mathematics

MDPI

Article

A New Robust and Secure 3-Level Digital Image Watermarking Method Based on G-BAT Hybrid Optimization

Kilari Jyothsna Devi [1], Priyanka Singh [1], Jatindra Kumar Dash [1], Hiren Kumar Thakkar [2], José Santamaría [3,*], Musalreddy Venkata Jayanth Krishna [1] and Antonio Romero-Manchado [4]

[1] Department of Computer Science and Engineering, SRM University, Amaravati 522240, Andhra Pradesh, India
[2] Department of Computer Science and Engineering, School of Technology, Pandit Deendayal Energy University, Gandhinagar 382007, Gujarat, India
[3] Department of Computer Science, University of Jaén, 23001 Jaén, Spain
[4] Department of Cartographic Engineering, Geodesy, and Photogrammetry, University of Jaén, 23001 Jaén, Spain
* Correspondence: jslopez@ujaen.es

Abstract: This contribution applies tools from the information theory and soft computing (SC) paradigms to the embedding and extraction of watermarks in aerial remote sensing (RS) images to protect copyright. By the time 5G came along, Internet usage had already grown exponentially. Regarding copyright protection, the most important responsibility of the digital image watermarking (DIW) approach is to provide authentication and security for digital content. In this paper, our main goal is to provide authentication and security to aerial RS images transmitted over the Internet by the proposal of a hybrid approach using both the redundant discrete wavelet transform (RDWT) and the singular value decomposition (SVD) schemes for DIW. Specifically, SC is adopted in this work for the numerical optimization of critical parameters. Moreover, 1-level RDWT and SVD are applied on digital cover image and singular matrices of LH and HL sub-bands are selected for watermark embedding. Further selected singular matrices S_{LH} and S_{HL} are split into 3×3 non-overlapping blocks, and diagonal positions are used for watermark embedding. Three-level symmetric encryption with low computational cost is used to ensure higher watermark security. A hybrid grasshopper–BAT (G-BAT) SC-based optimization algorithm is also proposed in order to achieve high quality DIW outcomes, and a broad comparison against other methods in the state-of-the-art is provided. The experimental results have demonstrated that our proposal provides high levels of imperceptibility, robustness, embedding capacity and security when dealing with DIW of aerial RS images, even higher than the state-of-the-art methods.

Keywords: aerial images; RDWT–SVD; digital image watermarking; random key; grasshopper; bat; hybrid optimization

MSC: 68U10; 68T20

Citation: Devi, K.J.; Singh, P.; Dash, J.K.; Thakkar, H.K.; Santamaría, J.; Krishna, M.V.J.; Romero-Manchado, A. A New Robust and Secure 3-Level Digital Image Watermarking Method Based on G-BAT Hybrid Optimization. *Mathematics* **2022**, *10*, 3015. https://doi.org/10.3390/math10163015

Academic Editor: Radu Tudor Ionescu

Received: 10 July 2022
Accepted: 13 August 2022
Published: 21 August 2022

1. Introduction

The widespread use of the Internet and ever-evolving telecommunications technologies has led to an increase in the sharing of multimedia content, such as audio, video and images. The revolution of social networks, electronic health care systems, electronic commerce and the IoT has further accelerated the exchange of multimedia content through the Internet. Moreover, specific remote sensing (RS) tasks involving aerial images are largely known of, e.g., the transmission of aerial RS images for tracking weather, volcanic eruptions, disaster management and growth of cities or forests, among others [1]. RS images contain sensitive data; therefore, it is necessary to ensure their authenticity and provide copyright protection. There is a high possibility that these contents can be easily

hacked and manipulated by unauthorized users. Furthermore, illegal distribution of RS images should be prevented. Therefore, the authenticity, integrity and confidentiality of images is very important. This issue can be addressed by using the digital image watermark (DIW) approach [2]. DIW is the process where a watermark such as text or an image is embedded in a cover image to get a watermarked image. Features such as robustness and imperceptibility are important for efficient watermarking schemes. The term imperceptibility represents the visual quality of the image. If the original and embedded images are the same in terms of visual perception, then it is very unnoticeable. Robustness refers to similarity between the original and extracted watermark images [3].

