Computers, Materials & Continua
DOI:10.32604/cmc.2022.019866
Article

Tech Science Press

A Robust Video Watermarking Scheme with Squirrel Search Algorithm

Aman Bhaskar[1], Chirag Sharma[1], Khalid Mohiuddin[2], Aman Singh[1,*], Osman A. Nasr[2] and Mamdooh Alwetaishi[3]

[1]Computer Science and Engineering, Lovely Professional University, Punjab, 144411, India
[2]Department of Management Information Systems, King Khalid University, Guraiger, Abha, 62529, Saudi Arabia
[3]Department of Civil Engineering, College of Engineering, Taif University, Taif, 21944, Saudi Arabia
*Corresponding Author: Aman Singh. Email: amansingh.x@gmail.com
Received: 29 April 2021; Accepted: 19 July 2021

Abstract: Advancement in multimedia technology has resulted in protection against distortion, modification, and piracy. For implementing such protection, we have an existing technique called watermarking but obtaining desired distortion level with sufficient robustness is a challenging task for watermarking in multimedia applications. In the paper, we proposed a smart technique for video watermarking associating meta-heuristic algorithms along with an embedding method to gain an optimized efficiency. The main aim of the optimization algorithm is to obtain solutions with maximum robustness, and which should not exceed the set threshold of quality. To represent the accuracy of the proposed scheme, we employ a popular video watermarking technique (DCT domain) having frame selection and embedding method for watermarking. A squirrel search algorithm is chosen as a meta-heuristic algorithm that utilizes the stated fitness function. The results indicate that quality constraint is fulfilled, and the proposed technique gives improved robustness against different attacks with several quality thresholds. The proposed technique could be practically implemented in several multimedia applications such as the films industry, medical imagery, OOT platforms, etc.

Keywords: Meta-heuristic algorithm; constrain optimization problem; fitness fiction; frame selection; squirrel search algorithm

1 Introduction

As the growth of internet prevailing multimedia usage is easy to be shared and accessed, editing tools and software are so readily available that their utilization is inevitable up to such an extent that no professional knowledge is required to manipulate digital multimedia. To protect the digital data in a shared medium from illegal manipulation, there is a need for a unique and popular technique known as digital watermarking. Digital watermarking is a process to embed an imperceptible watermark into a host/cover digital content in such a way that the spread of watermark bits is scattered in the host evenly. Also, it must be infeasible to alter the watermark bits and could be extracted through some computation. The main applications of digital watermarking are copyright protection, ownership identification [1]. Some of the other applications include broadcast monitoring, modification detection, authentication, and source tracking, etc reflected in Fig. 1a. This paper discusses video watermarking, which proposes authentication to the video by embedding watermark data into the cover video. There are many challenges in video watermarking. One of the major challenges is a design scheme that provides robustness to the watermarking and loss of quality is a challenge that is reflected in Fig. 1b. Whereas robustness is the tendency of the watermark to not getting destroyed during an attack makes it fault-tolerant. The significance of one is often highlighted over the other, as per the need for required quality in the application [2]. There is a tradeoff between imperceptibility (quality) and robustness (fault-tolerance) i.e., if one is required the second must be compromised. Imperceptibility significantly affects the visual quality of the video.

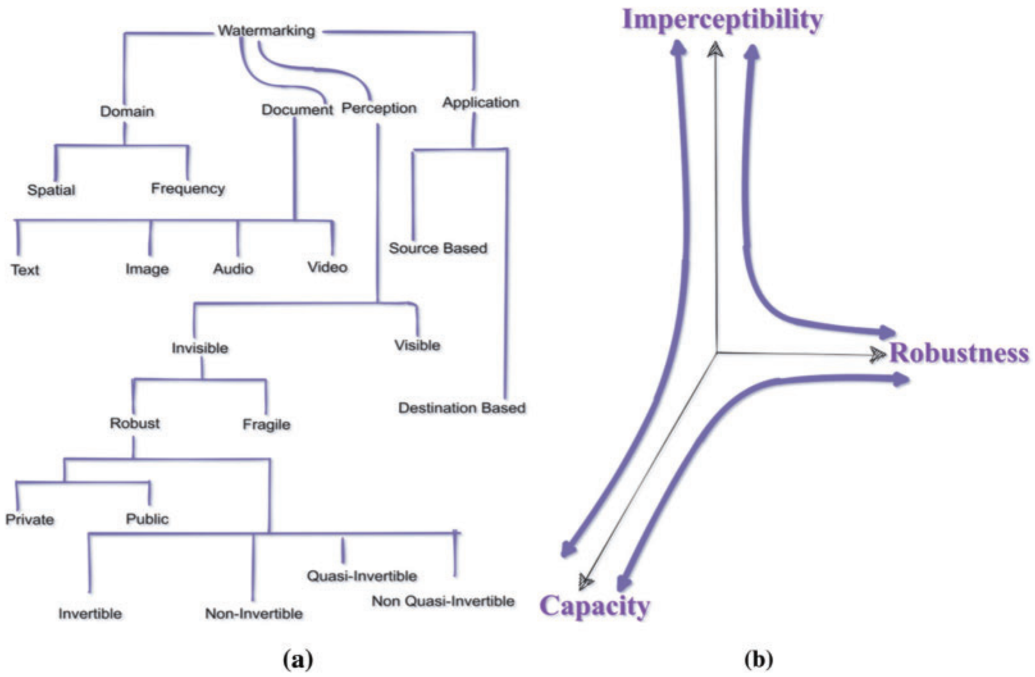


Figure 1: (a) Watermarking techniques (b) Relationship among watermark capacity, imperceptibility, and robustness

Meanwhile, the minimal modification would result in reduced robustness of the algorithm [3]. To have the balance between the three concepts, we employ a meta-heuristic optimization algorithm (SSA (Squirrel Search Algorithm)). These computed objective functions are then utilized to rank the solutions and find the optimal results. This will generate robust watermarking schemes with distortions below the sensitivity threshold [4].

Knowledge embedding in multimedia data is a field that has lately gotten a lot of attention [5]. The Major contribution of the paper is integrated into the following points.

(a) We have proposed the frame selection technique to chose optimal frames.
(b) We have proposed a novel technique for embedding a watermark that is based on DCT domain block-based conversion along with an optimization algorithm.
(c) We have proposed the calculation of efficiency and performance of the presented scheme against several image processing attacks.

Kavitha Soppari et al. have done more research studies in implementing robust watermarking through clustering techniques as most of the transform-domain watermarking approaches were build on Discrete Cosine Transform (DCT). This is because it is robust to JPEG lossy compression. Therefore, the author proposes an efficient digital watermarking mechanism based on an optimized clustering method. Mohammad Moosazadeh et al. proposed an embedding method that depends upon the relationships in between DCT coefficients. The watermarking parameters are optimally selected using a rarely used optimization algorithm of watermark embedding known as Teaching-learning-Based Optimization (TLBO). Vipul and Roohie stated the considerable time complexity of meta-heuristic search algorithms is not applicable for time-sensitive applications. For this, they utilize the DCT domain for embedding watermark followed by nature-based optimization algorithm ant colony optimization to search optimal parameters. Assem Mahmoud Abdelhakim et al. provided a smart scheme for watermarking by incorporating a meta-heuristic search technique known as Artificial Bee Colony along with an embedding function to achieve an optimized functionality. The quality of watermarking is insured by constraint optimization.

Squirrel Search Algorithm

There is no single nature-based optimization strategy that can solve all optimization problems in the best possible way. This means that an optimization algorithm is good at solving a certain set of problems but not so well at solving others. As a result, a new unconstrained numerical optimization algorithm called Squirrel Search is proposed. The foraging of flying squirrels can be divided into two scenarios [6].

(i) Assumptions in SSA

In a deciduous forest, there are n flying squirrels, and each squirrel is thought to be on one tree. Each flying squirrel uses a complex foraging behavior to hunt for food and make the best use of the available resources. There are only three types of trees in the forest: I Regular tree (ii) Oaktree (acorns) (abundant food source) hickory tree (iii) (alternate food source).

