



ARTICLE

Improving Video Watermarking through Galois Field $GF(2^4)$ Multiplication Tables with Diverse Irreducible Polynomials and Adaptive Techniques

Yasmin Alaa Hassan^{1,*} and Abdul Monem S. Rahma²

¹Department of Computer Science, College of Science, University of Baghdad, Baghdad, Iraq

²Department of Computer Science, College of Science, Al-Maarif University College, Al-Anbar, Iraq

*Corresponding Author: Yasmin Alaa Hassan. Email: yasmin.a@sc.uobaghdad.edu.iq

Received: 20 September 2023 Accepted: 29 November 2023 Published: 30 January 2024

ABSTRACT

Video watermarking plays a crucial role in protecting intellectual property rights and ensuring content authenticity. This study delves into the integration of Galois Field (GF) multiplication tables, especially $GF(2^4)$, and their interaction with distinct irreducible polynomials. The primary aim is to enhance watermarking techniques for achieving imperceptibility, robustness, and efficient execution time. The research employs scene selection and adaptive thresholding techniques to streamline the watermarking process. Scene selection is used strategically to embed watermarks in the most vital frames of the video, while adaptive thresholding methods ensure that the watermarking process adheres to imperceptibility criteria, maintaining the video's visual quality. Concurrently, careful consideration is given to execution time, crucial in real-world scenarios, to balance efficiency and efficacy. The Peak Signal-to-Noise Ratio (PSNR) serves as a pivotal metric to gauge the watermark's imperceptibility and video quality. The study explores various irreducible polynomials, navigating the trade-offs between computational efficiency and watermark imperceptibility. In parallel, the study pays careful attention to the execution time, a paramount consideration in real-world scenarios, to strike a balance between efficiency and efficacy. This comprehensive analysis provides valuable insights into the interplay of GF multiplication tables, diverse irreducible polynomials, scene selection, adaptive thresholding, imperceptibility, and execution time. The evaluation of the proposed algorithm's robustness was conducted using PSNR and NC metrics, and it was subjected to assessment under the impact of five distinct attack scenarios. These findings contribute to the development of watermarking strategies that balance imperceptibility, robustness, and processing efficiency, enhancing the field's practicality and effectiveness.

KEYWORDS

Video watermarking; galois field; irreducible polynomial; multiplication table; scene selection; adaptive thresholding

1 Introduction

In the rapidly evolving landscape of multimedia content, video watermarking stands as a crucial mechanism for safeguarding intellectual property rights, curbing unauthorized distribution, and ensuring the authenticity of digital content [1]. This field has witnessed significant advancements,



with researchers continuously exploring innovative techniques to strike a delicate balance between imperceptibility and robustness in watermarking applications [2,3].

Video watermarking is a technique used to embed hidden information, such as a logo, copyright notice, or owner's identification, into a video in a way that is imperceptible to human viewers [4,5]. The purpose of video watermarking is to protect intellectual property rights, prevent unauthorized distribution or piracy, and track the origin of the content [6]. One approach to video watermarking is by utilizing the mathematical properties of finite fields, specifically $GF(2^4)$, also known as the Galois field. A Galois field is a finite field with a prime number of elements, and $GF(2^4)$ is a particular type of Galois field with 16 elements [7,8]. It is represented using binary polynomials of degree 4. $GF(2^4)$ is a suitable choice for video watermarking due to its ability to perform bitwise operations, such as XOR, which are essential for manipulating the video data at the bit level. These operations allow for the seamless embedding of watermark information without significantly affecting the visual quality of the video [9].

Video watermarking has emerged as a pivotal technique for safeguarding multimedia content from unauthorized distribution and manipulation [10]. This study presents a comprehensive comparative analysis of video watermarking employing Galois Field (GF) multiplication tables with diverse irreducible polynomials in the context of $GF(2^4)$. In addition, the study incorporates an adaptive thresholding mechanism for scene selection, enhancing the watermarking process's efficiency.

The selection of an appropriate irreducible polynomial significantly influences the characteristics of the Galois Field, directly affecting the watermark embedding processes. To enhance the performance of watermarking schemes, an adaptive thresholding approach is introduced during scene detection, dynamically adjusting the threshold value based on the pixel intensity distribution within frame differences. This mechanism contributes to the identification of significant scene changes, optimizing the selection of frames for watermark embedding [11]. Watermarks are embedded in selected frames through XOR blending, utilizing watermark values represented via the chosen GF multiplication table. The study provides insights into the trade-offs between various irreducible polynomials, computational efficiency, and watermarking performance, considering adaptive scene selection.

The evaluation and validation of watermarking techniques are inherently multifaceted endeavors. The proposed approach leans on the Peak Signal-to-Noise Ratio (PSNR) as a yardstick to measure the watermark's imperceptibility and the resultant video quality. This metric provides a reliable gauge for understanding the subtle yet critical interplay between watermark imperceptibility, execution time, and computational complexity. In the realm of multimedia security and watermarking, several noteworthy developments have contributed to the evolution of techniques and methodologies. In one instance, a watermarking method, as discussed in [12], drew on the foundation of Discrete Wavelet Transform and Singular Value Decomposition (DWT-SVD) but elevated its capabilities by harnessing Particle Swarm Optimization (PSO). This adaptation sought to bolster the method's resistance to diverse attacks while preserving its imperceptibility. Furthermore, both the conventional and the enhanced algorithms exhibited a distinct diagonal line pattern in the extracted watermark image when subjected to specific noise attacks.

Meanwhile, reference [13] introduced an innovative approach that leveraged the Jigsaw Transform (JT) in tandem with the iterative cosine transform within a finite field for image encryption and decryption. The JT, characterized as a nonlinear operation, played a pivotal role in enhancing the security of encrypted images through the introduction of new encryption keys and decryption

mechanisms. This approach was firmly rooted in the principles of finite fields, where a finite set of integer numbers underpinned mathematical operations conducted using modular arithmetic.

The dynamic realm of video watermarking and scene change detection was explored in [14]. To address the challenge of detecting scene changes in complex environments, a novel approach was presented. Central to this approach was the introduction of a novel color descriptor, the local color difference pattern (LCDP), which remained resilient in the presence of shadows while effectively capturing both color and texture features at the pixel level. In addition, a scene change detection framework was introduced, built on the concept of sample consensus, seamlessly integrating LCDP and an innovative spatial model fusion mechanism.

Another pioneering method, as documented in [15], offered a compelling solution to alleviate the computational burden associated with video classification while simultaneously bolstering its accuracy. The heart of this approach lay in its intelligent frame selection process, which judiciously eliminated challenging-to-classify frames. Departing from the conventional practice of evaluating frames in isolation, this method undertook a collective assessment, resulting in a more efficient and balanced selection of ‘high-quality’ frames. The outcome resembled the creation of a coherent narrative through evenly distributed snapshots throughout the video. Reference [16] focused on the construction of S-boxes, particularly utilizing coset graphs and group actions involving the alternating group A_5 and the symmetric group S_{256} over finite fields. The core of the work involves analyzing the constructed S-boxes against various security criteria to assess their cryptographic strength. Additionally, the S-box’s performance in image encryption is evaluated, demonstrating its capability to enhance data security against cyber-attacks.

