**TELKOMNIKA Telecommunication Computing Electronics and Control**
Vol. 23, No. 6, December 2025, pp. 1476~1494
ISSN: 1693-6930, DOI: 10.12928/TELKOMNIKA.v23i6.27211 

# Watermarking on spread multi-frame data video using discrete wavelet transform hybrid and frame ratio message variance

**Ilham Firman Ashari, Sarwono Sutikno**
Department of Informatics Engineering, Faculty of Industrial Technology, Institut Teknologi Sumatera, Lampung Selatan, Indonesia

**Article Info**

***Article history:***

Received May 12, 2025
Revised Sep 23, 2025
Accepted Oct 19, 2025

***Keywords:***

Discrete wavelet transform
Frame ratio
Spread multi-frame
Video
Watermarking

**ABSTRACT**

The exponential growth of video sharing demands secure and imperceptible watermarking methods. This study presents a video watermarking framework using discrete wavelet transform (DWT) with hybrid sub-band embedding and multi-frame allocation to balance imperceptibility, capacity, and robustness. Watermark bits are adaptively distributed across low-low (LL), low-high (LH), and high-low (HL) sub-bands of selected frames, with uncompressed audio video interleave (AVI) ensuring coefficient integrity. Experiments on 640×360 videos show LL-only embedding achieves high imperceptibility (peak signal-to-noise ratio (PSNR) > 38.6 dB, structural similarity index measure (SSIM) ≥ 0.9945, and bit error rate (BER) = 0), while LL-dominant hybrids increase capacity with slight robustness trade-offs. Embedding in LH and high-high (HH) sub-bands raises distortion vulnerability. Under cropping, BER rises from 0.005 to 0.205 (0–50%), and normalized correlation (NC) drops from 0.998 to 0.802, remaining acceptable for ≤30% cropping. The scheme resists joint photographic experts' group (JPEG) compression quality factor 20–80 (Q20–Q80), resizing (≥70%), and mild Gaussian blur (3×3), maintaining efficient decoding under higher payloads. Future work may apply error-correction coding and redundancy-aware embedding for improved resilience. Overall, the proposed method offers a secure, adaptive, and efficient solution for video authentication and covert communication.



***Corresponding Author:***

Ilham Firman Ashari
Department of Informatics Engineering, Faculty of Industrial Technology, Institut Teknologi Sumatera
St. Terusan Ryacudu, Lampung Selatan, Indonesia
Email: firman.ashari@if.itera.ac.id

## 1. INTRODUCTION

In the contemporary digital landscape, the exponential growth of video content distribution across national and international platforms has heightened the demand for robust content protection mechanisms [1]. As video becomes a dominant medium for communication, education, entertainment, and commercial exchange, it also increasingly becomes a target for unauthorized duplication, tampering, and misuse [2]. This growing vulnerability underscores the critical need for advanced techniques capable of safeguarding intellectual property and verifying content authenticity.

Digital watermarking has emerged as a viable technological response to these challenges, offering a method to embed imperceptible identification data within multimedia content [3]. Particularly in the domain of video, red, green, blue (RGB) watermarking techniques have attracted significant attention due to their potential to preserve visual fidelity while providing layers of security and traceability [4], [5]. However, despite its promise, watermarking in video remains a complex task, confronted with issues such as perceptual degradation, fragility under common signal attacks (compression, scaling, frame manipulation), and the need for adaptive embedding schemes that balance robustness and invisibility [6], [7].

To address these concerns, contemporary research has increasingly explored hybrid and transform-based watermarking approaches. Among these, the discrete wavelet transform (DWT) has proven especially effective, owing to its multi-resolution representation and frequency localization properties [8], [9]. DWT-based schemes allow watermarks to be embedded in coefficients less prone to perceptual loss, thereby enhancing invisibility and robustness [10]. Steganography, in this context, refers to embedding hidden information within digital media such as images, audio, and video without attracting attention, differing from cryptography which focuses on encryption [11]. An effective system must ultimately provide high capacity, imperceptibility, and resilience against signal processing or deliberate attacks [12], [13].

Two major methodological branches in steganography and watermarking are the spatial domain and transform domain techniques [14], [15]. Spatial domain methods operate directly on pixel values or temporal samples, often offering simplicity but limited robustness [16]. Transform domain approaches, by contrast, embed data in transformed representations such as frequency or wavelet domains, thereby offering enhanced durability and improved imperceptibility [17]. Recent studies highlight deep learning as a powerful direction for video watermarking but also expose key limitations. Bistroń and Piotrowski [18] introduced a convolutional neural network (CNN)-based entropy-driven mapper that improved robustness against compression and geometric distortions, yet its embedding capacity was restricted and computational demand remained high. Likewise, Mansour *et al.* [19] proposed a mosaic generation approach that demonstrated strong resistance to temporal and collusion attacks, though occasional artifacts reduced imperceptibility.

The decision to use audio video interleave (AVI) format as the video carrier stems from its uncompressed or lightly compressed structure, which retains greater pixel-level fidelity compared to heavily compressed formats like MP4 (moving picture experts group (MPEG)-4 Part 14) [20], [21]. This characteristic is particularly valuable for transform-domain watermarking, where minute alterations in coefficient values are sensitive to re-compression artifacts. AVI also offers frame-by-frame accessibility, which is critical for precise control during watermark insertion and extraction across multiple frames. On the other hand, the watermark payload selected in this study is a plaintext message saved as a ".txt" file. The rationale for using plain text lies in its ubiquity, low computational overhead, and relevance to practical watermarking use cases such as ownership claims, metadata embedding, and confidential annotation [22].

The novelty of this study lies in the hybrid sub-band watermarking strategy applied over multiple video frames, where watermark bits are adaptively distributed across selected DWT sub-bands (low-low (LL), low-high (LH), high-low (HL)) according to frame perceptual and structural characteristics. This multi-frame allocation disperses the payload temporally, reducing detectability, mitigating perceptual artifacts, and increasing resilience against frame loss, cropping, or temporal attacks. In contrast to deep learning-based methods, the proposed frequency-domain approach achieves superior imperceptibility (peak signal-to-noise ratio (PSNR) > 38 dB, structural similarity index measure (SSIM) > 0.99 in LL-only) with lightweight computation. While mid- and high-frequency bands (LH / high-high (HH)) remain vulnerable to distortion, the framework provides an efficient and practical alternative for secure video authentication and covert communication.

To rigorously validate the proposed technique, a comprehensive evaluation was conducted across multiple performance axes [23]–[25]. Embedding capacity was measured for each DWT subband configuration, demonstrating how the hybrid allocation impacts total bits embedded per frame [26], [27]. Computational efficiency was assessed by calculating encoding and decoding runtimes for various frame message ratios and subband combinations [28]. The method was subjected to real-world signal distortions including lossy MPEG compression, Gaussian blur, resizing operations, and additive noise to examine robustness [29]. Accuracy was measured using the bit error rate (BER) to capture bit-level discrepancies between original and extracted messages [30]. PSNR was computed to quantify perceptual degradation, and SSIM was used to evaluate visual similarity per frame [31], [32].

Further, comparative performance was explored by simulating different hybrid subband configurations (LL-only, LH-only, HL-only, and mixed hybrids) and recording their respective encoding time, decoding time, and bit capacity. Experiments also included performance curves relating capacity and runtime to changes in the frame message ratio. Attack-specific evaluations were conducted to assess how each perturbation (compression, resize, noise, blur) influenced decoding success [33]. Metrics such as BER and the number of successfully decoded text lines were reported post-attack, with visual plots illustrating degradation trends. These comprehensive and reproducible evaluations affirm the reliability, adaptability, and practical usability of the proposed multiframe watermarking scheme based on DWT hybrid subband embedding and dynamic frame allocation.

## 2. METHOD

Figure 1 presents the end-to-end workflow of the subband-based video watermarking system, encompassing both embedding and extraction. In the embedding phase, the secret message is binarized and

dispersed across selected frames by modifying least significant bit (LSB) coefficients within DWT subbands. The modified frames are then reconstructed to generate the stego video while preserving visual quality.

During the extraction stage, target frames are identified and decomposed through DWT to retrieve the embedded bits. The recovered bitstream is then validated using an end of file (EOF) marker and converted back into American Standard Code for Information Interchange (ASCII) text to reconstruct the hidden message. This structured process ensures efficient data hiding with a balance of imperceptibility, robustness, and capacity.

### 2.1. Encode

Figure 2 presents a comprehensive overview of the frame selection process in the proposed subband-based video watermarking system. The diagram systematically depicts how the secret message, video properties, and embedding parameters interact to determine the target frames. This structured approach ensures that the message is embedded efficiently while maintaining video quality. Additionally, the dual frame selection mode random or sequential provides flexibility to optimize between robustness and computational cost.