However, it is very difficult to maintain all of these features (i.e., imperceptibility, robustness and embedding capacity) in a single watermark scheme, and there is always a trade-off between them. To face this, an optimal scaling factor (Γ) should be used in the embedding process in order to obtain quality DIW results. In the last decade, an approach called soft computing (SC) [4] has emerged to tackle with the latter issue by proposing near-optimal solutions to complex optimization problems. In particular, nature inspired optimization (NIO) algorithms have become a viable solution to tackle with this problem due to their successful results [5,6]. Our proposal makes use of SC paradigm to derive near-optimal solutions for the scaling factor. Watermark security is also one of the important features to look for in a DIW scheme. The watermark can be embedded in the spatial or spectral domain. During streaming, the security of the watermark is also crucial. To ensure high security, researchers have suggested hashing, compression, Arnold maps and chaotic maps techniques. However, these techniques are either less secure or have high computational costs for encryption and decryption tasks [7].

In the existing literature, limited research has been done to address issues related to the copyright protection, authentication and security problems of remote sensing images. The proposed scheme aims to address the latter limitations of the current DIW method for aerial RS images by providing high imperceptibility, robustness and security with the optimal computational cost. Here, a 3-level watermark security scheme is proposed to achieve high watermark security with a low computational cost. Furthermore, a grasshopper–BAT (G-BAT) hybrid NIO algorithm is proposed for the optimal computation of the scaling factor to balance the trade-off between watermarking characteristics.

The structure of the paper is as follows. Section 2 provides a brief review of the state-of-the-art of DIW, describing several of the contributions and limitations of the current methods in the field. This discussion is followed by the motivation and contribution of the proposed scheme. The proposed scheme is described in detail in Section 3, and Section 4 is focused on its experimental evaluation. Finally, Section 5 provides relevant comments on the results reported as a conclusion and outlines work that may be carried out in the future.

2. Literature Review

As stated, the DIW scheme came into being and has been widely used to secure the copyright of one's content, such as images, text, audio and video. The initial DIW schemes that exist focused primarily on imperceptibility and robustness by incorporating watermarking in the spatial and spectral domains. In particular, the schemes proposed in [8–10] used the spatial domain in order to embed the watermark in the image. In the spatial domain, embedding can be done directly by using pixel values of the image. This gives us high imperceptibility and less robustness. The majority of the researchers proposed methods dealing with DIW from the frequency domain for the watermark embedding [10–13] to ensure higher robustness and imperceptibility. Furthermore, to increase the imperceptibility and robustness of the watermark scheme, hybrid transformations are suggested [14–18]. For the secure transmission of aerial images in remote sensing applications, some researchers proposed the DIW schemes [19–23]. In order to incorporate a watermark in satellite images, the DIW approach suggested in [19] used a degradation and restoration model and a chaotic map for watermark security.

Furthermore, [20] describes a DWT transform scheme for watermarking remote sensing images. To secure the watermark in this scheme, Arnold cat maps are used. For secure transmission of compressive remote sensing images, a hybrid LWT–Hadamard–ternary watermark sequence DIW scheme is suggested in [21].

A DIW scheme propped in [22] employs a hybrid DWT–HD–SVD transform to safeguard the copyright of landside images. The ideal embedding factor is discovered using particle swarm optimization (PSO) and the firefly optimization approach. Additionally, the step space filling curve-based approach is used to achieve security. The DIW approach suggested in [23] used the scale-invariant feature and wavelet transform to include aerial images for copyright protection.

To balance the trade-offs in the watermark features, a scaling factor (i.e., Γ) is used for the watermark embedding process. Some of the researchers used a constant scaling factor to optimize all the images [10], which may not work effectively for all kinds of imaging modalities. Therefore, optimization of scaling factor plays a crucial role in the entire DIW process. Specifically, the DIW algorithms proposed in [14–16,18] adopted the use of NIO algorithms for scale factor optimization. Nevertheless, despite the good performances of these methods, the results in terms of imperceptibility and robustness can be improved. On the other hand, NIO algorithms alone are not very effective at balancing the exploration and exploitation of algorithm search behavior. Thus, there is a need to develop hybrid optimization algorithms to properly balance the search behavior. Additionally, the security of the watermark has not been given much importance by the field. The methods proposed in [10,15,17,23] have less focus on watermark security, and the schemes proposed in [14,16,19–21] used chaotic maps and Arnold maps for watermark security. However, these methods are less secure methods. It is suggested to use a pseudo-random key for watermark encryption in [18], but the computational cost of the key generation process is high. Therefore, developing high security approaches with lower computational costs is both a challenging and needed task.