In a world where multimedia content is ceaselessly evolving, and where security concerns parallel the pace of innovation, this study holds the potential to redefine watermarking strategies. By unraveling the symphony of GF multiplication tables, irreducible polynomials, scene selection, adaptive thresholding, imperceptibility, execution time, and PSNR, this exploration advances our understanding and fortifies the foundation for future multimedia security endeavors [17].

The subsequent sections of this paper have been organized as follows: [Section 1.1](#) introduces the fundamentals of $GF(2^4)$, [Section 1.2](#) discusses Scene Detection and Frame Selection, [Section 1.3](#) elaborates on Adaptive Thresholding for Frame Selection, [Section 1.4](#) outlines image and video watermarking: similarities and differences. The proposed method is detailed in [Section 2](#), and the experimental results and discussions are presented in [Section 3](#). The conclusion is provided at the end of this work ([Section 4](#)).

1.1 $GF(2^4)$ Multiplication Table and Its Transformative Role

The utilization of Galois Field $GF(2^4)$ multiplication tables in video watermarking indeed represents a unique and innovative approach in the field of digital watermarking. This concept provides a novel knowledge-based way to manipulate pixel values within video frames, facilitating subtle and effective watermark embedding. While digital watermarking techniques have been explored extensively, the specific application of $GF(2^4)$ multiplication tables for this purpose appears to be a pioneering idea, offering new avenues for research and development in the field of multimedia security and content protection. Further exploration and experimentation in this area could yield valuable insights and potentially lead to advancements in video watermarking technology.

1.2 Scene Detection and Frame Selection

Scene detection involves the identification of transitions between scenes in the video, such as cuts or fades, to determine significant changes in content. These transitions indicate potential scene changes, which are crucial for determining the appropriate frames to watermark. The algorithm analyzes the differences between consecutive frames, considering factors like frame intensity, to pinpoint scene change instances. Once these instances are detected, a predetermined percentage of frames, often around 20% of the total frames, is selected for watermarking. The rationale behind this selection is to ensure a representative sample while avoiding overloading the video with watermarked frames, which could affect the video's perceptual quality [18–20].

1.3 Adaptive Thresholding for Frame Selection

This section explores the concept of adaptive thresholding, a critical aspect of frame selection. Adaptive thresholding is a dynamic technique that calculates thresholds based on the specific characteristics of the data under analysis. In the context of video watermarking, it plays a vital role in distinguishing frames with substantial content changes from those with minor variations.

Rather than relying on a single fixed threshold, this approach utilizes adaptive thresholding, allowing it to compute local thresholds for distinct video frames. This adaptive strategy is context-aware and responsive to local frame features, ensuring that frames are chosen based on their content and context. This dynamic approach enhances scene change detection precision and results in the selection of frames with unique and substantial content. The adaptive thresholding technique is particularly valuable when dealing with videos that encompass varying lighting conditions, complex scenes, or rapid content alterations. It effectively overcomes the limitations associated with traditional frame selection methods [21,22].

Moreover, the use of adaptive thresholding is well-suited for video watermarking applications, as it helps ensure that watermarks are inserted into frames where they are least likely to disrupt the video's quality or visibility. This balances the imperceptibility and robustness of the watermark, a critical factor in ensuring viewer satisfaction and the preservation of the video's authenticity. The application of adaptive thresholding ensures that the scene detection process remains adaptable to variations in lighting, motion, and content complexity, resulting in a more precise and context-aware selection of frames for watermarking [23,24].

1.4 Image and Video Watermarking: Similarities and Differences

This section delves into the world of watermarking and explores the shared attributes and distinctions between watermarking techniques applied to both images and videos. By highlighting these similarities and differences, a comprehensive understanding of how watermarking strategies are adapted to suit the characteristics of images and videos is provided, offering valuable insights into the domain of multimedia content protection.

1.4.1 Similarities

- **Embedding Process:** Both photos and videos undergo an embedding process to insert a watermark. This typically involves altering the pixel values in the media to incorporate the watermark information.
- **Watermark Extraction:** Both photos and videos require an extraction process to retrieve the watermark. Special algorithms and keys are used to separate the watermark from the media.

- Purpose: The primary purpose of watermarking in both photos and videos is to claim ownership, protect copyright, and deter unauthorized use.

1.4.2 Differences

- Temporal Context: Frames in a video are temporally related, and it is essential to consider this temporal context when watermarking videos. Unlike images, videos involve a sequence of frames where each frame follows the preceding one.
- Scene Changes: Videos often contain scene changes, where the content or context may shift abruptly. These scene changes impact the watermarking process. In contrast, images do not have scene changes, making the watermarking process more straightforward.
- Efficiency: Frame processing allows to optimization of the algorithm by watermarking only selected frames, typically at scene changes or predetermined intervals. This approach reduces computational load and minimizes potential visual artifacts in the video.

In summary, while image processing deals with standalone images, frame processing considers the unique characteristics of video data, such as temporal context, scene changes, efficiency, and visual continuity. These distinctions are necessary for the effective and robust watermarking of videos, ensuring that the watermark is both imperceptible and resistant to removal [3,25–27].

2 Proposed Method

Watermarking, a pivotal facet of multimedia security, is tasked with embedding hidden information into digital content to safeguard intellectual property, combat unauthorized distribution, and ensure authenticity. However, watermarking presents challenges, including imperceptibility to human perception, and computational efficiency. This study seeks to tackle these challenges by creatively incorporating GF multiplication tables with scene selection techniques based on adaptive thresholding. The main goal is to improve the efficiency of watermarking by utilizing the unique properties offered by these GF multiplication tables. These tables, rooted in abstract algebra, offer properties conducive to robust watermarking, such as information hiding, error correction, and tamper detection. The methodology centers on harnessing these properties for watermark embedding, ultimately contributing to a more secure and authentic multimedia landscape. The effective utilization of GF multiplication tables hinges on their meticulous construction. This involves creating these tables and subsequently employing them as a transformative tool for embedding watermarks. The transformation involves XOR blending of watermark values derived from the selected GF multiplication table. The strategic choice of employing the Galois field with 16 elements ($GF(2^4)$) significantly influenced the research direction in this study. Deliberately prioritizing the 4 least significant bits (LSB) within this field was undertaken to mitigate any potential impact on video quality. It is important to note that the 4 LSB represent a relatively minor portion of the data, as opposed to the 4 most significant bits (MSB). Specifically, $GF(2^4)$ ($GF(16)$) was opted for to strike an equilibrium between computational complexity and imperceptibility. By concentrating on the 4 LSB, video quality was successfully maintained at an acceptable level while managing computational demands. Various irreducible polynomials within $GF(2^4)$, such as $(x^4 + x + 1)$ or $(x^4 + x^3 + 1)$ or $(x^4 + x^3 + x^2 + x + 1)$ have been used as a comparative exploration of watermarking process. Irreducible polynomials fundamentally influence the algebraic properties of GF multiplication tables, thereby affecting watermarking robustness and imperceptibility. The experiment examines these polynomials to identify the most effective one that yields optimal results, both in terms of secure embedding, good quality, and less execution time. Addressing the challenge of selecting frames for watermarking, an adaptive threshold-based approach

has been adopted that ensures both diversity and relevance. Specifically, approximately 20% of frames in a video have been chosen to add a watermark.