(ii) Implementation of SSA

The game begins with a random location of a flying squirrel (FS). In a d-dimensional search space, the location of FS is represented by a vector. The random initialization of ‘n’ number of FS in the forest and each vector specifies the direction of the ith FS. A matrix FS can be used to represent the location of all FS and shown by below equation

$$\text{‘n’ FS} = \begin{array}{cccccc} & & \text{‘d’ location vector} & & & \\ FS_{1,1} & FS_{1,2} & FS_{1,3} & \ldots. & \ldots. & FS_{1,d} \\ FS_{2,1} & FS_{2,2} & FS_{2,3} & \ldots. & \ldots. & FS_{2,d} \\ FS_{3,1} & FS_{3,2} & FS_{3,3} & \ldots. & \ldots. & FS_{3,d} \\ \| & \| & \| & \| & \| & \| \\ FS_{n,1} & FS_{n,2} & FS_{n,3} & \ldots. & \ldots. & FS_{n,d} \end{array} \tag{1}$$

The j^{th} dimension of i^{th} squirrel is defined by $FS_{i,j}$ equation

This provides a uniform allocation for allocating the initial position of each FS in the Forest.

$$FS_i = FSL + U(0,\ 1) \times (FSU - FSL) \tag{2}$$

FSL: Lower Bound of FS_i

FSU: Upper Bound of FS_i

U (0, 1): random no. between [0,1]

(iii) Fitness Evaluation

In the user's fitness function, the value of the decision variable (solution vector) and its related value is stored in 'f'. Some of the constants considered are dg:random gliding distance, R1: random no. [0,1], t: current iteration tree Gc: Constant to achieve balance b/w exploration and exploitation, dg: (hg/tan(Q)) hg: loss in height after gliding, Q: arctan(D/L), D: drag force produced, L: Sum of lift at equilibrium, L = 1/(2ÞCL V2S), Þ: 1.204 kgm^3 (air density), CL = Lift Coefficient in the range of [0.675, 1.5], V = 5.25 m/sec (speed of flying squirrel (avg)), S = 154 cm^2 (Surface area), D = 1/(2ÞCDV2S), CD: Drag Coefficient (0.60) [6] as mentioned in Eq. (3).

$$f = \begin{array}{llllll} f(FS_{1,1} & FS_{1,2} & FS_{1,3} & \ldots. & \ldots. & FS_{1,d}) \\ f(FS_{2,1} & FS_{2,2} & FS_{2,3} & \ldots. & \ldots. & FS_{2,d}) \\ f(FS_{3,1} & FS_{3,2} & FS_{3,3} & \ldots. & \ldots. & FS_{3,d}) \\ \| & \| & \| & \| & \| & \| \\ f(FS_{n,1} & FS_{n,2} & FS_{n,3} & \ldots. & \ldots. & FS_{n,d}) \end{array} \tag{3}$$

Case 1: FS on Acorn tree (FS_{at}) moving towards hickory tree (FS_{ht}). Location is updated as given in the following given Eq. (4).

$$FS_{at^{t+1}} = \begin{cases} FS_{at^t} + dg \times Gc \times (FS_{ht^t} - FS_{at^t}) & \text{if } R1 > PQ \\ \text{Random location} & \text{otherwise} \end{cases} \tag{4}$$

where R1 is a random number between [0, 1]

Case 2: FS on a normal tree (FS_{nt}) moving towards acorn tree (FS_{at}) location is updated as given in following given Eq. (5).

$$FS_{nt^{t+1}} = \begin{cases} FS_{nt^t} + dg \times Gc \times (FS_{at^t} - FS_{nt^t}) & \text{if } R2 > PQ \\ \text{Random location} & \text{otherwise} \end{cases} \tag{5}$$

where R2 is a random number between [0, 1]

Case 3: FS on a normal tree (FS_{nt}) and already passed acorn tree and now moving towards hickory tree (FS_{ht}), Location is updated asgiven in following given Eq. (6).

$$FS_{nt^{t+1}} = \begin{cases} FS_{nt^t} + dg \times Gc \times (FS_{ht^t} - FS_{nt^t}) & \text{if } R1 > PQ \\ \text{Random location} & \text{otherwise} \end{cases} \tag{6}$$

where R3 is a random number between [0, 1]

(iv) Seasonal Monitoring Condition

Since foraging in the winter was more expensive and dangerous, FS became less successful than in the autumn. Incorporating such behavior results in a more practical strategy that avoids being stuck in a locally optimal solution. Per iteration, calculate the seasonal constant (Sc) Eq. (7) [6].

$$Sc^{t} = \sqrt{\left(\sum_{k=1}^{d} (Fs_{at,k} - FS_{nt,k})^{2}\right)} \tag{7}$$

where t = 1, 2, 3 iterations

Check $Sct < Smin$ where Smin is the minimum value of Seasonal Constant Computed as Eq. (8) [6].

$$S_{min} = \frac{10E^{-6}}{((365)^{t}/(t_{m}2.5))} \tag{8}$$

$t <$ current iteration $tm <$ max iteration

If the condition is true, then randomly reallocating FS did not explore the forest by following Eq. (9) [6].

$$Fs_{nt}^{new} = FS_{L} + Levy(n) \times (FS_{U} - FS_{L}) \tag{9}$$

where Levy is random walk-through Levy distribution.

The discrete cosine transform is then applied to the blocks of a grayscale image used as the carrier image, and the DCT coefficients are quantized using the quantization table represented in the JPEG norms, which is used where an image requires a very high compression rate [7].

2 Proposed Fitness Function

Optimization problem to search for the optimum embedding strength parameter (k and p) for embedding watermark problem formulated in Eq. (10).

$$\begin{array}{ll} \text{MAX: } R([k,p]) & \{\text{Robustness}\} \\ \text{Subjectto: } Q([k,p]) >= Q^{th} & \{\text{Quality}\} \end{array} \tag{10}$$

It assigns a score to each solution based on how effective it is at solving the problem. Sorting, declaring, and selecting at random: The following three minimum sorted fitness values are known to have (normal food source), and they are progressing towards (optimal food source). The rest of the tree is regular. The rest FS fitness value is used to make a random pick, which means aiming towards the best food source. The proposed fitness function Algorithm 1 in pseudo-code is followed by a flow diagram given in Fig. 2.

Algorithm 1: Pseudocode for proposed fitness function

Input: Embedding-parameters ($\boldsymbol{\alpha}$,$\boldsymbol{\beta}$), Attacks: Gaussian Noise, Sharpening, Median filtering attack, Rotation attack

Output: Ranked Solution Population (**NS**)

```
Step 1    Initialize $Q^{th}$ := avg (PSNR) and $R^{t}$ := avg (INC);
Step 2    For all solutions in NS do:
            if $\beta$ < Qth then Add solution to the quality group (Q)
            else Add solution to the robustness group (R)
Step 3    end for
```

(Continued)

```
Step 4     Sort Q group and R group in descending order based on min value of (Qth-PSNR)
            and ($R^t$-INC) respectively.
Step 5     Merge Q group to R group and call resulted population as recombined population
            (RP).
Step 6     Initialize Rank list with length of RP-current index (i) of solution in RP.
Step 7     For all solution s in RP do:
            if s_i$^{th}$ $\epsilon$ R and s_j$^{th}$ $\epsilon$ Q then Rank(s_i$^{th}$) > Rank(s_j$^{th}$) end if
            else if s_i$^{th}$ $\epsilon$ R and s_j$^{th}$ $\epsilon$ R then
                  if s_i$^{th}$ (INC) > s_j$^{th}$ (INC) then Rank(s_i$^{th}$) > Rank(s_j$^{th}$)
                  else Rank(s_j$^{th}$) > Rank(s_i$^{th}$)
                  end if
                  end else
                  else
            if s_i$^{th}$ (PSNR) > s_j$^{th}$ (PSNR) then Rank(s_i$^{th}$) > Rank(s_j$^{th}$)
            else Rank(s_j$^{th}$) > Rank(s_i$^{th}$)
            end if
Step 8     end else
Step 9     end for
Step 10     Return RP
```

The robustness of the proposed method against various kinds of attacks such as median filtering, Gaussian noise [8], Sharpening, Rotation attack. According to the watermark experiment findings, the suggested scheme performs satisfactorily in terms of image distortion and embedded watermark retrieval, even after the original image has been damaged by several assaults [9]. This approach can effectively avoid rotating attacks, and only half of the points need to be scanned [10]. Spatial scanning, duplication, cropping, scaling, translation, compression, and rotating are examples of attacks that may be carried out deliberately or accidentally [11]. We checked the watermarking algorithm's resiliency against a series of four different attacks for various videos and enhanced the decoding BER (bit error rate) [12]. When constructing an appropriate architecture, we have a list of typical attacks or threats to remember. PSNR (peak signal-to-noise ratio), MSE (mean squared error), and NCC (normalized cross-correlation) are a few of the output metrics described briefly [13].

