proposed algorithm is compared with three video robust watermarking algorithms. The results are shown in Figs. 8 to 14 below.



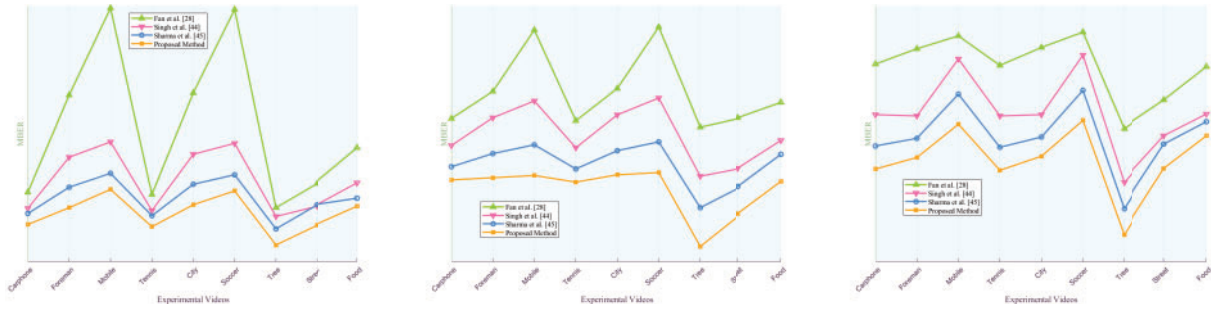
(**a**) MNC with attack strength 0.01 (**b**) MNC with attack strength 0.05 (**c**) MNC with attack strength 0.1

(**d**) MBERwith attack strength0.01 (**e**) MBER with attack strength 0.05 (**f**) MBER with attack strength 0.1

Figure 8: Comparison of MNC and MBER values of the proposed algorithm against three algorithms under various intensities of gaussian noise attack

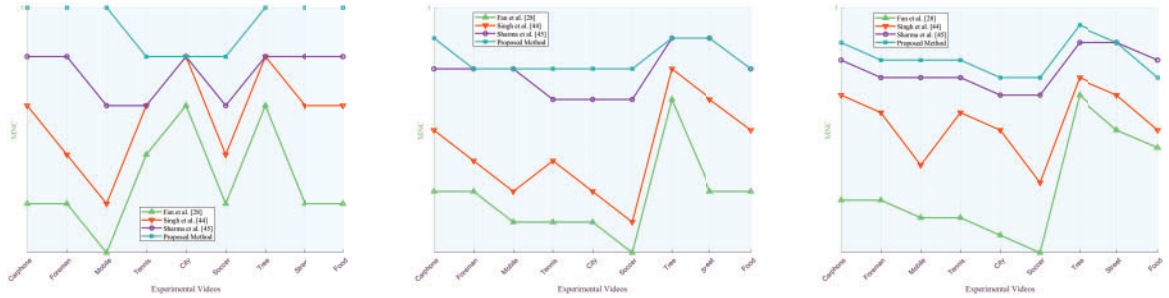


(**a**) MNC with attack strength 0.01 (**b**) MNC with attack strength 0.05 (**c**) MNC with attack strength 0.1

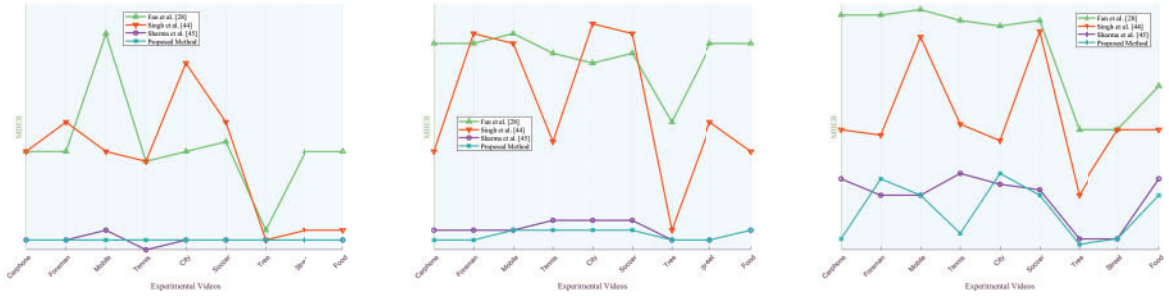


(**d**) MBER with attack strength 0.01 (**e**) MBER with attack strength 0.05 (**f**) MBER with attack strength 0.1

Figure 9: Comparison of MNC and MBER values of the proposed algorithm against three algorithms under various intensities of salt-and-pepper noise attack

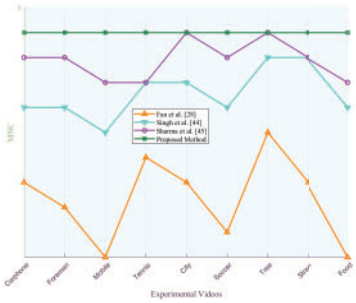


(**a**) MNC with attack strength [3,3,0.1] (**b**) MNC with attack strength [3,3,0.2] (**c**) MNC with attack strength [3,3,0.3]

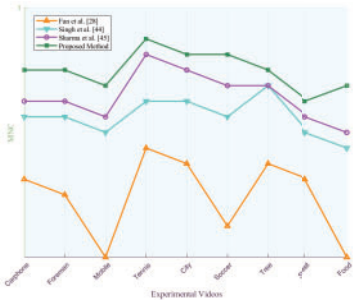


(**d**) MBER with attack strength [3,3,0.1] (**e**) MBER with attack strength [3,3,0.2] (**f**) MBER with attack strength [3,3,0.3]

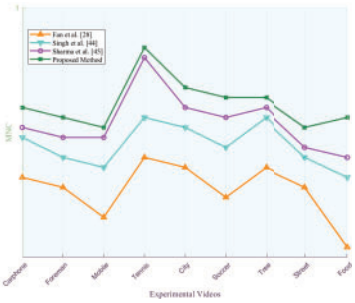
Figure 10: Comparison of MNC and MBER values of the proposed algorithm against three algorithms under various intensities of fuzzy attack