Overall, this algorithm performs video watermarking by selecting frames based on scene changes and then blending the watermark to the four corners of selected frames using a $GF(2^4)$ table. The watermarking process is controlled by the frame selection technique, which determines the number of frames to watermark based on the adaptive threshold which is determined automatically depending on the pixel intensity distribution in each video. The duration of the videos varies, with some being lengthy and comprising a substantial frame count, while others are shorter and encompass fewer frames. The number of frames to watermark is set to 20% of the total frames, making it a general approach that works for different videos without manually adjusting the threshold value. With this approach, the threshold is determined dynamically based on the pixel intensity distribution in each video. Therefore, it works reasonably well for different videos without the need to manually adjust the threshold value.

2.1 $GF(2^4)$ Multiplication Tables Construction

The construction of multiplication tables for $GF(2^4)$ with different irreducible polynomials offers a pivotal exploration into the realm of finite fields. These tables serve as fundamental tools for cryptographic operations and encoding techniques within $GF(2^4)$. This comparative study delves into the creation of three distinct multiplication tables using three specific irreducible polynomials. Through a systematic process of generating elements via polynomial reductions and multiplication operations, the distinct properties and attributes of these diverse multiplication tables have been revealed, contributing significant insights to the domain. The following algorithm illustrates the $GF(2^4)$ multiplication table construction using three irreducible polynomials that result in different three tables used as transformations in this work.

Algorithm: Multiplication table construction in $GF(2^4)$

Input: Irreducible Polynomial $(x^4 + x + 1)$, $(x^4 + x^3 + 1)$ and $(x^4 + x^3 + x^2 + x + 1)$

Output: Three multiplication tables of $GF(2^4)$ of size 15×15

1. **Begin**

2. Create two lists, r, and c, containing values from 1 to 15. These lists represent the rows and columns of the multiplication table.

3. Initialize a 2D list mulTABLE4 as a 15×15 matrix filled with zeros.

4. Define the mpy(x, y) function that multiplies two 8-bit values x and y using bitwise operations and reduction modulo the irreducible polynomial $(x^4 + x + 1)$ or $(x^4 + x^3 + 1)$ or $(x^4 + x^3 + x^2 + x + 1)$.

5. Iterate over the range of values in the list r.

For each r[i] value, iterate over the range of values in the list c.

- Access the corresponding cell in mulTABLE4 using mulTABLE4[i][j].
- Assign the product of r[i] and c[j] calculated using the mpy() function to mulTABLE4[i][j].

6. Print the resulting mulTABLE4, which represents the multiplication table.

7. **End.**

2.2 Frame Selection and Adaptive Thresholding

The calculation of the frame difference can be represented by the following equation:

$$frame_diff(i, j) = |I(i, j) - I_{prev}(i, j)|$$

where:

- $frame_diff(i, j)$: Pixel value of the frame difference at location (i, j)
- $I(i, j)$: Pixel value of the current frame at location (i, j)
- $I_prev(i, j)$: Pixel value of the previous frame at location (i, j)
- $||$: Absolute value

This equation calculates the absolute difference between the pixel values of the corresponding locations in the current frame and the previous frame, resulting in the frame difference image. It is a key step in detecting scene changes and determining where the watermark should be added.

The calculation of the adaptive threshold used for scene change detection can be represented by the following equation:

$$Threshold = \frac{hist_sum(k)}{total_pixels} \times (total_hist_sum - hist_sum(k))$$

- $Threshold$: represents the adaptive threshold used for scene change detection.
- $hist_sum(k)$: Cumulative histogram sum up to bin k
- $total_pixels$: Total number of pixels in the frame
- $total_hist_sum$: Cumulative histogram sum of the entire frame

In this equation, the adaptive threshold is calculated based on the pixel intensity distribution in the frame difference image. It helps in determining whether the current frame indicates a scene change by comparing the mean difference of the frame with the adaptive threshold.

2.3 Video Watermarking Using $GF(2^4)$ Multiplication Tables

The forthcoming elucidation delves into each sequential stage of this process, delving into how mathematical theory seamlessly intertwines with video files to attain an exceptional watermark embedding outcome.

Algorithm: Add a watermark to a video using the GF table as a transformation

Input: Cover video, watermark image, $GF(2^4)$ multiplication tables using irreducible polynomials $x^4 + x + 1$, $x^4 + x^3 + 1$, and $x^4 + x^3 + x^2 + x + 1$

Output: Watermarked video

Begin

1. Load the video file
 2. Get the properties of the video (frame width, frame height, frames per second, total frames)
 3. Define the percentage of frames to select for watermarking (e.g., 20%)
 4. Calculate the number of frames to select based on the total frames and the selected percentage
 5. Perform scene detection on the video to identify scene changes
 6. Create an empty list to store the scene change frames
 7. Loop through each frame in the video
 - 7.1. Read the frame
 - 7.2. Convert the frame to grayscale
 - 7.3. If it is not the first frame
 - 7.3.1. Calculate the absolute difference between the current frame and the previous frame
 - 7.3.2. Calculate the mean difference of the frame
-

(Continued)

Algorithm (continued)

-
- 7.3.3. Calculate an adaptive threshold based on the pixel intensity distribution in the frame difference.
 - 7.3.4. If the mean difference is greater than the threshold, add the current frame number to the list of scene changes
 - 7.3.5. Set the current frame as the previous frame for the next iteration
 8. Close the video
 9. Load the watermark image
 10. Convert the watermark pixel values to $GF(2^4)$ table values
 11. Create a VideoWriter object to write the output watermarked video
 12. Create variables to store PSNR values and execution times for each watermarked frame
 13. Loop through each frame in the video again
 - 13.1. Read the frame
 - 13.2. If the current frame number is in the list of scene changes and it's the nth frame to select (based on the calculated value in step 4)
 - 13.2.1. Record the start time for execution time calculation
 - 13.2.2. Add the watermark on the four corners to the selected frame using XOR blending with the converted watermark values
 - 13.2.3. Record the end time and calculate the execution time
 - 13.2.4. Compute the PSNR value between the original frame and the watermarked frame
 - 13.2.5. Save the watermarked frame to the output video
 - 13.2.6. Add the PSNR value to the list of PSNR values
 - 13.2.7. Add the execution time to the list of execution times
 - 13.3. Else, save the original frame without watermarking the output video
 14. Close the output video
 15. Calculate the average PSNR value and average execution time
 16. Display or save the average PSNR and average execution time
- End.**
-