3 Watermark Embedding

Divide the host image I, of the size P × Q, into 8 × 8 blocks. Transform each image block I p, q and its adjacent block I p+1, q into DCT coefficients as given in Eq. (11), and also we have given Algorithm 2 to represent in the form of pseudo-code.

$$I^{DCT}_{p,q} = DCT(I_{p,q});\ I^{DCT}_{p+1,q} = DCT(I_{p+1,q}) \tag{11}$$

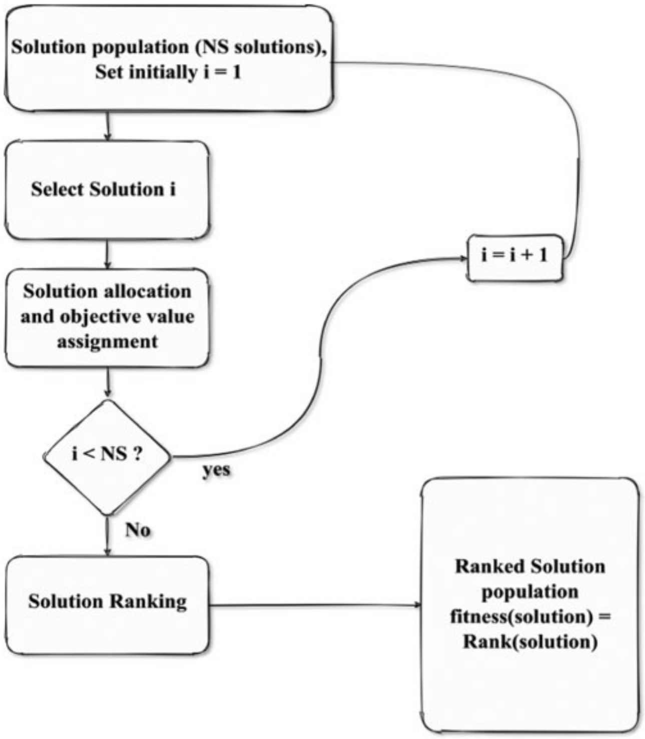

Solution population (NS solutions), Set initially i = 1
Select Solution i
Solution allocation and objective value assignment
i < NS ?
yes
No
i = i + 1
Solution Ranking
Ranked Solution population fitness(solution) = Rank(solution)

Figure 2: Fitness function algorithm flow diagram [14]

Adjust the value of two coefficients according to the watermark bit $W_{p,q}$, which is the bit location at the pth row & qth column of the watermark image. As shown in Eq. (12) below.

$$\begin{bmatrix} I^{DCT}_{p,q}(x,y) \\ I^{DCT}_{p+1,q}(x,y) \end{bmatrix} = \begin{cases} \begin{bmatrix} I^{DCT}_{p+1,q}(x,y) \\ I^{DCT}_{p,q}(x,y) \end{bmatrix}, \quad if\,(W_{p,n} = 1 \;\; and \\ (I^{DCT}_{p,q}(x,y) - \; I^{DCT}_{p+1,q}(x,y) < 0) or \\ \qquad (W_{p,n} = 0 \;\; and \\ (I^{DCT}_{p,q}(x,y) - I^{DCT}_{p+1,q}(x,y) > 0)) \\ \begin{bmatrix} I^{DCT}_{p,q}(x,y) \\ I^{DCT}_{p+1,q}(x,y) \end{bmatrix}, \quad Otherwise. \end{cases} \tag{12}$$

This scaling parameter is calculated by associating the imperceptibility quality of each watermark with the imperceptibility quality of the host image to mitigate the tradeoff between robustness and imperceptibility [15]. $\alpha \leftarrow$ Ensure absolute difference between modified coefficients large enough to provide the required robustness. $\beta \leftarrow$ Ensures how much proportion of the value $\boldsymbol{\alpha}$ can be used in a coefficient adjustment. As shown in Eq. (13) below.

$$\begin{bmatrix} I^{DCT}_{p,q}(x,y) \\ I^{DCT}_{p+1,q}(x,y) \end{bmatrix} = \begin{cases} \begin{bmatrix} I^{DCT}_{p+1,q}(x,y) + \alpha.\beta \\ I^{DCT}_{p,q}(x,y) - (1-\beta) \end{bmatrix}, \quad if(W_{p,n} = 1 \ and \\ |I^{DCT}_{p,q}(x,y) - I^{DCT}_{p+1,q}(x,y) < \beta| or \\ \qquad (W_{p,n} = 0 \ and \\ |I^{DCT}_{p,q}(x,y) - I^{DCT}_{p+1,q}(x,y) > \beta)| \\ \begin{bmatrix} I^{DCT}_{p,q}(x,y) \\ I^{DCT}_{p+1,q}(x,y) \end{bmatrix}, \quad Otherwise. \end{cases} \tag{13}$$

Algorithm 2: Pseudocode of Embedding Watermark

Input: Host video frame (**I**) and watermark image (**I_w**).

Output: Embedded watermark frame, PSNR, and INC values on different Robustness and quality parameters.

Input: Host video frame (**I**) and watermark image (**I_w**).

Output: Embedded watermark frame, PSNR, and INC values on different Robustness and quality parameters.

```
Step 1    Divide host watermark image I of size P × Q, into 8 × 8 blocks.
Step 2    Transform each image block I_p, q, and its adjacent block I_p+1, q into DCT
          coefficients, where p, q are rows and columns indices of a frame-block.
Step 3    Initialize $\alpha$: = 127 to 255, $\beta$: = 0.9 to 0.1 and MSE, INC embedding-parameters
          equal to zero. Also iterations counters p, q, a, b equal to 1. New frame I_e equals to
          zero matrix.
Step 4    while p < P and q < Q (till the end of frame)
              if I_w (p, q):=1 then
                  if I (p, q) – I (p+1, q) greater than $\alpha$ then
                      I_e (p, q) := I (p, q) + ($\alpha$ × $\beta$)
                      I_e (p+1, q) := I (p+1, q) – ($\alpha$ × (1-$\beta$))
                      MSE := MSE + CALL mean_square_error (I (p, q), I_e (p, q))
                      INC := INC + CALL normalized_correlation (I (p, q), I_e (p, q))
                      End if
                  Else
                      I_e (p, q) := I (p, q) - ($\alpha$ × $\beta$)
                      I_e (p+1, q) := I (p+1, q) + ($\alpha$ × (1-$\beta$))
                      MSE := MSE + CALL mean_square_error (I (p, q), I_e (p, q))
                      INC := INC + CALL normalized_correlation (I (p, q), I_e (p, q))
                      End else
                  End
              Else
                  I_e (p, q) := I (p, q)
                  I_e (p+1, q) := I (p+1, q)
                  End else
              End if
Step 5    End while
Step 6    Apply inverse DCT to obtain the watermarked image block on I_e
Step 7    CALL peak_signal_to_noise_ratio(MSE) to calculate PSNR value.
```

(Continued)

Step 8	Store $\boldsymbol{\alpha}, \boldsymbol{\beta}$ with respected PSNR and INC value to form a population.
Step 9	End

4 Intelligence Evolutionary Algorithm

With the rapid development of modern technology, the complex evolutionary problems are too difficult to be answered by conventional optimization algorithms. This challenge can be dealt with by incorporating an evolutionary intelligence algorithm. The implementation is done by embedding a watermark in the DCT domain, accompanied by a nature-based optimization watermark [16]. This is due to the resistance to lossy JPEG compression [17]. Attacks like frame lowering, averaging, cropping, and median filtering, as well as unintended attacks like noise and compression, can breach copyright information in a video [18]. The evolutionary intelligence algorithms can be divided into 3 categories as given below.

4.1 Evolutionary Heuristic Algorithms

The Evolutionary heuristic algorithms are inspired by biological evolution like reproduction, mutation, recombination, and selection. It plays a vital role in estimating accurate solutions or optimal solutions from a population of solutions. Moreover, it solves difficult problems by using a fitness function which is using the evolutionary algorithm as a tool for problem-solving also known as Evolutionary Computation (EC). Genetic Programming (GP): It is an extension of GA used for testing and selecting the best choices among the set of results [19]. And by integrating the MPEG specifications with the classic block-adaptive-video watermarking systems, can get better results [20].

4.2 Physics Heuristic Algorithms

These are originated by physical phenomena, which are then conceptually applied to obtain optimal solutions. After melting the energy is exceedingly high and the temperature slowly drops to a stable solid-state having minimum energy (optimal solution). Similarly, Gravitational, The magnetic optimization.