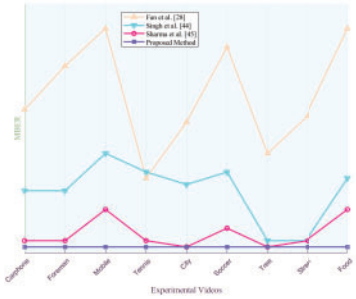
(**a**) MNC with attack strength 10%



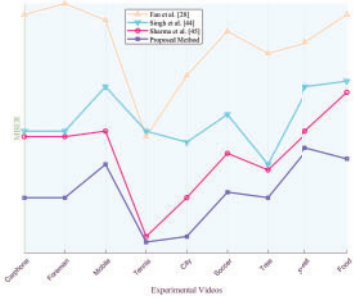
(**b**) MNC with attack strength 20%



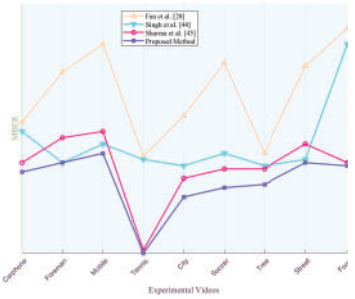
(**c**) MNC with attack strength 25%



(**d**) MBER with attack strength 10%

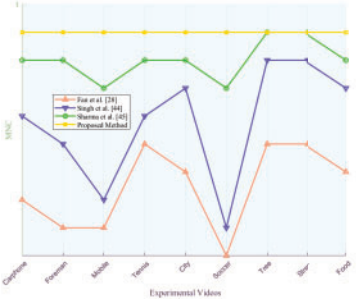


(**e**) MBER with attack strength 20%

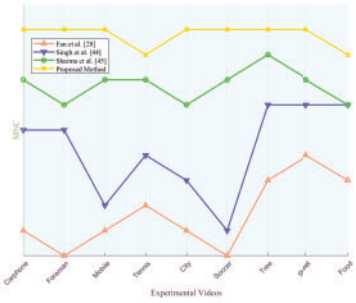


(**f**) MBER with attack strength 25%

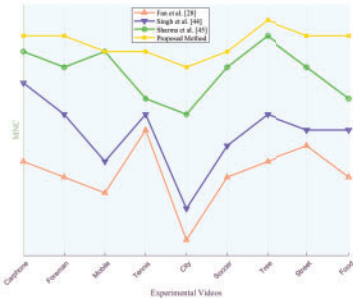
Figure 11: Comparison of MNC and MBER values of the proposed algorithm against three algorithms under various intensities of frame dropping attack



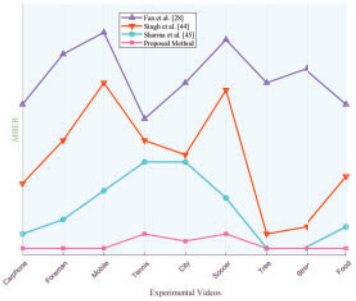
(a) MNC with attack strength 10%



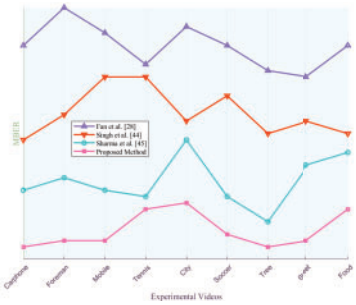
(b) MNC with attack strength 20%



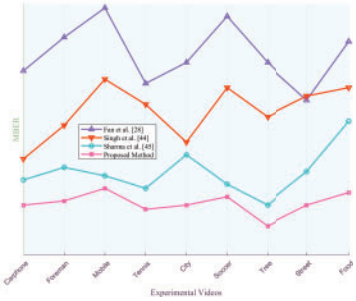
(c) MNC with attack strength 25%



(d) MBER with attack strength 10%

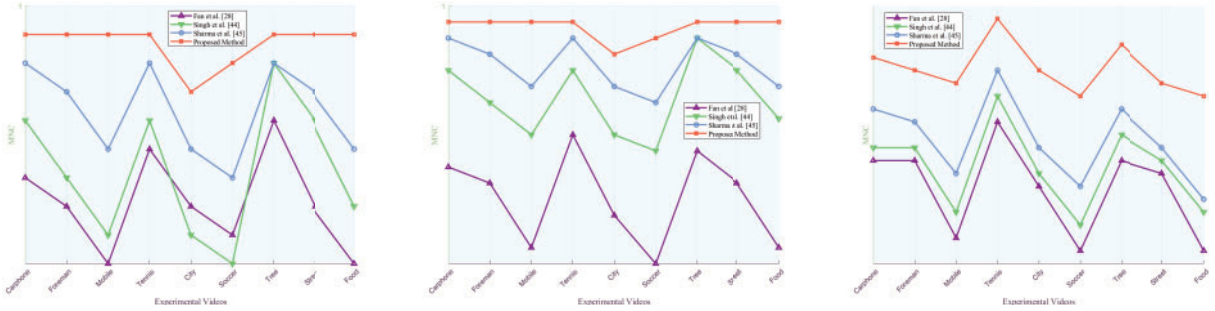


(e) MBER with attack strength 20%

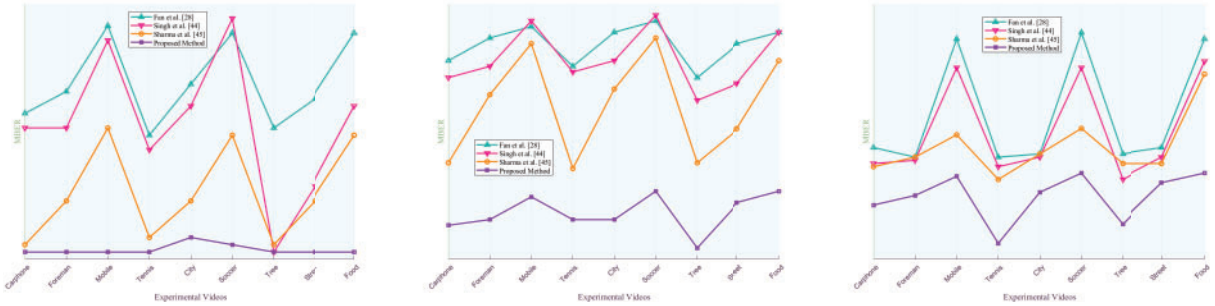


(f) MBER with attack strength 25%

Figure 12: Comparison of MNC and MBER values of the proposed algorithm against three algorithms under various intensities of frame average attack