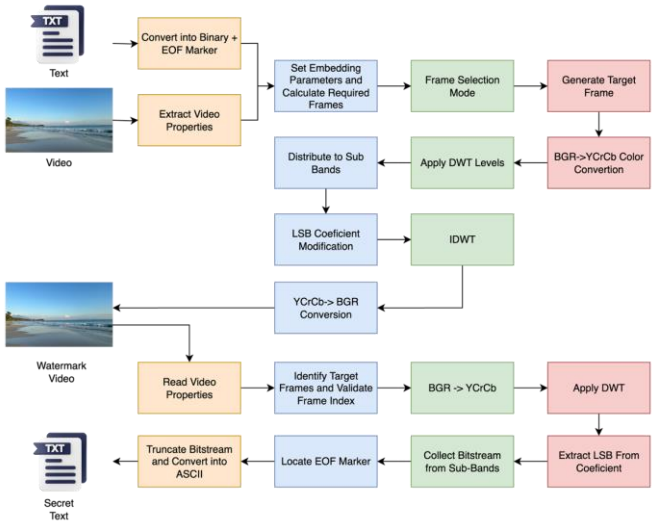



Figure 1. Overview of the proposed method

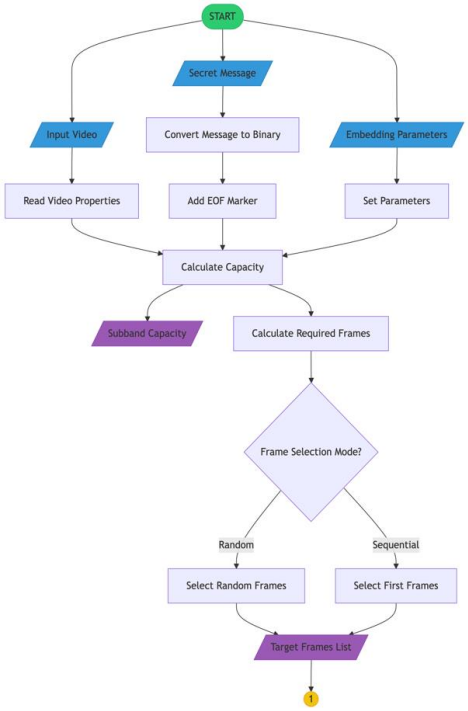



Figure 2. Frame selection process in subband-based video steganography system

The detailed procedure for determining which video frames will carry the embedded watermark is formalized in Algorithm 1, which outlines the steps for converting the message to binary, calculating embedding capacity, and selecting target frames based on predefined or random modes.

**Algorithm 1. Generate target frames (function to select target frames for embedding)**

```
Input: secretMessage, inputVideo, embeddingParams
Output: List of FrameIndex
 1:  Convert secretMessage to binary:
2:      For each character mᵢ in secretMessage:
3:          Convert mᵢ to 8-bit binary → bᵢ
4:      Concatenate all bᵢ and append 32-bit EOF marker
5:      Compute message length: L = 8n + 32
6:  Read video properties from inputVideo:
7:      Extract total frames (N), height (H), and width (W)
8:  Set embedding parameters from embeddingParams:
9:      α = frame message ratio
10:     r_HL, r_HH, r_LL, r_LH = sub-band allocation ratios
```

```
Calculate sub-band capacities:
    C_LL = floor ((H/2 × W/2) × α × r_LL)
    C_LH = floor ((H/2 × W/2) × α × r_LH)
    C_HL = floor ((H/2 × W/2) × α × r_HL)
    C_HH = floor ((H/2 × W/2) × α × r_HH)
    C_total = C_LL + C_LH + C_HL + C_HH
Compute required number of frames:
    N_req = ceil (L / C_total)
Select frame indices based on selection mode:
    If mode = "SEQUENTIAL":
        targetFrames = [0, 1, ..., N_req - 1]
    Else if mode = "RANDOM":
        targetFrames = random_sample ([0, 1, ..., N - 1], N_req, seed=s)
Return targetFrames
```

### 2.1.1. Frame-based relative motion vector calculation

In the proposed framework, the frame-based relative motion vector (FRMV) is employed to quantify temporal stability across consecutive video frames and to guide the selection of embedding regions. For each block in the current frame, a motion vector $MV_t(x,y) = (u,v)$ is estimated by block matching with the reference frame, and the relative motion vector is then defined as can be seen in (1).

$$RMV_t(x,y) = MV_t(x,y) - MV_{t-1}(x,y) \quad (1)$$

The overall FRMV value for frame $t$ is calculated as the mean magnitude of these relative vectors, expressed as can be seen in (2).

$$FRMV_t = \frac{1}{B}\sum_{(x,y)\in Frame} \| RMV_t(x,y) \| = \frac{1}{B}\sum_{(x,y)} \sqrt{(u_t(x,y) - u_{t-1}(x,y))^2 + \left(v_t(x,y) - v_{t-1}(x,y)\right)^2} \quad (2)$$

where $B$ denotes the total number of blocks. A lower FRMV indicates higher temporal consistency and thus provides more reliable candidates for watermark embedding.

Figure 3 illustrates the complete frame embedding procedure implemented in the proposed subband-based video watermarking system, leveraging the DWT. The process of embedding secret bits into selected video frames using DWT-based subband modification is outlined in Algorithm 2, which details the transformation, bit distribution, coefficient modification, and reconstruction steps for generating stego video content.

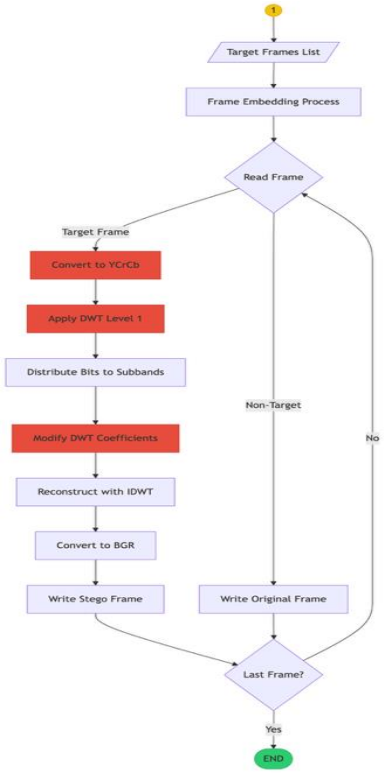



Figure 3. Frame embedding procedure using DWT in subband-based

```
Algorithm 2. Embed message in frames (function to embed secret bits in target frames)
Input: targetFrames, videoFrames, messageBits, embeddingParams
Output: stegoVideo
For each frame fᵢ in videoFrames:
    If fᵢ ∈ targetFrames then
        Convert fᵢ from BGR to YCrCb
        Extract Y, Cr, and Cb channels
        Apply 1-level DWT on Y → obtain LL, LH, HL, HH
        Distribute messageBits based on subband capacity:
            Bits_LL = messageBits[0 : C_LL]
            Bits_LH = messageBits[C_LL : C_LL + C_LH]
           Bits_HL = messageBits[C_LL + C_LH : C_LL + C_LH + C_HL]
        Flatten LL matrix to 1D array
        For each coefficient cᵢ in LL:
            I = floor (cᵢ × 255)
            F = (cᵢ × 255) - I
            I_new = (I AND 0xFE) OR bₖ        // Embed bit bₖ
            cᵢ_new = (I_new + F) / 255
        end for
        Reshape modified LL to 2D
        Reconstruct Y_stego using IDWT (LL_new, LH, HL, HH)
        Convert Y_stego × 255 to uint8 and clip to [0, 255]
        Merge with original Cr and Cb
        Convert YCrCb back to BGR
        Write stego frame to output
    else
        Write original frame to output
    end if
end for
Return stegoVideo
```

### 2.2. Decode

The decoding phase begins by identifying and validating frames within the stego video that potentially contain hidden information as illustrated in Figure 4. This process involves analyzing video properties and selecting specific frames either through predefined parameters or by automatically detecting unique frame features. Once the candidate frames are identified, they are validated to ensure only the correct target frames are used for extracting the hidden message.

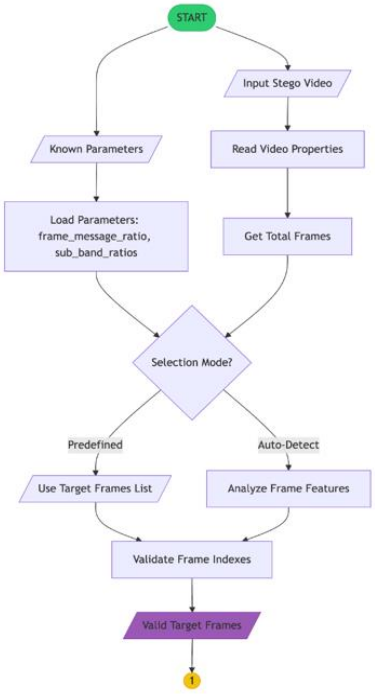



Figure 4. Target frame validation process in stego video analysis

The procedure for identifying which frames in a stego video are eligible for data extraction is detailed in Algorithm 3, which outlines validation steps based on predefined metadata or auto-detection using statistical analysis of frame features.

```
Algorithm 3. Identify valid target frames (function to determine valid frames for
extraction)
Input: stegoVideo, knownParams (optional)
Output: List of Valid Target Frames
Read stegoVideo as sequence of frames:
    V = {F₁, F₂, ..., Fₙ}
Extract video properties:
    - N = total frames = duration × frame rate
    - W, H = frame width and height
Load embedding parameters (if known):
    - α = frame_message_ratio
    - R = {r_LL, r_LH, r_HL, r_HH}, with ∑rᵢ = 1
Determine selection mode:
    If mode = "Predefined":
        T = {t₁, t₂, ..., t_k} from metadata or input
        For each tᵢ in T:
            If 0 ≤ tᵢ < N and rᵢ > 0:
                Add tᵢ to validTargetFrames
    Else if mode = "Auto-Detect":
        For each frame Fᵢ in V:
            Compute statistical features (e.g., DWT energy)
        Compute global mean μ and std deviation σ
        For each Fᵢ:
            If Feature (Fᵢ) > μ + kσ:
                Add i to validTargetFrames
 Return validTargetFrames
```

Once valid target frames have been identified, the system proceeds with extracting the embedded message from each frame by reversing the embedding operations. Figure 5 illustrates the bit extraction process performed on target frames within the proposed DWT-based video watermarking framework.