4.3 Nature Heuristic Algorithms

These are inspired by biological practices and phenomena in nature. There are several algorithms in this class such as Ant Colony Optimization that are based on naturally excrete pheromones in their path which act as an indicator better path. Another such algorithm is the particle swarm optimization algorithm, which is based on bird's practices in seeking food. Similarly, Artificial Bee Colony is based on the adaptation of the foraging practices of the honeybee. The bee's population is formed through 3 types of bees based on their work. Mainly Employee bee, onlooker bee and scout bee.

5 Frame Selection

We looked at a variety of image matching methods before settling on the histogram process. Histogram-based methods are ideal for determining the total changes in an image. An image's color histogram can be calculated by dividing a color space, such as RGB. The distinction between two pictures dependent on color histograms H1 and H2 can be calculated as in Eq. (14). Fig. 3 depicts the frame selection flow diagram to more clearly explain the algorithm. Algorithm 3 is also stated for the reader's reference.

$$d(l_1, l_2) = \sum_{k=1}^{n} |H_{1K}(x, y) - H_{2K}(x, y)| \tag{14}$$

Algorithm 3: Pseudo code for Frame Selection Algorithm

Input: Avideo having N number of frames. Initial value of integrator i and j equal to zero and v_diff vector equal to zero.

Output: Select frame(s) concerning a threshold, calculated through Scene Change detection using histogram.

```
Step 1 Start
Step 2 Initialize variable Curframe_i := First Frame (Video) and Curframe_j := Second Frame
       (Video).
Step 3 While Curframe_i not equal to the last frame do:
        Calculate P := Histogram(gray_( Curframe_i)) - Histogram(gray_( Curframe_j))
        Calculate Q := Maximum(Histogram(gray_( Curframe_i)), Histogram(gray_( Curframe_j)))
        Calculate Vector S (i) := Avg(Sum(P/Q))
Step 4 End While
Step 5 For i := 1 to (N−1) do:
        Calculate vector diff_v (i) := Abs(S(i+1) – S(i))
Step 6 End for
Step 7 Initialize SC:= 0 (Gives No. of Scene Change), K := 1 And vector SCV := 0 (Scene Change
       frame Value)
Step 8 For I := 1 to (N−1) do:
       $V^{th}$ := K ∗ (sum(diff_v)/(N−1) + ((max(diff_v)+min(diff_v))/2)
     If diff_v (i) > $V^{th}$:
         Increment SC := SC + 1
         Set Value K := SC
         SCV (SC) := i
     End if
Step 9 End for
Step 10 Return SC and SCV.
Step 11 End.
```

6 Quality Measures

It significantly affects the visual quality of the video. So, to ensure the quality of the video watermarking, we need to calculate a measure that can indicate the level of imperceptibility. It is calculated through a mathematical ratio known as Peak Signal to Noise ratio (PSNR) as per Eqs. (15)–(16). The term robustness refers to the embedded watermark's ability to withstand image manipulation and geometric attacks.

$$\mathrm{PSNR_k} = 10\log_{10}\frac{255^2}{\mathrm{MSE}}(\mathrm{db}) \tag{15}$$

$$\mathrm{MSE_k} = \frac{1}{P \times Q}\sum_{i=1}^{P}\sum_{j=1}^{Q}[f_k(i,j) - f_{k_w}(i,j)]^2 \tag{16}$$

$f_k \rightarrow$ original video frame $f_{kw} \rightarrow$ Watermarked video frame

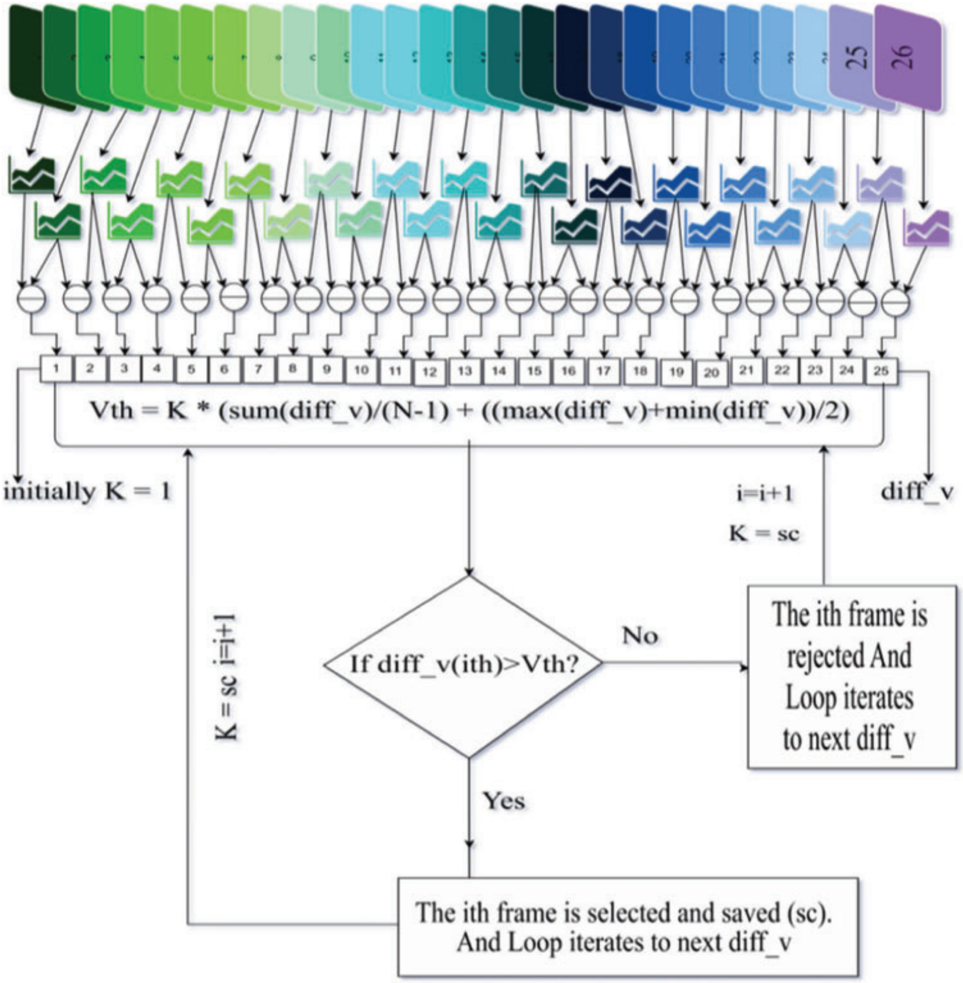

25
26
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
Vth = K * (sum(diff_v)/(N-1) + ((max(diff_v)+min(diff_v))/2)
initially K = 1
i=i+1
K = sc
diff_v
K = sc i=i+1
If diff_v(ith)>Vth?
No
The ith frame is rejected And Loop iterates to next diff_v
Yes
The ith frame is selected and saved (sc). And Loop iterates to next diff_v

Figure 3: Frame selection algorithm flow

7 Robustness Measures

Robustness is vital for granting capability to withstand different image processing and geometric attacks and the quality of digital content must not be degraded. Related attacks include smoothing, sharpening, compression, de-noising, filtering, scaling, cropping, and rotation. The robustness of watermarking algorithms is usually measured using normalized correlation (NC) and bit error rate (BER). NC is used to calculate the degree of resemblance between the derived and initial watermarks. It can be calculated through a mathematical measure known as Normalize Correlation that ranges between 0 to 1 [21]. A higher value of NC stronger the anti-attack ability, as given in Eq. (17).

$$I_{NC} = \sum_{i=1}^{P} \sum_{j=1}^{Q} \frac{W(i,j) \times W'(i,j)}{\sum_{i=1}^{P} \sum_{j=1}^{Q} W(i,j)^2} \tag{17}$$

8 Experimental Results

The results were evaluated in MATLAB 2014a using an i5 processor. The frame selection time and embedding time is dependent on the processing power of the processor and the size of the RAM. The observed results are based on the watermark embedding time, frame selection time, and optimization algorithm time. For testing the algorithm, a total of 5 CIF (Common Interchange Format) videos are utilized, and the frame selection technique used is based on scene change in the video. Fig. 4 shows the test videos list with the chosen frames from each video. The higher value the PSNR value indicates the proposed technique will have a minimum loss in quality of output. Similarly, for greater the value of NC, more resistance against attacks. The testing is started with converting videos in mp4 format followed by frame extraction. The frames extracted are in the PNG format to withstand quality loss. The selected frames are then embedded through embedding algorithms with hyperparameters values obtained through optimization algorithm. Finally, we bring the efficiency of the proposed work by applying several image processing attacks and then calculating NC values [22]. Tabs. 1–3 represents number of frames selected, quality metrics values and a comparison analysis of existing frame selection techniques.