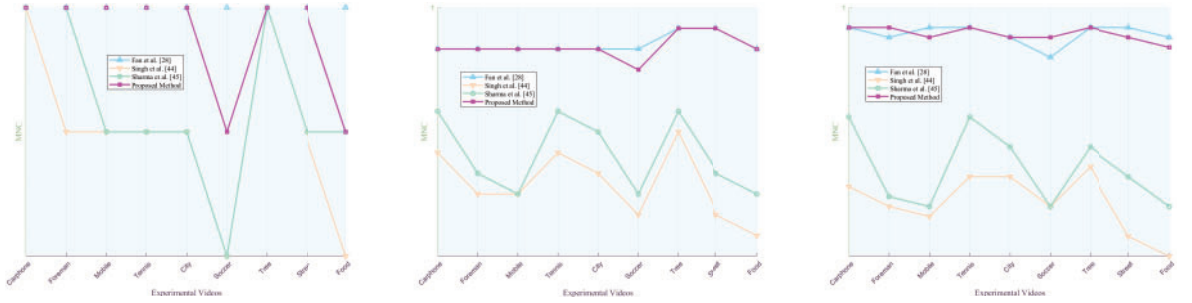


(**a**) MNC with attack strength 10% (**b**) MNC with attack strength 20% (**c**) MNC with attack strength 25%

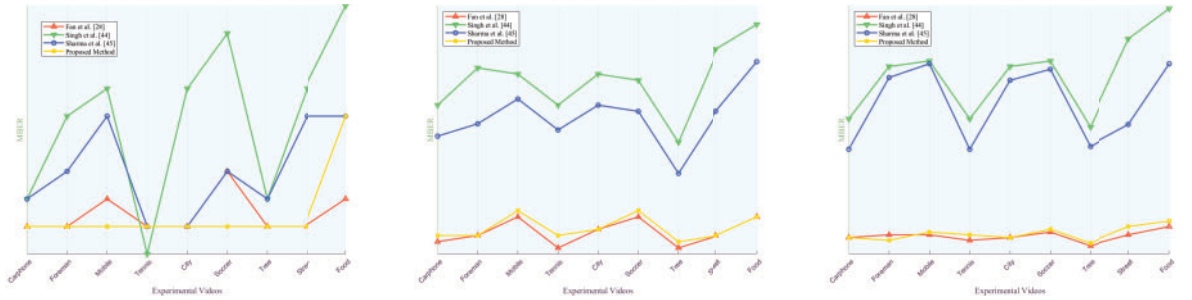


(**d**) MBER with attack strength 10% (**e**) MBER with attack strength 20% (**f**) MBER with attack strength 25%

Figure 13: Comparison of MNC and MBER values of the proposed algorithm against three algorithms under various intensities of frame swapping attack



(**a**) MNC with attack strength QP:26 (**b**) MNC with attack strength QP:27 (**c**) MNC with attack strength QP:28



(**d**) MBER with attack strength QP:26 (**e**) MBER with attack strength QP:27 (**f**) MBER with attack strength QP:28

Figure 14: Comparison of MNC and MBER values of the proposed algorithm against three algorithms under various intensities of recompression attack

Figs. 8 to 14 present the MNC and MBER values of the proposed algorithm and three comparative video watermarking algorithms under various attack scenarios, including Gaussian noise, salt and pepper noise, blur, Frame Dropping, Frame Averaging, Frame Swapping, and recompression. The analysis of these tables indicates that the algorithm introduced in this study exhibits superior performance in withstanding attacks on traditional videos such as Carphone, Foreman, Tennis, and the short video Tree, characterized by smoother scenes, slower changes, and less content changes. This superiority is evident in the algorithm's ability to minimize the decrease in MNC values and limit the increase in MBER values under adverse conditions, showcasing its robustness in preserving watermark integrity.

The algorithm under study demonstrates robustness against traditional noise attacks, particularly Gaussian noise and salt and pepper noise, as evidenced by MNC values consistently exceeding 0.9946 and MBER values remaining below 0.0571. Notably, the algorithm's resilience diminishes as noise intensity increases, reflected in the decreasing trend of MNC values and the escalating trend of MBER values. In the context of fuzzy attacks, the algorithm excels in robustness, achieving a peak MNC value of 1 and maintaining MBER values below 0.001. This heightened resilience can be attributed to the algorithm's ability to mitigate the impact of fuzzy attacks on pixel values within experimental video frames through frequency domain transformation, enabling accurate watermark extraction despite alterations in pixel values across different regions.