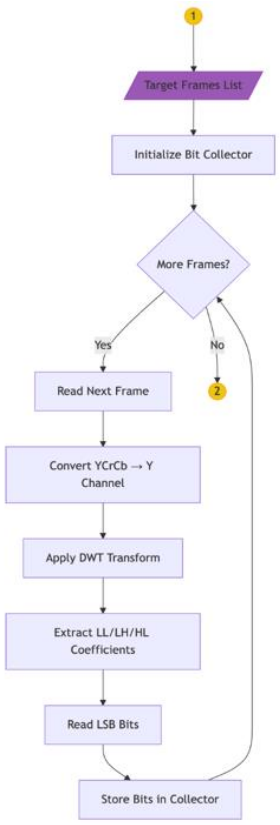



Figure 5. Bit extraction process from target frames in DWT-based video watermarking

The process of retrieving embedded bits from the selected target frames is presented in Algorithm 4, which describes the steps for luminance extraction, DWT decomposition, and LSB-based bit recovery from the LL, LH, and HL subbands.

**Algorithm 4. Extract bits from frames (function to recover embedded bits)**

```
Input: targetFrames, stegoVideo
Output: Bitstream B
Initialize empty bit collector:
    B ← ∅
For each frame index tᵢ in targetFrames:
    Read frame F_tᵢ from stegoVideo
    Convert F_tᵢ to YCrCb color space
    Extract Y channel: Y = ψ(F_tᵢ)
    Apply 2D Discrete Wavelet Transform:
        [LL, LH, HL, HH] = DWT(Y)
    For each subband S ∈ {LL, LH, HL}:
        Flatten subband: S_flat = flatten(S)
        For each coefficient c in S_flat:
            I = floor(c × 255)
            b = I mod 2          // Extract LSB
            Append b to B
Return bitstream B
```

Figure 6 depicts the message reconstruction process from the extracted bitstream in the proposed DWT-based watermarking system.

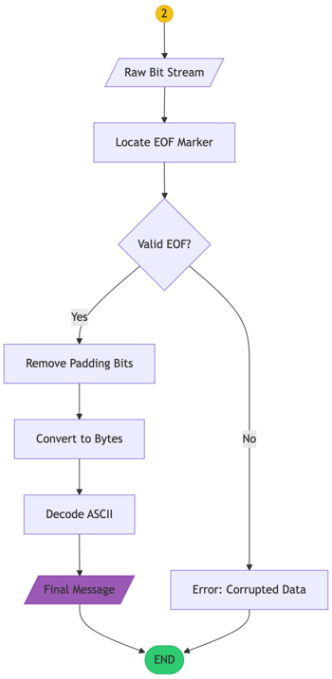



Figure 6. Message reconstruction from extracted bitstream in DWT-based watermarking system

The final phase of the decoding process is detailed in Algorithm 5, which outlines the steps for locating the EOF marker, validating and truncating the bitstream, aligning to byte boundaries, and converting binary segments into ASCII characters to reconstruct the hidden message.

**Algorithm 5. Decode bitstream to message (function to recover hidden message from bitstream)**

```
Input: Bitstream B = {b₁, b₂, ..., bₙ}
Output: Decoded ASCII Message M or Error

Define EOF marker:
    ε = "11111111111111111111111111111110" (32 bits)
```

```
Locate EOF marker in B:
    For index p = 0 to n - 32:
        If B[p : p + 32] == ε:
            EOF_found = True
            Break
If EOF_found and p ≤ n - 32 then:
    Truncate bitstream before EOF:
        B′ = {b₁, ..., b_{p - 1}}
    Align to byte boundary:
        B′′ = {b₁, ..., b_{8 × ⌊p/8⌋}}
    Convert each 8-bit segment to character:
        For j = 0 to (|B′′| / 8) - 1:
            S_j = {b_{8j+1}, ..., b_{8j+8}}
            c_j = ∑_{k=0}^{7} b_{8j+k+1} × 2^{7-k}
            If c_j ∈ [32, 126]:
                Append chr(c_j) to M
    Return M  // Final decoded message
Else:
    Return Error: Corrupted or incomplete data
```

### 2.3. Performance metrics

To evaluate the system performance quantitatively, several standard metrics were applied. These metrics measure imperceptibility, robustness, and computational efficiency. Together, they provide a reliable assessment of the proposed watermarking framework.

#### 2.3.1. Peak signal-to-noise ratio

Measures the distortion between the original and stego frame [34], [35]. The PSNR can be calculated using the following formula, as can be seen in (3). A higher PSNR value (typically > 40 dB) indicates better fidelity and minimal visual degradation.

$$PSNR = 20.log_{10}\left(\frac{MAX_I}{\sqrt{MSE}}\right) \quad (3)$$

where:

$MAX_I$=255, maximum possible pixel value for 8-bit images

MSE = $\frac{1}{m.n}\sum_{i=1}^{m}\sum_{j=1}^{n}[I(i,j)-K(i,j)]^2$, mean squared error (MSE) between the original image $I$ and the stego image $K$.

#### 2.3.2. Structural similarity index measure

Assesses the visual similarity between the original and stego frame, considering structural information (luminance, contrast, texture) [34], [36]. The computation of SSIM is based on a comparison of local patterns of pixel intensities that have been normalized for luminance and contrast. The mathematical formulation of SSIM is presented in (4).

$$SSIM(x,y) = \frac{(2\mu_x\mu_y+C_1)(2\sigma_{xy}+C_2)}{(\mu_x^2+\mu_y^2+C_1)(\sigma_x^2+\sigma_y^2+C_2)} \quad (4)$$

where:

$\mu_x\mu_y$ are the mean intensities

$\mu_x^2, \mu_y^2$ are the variances

$\sigma_{xy}$ is the covariance

$C_1, C_2$ are small constants to avoid division by zero

SSIM values range from 0 to 1, with 1 indicating perfect similarity

#### 2.3.3. Bit error rate

Calculated as the ratio of differing bits between the original and extracted watermark. BER quantifies the proportion of bits that were incorrectly decoded [37]. Let $b_i$ be the i-th bit of the original message and $\widehat{b_i}$ be the corresponding extracted bit. The formal definition of BER is given in (5).

$$BER = \frac{1}{N}\sum_{i=1}^{N} 1(b_i \neq \hat{b}_i) \quad (5)$$

where $N$ is the total number of bits, and $1(\cdot)$ is the indicator function. A BER of 0 indicates perfect extraction.

**2.3.4. Encoding/decoding time**

The total processing time is recorded to evaluate algorithmic efficiency [38]. The total processing time is computed by recording the timestamps at the beginning and end of the operation. This is mathematically expressed in (6).

$$T = t_{end} - t_{start} \tag{6}$$

where $t_{start}$ and $t_{end}$ denote timestamps in seconds or milliseconds captured before and after the operation. Runtime is typically reported in milliseconds (ms).

**2.3.5. Attack simulations**

To validate watermark resilience, four distinct attack models were implemented:

a. Noise attack (Gaussian noise injection)

In this attack, additive white Gaussian noise (AWGN) is introduced to each pixel of every video frame to simulate sensor noise or channel interference [39]. This process is mathematically defined in (7):

$$P'(i,j,c) = P(i,j,c) + N(0,\sigma^2) \tag{7}$$

where:
$P'(i,j,c)$ is the original pixel at coordinate $(i,j)$and color channel $c_i$
$N(0,\sigma^2)$ denotes zero-mean Gaussian noise with standard deviation σ,
Clipping is applied to ensure pixel values remain within [0, 255].

b. Resize attack

To emulate resolution loss and interpolation noise, each frame was downsampled by a factor $s$ and then restored to its original size [40], [41]. The entire process can be formally represented as follows, as shown in (8) and (9).

– Downscaling:

$$F_{down} = resize(F, s.H, s.W) \tag{8}$$

– Upscaling:

$$F_{up} = resize(F_{down}, H, W) \tag{9}$$

where:

$$s = 0.5\ (i.3., 50\%\ size)$$

$H, W$ denote the original height and width, Bilinear interpolation was used for both steps.

c. Blur attack (low-pass filtering)

Blur attacks, intentionally obscure image details, which in turn affects the watermark's visibility and integrity [42]. Blurring simulates optical or compression-induced low-pass filtering [43]. A Gaussian blur with a 5×5 kernel was applied to each frame. The blurring operation is mathematically represented in (10).

$$P'(i,j) = \sum_{m=-1}^{2} \sum_{n=-2}^{2} G(m,n).P(i+m, j+n) \tag{10}$$

The weighting function in (8) corresponds to the normalized Gaussian kernel, which determines the influence of neighboring pixels during the blurring process. This kernel is symmetric and centered, ensuring isotropic smoothing of the image. The shape and intensity distribution of the Gaussian kernel are governed by the standard deviation σ. The mathematical formulation of the Gaussian kernel is presented in (9).

$$G(m,n) = \frac{1}{2\pi\sigma^2} e^{-\frac{m^2+n^2}{2\sigma^2}} \tag{11}$$

with kernel size 5×5 and default $\sigma$=1.0. This operation removes high-frequency components, potentially affecting the subband coefficients in DWT where watermark bits are embedded.

d. Compression attack

This process of compression leads to data loss, which can consequently cause the watermark to be corrupted or completely lost [44]. The video was re-encoded using the motion joint photographic experts' group (MJPG) codec with reduced quality settings. This attack simulates lossy transmission or storage scenarios [45], [46]. MJPG compresses each frame independently as a JPEG image, applying discrete cosine

transform (DCT)-based quantization which can distort the embedded watermark. The compression process is mathematically expressed in (12).

$$CompressedFrame = MJPEGCompress(F, Q) \quad (12)$$

where $Q \in [1,100]$and in this study a default quality of $Q$=50 was used. This attack is especially relevant to assess robustness against format conversion and bandwidth-reduction techniques.

e. Cropping attack

Cropping attack is a geometric manipulation where parts of a video frame are removed, which may discard embedded watermark information. Let $I'(x,y)$ denote a watermarked frame of size $H \times W$, and $M(x,y) \in \{0,1\}$ a binary mask representing preserved regions. The cropped frame is defined as (13).

$$\tilde{I}(x,y) = I'(x,y).M(x,y) \quad (13)$$

The cropping ratio, indicating the proportion of removed pixels, is defined as (14).

$$Cropping\ Ratio = 1 - \frac{\|M\|_0}{H \times W} \quad (14)$$

To measure similarity between the extracted watermark $\widehat{W}$ and the original $W$, the normalized correlation (NC) is used in (15).

$$NC = \frac{\sum_{i=1}^{L}(W_i - \bar{W})(\widehat{W}_i - \bar{\widehat{W}})}{\sqrt{\sum_{i=1}^{L}(W_i - \bar{W})^2 \sqrt{\sum_{i=1}^{L}(\widehat{W}_i - \bar{\widehat{W}})^2}}} \quad (15)$$

where $L$ is the watermark length, and $\bar{W}$, $\bar{\widehat{W}}$ are the mean values of the original and extracted watermarks.