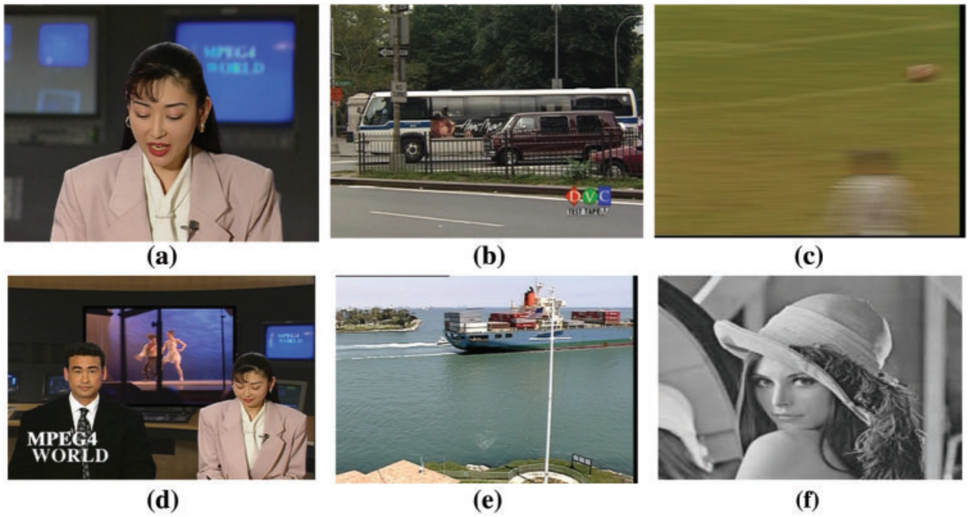


Figure 4: Selected frames from videos: (a) Akiyo #251(252), (b) Bus #126(127), (c) Football #74(218), (d) News #77(252), (e) Container #251(252), (f) Watermark

Table 1: Comparison of videos in terms of frame selection

Video name	No. of frames selected
Akiyo	1
Bus	1
Football	2
News	2
Container	1

Table 2: Results after embedding of watermark 1 on selected frames

Videos	K	P	PSNR	NC	BER
Akiyo	143	0.7	60.556	0.8214	0.01651353
Bus	214	0.7	68.676	0.9866	0.01456119
Football	159	0.6	59.655	0.8363	0.01676311
News	211	0.8	71.061	0.9477	0.01407236
Container	240	0.4	59.793	0.9392	0.01672425

Table 3: Comparative analysis of the robustness of our scheme with other algorithms

Author	PSNR (db)	Techniques
Dengpan Ye	44	Block classification
Hartung	35	MPEG 2 bitstream
Deepayan	40	VA-based watermarking
Pei	42.31	Motion Vectors
Venugopala	38.96	Scene Change Detection
Mangler	48.82	SIFT
Osama	44.83	SVD DWT-2 level
Doosep	61.04	DCT B-frames MPEG
Bhaskar	71.06	Block Based SSA

Experimental Tests for Time Complexity: Tab. 4 describes the processing time (in seconds) associated with the frame selection and embedding algorithm applied to the test videos. The duration is entirely based on hardware processing power. Also, it depends upon the number of selected frames and the total number of frames in the video. For football and news number of frames selected is 2 out of 218 and 252 frames respectively. The average frame selection time is nearly the same. Moreover, the total number of frames in the video also lead to an increase in the execution time due to which we can see the football video having 2 frames out of 252 frames have slightly higher embedding time (3.1747) to the Akiyo video having 252 frames and 1 frame selected (3.164). Figs. 5, 6 signify plots drawn against quality metrics and frame selection time, embedding time Figs. 7–10 signify plots drawn against gaussian noise, sharpening attack, rotation, median filtering attacks.

Table 4: Results of execution time using watermark

Video name	Frames selection time (sec)	Embedding time (sec)
Akiyo	1.873	3.164
Bus	0.942	3.132
Football	1.536	3.1747
News	1.897	3.1205
Container	1.888	3.116

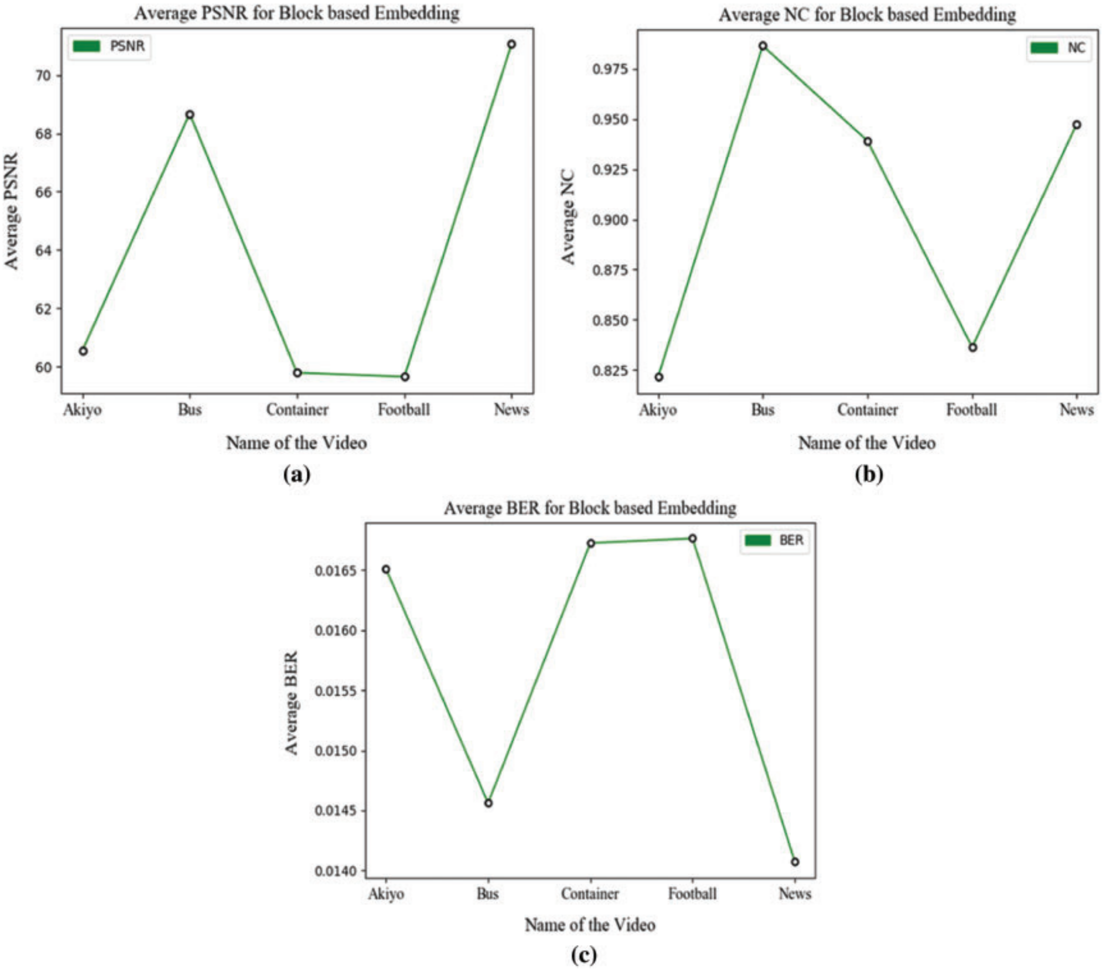


Figure 5: Plot of PSNR, NC, K, BER w.r.t the videos taken in the proposed work for watermark against no attack. (a) Average PSNR, (b) Average NC, (c) Average BER