When confronted with frame attacks in experimental videos, the methods proposed by Fan et al., Singh et al., Sharma et al., and our own method all exhibit enhanced resilience compared to traditional noise attacks. Each of these algorithms serves as a video watermarking technique designed to identify the most appropriate frame within the video for embedding, showcasing robustness against frame-based adversarial scenarios. Specifically, the algorithm introduced in this study excels in countering frame average attacks, achieving MNC values exceeding 0.9994 and MBER values below 0.0016, thereby demonstrating substantial robustness. Despite the pixel averaging process employed in frame average attacks, the influence on the watermark post-frequency domain transformation remains relatively minor, as the errors introduced by the attack fall within the error correction capabilities of the error correcting code. When faced with escalating Frame Dropping attacks, the algorithm presented in this study exhibits a noticeable weaken in both MNC and MBER values. Notably, among the various video types analyzed, the experimental results for traditional videos such as Mobile and short video Food show a more pronounced weakening effect. This can be attributed to the rapid scene transformations, lower image smoothness, and closer proximity of hidden frames within these particular videos. As the number of frames extracted increases, the likelihood of the hidden frame containing the watermark being compromised rises, consequently elevating the risk of losing the marked frame. Such occurrences can result in misalignment during subsequent watermark extraction processes. While errors remain within correctable limits, misalignment can impact extraction accuracy to some extent, leading to a reduction in MNC values and an increase in MBER values. Nonetheless, the minimum MNC value remains above 0.9988, and the maximum MBER value stays below 0.0042. When confronted with varying intensities of Frame Swapping attacks, the algorithm under study also demonstrates a weakening trend, albeit with a smaller magnitude compared to Frame Dropping attacks. Frame Dropping attacks result in the loss of hidden frames within the video, leading to potential misalignment and errors in subsequent watermark extraction processes. Specifically, the loss of a hidden frame can cause subsequent frames to shift forward, disrupting the embedding order before and after the missing frame. As the intensity of Frame Dropping attacks escalates, the misalignment of subsequent hidden frames may occur, impacting the accuracy of watermark extraction. In contrast, when facing Frame Swapping attacks, the total number of frames remains constant, mitigating the

risk of losing hidden frames. The algorithm's robustness against Frame Swapping attacks surpasses that against Frame Dropping attacks, as the consistent number of frames ensures that hidden frames are preserved to a certain extent. This results in MNC values above 0.9989 and MBER values below 0.0037, indicating a higher level of resilience against Frame Swapping attacks.

The experimental results demonstrate the robustness of the video watermarking algorithm against recompression attacks of various QP. Notably, when comparing the recompression effects of QP = 26 and QP = 28, it is observed that the recompression with QP = 26 yields better results due to the lower quantization steps resulting in less distortion. The final MNC value remains above 0.9996, while the MBER value stays below 0.0012, indicating the algorithm's resilience to QP recompression attacks. Specifically, the original QP of 26 showcases superior performance compared to QP 28, highlighting the importance of quantization parameters in maintaining watermark integrity during recompression processes. These findings underscore the algorithm's effectiveness in preserving watermark robustness under varying QP settings.

In order to evaluate the performance superiority of the proposed algorithm compared with the current development, the proposed algorithm was compared with three reference video robust watermarking algorithms [28,44,45], and the attack intensity and related parameter Settings were consistent with those in Subsection 5.2. Using the experimental video Carphone as a case study, Figs. 15 and 16 illustrate the comparative summary of average MNC and MBER values for the algorithm proposed in this paper and the three comparison algorithms across various attack intensities. Each numerical column denotes the average MNC and MBER values of the watermarking algorithm under three distinct attack intensities.

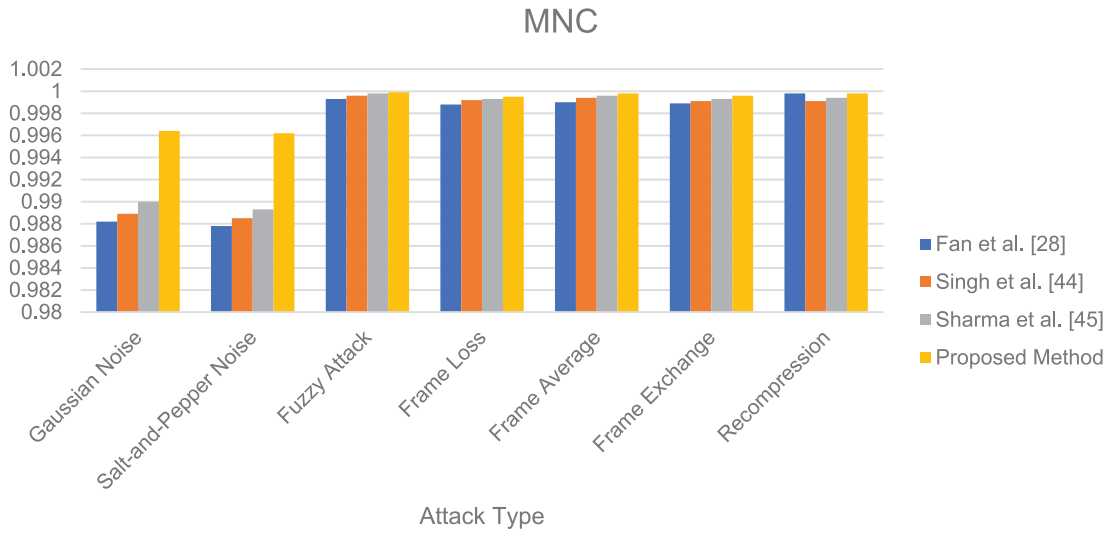


Figure 15: Analysis of average MNC values for the algorithm and three comparison algorithms in response to varied attack intensities using the experimental video "Carphone" as a case study