Fig. 1 illustrates the block diagram of the proposed algorithm visually. This algorithm begins by loading a video and extracting its properties. It then intelligently selects a subset of frames for watermarking based on a user-defined percentage and scene change detection. The watermark, converted into $GF(2^4)$ values, is embedded into the selected frames, and the process is timed and assessed using the Peak Signal-to-Noise Ratio (PSNR). The watermarked frames are saved, and the algorithm calculates the average PSNR and execution time for performance evaluation. This approach optimizes the watermarking process for video while considering scene dynamics, ensuring robustness and content preservation. The algorithm seamlessly integrates the watermark with the frames by employing bitwise operations and a carefully selected $GF(2^4)$ multiplication table. It prioritizes the 4 least significant bits (LSB), which represent a small fraction of the video data, ensuring the imperceptibility of the watermark while preserving the overall video quality.

Overall, this algorithm performs video watermarking by selecting frames based on scene changes and then blending the watermark in selected frames using a $GF(2^4)$ table. The more frames you select for watermarking, the lower the PSNR is likely to be due to the increased modifications to the video content. Conversely, if you select too few frames, the watermark might not be robust enough. You can adjust this value based on your requirements and the desired trade-off between watermark imperceptibility and robustness.

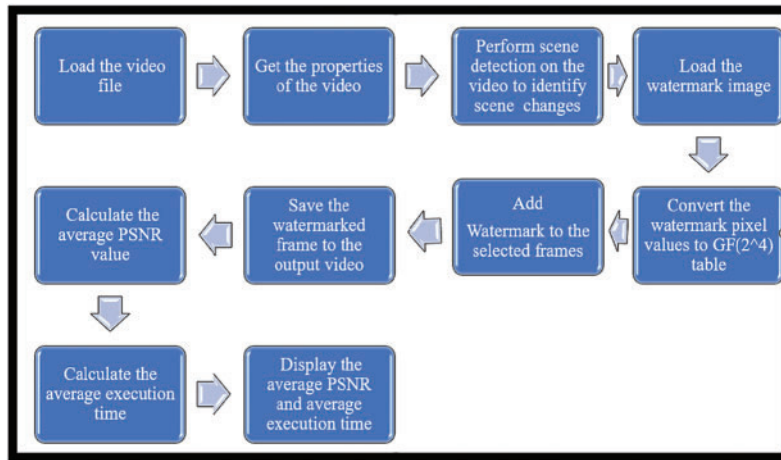


Figure 1: Block diagram of the proposed system

2.4 Image Watermarking Using $GF(2^4)$ Multiplication Table

In contrast, image watermarking primarily deals with individual frames, requiring unique strategies. While image watermarking may lack the complexities tied to scene changes, it places greater emphasis on spatial fidelity and perceptual quality over temporal considerations.

Algorithm: Image watermarking using $GF(2^4)$ multiplication table

Input: multiplication table of $GF(2^4)$, cover image, watermark image.

Output: Watermarked image.

1. **Begin**

2. Load the cover image and watermark image.

3. Define the $GF(2^4)$ table containing the values for the desired mapping.

4. Create a copy of the cover image as the watermarked image.

5. Resize the watermark image to match the desired size if needed.

6. Convert the pixel values of the watermark image to their corresponding values in the $GF(2^4)$ table:

-Iterate over the pixels of the watermark image.

-Extract the lower 4 bits (y) and upper 4 bits (x) of each pixel value.

-Map the values x and y to their corresponding values in the $GF(2^4)$ table using `gf_table[x][y]`.

-Update the corresponding pixels in the converted watermark image with the mapped values.

7. Embed the converted watermark in the corners of the cover image:

-Iterate over the pixels of the converted watermark image.

-For each pixel in the converted watermark image:

-Retrieve the corresponding pixel in the watermarked image.

-Perform a blending operation using XOR between the watermarked image pixel and the converted watermark pixel using the $GF(2^4)$ table.

-Update the corresponding pixel in the watermarked image with the blended value.

8. Save the watermarked image and the robustness evaluation results as desired.

9. **End.**









This algorithm ensures that the watermark is embedded in a manner that minimizes its visibility to the human eye while utilizing the $GF(2^4)$ table for robustness against image processing operations.

A $GF(2^4)$ multiplication table has been employed in this investigation along with the irreducible polynomial $(x^4 + x + 1)$. Image watermarking, unlike video watermarking, does not encompass the intricacies of scene selection, making the process significantly more straightforward. The absence of temporal considerations in image watermarking allows for a more direct focus on the spatial attributes of individual frames, eliminating the need for scene change detection and adaptive thresholding. Instead, image watermarking emphasizes spatial fidelity, ensuring that the watermark is subtly embedded within the image to maintain its quality and visibility, all within the confines of a single, static frame.

3 Experimental Results and Discussions

The experimental results obtained from the conducted study shed light on the effectiveness and potential of utilizing Galois Field (GF) multiplication tables for video watermarking. Through the utilization of various irreducible polynomials in the construction of these tables, the intricate interplay between mathematical foundations and practical applications was sought to be uncovered. Through rigorous experimentation and analysis, our study delved into the robustness, imperceptibility, and computational complexity aspects of the video watermarking process. The obtained results offer valuable insights into the influence of different irreducible polynomials on watermarking performance, contributing to a deeper understanding of how algebraic structures can be harnessed to enhance multimedia security solutions. In this study, a range of video files were employed, (v1, v2, and v3) have been obtained from a personal camera to ensure the preservation of others' rights. These videos encompass indoor and outdoor scenes, impacting both the content and specific details within the videos. Furthermore, the comparison entails the utilization of three conventional video files (Foreman, Akiyo, and Coastguard), which are commonly employed in academic literature. The videos utilized were in the (.mp4) format, while the watermark image was in (.png) format, using two resolutions for the same image as the watermark named watermark1 and watermark 2, featuring a resolution of (225×225) and (100×100) . The experimentation was conducted on the Python platform version 3.8, running on a Windows 10 system with a Core i7 CPU and 16 GB of RAM. [Table 1](#) illustrates the cover video data and watermark images used in this study.

Table 1: The cover videos and the watermark image utilized in this study

Cover videos			Watermark image
V1	V2	V3	225×225
			
Foreman	Akiyo	Coastguard	100×100
			

Tables 2–4 illustrate the constructed $GF(2^4)$ multiplication tables based on the three irreducible polynomials.