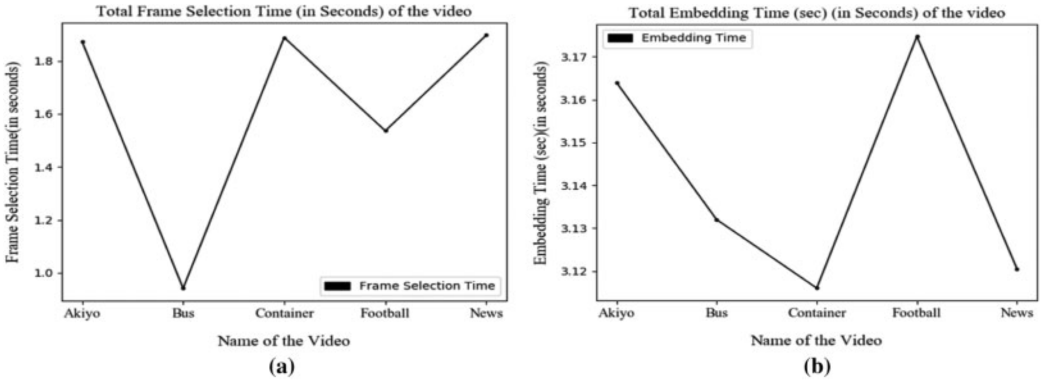
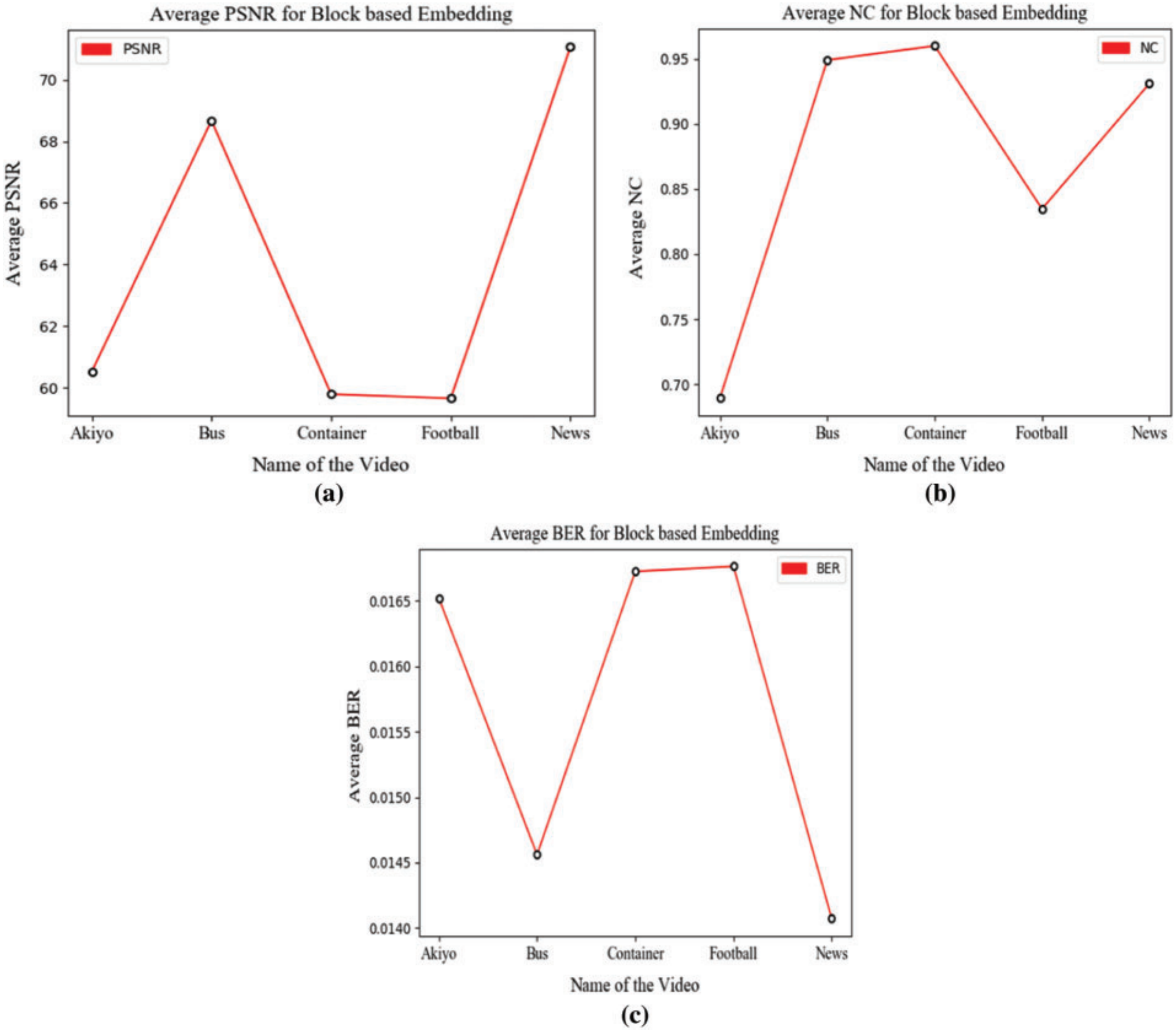


Figure 6: Plot of total frame selection time and total embedding time (in seconds) for 5 videos. (a) Total frame selection time (in seconds). (b) Total embedding time (in seconds)


Average PSNR for Block based Embedding

(a)

Average NC for Block based Embedding

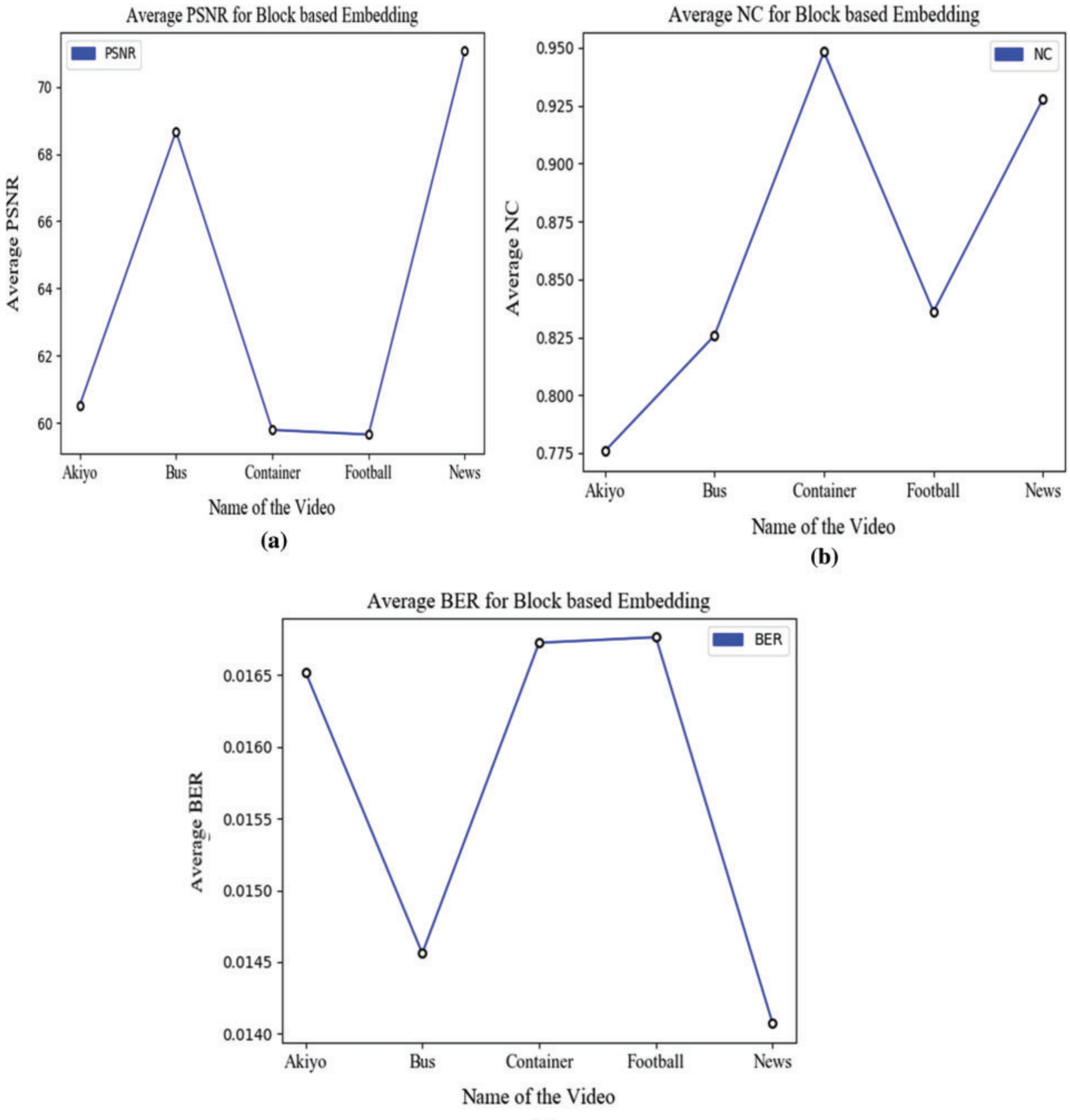
(b)

Average BER for Block based Embedding

(c)

Figure 7: Plot of PSNR, NC, P, BER w.r.t the videos taken in the proposed work for watermark against Gaussian attack. (a) Average PSNR, (b) Average NC, (c) Average BER

The fault tolerance of the proposed technique is tested against several attack scenarios like Gaussian Noise in Tab. 5, Sharpening in Tab. 6, Median filtering attack in Tab. 7, and Rotation attack in Tab. 8. To account for the lost number of attacks on every watermarked frame to measure the quality of the proposed technique [22]. The second attack on the selected frames will stretch the pixel distribution in the image and highlights the fine image details [22].