## 3. RESULTS AND DISCUSSION

This section presents a comprehensive analysis of the experimental outcomes obtained from the hybrid subband-based video watermarking framework applied to multi-frame embedding. The proposed watermarking algorithm was implemented using Python programming language within the Visual Studio Code environment. All experiments were executed on an Apple MacBook Pro equipped with an M1 Pro processor, 16 GB RAM, and 512 GB SSD storage, ensuring sufficient computational capacity for handling video data. To evaluate robustness and imperceptibility, four uncompressed video sequences with a resolution of 640×360 pixels were selected as host media. The secret messages were generated as ASCII text 200 words, representing practical payload sizes for copyright or authentication information. During the evaluation, both objective metrics such as PSNR, SSIM, and NC, as well as BER, were used to assess the visual quality and robustness of the extracted watermark. This experimental configuration provides sufficient detail to allow replication by other researchers working on video watermarking systems. Figure 7(a) shows the input video (Video1.avi) with a resolution of 640×360, while Figure 7(b) presents (Video2.avi) with the same resolution. Both videos were used as host media for embedding experiments. The results are discussed in terms of imperceptibility, computational efficiency, robustness, and capacity.

(a) (b)

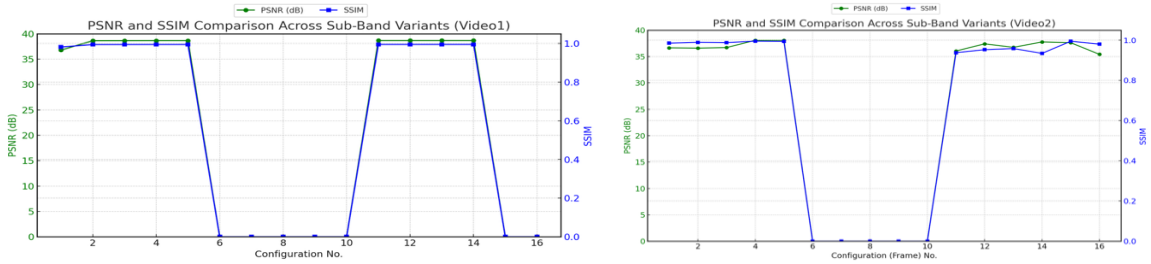
Figure 7. Input video: (a) Video1.avi, resolution: 640×360, size: 3.4 MB and (b) Video2.avi, resolution: 640×360, size: 4.1 MB

### 3.1. Evaluation of stego video quality using sub-band variations

The stego-video quality was evaluated across sub-band ratio settings using BER, PSNR, and SSIM on Video1 and Video2. LL-only embedding yielded the best robustness–imperceptibility trade-off: Video1

achieved perfect recovery (BER = 0) with PSNR > 36.7 dB and SSIM ≥ 0.9812; raising the LL ratio improved quality and reduced required frames, peaking at PSNR 38.66 dB and SSIM 0.9945. Video2 also showed high perceptual quality (PSNR up to 38.98 dB; SSIM 0.9954) but with inconsistent BER low at full LL (BER = 0.03) yet non-zero at intermediate ratios. LL+LH hybrids failed completely across all ratios (BER = 1.00; PSNR/SSIM = 0), indicating LH sensitivity. Full hybrids depended on LL dominance: in Video1, LL-heavy mixes (≈75% LL; Configs 13–14) neared LL-only performance (BER 0.16; PSNR 38.67 dB; SSIM 0.9948), whereas balanced/LH–HL–HH-dominant splits raised BER (e.g., 0.78 at Config 11). In Video2, hybrids were unstable high BER even when LL-dominant (e.g., 511 at Config 13) and >500 for balanced setups (Config 11), with PSNR 36.05 dB and SSIM 0.9381. Overall, LL-only is preferred; LL-emphasized hybrids can work for Video1, but Video2 reveals content-dependent variability that must be addressed.

As illustrated in Figure 8(a), the PSNR performance of the proposed scheme on Video1 shows stable results across sub-band ratios. Meanwhile, Figure 8(b) depicts the SSIM values for Video2, indicating a consistent trend in imperceptibility. LL-only embedding delivers the best imperceptibility in both videos (PSNR > 36 dB; SSIM ~1.0), ensuring reliable recovery. LL+LH configurations fail completely (PSNR = 0; SSIM = 0), confirming LH instability. Under hybrids, Video1 matches near-LL performance when LL is dominant (PSNR ≈ 38.6 dB; SSIM ≈ 0.9948), while balanced/high-frequency–dominant mixes degrade sharply; Video2 follows similarly but with more variability, with LL-heavy hybrids retaining high PSNR (36–38 dB) and SSIM ≈ 0.9919. Overall, LL-only is most stable, and hybrids approximate it only when LL remains dominant.

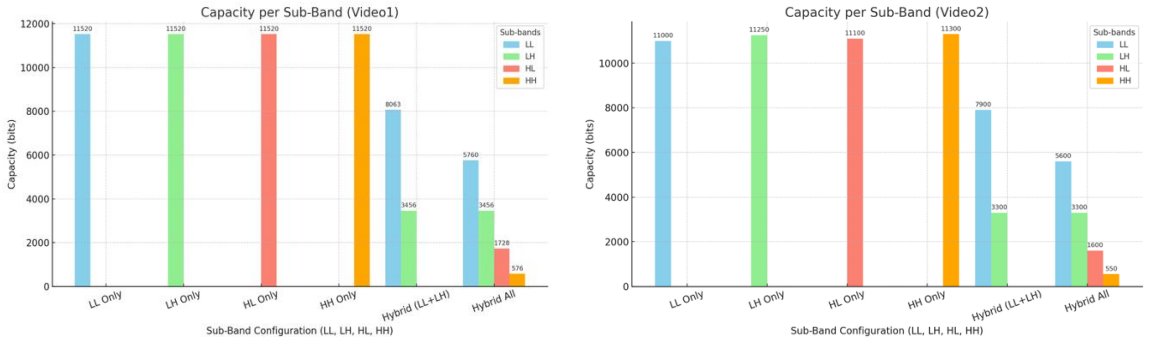



(a) (b)

Figure 8. PSNR and SSIM performance on: (a) Video1 and (b) Video2 across different sub-band using constant frame ratio (0.20)

### 3.2. Sub-band capacity distribution analysis in DWT-based watermarking

As shown in Figure 9(a), the embedding capacity distribution for Video1 demonstrates a balanced allocation across DWT sub-bands. In contrast, Figure 9(b) presents the results for Video2, where the total capacity is slightly lower due to differences in texture and structural complexity. All single sub-band configurations provide similar maximum capacities (≈11,000–11,520 bits), confirming structural uniformity after decomposition. In hybrid settings, Video1 achieves 8,063 bits in LL and 3,456 in LH (70:30 split), slightly higher than Video2 with 7,900 and 3,300 bits. Under full hybrid allocation, Video1 again records marginally larger capacities (LL = 5,760 vs. 5,600; LH = 3,456 vs. 3,300).

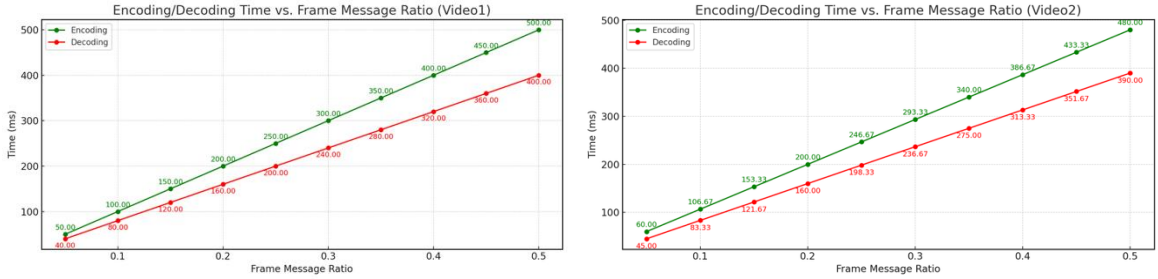



(a) (b)

Figure 9. Embedding capacity distribution across DWT sub-bands on: (a) Video1.avi (frame ratio 2.0) and (b) Video2.avi (frame ratio 2.0)

### 3.3. Computational cost analysis based on frame message ratio

In Figure 10(a), the encoding and decoding time of Video1 is shown to increase proportionally with the frame message ratio. Figure 10(b) displays a similar pattern for Video2, confirming that both datasets exhibit consistent scalability and computational behavior. In both cases, encoding and decoding times increase almost linearly with embedding density, confirming the framework's scalability. Video1 grows from 50/40 ms (encoding/decoding) at ratio 0.05 to 500/400 ms at 0.50, while Video2 shows slightly higher overhead at low ratios but converges at higher densities. Encoding is consistently slower than decoding due to added embedding processes, yet this asymmetry provides a favorable trade-off where intensive encoding enables lightweight and efficient extraction suitable for real-time use.