Table 2: Multiplication table of $GF(2^4)$ with irreducible polynomial $x^4 + x + 1$

Multiplication Table with $x^4 + x + 1$
[1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 5],
[2, 4, 6, 8, 10, 12, 14, 3, 1, 7, 5, 11, 9, 15, 13],
[3, 6, 5, 12, 15, 10, 9, 11, 8, 13, 14, 7, 4, 1, 2],
[4, 8, 12, 3, 7, 11, 15, 6, 2, 14, 10, 5, 1, 13, 9],
[5, 10, 15, 7, 2, 13, 8, 14, 11, 4, 1, 9, 12, 3, 6],
[6, 12, 10, 11, 13, 7, 1, 5, 3, 9, 15, 14, 8, 2, 4],
[7, 14, 9, 15, 8, 1, 6, 13, 10, 3, 4, 2, 5, 12, 11],
[8, 3, 11, 6, 14, 5, 13, 12, 4, 15, 7, 10, 2, 9, 1],
[9, 1, 8, 2, 11, 3, 10, 4, 13, 5, 12, 6, 15, 7, 14],
[10, 7, 13, 14, 4, 9, 3, 15, 5, 8, 2, 1, 11, 6, 12],
[11, 5, 14, 10, 1, 15, 4, 7, 12, 2, 9, 13, 6, 8, 3],
[12, 11, 7, 5, 9, 14, 2, 10, 6, 1, 13, 15, 3, 4, 8],
[13, 9, 4, 1, 12, 8, 5, 2, 15, 11, 6, 3, 14, 10, 7],
[14, 15, 1, 13, 3, 2, 12, 9, 7, 6, 8, 4, 10, 11, 5],
[15, 13, 2, 9, 6, 4, 11, 1, 14, 12, 3, 8, 7, 5, 10]]

Table 3: Multiplication table of $GF(2^4)$ with irreducible polynomial $x^4 + x^3 + 1$

Multiplication table $x^4 + x^3 + 1$
[1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15],
[2, 4, 6, 8, 10, 12, 14, 9, 11, 13, 15, 1, 3, 5, 7],
[3, 6, 5, 12, 15, 10, 9, 1, 2, 7, 4, 13, 14, 11, 8],
[4, 8, 12, 9, 13, 1, 5, 11, 15, 3, 7, 2, 6, 10, 14],
[5, 10, 15, 13, 8, 7, 2, 3, 6, 9, 12, 14, 11, 4, 1],
[6, 12, 10, 1, 7, 13, 11, 2, 4, 14, 8, 3, 5, 15, 9],
[7, 14, 9, 5, 2, 11, 12, 10, 13, 4, 3, 15, 8, 1, 6],
[8, 9, 1, 11, 3, 2, 10, 15, 7, 6, 14, 4, 12, 13, 5],
[9, 11, 2, 15, 6, 4, 13, 7, 14, 12, 5, 8, 1, 3, 10],
[10, 13, 7, 3, 9, 14, 4, 6, 12, 11, 1, 5, 15, 8, 2],
[11, 15, 4, 7, 12, 8, 3, 14, 5, 1, 10, 9, 2, 6, 13],
[12, 1, 13, 2, 14, 3, 15, 4, 8, 5, 9, 6, 10, 7, 11],
[13, 3, 14, 6, 11, 5, 8, 12, 1, 15, 2, 10, 7, 9, 4],
[14, 5, 11, 10, 4, 15, 1, 13, 3, 8, 6, 7, 9, 2, 12],
[15, 7, 8, 14, 1, 9, 6, 5, 10, 2, 13, 11, 4, 12, 3]]

Table 4: Multiplication table of $GF(2^4)$ with irreducible polynomial $x^4 + x^3 + x^2 + x + 1$

Multiplication table with IR $x^4 + x^3 + x^2 + x + 1$	
[1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15],	
[2, 4, 6, 8, 10, 12, 14, 15, 13, 11, 9, 7, 5, 3, 1],	
[3, 6, 5, 12, 15, 10, 9, 7, 4, 1, 2, 11, 8, 13, 14],	
[4, 8, 12, 15, 11, 7, 3, 1, 5, 9, 13, 14, 10, 6, 2],	
[5, 10, 15, 11, 14, 1, 4, 9, 12, 3, 6, 2, 7, 8, 13],	
[6, 12, 10, 7, 1, 11, 13, 14, 8, 2, 4, 9, 15, 5, 3],	
[7, 14, 9, 3, 4, 13, 10, 6, 1, 8, 15, 5, 2, 11, 12],	
[8, 15, 7, 1, 9, 14, 6, 2, 10, 13, 5, 3, 11, 12, 4],	
[9, 13, 4, 5, 12, 8, 1, 10, 3, 7, 14, 15, 6, 2, 11],	
[10, 11, 1, 9, 3, 2, 8, 13, 7, 6, 12, 4, 14, 15, 5],	
[11, 9, 2, 13, 6, 4, 15, 5, 14, 12, 7, 8, 3, 1, 10],	
[12, 7, 11, 14, 2, 9, 5, 3, 15, 4, 8, 13, 1, 10, 6],	
[13, 5, 8, 10, 7, 15, 2, 11, 6, 14, 3, 1, 12, 4, 9],	
[14, 3, 13, 6, 8, 5, 11, 12, 2, 15, 1, 10, 4, 9, 7],	
[15, 1, 14, 2, 13, 3, 12, 4, 11, 5, 10, 6, 9, 7, 8]]	

3.1 Evaluation Metrics

Peak Signal-to-Noise Ratio (PSNR) is a metric widely employed to assess the imperceptibility and quality of images or videos after applying various transformations or watermarking techniques. It quantifies the difference between the original and the modified image, giving insight into the level of distortion introduced by the process.

Mathematically, PSNR is calculated using the Mean Squared Error (MSE) between the original and the modified images, which measures the average squared difference between corresponding pixel values:

$$MSE = \frac{1}{M \times N} \sum_{i=1}^M \sum_{j=1}^N [I(i,j) - K(i,j)]^2$$

- $I(i,j)$ represents the pixel value of the original image at location (i,j) ,
- $K(i,j)$ represents the pixel value of the modified image (e.g., watermarked image) at the same location,
- M and N are the dimensions of the image.

The PSNR is then calculated using the MSE as follows:

$$PSNR = 10 \times \log_{10} \left(\frac{MAX^2}{MSE} \right)$$

where: MAX is the maximum possible pixel value (e.g., 255 for 8-bit images).

In the context of watermarking, comparing the PSNR values of watermarked images with their corresponding original images helps to quantify the perceptual quality of the watermarking process,

providing insight into how well the watermark is embedded while maintaining the visual integrity of the content.