Motivation and Contributions:

A review of these watermark schemes revealed that the majority of existing DIW schemes have less efficacious embedding scaling factors and overlooked watermark security. To address the security issue, a 3-level watermark security scheme is proposed in this paper to achieve high watermark security with low computational cost, tackling aerial RS images. Furthermore, our proposal is based on the design of a hybrid optimization algorithm for the optimal computation of the scaling factor by using a NIO algorithm, i.e., G-BAT. An outline of the main contribution of our work is as follows:

I. Higher security with low computational cost: Three levels of watermark security, i.e., shuffling, substitution and partitioning, are suggested in the proposed scheme to ensure high watermark security with low computational cost. At the 1st level, row-column-zigzag (RCZ) shuffling is applied to obtain a shuffled watermark. Then, using the Lehmer Random Number Generator (LRNG), substitution is performed on the RCZ random watermark to obtain an encrypted watermark. An additional encrypted watermark is divided into odd and even position pixels for embedding.

II. Scaling factor optimization using hybrid G-BAT: Scale factor optimization using a hybrid NIO-based G-BAT algorithm to balance trade-offs in watermark features and to balance exploration and exploitation search behavior of watermarks.

III. High imperceptibility and robustness: Employment of a hybrid RDWT–SVD transformation for high imperceptibility and robustness. Due to its change-invariant nature, RDWT is resistant to geometric attacks. SVD, on the other hand, is resistant to filtering and noise attacks.

IV. Application area: The proposed DIW scheme is used to address security concerns in the transmission of digital images of aerial RS images over the Internet. Furthermore, it combats various threats in the transmission of RS images, such as copyright protection, copy control and unauthorized access.

3. Materials and Methods

A hybrid RDWT–SVD based DIW scheme is proposed for secure transmission of aerial RS images through digital communications channels. The proposed scheme ensures high robustness, imperceptibility, security and embedding capacity. Sections 3.1–3.4 are devoted to elucidate the proposed scheme.The flowchart of the proposed scheme is provided in Figure 1.

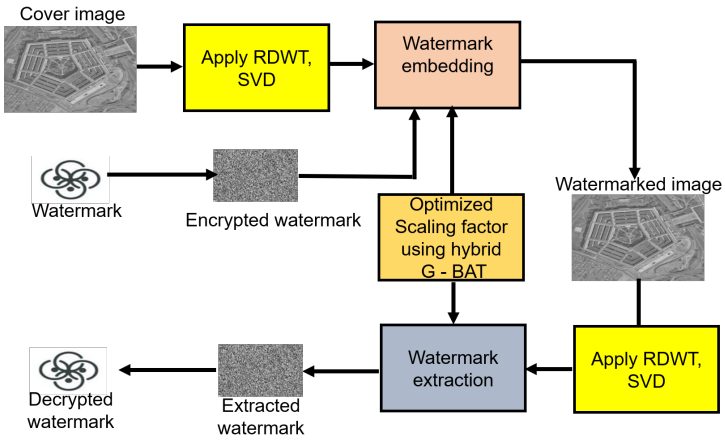

Figure 1. Flowchart of the proposed scheme.