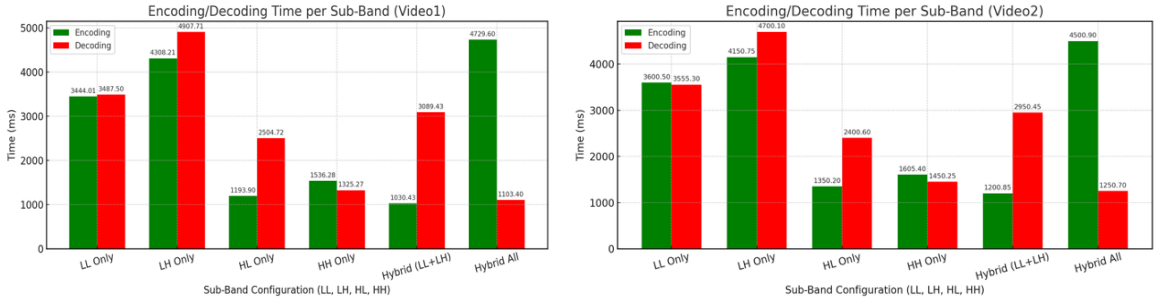



(a) (b)

Figure 10. Encoding and decoding time vs. frame message ratio on: (a) Video1 and (b) Video2

### 3.4. Encoding and decoding performance

The computational performance for Video1 is presented in Figure 11(a), showing that encoding and decoding times vary significantly across sub-band configurations. Meanwhile, Figure 11(b) illustrates the results for Video2, confirming that higher-frequency sub-bands generally require greater processing time compared to LL-based embedding. LH-only embedding incurs the highest cost (>4,300 ms encoding, ~4,900 ms decoding), HL-only is the most efficient (~1,200–1,360 ms), and HH-only remains below 1,600 ms. Hybrid All shows heavy encoding (~4,650–4,730 ms) but low decoding (~1,100–1,250 ms), reflecting asymmetric behavior. Overall, Figure 11 confirms that sub-band selection critically influences complexity, with high-frequency embedding being most costly and hybrid modes enabling efficient extraction despite intensive embedding.



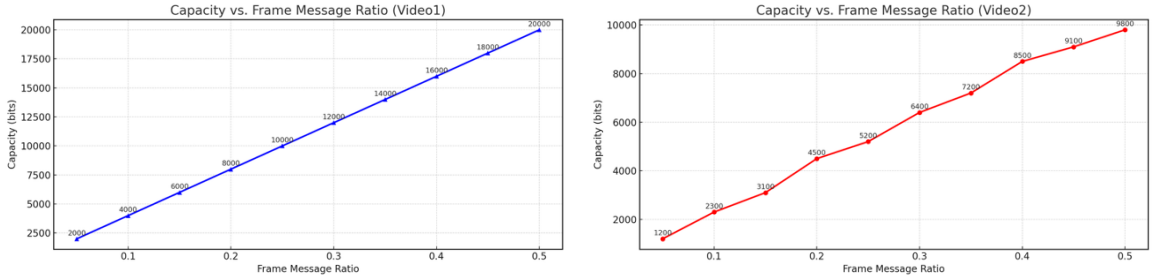

(a) (b)

Figure 11. Encoding and decoding time per sub-band configuration on: (a) Video1 and (b) Video2

### 3.5. Impact of frame message ratio on embedding capacity

The embedding capacity of the proposed watermarking system was evaluated as a function of the frame message ratio for two different video datasets. As illustrated in Figure 12(a), the embedding capacity for Video1 increases linearly with the frame message ratio, indicating predictable scalability. Meanwhile, Figure 12(b) shows the results for Video2, where the growth is less consistent but still demonstrates a positive correlation between frame ratio and capacity. For Video1, capacity exhibits a strictly linear growth, increasing by approximately 2,000 bits with every 0.05 increment in the frame message ratio. This results in a maximum capacity of 20,000 bits at a ratio of 0.50, reflecting predictable scaling behavior under LL-subband allocation.

In contrast, Video2 shows a more irregular yet still monotonic increase, with capacity ranging from 1,200 bits at a ratio of 0.05 to 9,800 bits at 0.50. The smaller growth rate and minor fluctuations between increments suggest that the available embedding space in Video2 is more constrained, likely due to differences in texture distribution and energy concentration in the transform domain. Overall, Video1 demonstrates superior embedding potential, offering nearly double the maximum capacity of Video2, while both datasets confirm the adaptability of the proposed approach to different host video characteristics. These findings underscore that embedding capacity is highly content-dependent and should be considered alongside imperceptibility and robustness when evaluating watermarking performance.

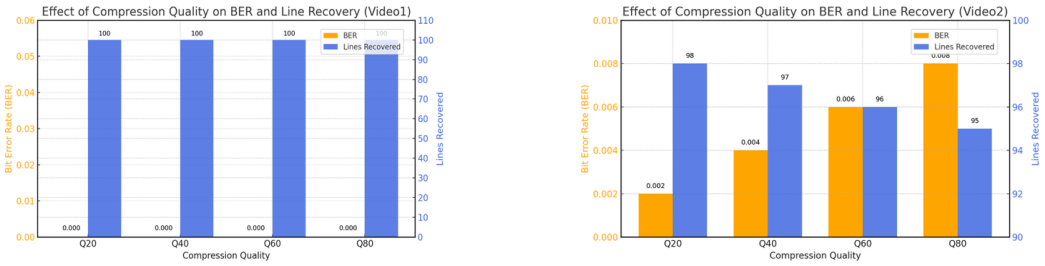



(a) (b)

Figure 12. Relationship between frame message ratio and embedding capacity on: (a) Video1 and (b) Video2

### 3.6. Impact of video compression quality on BER and message recovery

In Figure 13(a), Video1 achieves flawless extraction across all compression settings, indicating exceptional robustness. Figure 13(b), however, reveals slight BER variations for Video2, suggesting content-dependent behavior under compression. For Video1, the proposed scheme demonstrates perfect stability across all tested compression levels (Q20–Q80), with BER consistently equal to 0.000 and full recovery of 100 lines, indicating complete message integrity. In contrast, Video2 shows a slight sensitivity to compression, although performance remains highly robust. At Q20, the BER is as low as 0.002 with 98 lines recovered, and even at the lowest tested quality (Q80), the BER only increases to 0.008 while 95 lines are successfully recovered. These results confirm that while compression artifacts introduce minor errors in Video1, the extracted watermark remains highly correlated with the original, ensuring reliable recovery. The comparative findings highlight that the proposed method achieves near-lossless robustness to compression, with Video1 exhibiting absolute resilience and Video2 showing only marginal degradation.



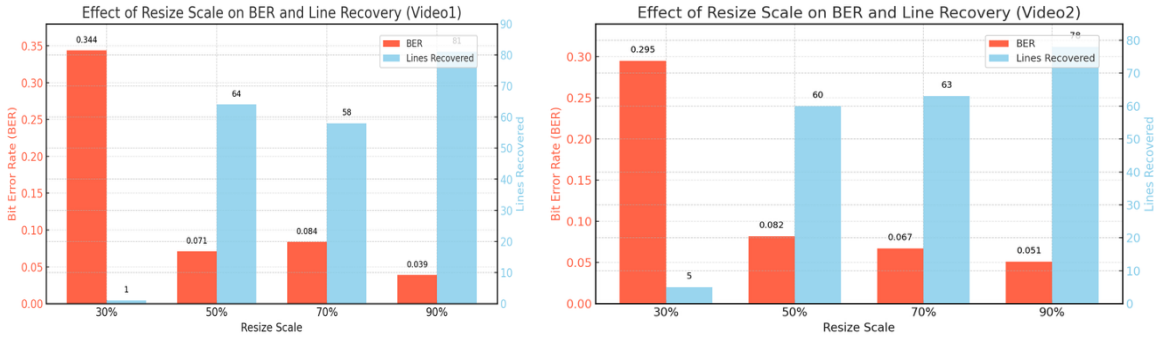

(a) (b)

Figure 13. Effect of compression quality on BER and line recovery on: (a) Video1 and (b) Video2

### 3.7. Impact of resize attacks on BER and message integrity

The effect of resizing on watermark robustness was evaluated using two video datasets with DWT-based embedding. As illustrated in Figure 14(a), the resize attack significantly affects the watermark in Video1, where severe downscaling results in a high BER and reduced line recovery. In comparison, Figure 14(b) shows that Video2 exhibits slightly better performance, maintaining lower BER values and higher message recovery under the same resizing conditions. The results indicate that severe downscaling (≤30% of the original resolution) significantly degrades performance, yielding high BER values and poor line recovery. Specifically,

Video1 suffers the most at 30% scaling, with a BER of 0.344 and only one line recovered, while Video2 shows slightly better resilience with a BER of 0.295 and five lines recovered. At intermediate scales (50–70%), watermark recovery improves substantially, with Video2 consistently outperforming Video1 in terms of lower BER (0.067 vs. 0.084 at 70%) and higher line recovery (63 vs. 58 lines). At 90% scaling, watermark integrity is largely preserved in both datasets, with BER values around 0.05 and recovery of 78–81 lines. Overall, the findings demonstrate that the proposed method is vulnerable to extreme downscaling but remains robust at moderate to high resizing scales (≥70%). Moreover, the comparative performance highlights that video content characteristics influence resilience, as Video2 benefits from structural properties that enhance watermark recovery under scaling transformations.