(a)

(b)

(c)

Figure 8: Plot of PSNR, NC, K, BER w.r.t the videos taken in the proposed work for watermark against Sharpening attacked. (a) Average PSNR, (b) Average NC, (c) Average BER

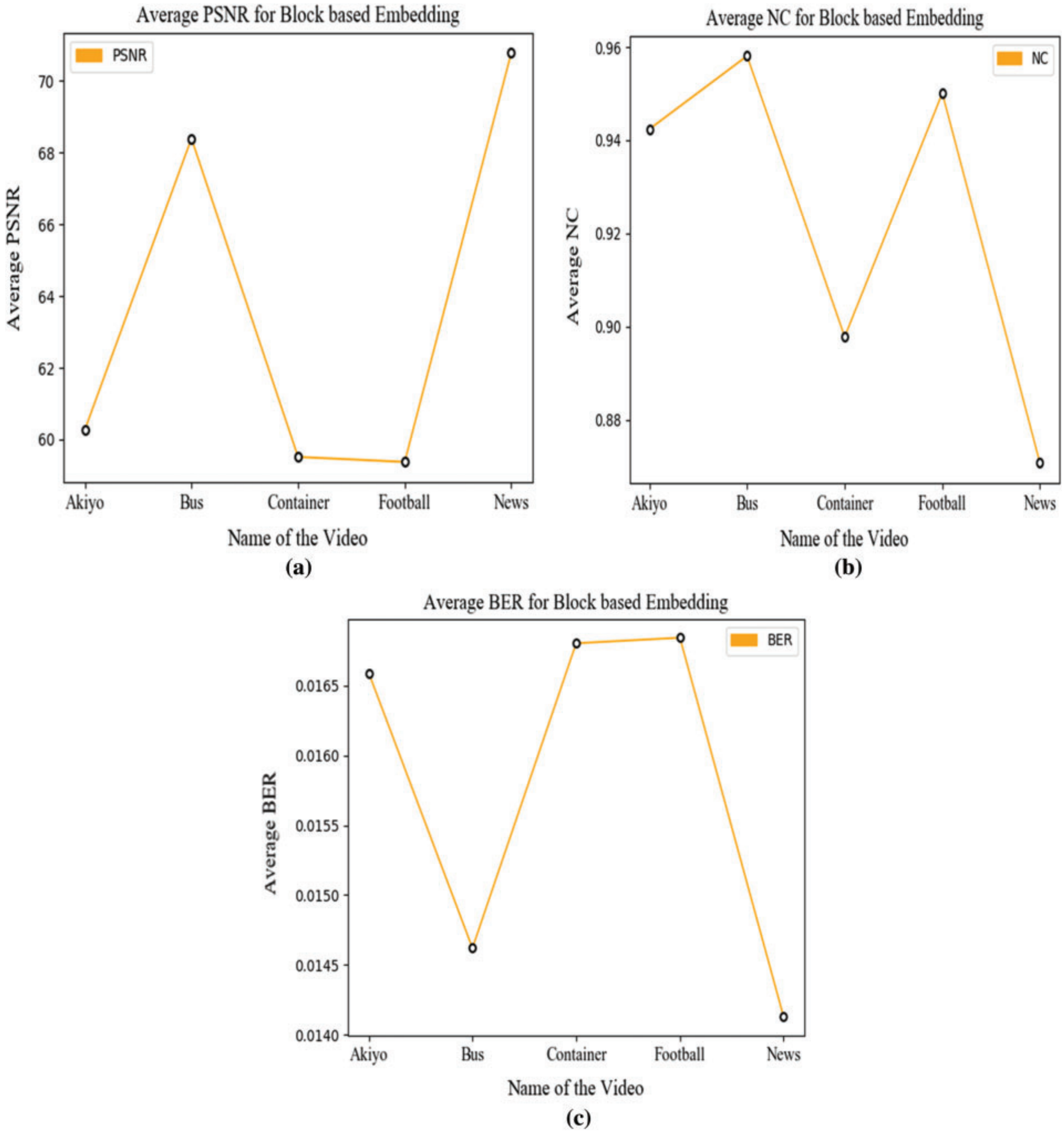


Figure 9: Plot of PSNR, NC, BER w.r.t the videos taken in the proposed work for watermark against Rotation attack. (a) Average PSNR, (b) Average NC, (c) Average BER

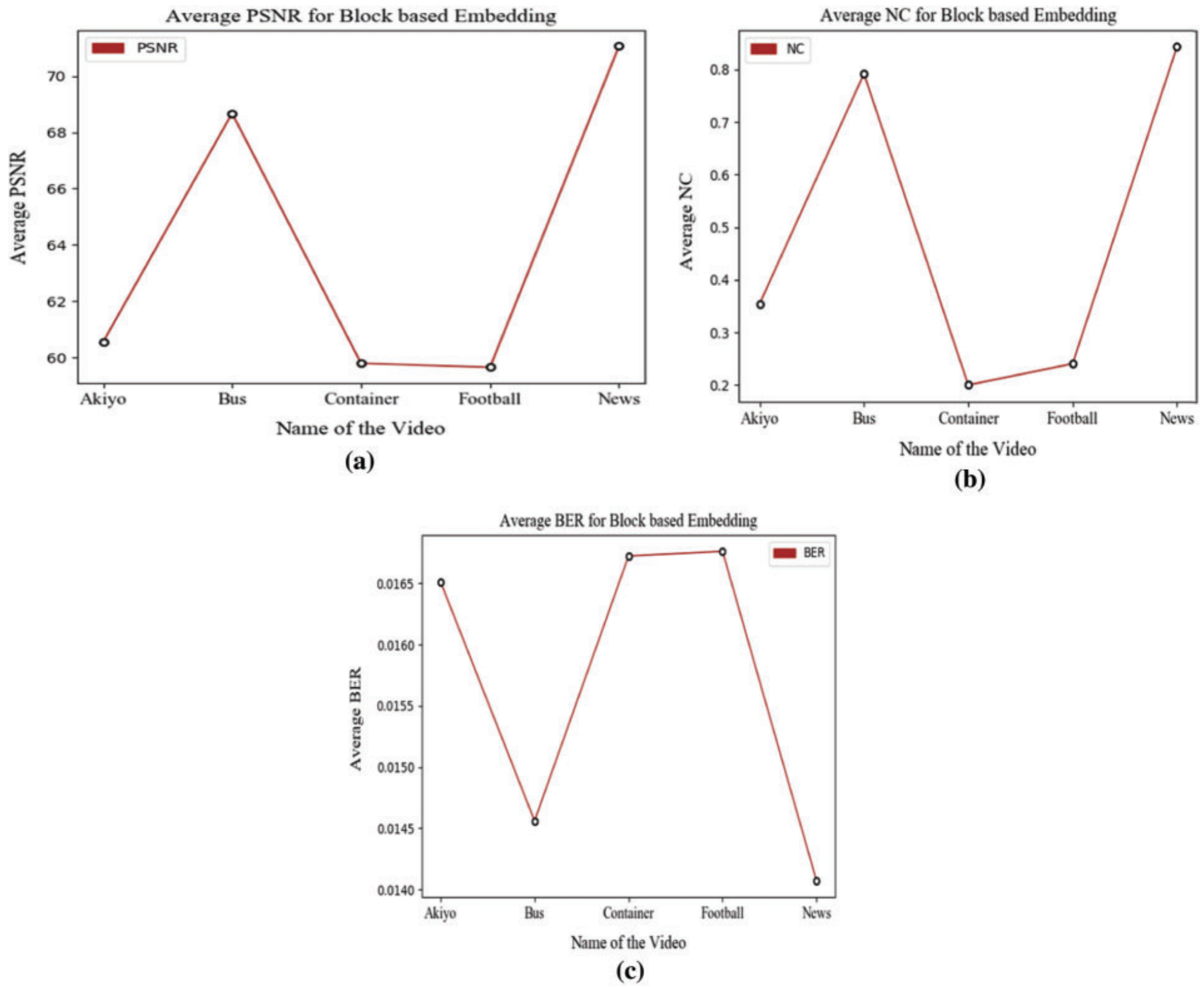


Figure 10: Plot of PSNR, NC, BER w.r.t the videos taken in the proposed work for watermark against median filter attacked. (a) Average PSNR, (b) Average NC, (c) Average BER