3.1. Watermark Embedding

A cover image (C) of size $X \times Y$ and a binary watermark (W) of size $A \times B$ are considered in the suggested scheme. First, 1-level RDWT is applied on C to decompose it into four sub-bands: LL, LH, HL and HH. Sub-bands LH and HL are chosen for watermark embedding, as they undergo less distortions due to embedding than LL and HH. Since RDWT is shift invariant and avoids downsampling in its sub-bands, each sub-band is of the same size as C. RDWT is more reliable in conveying watermarks with a high embedding capacity by this property. However, it is more vulnerable to noise and filtering attacks. To mitigate this pitfall, the proposed scheme employs a hybrid RDWT-SVD algorithm. SVD is applied on LH and HL, which decomposes into U_{LH}, S_{LH}, V_{LH}, U_{HL}, S_{HL} and V_{HL} sub-matrices, respectively. U_{LH} U_{HL} V_{LH} and V_{HL} represent the column and row orthogonal matrices; and S_{LH} S_{HL} represents singular diagonal matrices. Since the tiny fluctuation in singular values is unlikely to impact the image's visual perception, the watermark is embedded by altering these singular values that show high robustness. Therefore, S_{LH} and S_{HL} matrices are selected and partitioned into 3×3 non-overlapping blocks (β_{LH}, β_{HL}) for embedding. To ensure high imperceptibility and robustness, it uses an optimized scaling factor (Γ); an encrypted even (W_e) and odd (W_o) watermark are embedded using Equations (1) and (2) into diagonal positions of β_{LH} and β_{HL}, respectively. The process of watermark encryption as elucidated in Section 3.3 and scaling factor optimization using G-BAT is described in Section 3.4. Finally, inverse SVD and RDWT are applied to get watermarked image (C').

$$\beta_{LH}{}'(k,k) = \beta_{LH}(k,k) + \Gamma * W_e \quad (1)$$

$$\beta_{HL}{}'(k,k) = \beta_{HL}(k,k) + \Gamma * W_o \quad (2)$$

where k = 1, 2 and 3; and Γ is optimized scaling factor.

The proposed watermark embedding, encryption and scaling factor optimization are shown in Figure 2. The process of watermark extraction is explained in detail in the next Section 3.2.

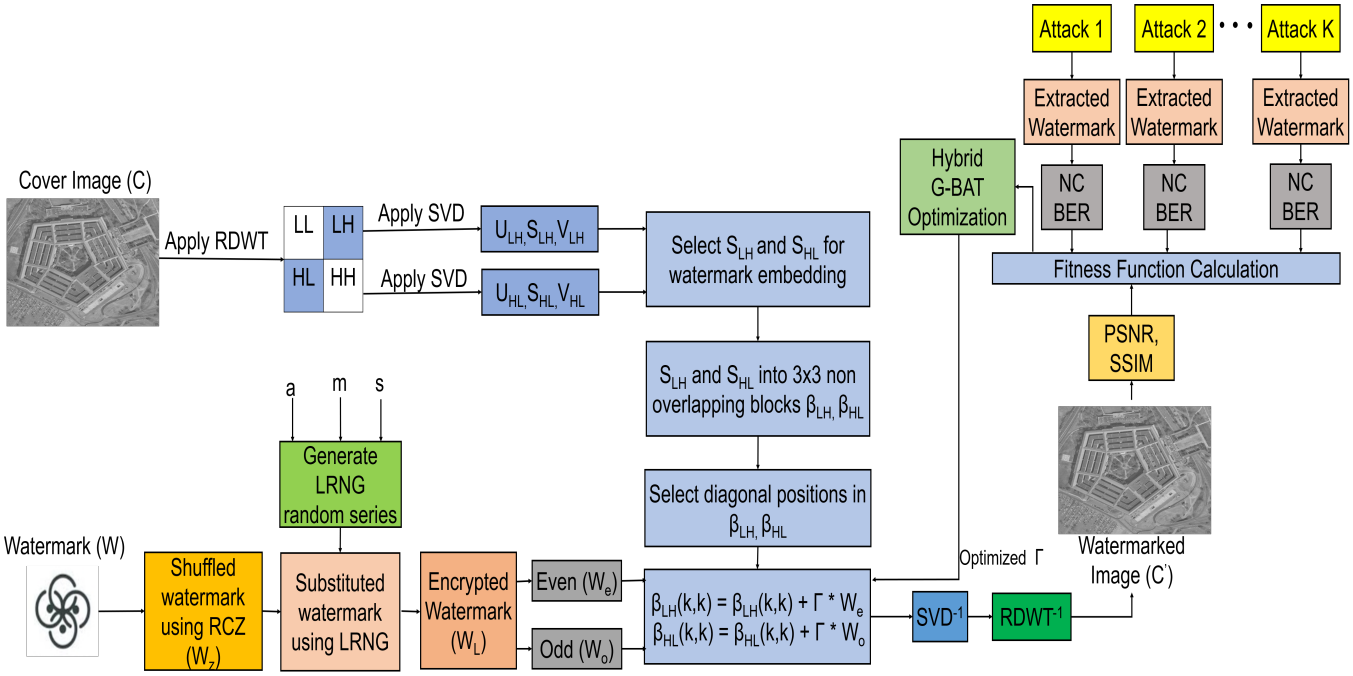

Figure 2. Block diagram for watermark encryption and embedding.