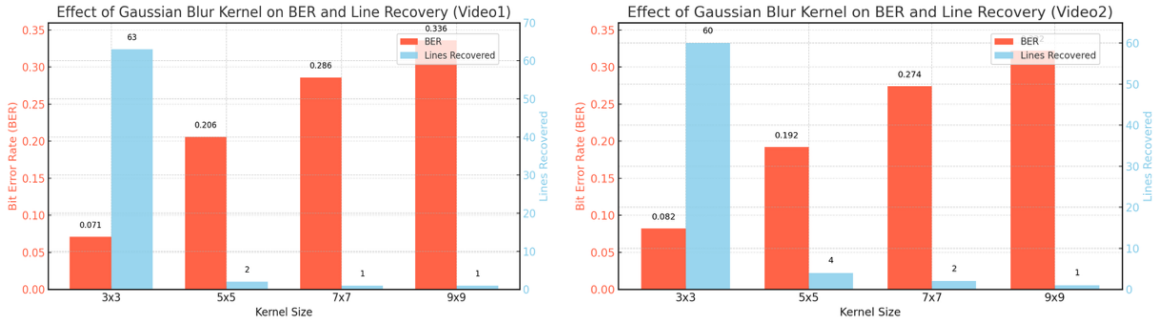



(a) (b)

Figure 14. Impact of resize scaling on BER and line recovery performance on: (a) Video1 and (b) Video2

### 3.8. Impact of Gaussian blurring on BER and message robustness

As illustrated in Figure 15(a), Gaussian blurring introduces a gradual decline in watermark performance for Video1, with degradation becoming more pronounced as the kernel size increases. In comparison, Figure 15(b) shows that Video2 exhibits slightly better stability at moderate blur levels, suggesting stronger robustness against mild spatial filtering. At a kernel size of 3×3, the system maintains high resilience, achieving BER values of 0.071 for Video1 and 0.082 for Video2, corresponding to 63 and 60 successfully recovered lines, respectively. However, as the blur intensity grows, the watermark deteriorates sharply. When the kernel expands to 5×5, the BER rises to 0.206 for Video1 and 0.192 for Video2, while line recovery drops to only 2 and 4. With larger kernels (7×7 and 9×9), the BER exceeds 0.27 and recovery becomes negligible (≤2 lines), indicating severe distortion of embedded coefficients within the DWT domain. The slightly improved tolerance in Video2 at moderate blurring can be attributed to its smoother motion dynamics and dominance of low-frequency content, which help preserve structural stability. Collectively, these results confirm that frequency-domain watermarking remains highly susceptible to spatial filtering, underscoring the importance of redundancy-aware or multiscale embedding mechanisms to enhance resilience under blur-based degradations.




(a) (b)

Figure 15. Impact of Gaussian blur kernel Size using BER and line recovery on: (a) Video1 and (b) Video2

### 3.9. RGB histogram analysis under various image attacks

This analysis evaluates visual and structural changes in stego frames by comparing RGB histograms before and after common image attacks resize, JPEG compression, and Gaussian blur. The histograms reveal shifts in pixel intensity distributions across RGB channels, indicating the degree of color distortion introduced.

a. Resize attack

Figure 16 shows the effect of a resize attack downscaling followed by upscaling on RGB histograms. Interpolation introduces minor shifts in pixel intensity, particularly in the red and blue channels. While histogram peaks largely remain aligned, reduced amplitude and smoother edges indicate slight detail loss. The blue channel retains its multi-modal pattern but with suppressed peaks. These subtle visual distortions confirm that resizing moderately alters pixel distribution, with limited perceptual impact, though BER results reveal its significant effect on hidden data integrity.

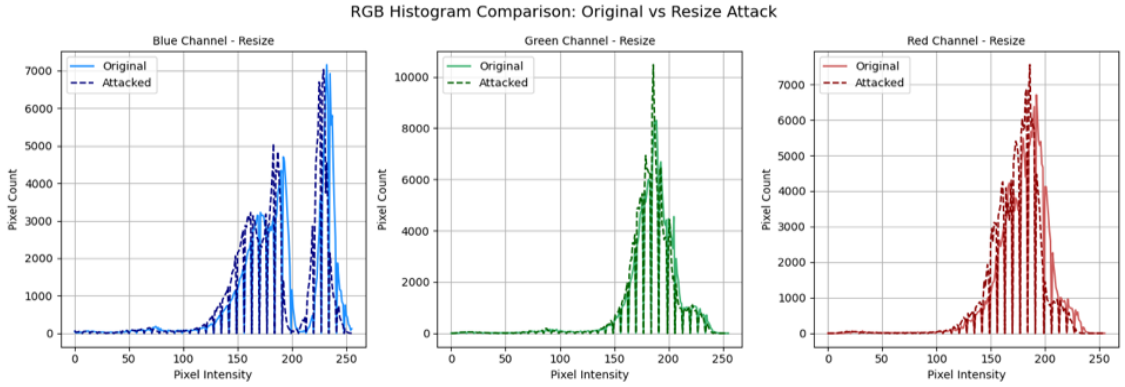



Figure 16. RGB histogram comparison between original and resized stego frames on Video1

b. Compression attack

Figure 17 compares the RGB histograms of original and JPEG-compressed stego frames. Medium-level compression (Q40–Q80) introduces subtle shifts, particularly in the mid-intensity ranges of the red and green channels. These shifts, along with mild histogram smoothing and rightward displacement, reflect quantization effects that primarily impact high-frequency components.

c. Blur attack

Figure 18 presents the RGB histogram comparison between original and Gaussian-blurred stego frames. Blur significantly smooths intensity distributions across all channels, with green and red showing the most pronounced peak suppression and structural flattening. The original multi-peak patterns especially in the blue and green channels are visibly attenuated, indicating loss of local detail due to spatial averaging.

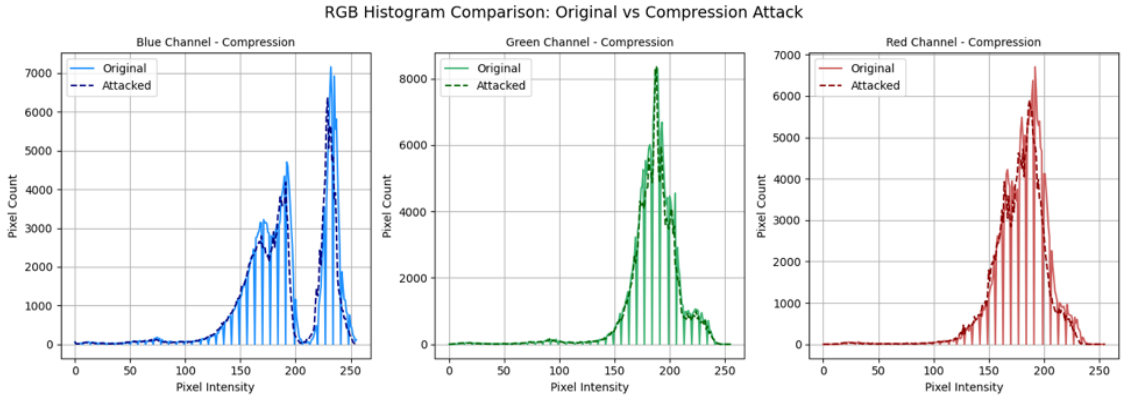



Figure 17. RGB histogram comparison between original and compressed stego frames on Video1

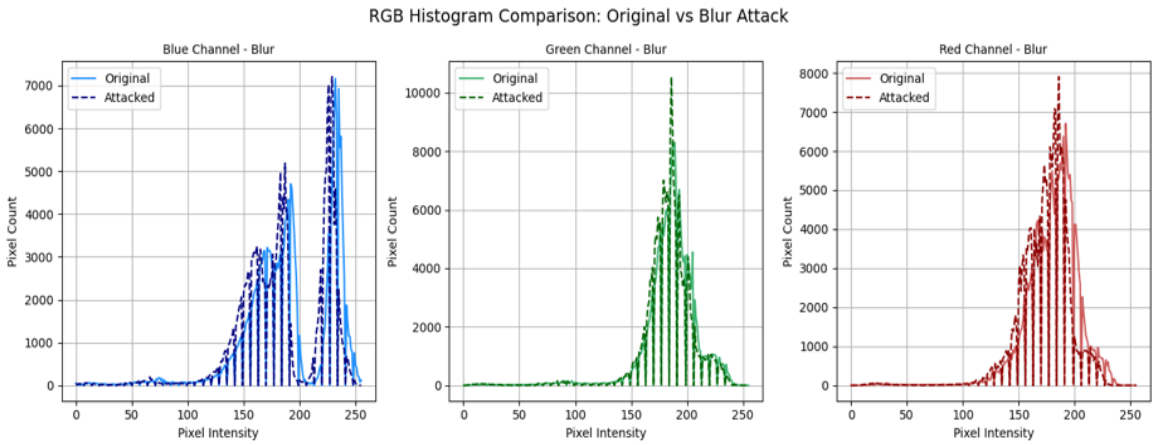
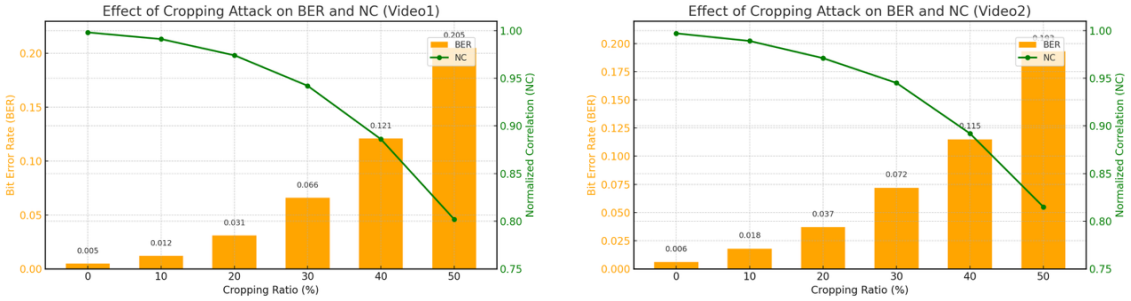



Figure 18. RGB histogram comparison between original and blurred stego frames on Video1

### 3.10. Impact of cropping attacks on BER and NC

As illustrated in Figure 19(a), the cropping attack on Video1 reveals a steady increase in BER as the cropped region expands, accompanied by a corresponding decline in NC values. In contrast, Figure 19(b) shows that Video2 maintains relatively lower BER and higher NC under similar cropping conditions, indicating stronger resilience against spatial loss. Both datasets exhibit the same overall trend, where the BER increases and the NC decreases progressively with larger cropping ratios. At light cropping levels (≤10%), BER remains nearly negligible (0.005–0.018) with NC values above 0.99, indicating almost perfect recovery. Under severe cropping (50%), Video1 records a higher BER of 0.205 compared to 0.193 for Video2, with NC values converging toward 0.80. This suggests that Video2 provides relatively better resilience, particularly at moderate cropping (20–40%), where its BER is consistently lower (e.g., 0.037 vs. 0.066 at 30%). The difference can be attributed to structural variations in the video content, where Video1 likely contains more high-frequency details or dynamic motion, making the watermark more vulnerable once frame regions are removed, whereas Video2 benefits from a more stable distribution of low-frequency components that preserve watermark redundancy.