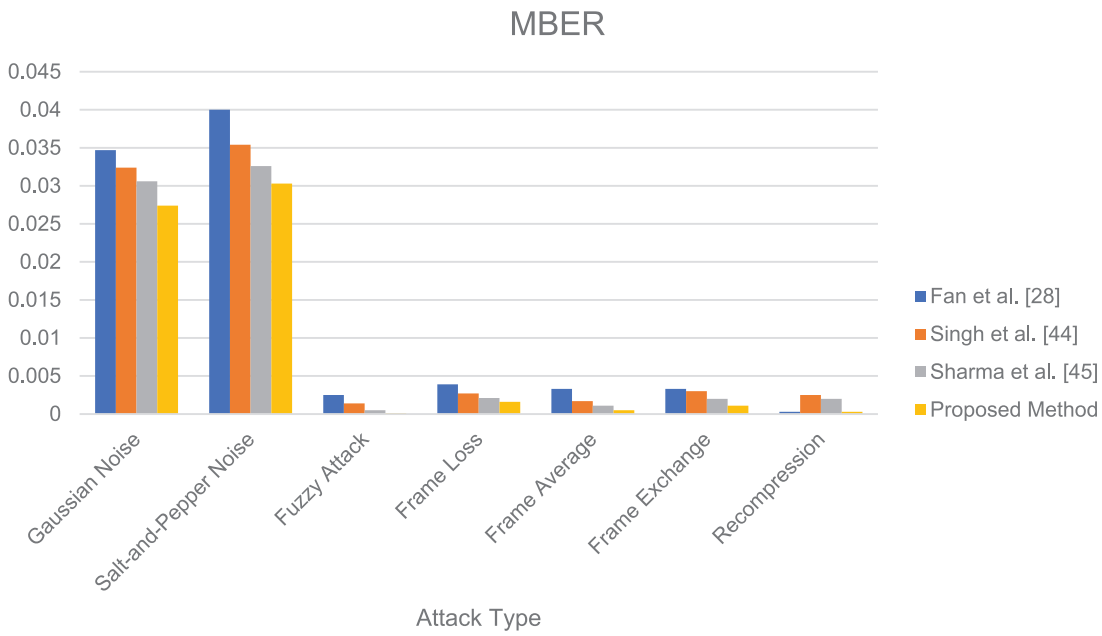


Figure 16: Analysis of average MBER values for the algorithm and three comparison algorithms in response to varied attack intensities using the experimental video "Carphone" as a case study

The results presented in Figs. 8 to 16 indicate that our proposed algorithm outperforms the three comparison algorithms in terms of MNC and MBER values under equivalent attack types and intensities, showcasing superior robustness. Notably, Fan et al.'s method exhibits comparable robustness against re-compression attacks and outperforms other attack types compared to our algorithm. This can be attributed to Fan's approach of selecting sub-blocks for watermark embedding based on video frequency and watermark characteristics, enhancing robustness, and maintaining invisibility in video watermarking. The key distinction between our algorithm and Fan et al. lies in the sub-block selection criteria: Fan's method is based on chromaticity and modulation, while our algorithm detects, and groups frame smoothness based on brightness and chromaticity histogram differences. Ultimately, our algorithm selects the frame with the least disparity in brightness and chromaticity histograms within the same group for embedding, achieving heightened robustness and invisibility in video watermarking.

5.2.2 Robustness Analysis of Online Attacks

In order to evaluate the robustness of the proposed algorithm against online attacks, this subsubsection applies Class II collusion attacks in Subsection 5.2 to nine experimental videos based on the proposed algorithm. Specific practices are as follows:

1. First, the video frames in the same frame group of the experimental video obtained through SSGM are obtained and approximate as the carrier and corresponding copy in the Class II collusion attack. Note that the experimental video at this time has been marked with dual-domain watermarks.

2. Use pseudorandom number generator [46]. Generate pseudo-random bit strings as pseudo-watermarks in Class II collusion attacks. Note that the number of pseudo-watermarks generated should be the same as the number of video frames in the experimental video.

3. Embed the generated watermark into each frame of each frame group one by one to obtain the damaged video. Note that the embedding method is the same as the robust splicing technique used in this algorithm.

4. Extract the robust watermark from the distorted video, calculate the NC value and BER between the extracted robust watermark and the original robust watermark in each hidden frame, and finally calculate the average NC value and the average BER value of all hidden frames, MNC and MBER, as standards to evaluate the robustness of the proposed algorithm against Class II collusion attacks.

The experimental results obtained are shown in Table 5.

Table 5: Under Class II collusion attacks, the MNC and MBER values of the three comparison algorithms are compared

Video	Fan et al. [28]		Singh et al. [44]		Sharma et al. [45]		Proposed method	
	MNC	MBER	MNC	MBER	MNC	MBER	MNC	MBER
Carphone	0.8677	0.1835	0.8784	0.1832	0.8875	0.1784	0.8864	0.1789
Foreman	0.8658	0.2216	0.8761	0.2209	0.8847	0.2168	0.8841	0.2171
Mobile	0.8587	0.2799	0.8717	0.2786	0.8797	0.2755	0.8791	0.2732
Tennis	0.8702	0.1747	0.8817	0.1743	0.8880	0.1703	0.8894	0.1686
City	0.8626	0.2176	0.8735	0.2165	0.8816	0.2132	0.8827	0.2109
Soccer	0.8552	0.2694	0.8670	0.2684	0.8744	0.2624	0.8753	0.2627
Tree	0.8767	0.1997	0.8872	0.1991	0.8952	0.1931	0.8952	0.1922
Street	0.8639	0.2591	0.8743	0.2578	0.8839	0.2527	0.8842	0.2524
Food	0.8604	0.2944	0.8711	0.2927	0.8798	0.2866	0.8799	0.2864

In the face of online attacks, the MNC value between the extracted watermark and the original watermark is 0.8840, and the MBER value is 0.2269. As can be seen from the data in Table 5, the weakening of MNC value and MBER value of the proposed algorithm in the face of Class II collusion attacks is much greater than that of MNC value and MBER value in offline attacks of Subsection 5.2.1, which is because, a Type II collusion attack targets all video frames in an experimental video, affecting all initially embedded legal dual-domain watermark bits. In contrast, offline attacks do not impact all the legal watermarks due to variations in attack strengths. Notably, the embedding method of pseudo-random watermark bits aligns with that of legal dual-domain watermarks, leading to the pseudo-random watermark bits overriding the legal ones. Consequently, Type II collusion attacks prove more effective against legitimate dual-domain watermarks compared to offline attacks. This maximizes the impact of the watermark, significantly reducing the proposed algorithm's robustness against online attacks compared to offline attacks. However, according to the performance range of NC value and BER value, when NC value is greater than 0.85 and BER value is less than 0.35, it indicates that the extracted watermark still has strong robustness and can be clearly identified [47]. Therefore, the proposed algorithm is still robust against Class II collusion attacks. It is worth noting that the performance of the three compared robust watermarking algorithms is not bad, because the three comparison algorithms, like the proposed algorithms, are all algorithms that select video frames for robust watermark embedding. A prerequisite for the successful attack of Class

II collusion attacks is that the attacked algorithm needs to embed a large number of video frames. Therefore, the three algorithms are also robust to collusion attacks to a certain extent.