3.2. Watermark Extraction

The proposed DIW scheme is blind, as the original image is not required for watermark extraction. The secret keys (Γ, S_{LH}, S_{HL}, a, m, s) are used in the proposed scheme to extract the watermark from the watermarked image. The watermark extraction process is the reverse of the embedding process. For watermark extraction, RDWT is applied on the watermarked image (C') to obtain sub-bands LL', LH', HL' and HH'. On sub-bands LH' and HL', the SVD transform is applied, and it further decomposes into ${U_{LH}}'$, ${S_{LH}}'$ and ${V_{LH}}'$; and ${U_{HL}}'$, ${S_{HL}}'$ and ${V_{HL}}'$. Select ${S_{LH}}'$ and ${S_{HL}}'$ and S_{LH} and S_{HL} singular arrays and split them into 3×3 non-overlapping blocks (${\beta_{LH}}'$ and ${\beta_{HL}}'$; and β_{LH}, β_{HL} and respectively). The encrypted ${W_e}'$ and ${W_o}'$ are extracted from ${\beta_{LH}}'$ and ${\beta_{HL}}'$ using Equations (3) and (4).

$$ {W_e}' = ({\beta_{LH}}'(k,k) - \beta_{LH}(k,k))/\Gamma \tag{3} $$

$$ {W_o}' = ({\beta_{HL}}'(k,k) - \beta_{HL}(k,k))/\Gamma \tag{4} $$

where k = 1, 2 and 3; and Γ is the optimized scaling factor.

The block diagram for watermark extraction and decryption is shown in Figure 3. The extracted ${W_o}'$ and ${W_e}'$ are decrypted using a decryption approach, as explained in Section 3.3.2, to obtain the watermark image (W'). The following subsections describe the watermark encryption and decryption process.

3.3. Watermark Encryption and Decryption

In the transmission of images over the Internet, ensuring the security of the watermark, is very important. In the proposed scheme, the security of the watermark is achieved by using a symmetric cryptographic approach. The proposed scheme ensures watermark security at 3 levels: (1) RCZ shuffle, (2) LRNG substitution and (3) odd-even partitioning.