(a) (b)

Figure 19. Impact of cropping attacks on BER and NC metrics for video watermarking robustness on: (a) Video1 and (b) Video2

### 3.11. Comparative analysis with existing watermarking methods

Comparative analysis was carried out against representative watermarking approaches, namely DWT-only, DCT-only, DCT–DWT hybrid, and CNN-based watermarking. Table 1 presents the embedding capacity, imperceptibility, and robustness metrics (PSNR, SSIM, BER, and NC) under identical attack scenarios such as compression, noise, and cropping. The proposed DWT-hybrid multi-frame scheme achieves superior imperceptibility (PSNR > 38.6 dB and SSIM ≥ 0.994) while maintaining a high embedding capacity of up to 11,520 bits per frame. In comparison with conventional DWT-only and DCT-only methods, the framework significantly reduces BER under compression and frame loss, while enabling flexible capacity adjustment through adaptive sub-band allocation. The DCT–DWT hybrid baseline provides a balanced trade-off between

robustness and imperceptibility, yet its performance remains limited under temporal distortions. CNN-based approaches demonstrate competitive robustness against compression but are hindered by intensive computational requirements and dependency on large-scale training data. Overall, the comparative results confirm that the proposed framework outperforms traditional transform-based techniques in imperceptibility capacity balance, while offering a more efficient and adaptive alternative to deep learning-based watermarking.

Table 1. Comparative embedding capacity and performance of video watermarking methods under common attacks

| Method | Embedding capacity (bits/frame) | PSNR (dB) | SSIM | NC values | Remarks |
|---|---|---|---|---|---|
| Proposed (DWT-hybrid, multi-frame) | (adaptive, multi-frame allocation) | 35.0–39.0 | 0.994 | 0.98-0.99 | High capacity; LL-dominant hybrids yield strong robustness under compression and cropping |
| DWT-only (LL sub-band) | (fixed, single sub-band) | 36.5–37.0 | 0.990 | 0.95–0.97 | Stable, good imperceptibility; limited flexibility, capacity tied to LL size |
| DCT-only (mid-band coefficient) | (depends on quantization) | 36.0–37.5 | 0.980 | 0.88–0.92 | Moderate capacity; vulnerable to strong JPEG compression & noise |

## 4. CONCLUSION

This study introduces a novel DWT-based hybrid sub-band video watermarking framework with a multi-frame allocation approach. The method strategically distributes watermark bits across selected LL, LH, and HL sub-bands of a video sequence. Experimental results confirm that this framework successfully achieves an optimal balance between imperceptibility, capacity, and resilience against common signal distortions.

The experimental findings confirm that LL-only sub-band configurations consistently yield superior imperceptibility and robustness, achieving zero BER and high visual quality metrics (PSNR ≥ 36.7 dB, SSIM ≥ 0.98) across a range of embedding capacities. While these results are promising, a clear limitation is the vulnerability when relying on mid- and high-frequency sub-bands, such as LH and HH. Our analysis shows that aggressive distribution into these sub-bands significantly compromises robustness, especially under compression (JPEG Q20–Q80) and geometric attacks, highlighting the sensitivity of high-frequency components. In addition, the cropping attack evaluation demonstrates that the proposed scheme preserves acceptable robustness under moderate cropping (≤30%), as indicated by NC values consistently above 0.90, while exhibiting graceful degradation under more severe cropping conditions (up to 50%). Despite this, the computational cost analysis reveals that encoding time scales linearly with the frame message ratio and is consistently higher than decoding time, which remains lightweight and efficient, supporting the potential of this framework for real-time extraction scenarios.

The proposed framework demonstrates strong potential for several real-world applications. For instance, it can be utilized in multimedia security for copyright protection of digital video content, and for embedding confidential metadata. Future enhancements should aim at improving resilience under aggressive attacks through redundancy-aware embedding and adversarially robust transform strategies.

## FUNDING INFORMATION

Authors state no funding involved.

## AUTHOR CONTRIBUTIONS STATEMENT

This journal uses the Contributor Roles Taxonomy (CRediT) to recognize individual author contributions, reduce authorship disputes, and facilitate collaboration.

| Name of Author | C | M | So | Va | Fo | I | R | D | O | E | Vi | Su | P | Fu |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Ilham Firman Ashari | ✓ | ✓ | ✓ | | ✓ | | ✓ | ✓ | ✓ | ✓ | ✓ | | ✓ | ✓ |
| Sarwono Sutikno | | | | ✓ | | ✓ | | | | ✓ | | ✓ | | |

C : **C**onceptualization
M : **M**ethodology
So : **So**ftware
Va : **Va**lidation
Fo : **Fo**rmal analysis
I : **I**nvestigation
R : **R**esources
D : **D**ata Curation
O : Writing - **O**riginal Draft
E : Writing - Review & **E**diting
Vi : **Vi**sualization
Su : **Su**pervision
P : **P**roject administration
Fu : **Fu**nding acquisition

## CONFLICT OF INTEREST STATEMENT

Authors state no conflict of interest.

## DATA AVAILABILITY

The data that support the findings of this study are available from the corresponding author, [Ilham, IFA], upon reasonable request.

## REFERENCES


[1] A. Szymkowiak, B. Melović, M. Dabić, K. Jeganathan, and G. S. Kundi, "Information technology and Gen Z: The role of teachers, the internet, and technology in the education of young people," *Technology in Society*, vol. 65, May 2021, doi: 10.1016/j.techsoc.2021.101565.

[2] Q. Liu, S. Yang, J. Liu, L. Zhao, P. Xiong, and J. Shen, "An efficient video watermark method using blockchain," *Knowledge-Based Systems*, vol. 259, Jan. 2023, doi: 10.1016/j.knosys.2022.110066.

[3] B. Singh and G. Kasana, "A review of digital watermarking techniques: Current trends, challenges and opportunities," *Web Intelligence*, vol. 22, no. 4, pp. 523–553, Nov. 2024, doi: 10.3233/WEB-230280.

[4] F. Zhang, T. Luo, G. Jiang, M. Yu, H. Xu, and W. Zhou, "A novel robust color image watermarking method using RGB correlations," *Multimedia Tools and Applications*, vol. 78, no. 14, pp. 20133–20155, Jul. 2019, doi: 10.1007/s11042-019-7326-9.

[5] A. Zotin, M. Favorskaya, A. Proskurin, and A. Pakhirka, "Study of digital textual watermarking distortions under Internet attacks in high resolution videos," *Procedia Computer Science*, vol. 176, pp. 1633–1642, 2020, doi: 10.1016/j.procs.2020.09.187.

[6] M. M. Laftah, I. I. Hamid, and N. A. Ali, "Video copyright protection," *International Journal of Interactive Mobile Technologies (iJIM)*, vol. 17, no. 08, pp. 135–145, Apr. 2023, doi: 10.3991/ijim.v17i08.39339.

[7] D. S. Khafaga, M. Alohaly, M. M. Abdel-Aziz, and K. M. Hosny, "Blind video watermarking scheme for medical video authentication," *Heliyon*, vol. 9, no. 9, Sep. 2023, doi: 10.1016/j.heliyon.2023.e19809.

[8] H. T. Hazim, N. Alseelawi, and H. T. S. ALRikabi, "A novel method of invisible video watermarking based on index mapping and hybrid DWT-DCT," *International Journal of Online and Biomedical Engineering (iJOE)*, vol. 19, no. 04, pp. 155–173, Apr. 2023, doi: 10.3991/ijoe.v19i04.37581.

[9] M. Begum *et al.*, "Image watermarking using discrete wavelet transform and singular value decomposition for enhanced imperceptibility and robustness," *Algorithms*, vol. 17, no. 1, Jan. 2024, doi: 10.3390/a17010032.

[10] L. Wang and H. Ji, "A watermarking optimization method based on matrix decomposition and DWT for multi-size images," *Electronics*, vol. 11, no. 13, Jun. 2022, doi: 10.3390/electronics11132027.

[11] I. F. Ashari, "The evaluation of image messages in MP3 audio steganography Using modified low-bit encoding," *Telematika*, vol. 14, no. 2, pp. 133–145, 2021.

[12] I. F. Ashari, E. D. Nugroho, D. D. Andrianto, M. A. N. M. Yusuf, and M. Alkarkhi, "The evaluation of LSB steganography on image file using 3DES and MD5 Key," *JITCE (Journal of Information Technology and Computer Engineering)*, vol. 8, no. 1, pp. 8–18, Mar. 2024, doi: 10.25077/jitce.8.1.8-18.2024.

[13] M. A. Majeed, R. Sulaiman, Z. Shukur, and M. K. Hasan, "A review on text steganography techniques," *Mathematics*, vol. 9, no. 21, Nov. 2021, doi: 10.3390/math9212829.