It is worth mentioning that Class II collusion attacks require the following assumptions:

1. Each hidden frame has the same mean and variance.

2. Robust watermarking follows Gaussian distribution, and the mean value is 0.

3. Hidden frame and robust watermark are independent of each other.

From the idea of the proposed algorithm, the robust watermark bits are embedded in the DC coefficients of all macroblocks in several hidden frames, which will suffer loss after H.264 decoding, so the assumption of Gaussian distribution will not be satisfied. Therefore, the attacker cannot theoretically obtain an estimated version of the video that completely removes the robust watermark, and the premise of "collusion" is lost.

Finally, according to the above analysis, the algorithm proposed in this paper can effectively resist Class II collusion attacks.

5.3 Computational Complexity Assessment

In assessing the practical efficiency of the proposed algorithm for real-world applications, this subsection delves into the computational complexity of the algorithm. The complexity analysis experiment involved measuring the time taken for encoding and decoding both before and after watermark embedding in seconds. Table 6 presents a comparison of the encoding and decoding times for the original video and the watermarked video. The results reveal that the encoding time difference for the experimental video is minimal, at less than 5.462 s, both before and after watermark embedding. Similarly, the decoding time difference does not exceed 3.758 s, indicating a low level of complexity.

Table 6: Comparison of encoding and decoding times and differences between original and watermarked videos

Video	Original video encoding time	Watermarked video encoding time	Difference	Original video decoding time	Watermarked video decoding time	Difference
Carphone	137.714	140.638	2.924	49.363	50.050	0.687
Foreman	139.143	141.832	2.689	51.427	52.096	0.669
Mobile	142.276	144.619	2.343	55.912	57.371	1.459
Tennis	139.372	144.834	5.462	51.411	54.004	2.593
City	140.617	145.921	5.304	52.799	55.378	2.579
Soccer	145.559	149.485	3.926	58.131	60.889	2.758
Tree	238.547	243.396	4.849	115.543	118.252	2.709
Street	240.723	244.189	3.466	118.376	122.134	3.758
Food	243.893	248.337	4.444	122.751	125.994	3.243

6 Conclusion

This article introduces a robust reversible watermarking algorithm tailored for frame-grouped videos based on scene smoothness. The algorithm preprocesses the watermark and auxiliary

information and embeds the high-frequency and low-frequency coefficients of the U channel of the transformation coefficients to achieve robust reversible watermark embedding. By employing separate embedding stages for low and high frequencies, the algorithm ensures high robustness in the initial stage while maintaining low distortion in the subsequent stage. This approach enables high error correction capabilities for watermarks, mitigates the impact of reversible embedding on watermark robustness, and efficient extraction of robust watermarks even in the presence of attacks. Compared with the three comparison algorithms, the performance of PSNR and SSIM is improved by 7.59% and 0.4% on average, while the performance of NC and BER is enhanced by 1.27% and 18.16% on average. The experimental findings demonstrate that the proposed algorithm outperforms several state-of-the-art keyframe embedding video watermarking algorithms, meeting the requirements of practical applications, and offering effective copyright protection for videos. A significant advantage lies in the algorithm's ability to achieve high fidelity across videos of varying sizes through scene smoothness grouping, coupled with low computational complexity for convenient and rapid application to existing videos. Future research directions may explore the design of more sophisticated mechanisms for frame grouping and embedding selection to enhance robustness further.

In the future work, we will try to design a smoother frame grouping standard and a more reasonable and accurate embedding frame selection mechanism, and the embedding position is accurate to the block, further reducing the amount of modifications to the unit frame, and achieving progress in invisibility. Although robust watermark splicing technology has strong resistance to aggression, it needs threshold to control the embedding strength, and there is a problem that the amount of original frame modification per unit watermark bit is large. Therefore, the subsequent work will also take the improvement of robust watermark embedding technology as another focus of development and propose a watermark embedding technology with a smaller amount of modification per unit watermark bit. At the same time, strengthen its robustness, and better balance the contradiction and balance between invisibility and robustness. At the same time, it is planned to find a frequency domain transformation with less computational complexity, which can better reduce the computational complexity and strengthen the development prospect and status of future work in practical applications.

Acknowledgement: We thank all the members who have contributed to this work with us.

Funding Statement: This work was supported in part by the National Natural Science Foundation of China under Grants 62202496, 62272478 and the Basic Frontier Innovation Project of Engineering university of People Armed Police under Grants WJY202314, WJY202221.

Author Contributions: Conceptualization: Yucheng Liang, Ke Niu; Experimental operation and data proofreading: Yucheng Liang, Yingnan Zhang; Analysis and interpretation of results: Yucheng Liang, Ke Niu, Yingnan Zhang; Draft Manuscript preparation: Yucheng Liang, Yifei Meng; Figure design and drawing: Yucheng Liang, Ke Niu, Yifei Meng. All authors read and approved the final manuscript.

Availability of Data and Materials: Given that the source code and data contain research findings that have not yet been publicly disseminated by our experimental team, we currently face challenges in making them openly available. Moreover, our academic institution is governed by confidentiality protocols that necessitate the disclosure of the source code and data only after the stipulated decryption period has been met.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1] B. Ma, K. Li, J. Xu, C. Wang, J. Li and L. W. Zhang, "Enhancing the security of image steganography via multiple adversarial networks and channel attention modules," *Digit. Signal Process.*, vol. 141, no. 7, pp. 104121, 2023. doi: 10.1016/j.dsp.2023.104121.

[2] W. Diffie and M. E. Hellman, "New directions in cryptography," in *Democratizing Cryptography: The Work of Whitfield Diffie and Martin Hellman*. New York, NY, USA: Association for Computing Machinery, 2022, pp. 365–390. doi: 10.1145/3549993.3550007.