Normalized Correlation (NC) is another essential metric for evaluating the quality of watermarked images. It quantifies the degree of similarity or correlation between the original and watermarked images. NC ranges from -1 (perfect negative correlation) to 1 (perfect positive correlation), with 0 indicating no correlation. The Normalized cross-Correlation (NC) between two signals or images can be computed using the following equation:

$$NC = \frac{\sum(I_1 - I_1')(I_2 - I_2')}{\sqrt{(\sum(I_1 - I_1')^2) \sum(I_2 - I_2')^2}}$$

where:

- I_1 and I_2 are the two signals or images being compared.
- I_1' and I_2' are the means (averages) of I_1 and I_2 , respectively.
- The numerator represents the cross-covariance between I_1 and I_2 .
- The denominator is the product of the square roots of the variances of I_1 and I_2 .

3.2 Results of Video Watermarking Using $GF(2^4)$ Multiplication Tables

In Fig. 2, a series of frames is provided as an illustrative example of a watermarked video. These frames demonstrate the process of watermark embedding at all four corners of the video frames. While the watermark in the algorithm is imperceptible, it has been deliberately made visible in this example solely for illustrative purposes. The first row of frames depicts the original cover video frames before the watermark is embedded. In the second row, you can see the watermark applied to all frames without considering scene selection. However, it is important to note that this approach can be time-consuming and may consume significant storage space.



Figure 2: An example of video frames before and after watermarking

On the other hand, when employing a scene selection process, the watermarking scheme becomes more efficient. This streamlined approach not only accelerates the watermarking process but also enhances its practicality for real-world applications.

The proposed approach's application using $GF(2^4)$ multiplication tables, established through three distinct irreducible polynomials, is demonstrated in Tables 5–10. Tables 5–7 use watermark 1 with resolution 225×225 and Tables 8–10 use watermark 2 with resolution 100×100 . These tables showcase the utilization of the scheme across varying video lengths. Notably, the No. of watermarked frames pertains to a selection of 20% of the entire video frames. This selection is accomplished using a frame selection technique that focuses exclusively on significant scenes. The integration of an adaptive threshold within this process expedites watermark embedding.

Table 5: Results of the proposed scheme using $GF(2^4)$ with irreducible polynomial $x^4 + x + 1$ and watermark 1

Video name	No. of frames	No. of watermarked frames (20%)	PSNR	Execution time of the embedding process in seconds
V1	2744	548	45.6145	0.3689
V2	3893	778	43.6996	0.3565
V3	1811	362	49.0775	0.3774
Foreman	301	60	53.1382	0.3736
Akiyo	288	57	54.0502	0.3824
Coastguard	946	189	52.3150	0.3900

Table 6: Results of the proposed scheme using $GF(2^4)$ with irreducible polynomial $x^4 + x^3 + 1$ and watermark 1

Video name	No. of frames	No. of watermarked frames (20%)	PSNR	Execution time of the embedding process in seconds
V1	2744	548	45.8208	0.3701
V2	3893	778	43.9309	0.3566
V3	1811	362	49.2646	0.3762
Foreman	301	60	53.3907	0.3665
Akiyo	288	57	54.3731	0.3410
Coastguard	946	189	53.6556	0.3699

Table 7: Results of the proposed scheme using $GF(2^4)$ with irreducible polynomial $x^4 + x^3 + x^2 + x + 1$ and watermark 1

Video name	No. of frames	No. of watermarked frames (20%)	PSNR	Execution time of the embedding process in seconds
V1	2744	548	45.7157	0.3703
V2	3893	778	43.7964	0.3749
V3	1811	362	49.1502	0.3784
Foreman	301	60	53.2218	0.3534
Akiyo	288	57	54.1399	0.3770
Coastguard	946	189	53.4087	0.3903

Table 8: Results of the proposed scheme using $GF(2^4)$ with irreducible polynomial $x^4 + x + 1$ and watermark2

Video name	No. of frames	No. of watermarked frames (20%)	PSNR	Execution time of the embedding process in seconds
V1	2744	548	47.6463	0.0878
V2	3893	778	45.5185	0.0773
V3	1811	362	51.8318	0.0909
Foreman	301	60	54.1104	0.0721
Akiyo	288	57	54.0287	0.0803
Coastguard	946	189	53.0444	0.0830

Table 9: Results of the proposed scheme using $GF(2^4)$ with irreducible polynomial $x^4 + x^3 + 1$ and watermark2

Video name	No. of frames	No. of watermarked frames (20%)	PSNR	Execution time of the embedding process in seconds
V1	2744	548	47.8708	0.0814
V2	3893	778	45.9658	0.0769
V3	1811	362	51.5688	0.0883
Foreman	301	60	55.7475	0.0732
Akiyo	288	57	54.6825	0.0871
Coastguard	946	189	52.9347	0.0731







Table 10: Results of the proposed scheme using $GF(2^4)$ with irreducible polynomial $x^4 + x^3 + x^2 + x + 1$ and watermark2

Video name	No. of frames	No. of watermarked frames (20%)	PSNR	Execution time of the embedding process in seconds
V1	2744	548	46.5522	0.0826
V2	3893	778	44.3455	0.0886
V3	1811	362	50.7278	0.0811
Foreman	301	60	54.9508	0.0743
Akiyo	288	57	54.9256	0.0768
Coastguard	946	189	52.7517	0.0981

3.3 Results of Image Watermarking Using $GF(2^4)$ Multiplication Table

In Table 11, the results of embedding an invisible mark or signal into digital images using the image watermarking process to protect intellectual property, authenticate users, and identify the content of the image. Utilizing $GF(2^4)$, one of the mathematical characteristics of GFs , as a transformation mechanism is one way to accomplish this.

Table 11: Image watermarking using $GF(2^4)$

Cover image	Watermark image	PSNR
		58.5504
		58.8200
		58.6782

3.4 Robustness Measurements

In [Table 12](#), the result of robustness experiments that have been presented and conducted to evaluate the performance of the watermarking algorithm under a variety of real-world attacks. It presents the results of various attacks on three video sequences (Akiyo, Foremen, Coastguard), focusing on the measurements of PSNR (Peak Signal-to-Noise Ratio) and NC (Normalized Correlation) for each attack. These metrics are commonly used in video watermarking to assess the quality and robustness of watermarked videos under different attacks. The results show the trade-off between video quality (PSNR) and the robustness of the watermark to various attacks. While some attacks result in reduced PSNR values, indicating a quality loss, the watermark remains relatively robust, as suggested by the high NC values. This implies that even under these attacks, the watermark can still be reliably detected. The choice of attack should align with the desired balance between video quality and watermark robustness in a specific application.