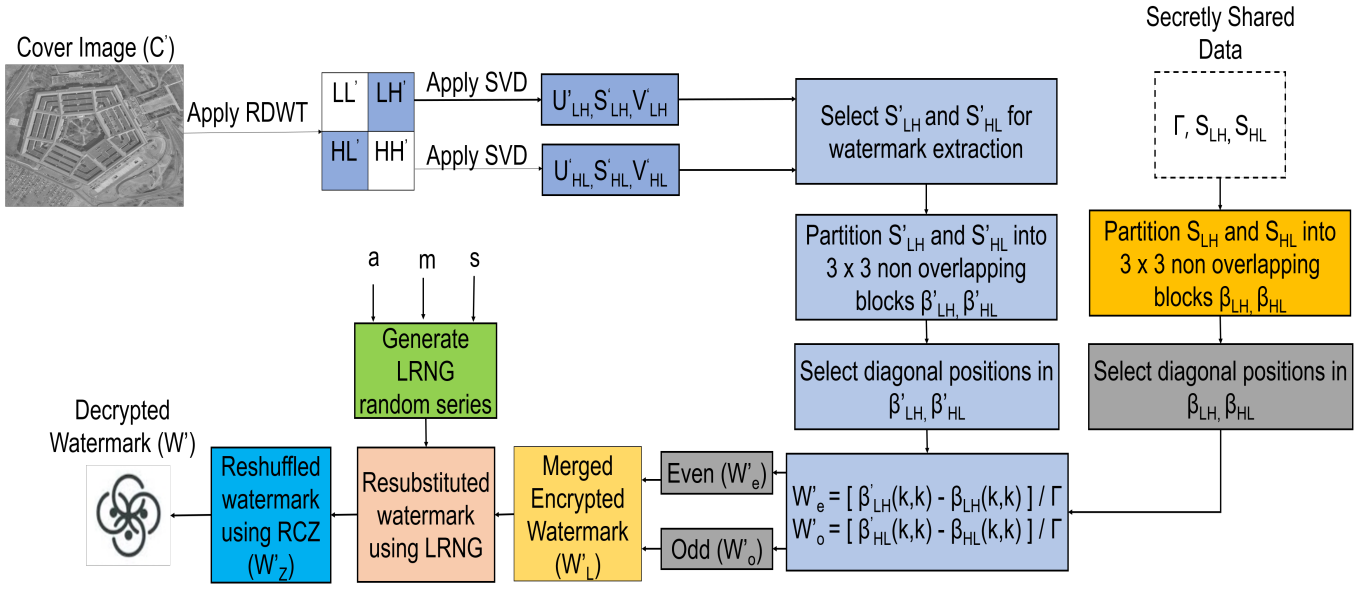

Figure 3. Block diagram for watermark extraction and decryption.

3.3.1. Watermark Encryption

Shuffling is done along the rows and columns and in a zigzag pattern (RCZ). The Lehmer Random Number Generator (LRNG) is used for the substitution process. Finally, the resulting encrypted watermark is divided into two parts. To secure the watermark, the RCZ is used in conjunction with the LRNG. This provides the encrypted watermark image, which is split into odd and even pixels and embedded in the cover image. In the RCZ shuffling process, initially the shuffling is done along the rows, followed by the columns, and then in a zigzag fashion. The procedural steps for watermark encryption are explained in Algorithm 1, and the 3-level encryption process is explained next.

Algorithm 1: Algorithm for watermark encryption.

Require: Watermark image (W)

Ensure: Even (W_e) and Odd (W_o) watermarks

1: Apply RCZ shuffling process on W to get 1st level encrypted watermark image (W_z) using the steps from 2 to 4.
2: Apply row shuffling: Even and odd rows in W is shuffled separately. The ith even row is shuffled to $(i+2)$th row and ith odd row is shuffled to $(i+2)$th row with a total of $M/2$ even, odd rows. Where M is the total number of rows in W and i is in the range of $[1, M/2]$. The resulting W is W_r
3: Apply column shuffling: The jth even column in W_r is shuffled to $(j+2)$th column and jth odd column is shuffled to $(j+2)$th column with a total of N$/2$ even, odd columns and j is in the range of $[1, N/2]$. Where N is the number of columns in W_r. The resulting is W_c.
4: Apply zigzag shuffling: The pixels positions in the W_c are shuffled in the zigzag manner. Then the final 1st level encrypted W is W_z.
5: Generate a LRNG decimal random series. Furthermore, convert each decimal value in random series with equivalent 8-bit binary format (R_b).
6: Then substitute each pixel value in W_z with R_b using Xor operation to generate final encrypted W (W_L).
7: Further partitioning the W_L into even (W_e) and odd (W_o) pixel positions using Equations (5) and (6).

Initially, in row shuffling, even rows, i.e., row ith, are shuffled into row $(i+2)$th, as shown in Figure 4a. Furthermore, similarly, odd rows, i.e., row ith, are shuffled into row $(i+2)$th to get W_r. Similarly, in column shuffling, odd and even columns are shuffled into W_r; i.e., column jth is shuffled into column $(j+2)$th, as shown in Figure 4b, to get W_c. In zigzag shuffling, shuffling starts from the position pixel in the upper left corner of the image and traverses the zigzag path at W_c, as shown in Figure 4c. The pixels that have been traversed are stored in the one-dimensional array, which is then converted to the 2D array for a mixed watermark (W_z). Furthermore, W_z is substituted using a binary random key (R_b) generated from LRNG.