[3] J. Fridrich, *Steganography in Digital Media: Principles, Algorithms, and Applications*. Cambridge, UK: Cambridge University Press, 2009. doi: 10.1017/CBO9781139192903.

[4] I. Cox, M. Miller, and J. Bloom, *Digital Watermarking and Steganography*. San Francisco, CA, USA: Morgan Kaufmann, 2007. doi: 10.1016/B978-0-12-372585-1.X5001-3.

[5] S. Subramani and S. K. Svn, "Review of security methods based on classical cryptography and quantum cryptography," *Cybern. Syst.*, vol. 2, no. 2, pp. 1–19, 2023. doi: 10.1080/01969722.2023.2166261.

[6] S. Rustad, P. N. Andono, and G. F. Shidik, "Digital image steganography survey and investigation (goal, assessment, method, development, and dataset)," *Signal Process.*, vol. 206, no. 3, pp. 108908, 2023. doi: 10.1016/j.sigpro.2022.108908.

[7] R. Singh, M. Saraswat, and A. Ashok, "From classical to soft computing based watermarking techniques: A comprehensive review," *Future Gener. Comput. Syst.*, vol. 141, no. 2, pp. 738–754, 2023. doi: 10.1016/j.future.2022.12.015.

[8] E. Gul and A. N. Toprak, "Contourlet and discrete cosine transform based quality guaranteed robust image watermarking method using artificial bee colony algorithm," *Expert Syst. Appl.*, vol. 212, no. 1, pp. 118730, 2023. doi: 10.1016/j.eswa.2022.118730.

[9] A. Soualmi, L. Laouamer, and A. Adel, "A blind watermarking approach based on hybrid imperialistic competitive algorithm and SURF points for color images' authentication," *Biomed. Signal Process. Control*, vol. 84, no. 3, pp. 105007, 2023. doi: 10.1016/j.bspc.2023.105007.

[10] A. Soualmi, A. Benhociner, and I. Midoun, "Artificial bee colony-based blind watermarking scheme for color images alter detection using BRISK features and DCT," *Arab. J. Sci. Eng.*, vol. 49, no. 3, pp. 3253–3266, 2023. doi: 10.1007/s13369-023-07958-8.

[11] P. Aberna and L. Agilandeeswari, "Digital image and video watermarking: Methodologies, attacks, applications, and future directions," *Multimed. Tools Appl.*, vol. 83, no. 2, pp. 5531–5591, 2024. doi: 10.1007/s11042-023-15806-y.

[12] X. Yu, C. Wang, and X. Zhou, "A survey on robust video watermarking algorithms for copyright protection," *Appl. Sci.*, vol. 8, no. 10, pp. 1891, 2018. doi: 10.3390/app8101891.

[13] M. Asikuzzaman and M. R. Pickering, "An overview of digital video watermarking," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 28, no. 9, pp. 2131–2153, 2017. doi: 10.1109/TCSVT.2017.2712162.

[14] J. Ouyang, J. Huang, and X. Wen, "A semi-fragile reversible watermarking method based on qdft and tamper ranking," *Multimed. Tools Appl.*, vol. 83, no. 14, pp. 41555–41578, 2023. doi: 10.1007/s11042-023-16963-w.

[15] L. Tanwar and J. Panda, "Hybrid reversible watermarking algorithm using histogram shifting and pairwise prediction error expansion," *Multimed. Tools Appl.*, vol. 83, no. 8, pp. 22075–22097, 2024. doi: 10.1007/s11042-023-15508-5.

[16] R. Ogla, E. S. Mahmood, R. I. Ahmed, and A. M. S. Rahma, "New fragile watermarking technique to identify inserted video objects using H.264 and color features," *Comput. Mater. Contin.*, vol. 76, no. 3, pp. 1–22, 2023. doi: 10.32604/cmc.2023.039818.

[17] Y. Q. Shi, X. Li, X. Zhang, H. T. Wu, and B. Ma, "Reversible data hiding: Advances in the past two decades," *IEEE Access*, vol. 4, pp. 3210–3237, 2016. doi: 10.1109/ACCESS.2016.2573308.