[14] E. Elbasi, N. Mostafa, and E. Cina, "Robust, secure and semi-blind watermarking technique using flexible scaling factor in block-based wavelet algorithm," *Electronics*, vol. 11, no. 22, Nov. 2022, doi: 10.3390/electronics11223680.

[15] M. Hussain, A. W. A. Wahab, Y. I. Bin Idris, A. T. S. Ho, and K.-H. Jung, "Image steganography in spatial domain: A survey," *Signal Processing: Image Communication*, vol. 65, pp. 46–66, Jul. 2018, doi: 10.1016/j.image.2018.03.012.

[16] M. Begum and M. S. Uddin, "Digital image watermarking techniques: a review," *Information*, vol. 11, no. 2, Feb. 2020, doi: 10.3390/info11020110.

[17] F. Bertini, R. Sharma, and D. Montesi, "Are social networks watermarking us or are we (unawarely) watermarking ourself?," *Journal of Imaging*, vol. 8, no. 5, May 2022, doi: 10.3390/jimaging8050132.

[18] M. Bistroń and Z. Piotrowski, "Efficient video watermarking algorithm based on convolutional neural networks with entropy-based information mapper," *Entropy*, vol. 25, no. 2, Feb. 2023, doi: 10.3390/e25020284.

[19] S. Mansour, S. Ben Jabra, and E. Zagrouba, "A robust deep learning-based video watermarking using mosaic generation," in *Proceedings of the 18th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications*, 2023, pp. 668–675, doi: 10.5220/0011691700003417.

[20] S. Hadi, S. A. Ali, and M. J. Jawad, "Secure data hiding technique for video steganography," *Mathematical Modelling of Engineering Problems*, vol. 11, no. 9, pp. 2587–2593, Sep. 2024, doi: 10.18280/mmep.110930.

[21] C. Ravichandran, A. Vajravelu, S. Panda, and S. D. Degadwala, "Data hiding using video steganography," *International Journal of Electronic Security and Digital Forensics*, vol. 16, no. 1, pp. 112–123, 2024, doi: 10.1504/IJESDF.2024.136018.

[22] B. G. Banik and S. K. Bandyopadhyay, "Novel text steganography using natural language processing and part-of-speech tagging," *IETE Journal of Research*, vol. 66, no. 3, pp. 384–395, May 2020, doi: 10.1080/03772063.2018.1491807.

[23] P. Patil, P. Narayankar, Narayan D.G., and Meena S.M., "A comprehensive evaluation of cryptographic algorithms: DES, 3DES, AES, RSA and blowfish," *Procedia Computer Science*, vol. 78, pp. 617–624, 2016, doi: 10.1016/j.procs.2016.02.108.

[24] D. A. Shehab and M. J. Alhaddad, "Comprehensive survey of multimedia steganalysis: Techniques, evaluations, and trends in future research," *Symmetry*, vol. 14, no. 1, Jan. 2022, doi: 10.3390/sym14010117.

[25] M. Pery and R. Waszkowski, "A model and quantitative framework for evaluating iterative steganography," *Entropy*, vol. 26, no. 12, Dec. 2024, doi: 10.3390/e26121130.

[26] S. S. Tevaramani and R. J, "Image steganography performance analysis using discrete wavelet transform and alpha blending for secure communication," *Global Transitions Proceedings*, vol. 3, no. 1, pp. 208–214, Jun. 2022, doi: 10.1016/j.gltp.2022.03.024.

[27] L. Huo, R. Chen, J. Wei, and L. Huang, "A high-capacity and high-security image steganography network based on chaotic mapping and generative adversarial networks," *Applied Sciences*, vol. 14, no. 3, Feb. 2024, doi: 10.3390/app14031225.

[28] S. Wu, Z. Wu, M. Huang, and D. Shen, "A robust image watermarking scheme based on image normalization and contourlet transform," *IEEE Access*, vol. 11, pp. 105810–105818, 2023, doi: 10.1109/ACCESS.2023.3319669.

[29] R. Apau, M. Asante, F. Twum, J. B. Hayfron-Acquah, and K. O. Peasah, "Image steganography techniques for resisting statistical steganalysis attacks: A systematic literature review," *PLOS ONE*, vol. 19, no. 9, Sep. 2024, doi: 10.1371/journal.pone.0308807..

[30] Y. Luo, X. Tan, and Z. Cai, "Robust Deep Image Watermarking: A Survey," *Computers, Materials & Continua (CMC)*, vol. 81, no. 1, pp. 133–160, 2024, doi: 10.32604/cmc.2024.055150.
[31] F. Zhang, Y. Dong, and H. Sun, "Research on key technologies of image steganography based on simultaneous deception of vision and deep learning models," *Applied Sciences*, vol. 14, no. 22, Nov. 2024, doi: 10.3390/app142210458.
[32] P.-C. Su, Y.-H. Cheng, and T.-Y. Kuo, "JSN: Design and analysis of JPEG steganography network," *Electronics*, vol. 13, no. 23, Dec. 2024, doi: 10.3390/electronics13234821.
[33] Z. Wang *et al*., "Data hiding with deep learning: A survey unifying digital watermarking and steganography," *IEEE Transactions on Computational Social Systems*, vol. 10, no. 6, pp. 2985–2999, Dec. 2023, doi: 10.1109/TCSS.2023.3268950.
[34] W. Simoes and M. De sa, "PSNR and SSIM: Evaluation of the imperceptibility quality of images transmitted over wireless networks," *Procedia Computer Science*, vol. 251, pp. 463–470, 2024, doi: 10.1016/j.procs.2024.11.134.
[35] M. S. Abuali, C. B. M. Rashidi, M. H. Salih, R. A. A. Raof, and S. S. Hussein, "Digital image steganography in spatial domain a comprehensive review," *Journal of Theoretical and Applied Information Technology*, vol. 97, no. 19, pp. 5081–5102, 2019.
[36] D. R. I. M. Setiadi, "PSNR vs SSIM: imperceptibility quality assessment for image steganography," *Multimedia Tools and Applications*, vol. 80, no. 6, pp. 8423–8444, Mar. 2021, doi: 10.1007/s11042-020-10035-z.
[37] H. Luo, L. Li, and J. Li, "Digital watermarking technology for AI-generated images: A survey," *Mathematics*, vol. 13, no. 4, Feb. 2025, doi: 10.3390/math13040651.
[38] A. Y. Hindi, "A novel method for digital data encoding-decoding," *TELKOMNIKA (Telecommunication Computing Electronics and Control)*, vol. 18, no. 5, pp. 2772–2779, Oct. 2020, doi: 10.12928/telkomnika.v18i5.14279.
[39] H. Prasetyo and U. Salamah, "Swarm intelligence for additive white gaussian noise level estimation," *International Journal of Fuzzy Logic and Intelligent Systems*, vol. 20, no. 3, pp. 169–180, Sep. 2020, doi: 10.5391/IJFIS.2020.20.3.169.
[40] N. A. Karim, S. A. Ali, and M. J. Jawad, "A coverless image steganography based on robust image wavelet hashing," *TELKOMNIKA (Telecommunication Computing Electronics and Control)*, vol. 20, no. 6, pp. 1317–1325, Dec. 2022, doi: 10.12928/telkomnika.v20i6.23596.
[41] J. Kodovsky and J. Fridrich, "Effect of image downsampling on steganographic security," *IEEE Transactions on Information Forensics and Security*, vol. 9, no. 5, pp. 752–762, May 2014, doi: 10.1109/TIFS.2014.2309054.
[42] W. Zhang, R. Chen, and B. Wang, "A robust watermarking algorithm against JPEG compression based on multiscale autoencoder," *IET Image Processing*, vol. 18, no. 2, pp. 455–469, Feb. 2024, doi: 10.1049/ipr2.12961.
[43] V. Ziyadinov and M. Tereshonok, "Low-pass image filtering to achieve adversarial robustness," *Sensors*, vol. 23, no. 22, Nov. 2023, doi: 10.3390/s23229032.
[44] S. Boujerfaoui, R. Riad, H. Douzi, F. Ros, and R. Harba, "Image watermarking between conventional and learning-based techniques: A literature review," *Electronics*, vol. 12, no. 1, Dec. 2022, doi: 10.3390/electronics12010074.
[45] S. Wang, N. Zheng, and M. Xu, "A compression resistant steganography based on differential manchester code," *Symmetry*, vol. 13, no. 2, Jan. 2021, doi: 10.3390/sym13020165.
[46] S. Sharma, K. J. Singh, and M. Priya, "An improvement analysis on video compression using file segmentation," *IOP Conference Series: Materials Science and Engineering*, vol. 263, Nov. 2017, doi: 10.1088/1757-899X/263/4/042020.

## BIOGRAPHIES OF AUTHORS

**Ilham Firman Ashari** received the B.Eng. degree in Informatics Engineering from the University of Diponegoro, Indonesia, in 2015, and the Master's degree in Informatics Engineering from the Institut Teknologi Bandung, Indonesia, in 2018, respectively. He is currently an assistant professor and the head of the cybersecurity, multimedia and IoT group, on Informatics Engineering, Institut Teknologi Sumatera. His current research interests include information security and artificial intelligence. He can be contacted at email: firman.ashari@if.itera.ac.id.

**Sarwono Sutikno** received the B.Eng. degree in Electrical Engineering from Institut Teknologi Bandung (ITB), Indonesia, and the M.Eng. and Dr.Eng. degrees in Electrical Engineering from Tokyo Institute of Technology. He is currently a professor and faculty member at the Informatics Engineering, Institut Teknologi Sumatera (ITERA), Indonesia. His research interests include cyber security. He can be contacted at email: sarwono.sutikno@if.itera.ac.id.