Table 5: Results after embedding of watermark on Gaussian attacked selected frames

Videos	K	P	PSNR	NC	BER
Akiyo	143	0.7	60.5279	0.6893	0.01652131
Bus	214	0.7	68.6757	0.9491	0.01456119
Football	159	0.6	59.6548	0.83475	0.01676311
News	211	0.8	71.0613	0.931	0.01407236
Container	240	0.4	59.7934	0.96	0.0167242

Table 6: Results after embedding of watermark on Sharpening attacked selected frames

Videos	K	P	PSNR	NC	BER
Akiyo	143	0.7	60.5279	0.7758	0.01652131
Bus	214	0.7	68.6757	0.8257	0.01456119
Football	159	0.6	59.6548	0.8361	0.01676311
News	211	0.8	71.0613	0.928	0.01407236
Container	240	0.4	59.7934	0.9482	0.01672425

Table 7: Results after embedding of watermark on Rotation attacked on selected frames

Videos	K	P	PSNR	NC	BER
Akiyo	143	0.7	60.274	0.9424	0.016591012
Bus	214	0.7	68.395	0.9581	0.014621059
Football	159	0.6	59.374	0.9501	0.016842389
News	211	0.8	70.78	0.8707	0.014128305
Container	240	0.4	59.514	0.8978	0.016802769

Table 8: Results after embedding of watermark on Median filter attacked on selected frames

Videos	K	P	PSNR	NC	BER
Akiyo	143	0.7	60.556	0.3544	0.01651364
Bus	214	0.7	68.676	0.7917	0.014561128
Football	159	0.6	59.6549	0.24055	0.016763082
News	211	0.8	71.0613	0.8429	0.014072357
Container	240	0.4	59.7934	0.2001	0.016724254

9 Conclusion and Future Work

We proposed a unique block-based embedding technique to entertain quality loss in watermarking process. For efficient time and space complexity, we utilize frame selection for the application of watermark embedding, which will be a time-consuming and space-expensive method. This frame selection is based on the number of scenes that change in a particular test video. This is followed by block-based application of DCT2 and searching for optimal parameters for embedding based on black and white watermark bits. The approach is tested against several image processing attacks such as Gaussian Noise, Image Sharpening, Rotation attack, Median filter attack, etc. The results indicate good levels of robustness against many attacks. Further security can be added in future work such as watermark encryption and more suitable frame selection for videos having no detectable scene change.

Acknowledgement: The authors would like to express their gratitude to King Khalid University, Saudi Arabia for providing administrative and technical support. This work was supported by the Deanship of Scientific Research, King Khalid University, under Grant GRP/358/42 and would also like to thank

Taif University Researchers Support Project Number (TURSP-2020/196) Taif University, Taif, Saudi Arabia.

Funding Statement: The authors would like to express their gratitude to King Khalid University, Saudi Arabia for providing administrative and technical support. This work was supported by the Deanship of Scientific Research, King Khalid University, under Grant GRP/358/42 and the study was also funded by Taif University Researchers Support Project Number (TURSP-2020/196) Taif University, Taif, Saudi Arabia.

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

References

[1] A. M. Abdelhakim, H. I. Saleh and A. M. Nassar, “A quality guaranteed robust image watermarking optimization with artificial bee colony,” *Expert Systems with Applications*, vol. 72, pp. 1–10, 2016.

[2] L. Agilandeeswari and K. Ganesan, “A robust color video watermarking scheme based on hybrid embedding techniques,” *Multimedia Tools and Applications*, vol. 75, pp. 8745–8780, 2016.

[3] D. Bhowmik, M. Oakes and C. Abhayaratne, “Visual attention-based image watermarking,” *Institute of Electrical and Electronics Engineers Access*, vol. 2016, pp. 1–17, 2016.

[4] A. Cedillo-Hernandez, M. Cedillo-Hernandez, M. N. Miyatake and H. P. Meana, “A spatiotemporal saliency-modulated JND profile applied to video watermarking,” *Journal of Visual Communication and Image Representation*, vol. 52, no. 18, pp. 106–117, 2018.

[5] F. Hartung and B. Girod, “Digital watermarking of Raw and compressed video,” in *Proc. of the Digital Compression Technologies and Systems for Video Communications*, Milan, vol.1996, pp. 205–213, 1996.

[6] M. Jain, V. Singh and A. Rani, “A novel nature-inspired algorithm for optimization: Squirrel search algorithm,” *Swarm and Evolutionary Computation*, vol. 2019, no. 49, pp. 148–175, 2019.

[7] L. Guang-Qi, Z. Xiao-Shi, Z. Yan-Ling and L. Na, “A robust digital video watermark algorithm based on DCT domain,” in *Proc of the Int. Conf. on Computer Application and System Modeling*, Taiyuan, vol. 2010, pp. 1–26, 2010.

[8] M. Masoumi and S. Amiri, “A blind scene-based watermarking for video copyright protection,” *AEU-International Journal of Electronics and Communications*, vol. 67, no. 6, pp. 528–535, 2013.

[9] M. Moosazadeh and G. Ekbatanifard, “A new DCT-based robust image watermarking method using teaching-learning-based optimization,” *Journal of Information Security and Applications*, vol. 47, pp. 28–38, 2019.

[10] W. Pei, Z. Zheng and J. Ying, “Novel video watermark technique in motion vectors,” *Institute of Electrical and Electronics Engineers Access*, vol. 2008, no. 8, pp. 1–23, 2008.

[11] Poonam and S. M. Arora, “DWT-SVd based robust digital watermarking for digital image,” *Procedia Computer Science*, vol. 132, pp. 1441–1448, 2018.

[12] R. O. Preda and D. N. Vizireanu, “A robust digital watermarking scheme for video copyright protection in the wavelet domain,” *Measurement*, vol. 43, pp. 1720–1726, 2010.

[13] H. Sahu and E. Patil, “A review on digital video watermarking,” *International Research Journal of Engineering and Technology*, vol. 7, no. 5, pp. 5432–5436, 2020.

[14] C. Sharma and A. Bhaskar, “A review on video watermarking techniques for compressed domain with optimization algorithms,” *Materials Today: Proceedings*, vol. 2020, pp. 1–9, 2020.

[15] S. Sharma, H. Sharma and J. B. Sharma, “An adaptive color image watermarking using RDWT-SVD and artificial bee colony-based quality metric strength factor optimization,” *Applied Soft Computing Journal*, vol. 84, pp. 1–18, 2019.

[16] V. Sharma and R. N. Mir, “An enhanced time efficient technique for image watermarking using ant colony optimization and light gradient boosting algorithm,” *Journal of King Saud University*, vol. 2019, pp. 1–12, 2019.

[17] K. S. Soppari and N. S. Chandra, “Development of improved whale optimization-based FCM clustering for image watermarking,” *Computer Science Review*, vol. 37, pp. 1–21, 2020.

[18] P. S. Venugopala, H. Sarojadevi, N. N. Chiplunkar and V. Bhat, “Video watermarking by adjusting the pixel values and using scene change detection,” in *Proc 2014 Fifth Int. Conf. on Signal and Image Processing*, Bangalore, vol. 13, pp. 259–264, 2013.

[19] Y. Wang and T. Du, “An improved squirrel search algorithm for global function optimization,” *Algorithms*, vol. 80, no. 12, pp. 1–29, 2019.

[20] D. Ye, C. Zou, Y. Dai and Z. Wang, “A new adaptive watermarking for real-time MPEG videos,” *Applied Mathematics and Computation*, vol. 185, no. 2, pp. 907–918, 2007.

[21] X. Yu, C. Wang and X. Zhou, “A review on robust video watermark algorithms for copyright protection,” *Applied Science*, vol. 8, no. 10, pp. 1–26, 2018.

[22] C. Sharma, A. Bagga, R. Sobti, T. R. Lohani and M. Shabaz, “A secured frame selection based video watermarking technique to address quality loss of data: Combining graph based transform, singular valued decomposition, and hyperchaotic encryption,” *Security and Communication Networks*, vol. 2021, no. 5536170, pp. 1–19, 2021.