[18] Y. F. Wang, Y. M. Zhou, Z. X. Qian, S. Li, and X. Zhang, "Review of robust video watermarking," *J. Chin. Image Graphics*, vol. 27, no. 1, pp. 27–42, 2022.
[19] T. S. Nguyen, "Reversible data hiding scheme based on coefficient pair mapping for videos H.264/AVC without distortion drift," *Symmetry*, vol. 14, no. 9, pp. 1768, 2022. doi: 10.3390/sym14091768.
[20] D. Coltuc, "Towards distortion-free robust image authentication," *J. Phys.*, vol. 77, no. 1, pp. 012005, 2007. doi: 10.1088/1742-6596/77/1/012005.
[21] D. Coltuc and J. M. Chassery, "Distortion-free robust watermarking: A case study," presented at the Secur., Steganogr., Watermark. Multimed. Contents IX, San Jose, USA, Feb. 27, 2007.
[22] X. Wang, X. Li, and Q. Pei, "Independent embedding domain based two-stage robust reversible watermarking," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 30, no. 8, pp. 2406–2417, 2019. doi: 10.1109/TCSVT.2019.2915116.
[23] G. Gao, M. Wang, and B. Wu, "Efficient robust reversible watermarking based on ZMs and integer wavelet transform," *IEEE Trans. Ind. Inform.*, vol. 20, no. 3, pp. 4115–4123, 2023. doi: 10.1109/TII.2023.3321101.
[24] Y. Tang, C. Wang, S. Xiang, and Y. Cheung, "A robust reversible watermarking scheme using attack-simulation-based adaptive normalization and embedding," *IEEE Trans. Inf. Forensics Secur.*, vol. 19, no. 1, pp. 4114–4129, 2024. doi: 10.1109/TIFS.2024.3372811.
[25] B. Ma, Z. Tao, R. Ma, C. Wang, J. Li and X. Li, "A high-performance robust reversible data hiding algorithm based on polar harmonic fourier moments," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 34, no. 4, pp. 2763–2774, 2023. doi: 10.1109/TCSVT.2023.3311483.
[26] H. T. Hazim, N. Alseelawi, and H. T. H. ALRikabi, "A novel method of invisible video watermarking based on index mapping and hybrid DWT-DCT," *Int. J. Online Biomed. Eng.*, vol. 19, no. 4, pp. 155–173, 2023.
[27] S. Chen, A. Malik, X. Zhang, G. Feng, and H. Wu, "A fast method for robust video watermarking based on Zernike moments," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 33, no. 12, pp. 7342–7353, 2023. doi: 10.1109/TCSVT.2023.3281618.
[28] D. Fan, H. Zhao, C. Zhang, H. Liu, and X. Wang, "Anti-recompression video watermarking algorithm based on H. 264/AVC," *Mathematics*, vol. 11, no. 13, pp. 2913, 2023.
[29] S. Takale and A. Mulani, "DWT-PCA based video watermarking," *J. Electron., Comput. Netw. Appl. Math.*, vol. 2, no. 6, pp. 1–7, 2022. doi: 10.55529/jecnam.
[30] E. Farri and P. Ayubi, "A robust digital video watermarking based on CT-SVD domain and chaotic DNA sequences for copyright protection," *J. Ambient Intell. Humaniz. Comput.*, vol. 14, no. 10, pp. 13113–13137, 2023. doi: 10.1007/s12652-022-03771-7.
[31] X. Y. Tao, L. Z. Xiong, and X. Zhang, "Robust video watermarking scheme based on QDCT global equalization strategy," *Comput. Sci.*, vol. 50, no. 11, pp. 168–176, 2023.
[32] Y. Zhou, Q. Ying, Y. Wang, X. Zhang, Z. Qian and X. Zhang, "Robust watermarking for video forgery detection with improved imperceptibility and robustness," presented at the 2022 IEEE 24th Int. Workshop Multimed. Signal Process (MMSP), Shanghai, China, Sep. 26–28, 2022.
[33] X. H. Huang, "The research on digital video watermarking algorithm based on H.264 Bitstream," M.S. thesis, Overseas Chinese Univ., Quanzhou, China, 2024.
[34] W. Jin, J. D. Liu, and H. Q. Xu, "H.264/AVC video encryption algorithm based on integer dynamic cross-coupling tent mapping model," *Multimed. Tools Appl.*, vol. 83, no. 5, pp. 13369–13393, 2023. doi: 10.1007/s11042-023-15448-0.
[35] I. S. Reed and G. Solomon, "Polynomial codes over certain finite fields," *J. Soc. Ind. Appl. Math.*, vol. 8, no. 2, pp. 300–304, 1960. doi: 10.1137/0108018.
[36] A. J. Calabria and M. D. Fairchild, "Perceived image contrast and observer preference I. The effects of lightness, chroma, and sharpness manipulations on contrast perception," *J. Imaging Sci. Technol.*, vol. 47, no. 6, pp. 479–493, 2003. doi: 10.2352/J.ImagingSci.Technol.2003.47.6.art00006.
[37] P. Kadian, S. M. Arora, and N. Arora, "Robust digital watermarking techniques for copyright protection of digital data: A survey," *Wirel. Pers. Commun.*, vol. 118, no. 4, pp. 3225–3249, 2021. doi: 10.1007/s11277-021-08177-w.

[38] Q. Wang *et al.*, "An overview on digital content watermarking," presented at the Signal Inf. Process., Netw. Comput., Shangdong, China, Sep. 13–17, 2021.
[39] Y. A. Hassan and A. M. S. Rahmah, "An overview of robust video watermarking techniques," *Iraqi J. Sci.*, vol. 64, no. 7, pp. 4513–4524, 2023. doi: 10.24996/ijs.2023.64.7.38.
[40] H. Lakshmi and S. Borra, "Digital video watermarking tools: An overview," *Int. J. Inf. Comput. Secur.*, vol. 16, no. 1–2, pp. 1–19, 2021. doi: 10.1504/IJICS.2021.117391.
[41] P. Garg and R. R. Kishore, "Performance comparison of various watermarking techniques," *Multimed. Tools Appl.*, vol. 79, no. 35–36, pp. 25921–25967, 2020. doi: 10.1007/s11042-020-09262-1.
[42] V. B. Giffen, D. Herhausen, and T. Fahse, "Overcoming the pitfalls and perils of algorithms: A classification of machine learning biases and mitigation methods," *J. Bus. Res.*, vol. 144, no. 6, pp. 93–106, 2022. doi: 10.1016/j.jbusres.2022.01.076.
[43] Y. Hu, H. K. Lee, and J. Li, "DE-based reversible data hiding with improved overflow location map," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 19, no. 2, pp. 250–260, 2008.
[44] R. Singh, H. Mittal, and R. Pal, "Optimal keyframe selection-based lossless video-watermarking technique using IGSA in LWT domain for copyright protection," *Complex Intell. Syst.*, vol. 8, no. 2, pp. 1047–1070, 2022. doi: 10.1007/s40747-021-00569-6.
[45] C. Sharma, B. Amandeep, R. Sobti, T. K. Lohani, and M. Shabaz, "A secured frame selection based video watermarking technique to address quality loss of data: Combining graph based transform, singular valued decomposition, and hyperchaotic encryption," *Secur. Commun. Netw.*, vol. 2021, pp. 5536170, 2021. doi: 10.1155/2021/5536170.
[46] B. A. Wichmann and I. D. Hill, "Generating good pseudo-random numbers," *Comput. Stat. Data Anal.*, vol. 51, no. 3, pp. 1614–1622, 2006. doi: 10.1016/j.csda.2006.05.019.
[47] S. Chen, Q. Su, and H. Wang, "An improved blind watermarking method facing dual color images based on Hadamard transform," *Soft Comput.*, vol. 27, no. 17, pp. 12517–12538, 2023. doi: 10.1007/s00500-023-07898-3.