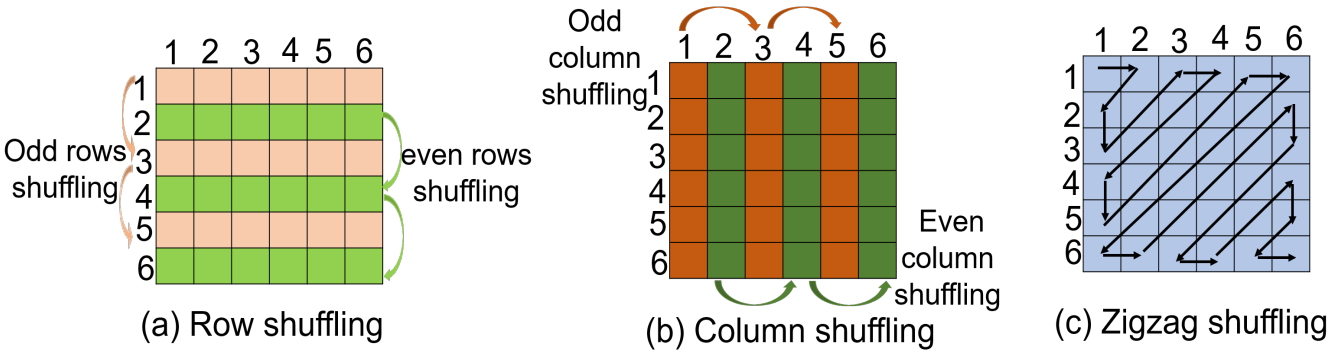

Figure 4. RCZ shuffling.

LRNG is a pseudo-random number generator used to generate decimal random numbers within a given range. The generation of random series uses LRNG as mentioned in [24]. After generation of the LRNG random string, each member of the string is converted to an equivalent 8-bit binary format to generate a binary random key (R_b) of size equal to the size of W. Then, W_z is combined with R_b to get encrypted W (W_L). Additionally, W_L is divided into even (W_e) and odd (W_o) position pixels of size $(A \times B)/2$ using Equations (5) and (6).

$$\sum_{k=1}^{(A\times B)/2} W_e(k) = \begin{cases} \sum_{i=1}^{A}\sum_{j=1}^{B} W_L(i,j), & if\ mod(j,2) = 0 \\ ignored, & Otherwise \end{cases} \quad (5)$$

$$\sum_{k=1}^{(A\times B)/2} W_o(k) = \begin{cases} \sum_{i=1}^{A}\sum_{j=1}^{B} W_L(i,j), & if\ mod(j,2) \neq 0 \\ ignored, & Otherwise \end{cases} \quad (6)$$

where $A \times B$ is the size of W_L.

The process of watermark encryption is elucidated with an example in Figure 5a. The process of watermark decryption is explained in the next sub-section.

3.3.2. Watermark Decryption

Watermark decryption is a reverse process to watermark encryption. First, the extracted even (W_e') and odd (W_o') position pixels are merged to obtain an encrypted watermark (W_L'). The process of extracting W_e' and W_o' from C' is explained in Section 3.2. More LRNG random strings are generated using secret keys (a, m, s) and convert them to the equivalent binary format (R_b'). Next, we XOR between W_L' and R_b' to get an encrypted watermark of level 1st (W_z'). Furthermore, we apply the RCZ process to get the original watermark (W'). The watermark decryption process is clarified with an example in Figure 5b.

3.4. Scaling Factor Optimization Using Hybrid G-BAT

DIW characteristics such as imperceptibility, robustness and embedding capacity are always trade-offs in designing a DIW scheme. A good DIW scheme must achieve a balance between the latter three traits. The balancing of these three features is aided by the scaling

factor (i.e., Γ), and using a suitable scaling factor is critical for retrieving quality DIW outcomes. The majority of the researchers recommended using optimization algorithms to find the optimum scaling factor. Specifically, the NIO algorithms, such as evolutionary algorithms (i.e., GA, DE and JAYA) and swarm intelligence (PSO, ABC, grasshopper, Bat, firefly), have been suggested in the last few years. However, a good balance of exploration and exploitation of search behavior for NIO is also critical. In our proposed scheme, a hybrid grasshopper–BAT (G-BAT) optimization algorithm is adopted to reconcile the search behaviors of exploration (search agents are encouraged to move abruptly) and exploitation (search agents tend to move abruptly).