Table 12: PSNR and NC of applying geometric attacks on watermarked video

Attacks	Akiyo		Foremen		Coastguard	
	PSNR	NC	PSNR	NC	PSNR	NC
Compression $q = 90$	42.649	0.995	40.310	0.974	36.449	0.984
Compression $q = 70$	39.462	0.989	36.611	0.961	34.724	0.977
Cropping	28.908	0.998	28.603	0.985	28.551	0.996
Frame removal $N = 20$	36.129	0.978	33.207	0.952	28.597	0.969
Frame removal $N = 40$	35.695	0.914	32.588	0.892	28.393	0.897
Rotation $Th = 30$	28.265	0.889	28.157	0.833	28.121	0.866
Rotation $Th = 60$	27.943	0.852	28.140	0.835	27.990	0.836
Rotation $Th = 90$	27.926	0.814	28.137	0.827	27.992	0.811
Salt & pepper $p_1 = 0.1, p_2 = 0.1$	31.150	0.974	31.707	0.961	31.435	0.965

(Continued)

Table 12 (continued)

Attacks	Akiyo		Foremen		Coastguard	
	PSNR	NC	PSNR	NC	PSNR	NC
Salt & pepper $p_1 = 0.01, p_2 = 0.01$	40.039	0.988	40.586	0.978	40.326	0.982
Salt & pepper $p_1 = 0.001, p_2 = 0.001$	45.950	0.991	44.485	0.992	46.222	0.994

Note: *Q = quality; N = no. of removed frames; Th = angle; p_1, p_2 = probability 1 and probability 2.

In [Table 13](#), a comparison of average PSNR values between the proposed method and other approaches using the standard videos (foreman, Akiyo, coastguard). The average PSNR of the proposed method was computed using the three irreducible polynomials with two watermarks. By analyzing the Peak Signal-to-Noise Ratio values, the performance and efficacy of our approach in video watermarking are assessed in comparison to alternative methods. This evaluation allows for insights into the robustness and quality of the proposed technique, highlighting its potential superiority in upholding watermark imperceptibility while contending with various challenges posed by different scenarios.

Table 13: Comparison of the average PSNR values of the proposed method with other approaches

Video name	[28]	[29]	[30]	Proposed method with watermark1	Proposed method with watermark2
Foreman	53.0568	47.1679	51.37	53.2502	54.9362
Akiyo	54.0078	48.0193	53.96	54.1876	54.5456
Coastguard	52.5768	–	49.25	53.1264	52.9103

Upon careful examination of the data presented in [Tables 4–6](#), a clear trend emerges: the utilization of the second irreducible polynomial ($x^4 + x^3 + 1$) consistently leads to the most favorable and promising outcomes. This polynomial exhibits a remarkable ability to effectively integrate with the *GF* multiplication tables and significantly enhance the watermarking process. The watermark embedding results achieved through the application of this polynomial showcase higher levels of imperceptibility and robustness when compared to the other irreducible polynomials. This superiority can be attributed to the specific mathematical characteristics inherent in the $x^4 + x^3 + 1$ polynomial, which enables it to facilitate a more seamless and efficient transformation mechanism for watermark embedding.

The algorithm exhibits linear time complexity concerning the number of frames the complexity of scene change detection, and the dimensions of video frames, and it requires additional space for scene change frames and watermark storage. The exact computational complexity may vary depending on the specific implementation and optimizations. The overall computational complexity is approximately $O(NM + WH + K)$, where the specific values of N (number of frames), M (number of scenes detected), W (width), H (height), and K (denotes the space required for watermark storage) depend on the video and watermark characteristics. As these values increase, the algorithm's computational demands will also increase accordingly.

As the number of frames chosen for watermarking increases, the likelihood of a decrease in PSNR due to heightened alterations in video content becomes evident. On the contrary, opting for a minimal

number of frames may lead to insufficient watermark robustness. This value can be adapted by the requisites and the intended balance between watermark imperceptibility and resilience (1).

4 Conclusion

Video watermarking using $GF(2^4)$ is a powerful technique for embedding hidden information in video streams. By leveraging the mathematical properties of Galois fields, this approach provides an effective means of protecting intellectual property rights, ensuring content integrity, and enabling the traceability of videos in various applications.

The frame selection and adaptive thresholding with $GF(2^4)$ tables using different irreducible polynomials to blend the watermark with the most important frames, ensuring that the watermark is imperceptible while minimizing execution time. The outcomes of this investigation demonstrate favorable PSNR values, particularly notable when employing the second irreducible polynomial ($x^4 + x^3 + 1$).

Overall, this algorithm performs video watermarking by selecting frames based on scene changes and then blending the watermark in selected frames using a $GF(2^4)$ table. The more frames have been selected for watermarking, the lower the PSNR is likely to be due to the increased modifications to the video content. Conversely, if the selected frames were too few, the watermark might not be robust enough. Therefore, it is important to adjust this value based on the requirements and the desired trade-off between watermark imperceptibility and robustness. The limitation arises when working with very short videos, as the outcomes tend to be ambiguous and unclear due to the inadequate presence of sufficient scenes for robust watermarking evaluation. The proposed method shows good results that are superior to the others in this field. In a world where multimedia content is ceaselessly evolving, and where security concerns parallel the pace of innovation, this study holds the potential to redefine watermarking strategies. By unraveling the symphony of GF multiplication tables, irreducible polynomials, scene selection, adaptive thresholding, imperceptibility, execution time, and PSNR, this exploration advances our understanding and fortifies the foundation for future multimedia security endeavors. Regarding future work, efforts will be made to investigate techniques aimed at further optimizing watermarking systems, with a particular emphasis on real-time applications such as live streaming and video conferencing.

Acknowledgement: None.

Funding Statement: The authors received no specific funding for this study.

Author Contributions: The authors confirm contribution to the paper as follows: study conception and design: Yasmin Alaa Hassan and Abdul Monem S. Rahma; data collection, analysis and interpretation of results, draft manuscript preparation: Yasmin Alaa Hassan. Both authors reviewed the results and approved the final version of the manuscript.

Availability of Data and Materials: Using two different sources of video data in this study: 1. Private Camera Data: This refers to video data that is obtained from private sources, such as personal cameras or non-publicly accessible sources. This data may not be readily available to the general public and might require specific permissions or access. 2. Publicly Available Web Data: This indicates that another portion of video data comes from publicly available sources on the internet, such as websites or platforms where videos are shared openly and can be accessed by anyone without restrictions.

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

References

- [1] P. Chavan, D. Jadhav and G. M. Borkar, “Challenges to multimedia privacy and security over social media,” in *Handbook of Research on Cyber Crime and Information Privacy*, Hershey, Pennsylvania, USA: IGI Global, pp. 118–131, 2021.
- [2] L. Kumar, K. U. Singh and I. Kumar, “A comprehensive review on digital image watermarking techniques,” in *2023 Int. Conf. on Computational Intelligence and Sustainable Engineering Solutions (CISES)*, Greater Noida, India, IEEE, pp. 737–743, 2023.
- [3] Y. A. Hassan and A. M. S. Rahmah, “An overview of robust video watermarking techniques,” *Iraqi Journal of Science*, vol. 64, no. 7, pp. 4513–4524, 2023.
- [4] H. Lakshmi and S. Borra, “Digital video watermarking tools: An overview,” *International Journal of Information and Computer Security*, vol. 16, no. 1–2, pp. 1–19, 2021.
- [5] A. D. Salman and H. B. Abdulwahab, “Proposed copyright protection systems for 3D video based on key frames,” *Iraqi Journal of Computers, Communications, Control and Systems Engineering*, vol. 20, no. 3, pp. 1–15, 2020.
- [6] R. Hamza and H. Pradana, “A survey of intellectual property rights protection in big data applications,” *Algorithms*, vol. 15, no. 11, pp. 418, 2022.
- [7] S. David Müzel, E. P. Bonhin, N. M. Guimarães and E. S. Guidi, “Application of the finite element method in the analysis of composite materials: A review,” *Polymers*, vol. 12, no. 4, pp. 818, 2020.
- [8] A. Broumandnia, “Image encryption algorithm based on the finite fields in chaotic maps,” *Journal of Information Security and Applications*, vol. 54, pp. 102553, 2020.
- [9] W. El-Shafai, M. A. Fouda, E. S. M. El-Rabaie and N. A. El-Salam, “A comprehensive taxonomy on multimedia video forgery detection techniques: Challenges and novel trends,” *Multimedia Tools and Applications*, vol. 82, pp. 1–67, 2023. <https://doi.org/10.1007/s11042-023-15609-1>
- [10] Y. A. Hassan and A. M. S. Rahmah, “The most common characteristics of fragile video watermarking,” *ARO-The Scientific Journal of Koya University*, vol. 11, no. 1, pp. 99–104, 2023.
- [11] K. Juneja and S. Bansal, “Frame selective and dynamic pattern based model for effective and secure video watermarking,” *International Journal of Computing*, vol. 18, no. 2, pp. 207–219, 2019.
- [12] F. Thakkar and V. K. Srivastava, “An adaptive, secure and imperceptible image watermarking using swarm intelligence, Arnold transform, SVD and DWT,” *Multimedia Tools and Applications*, vol. 80, pp. 12275–12292, 2021.
- [13] J. M. Vilarly O., L. Barba J. and C. O. Torres M., “Image encryption and decryption systems using the jigsaw transform and the iterative finite field cosine transform,” *Photonics*, vol. 6, no. 4, pp. 121, 2019.
- [14] H. F. Ng and C. Y. Chin, “Effective scene change detection in complex environments,” *International Journal of Computational Vision and Robotics*, vol. 9, no. 3, pp. 310–328, 2019.
- [15] S. N. Gowda, M. Rohrbach and L. Sevilla-Lara, “Smart frame selection for action recognition,” in *Proc. of the AAAI Conf. on Artificial Intelligence*, California, USA, vol. 35, no. 2, pp. 1451–1459, 2021.
- [16] A. Mahboob, M. Asif, R. M. Zulqarnain, I. Siddique, H. Ahmad *et al.*, “An innovative technique for constructing highly non-linear components of block cipher for data security against cyber attacks,” *Computer Systems Science & Engineering*, vol. 47, no. 2, pp. 2547–2562, 2023. <https://doi.org/10.32604/csse.2023.040855>
- [17] A. Gutub and E. Almeahadi, “Advancing partial verification of watermarking for Arabic text via utilization of innovative counting-based secret sharing,” *Arabian Journal for Science and Engineering*, vol. 48, no. 8, pp. 9963–9989, 2023.
- [18] K. R. Raval and M. M. Goyani, “A survey on event detection based video summarization for cricket,” *Multimedia Tools and Applications*, vol. 81, no. 20, pp. 29253–29281, 2022.

- [19] B. Rashmi and H. Nagendraswamy, "Video shot boundary detection using block based cumulative approach," *Multimedia Tools and Applications*, vol. 80, pp. 641–664, 2021.
- [20] F. Harun and N. H. Ali, "Video structure extraction using shot boundary detection for authentication detection," in *AIP Conf. Proc.*, vol. 2484, no. 1, Kuala Terengganu, Malaysia, pp. 060021, 2023.
- [21] R. Zhou, N. Konstantin, S. Mariya, R. Natalya and X. Zhang, "Motion algorithm for unmanned aerial vehicle landing on a car," in *2020 27th Saint Petersburg Int. Conf. on Integrated Navigation Systems (ICINS)*, St. Petersburg, Russia, IEEE, pp. 1–3, 2020.
- [22] S. K. Siri, S. P. Kumar and M. V. Latte, "Threshold-based new segmentation model to separate the liver from CT scan images," *IETE Journal of Research*, vol. 68, no. 6, pp. 4468–4475, 2022.
- [23] J. Song, W. Jiao, K. Lankowicz, Z. Cai and H. Bi, "A two-stage adaptive thresholding segmentation for noisy low-contrast images," *Ecological Informatics*, vol. 69, pp. 101632, 2022.
- [24] A. A. Gharahbagh, V. Hajhashemi, M. C. Ferreira, J. J. Machado and J. M. R. Tavares, "Best frame selection to enhance training step efficiency in video-based human action recognition," *Applied Sciences*, vol. 12, no. 4, pp. 1830, 2022.
- [25] M. R. Jiang, X. F. Feng, C. P. Wang, Z. Y. Li and H. Zhang, "Robust and blind watermarking algorithm for chaotic system encryption based on discrete cosine transform-singular value decomposition," *Journal of Electronic Imaging*, vol. 32, no. 1, pp. 013020, 2023.
- [26] M. R. Jiang, X. F. Feng, C. P. Wang and H. Zhang, "Robust color image watermarking algorithm based on synchronization correction with multi-layer perceptron and Cauchy distribution model," *Applied Soft Computing*, vol. 140, pp. 110271, 2023.
- [27] P. Garg and R. R. Kishore, "Performance comparison of various watermarking techniques," *Multimedia Tools and Applications*, vol. 79, pp. 25921–25967, 2020.
- [28] C. Sharma, A. Bagga, B. K. Singh and M. Shabaz, "A novel optimized graph-based transform watermarking technique to address security issues in real-time application," *Mathematical Problems in Engineering*, vol. 2021, pp. 1–27, 2021.
- [29] D. Fan, X. Zhang, W. Kang, H. Zhao and Y. Lv, "Video watermarking algorithm based on NSCT, pseudo 3D-DCT and NMF," *Sensors*, vol. 22, no. 13, pp. 4752, 2022.
- [30] Q. Chang, L. Huang, S. Liu, H. Liu, T. Yang *et al.*, "Blind robust video watermarking based on adaptive region selection and channel reference," in *Proc. of the 30th ACM Int. Conf. on Multimedia*, Melbourne, VIC, Australia, pp. 2344–2350, 2022.