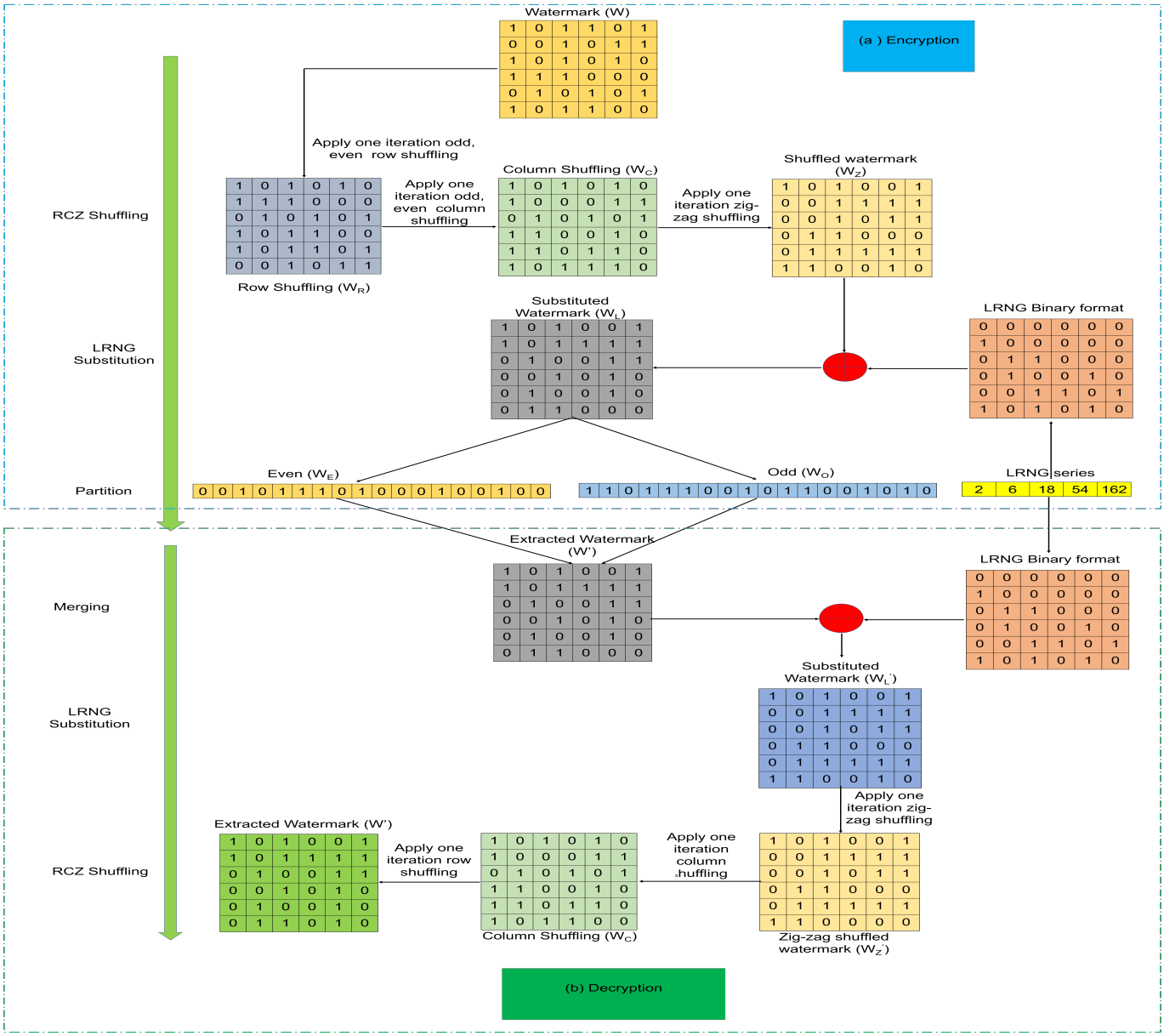

Figure 5. Three-level watermark encryption and decryption process.

The grasshopper optimization (GO) algorithm is a simple and fast swarm intelligence-based NIO algorithm that mimics the behavior of grasshoppers [25]. In terms of search behavior, GO strikes a good balance between exploration and exploitation, and also avoids getting stuck in local optima. As well as its advantages, it also has some limitations: grasshoppers reach their comfort zones faster, but the swarm does not join a key point, which results in low search accuracy and premature convergence [26]. To address the

shortcomings of the GO algorithm and improve convergence rates and search accuracy, the proposed scheme integrates GO with the BAT algorithm. BAT is also a swarm intelligence optimization algorithm based on NIO [27]. The unique feature of the BAT algorithm is that it balances exploration and exploitation behavior through echolocation frequency tuning control and auto-zoom capability. It also has the advantage of requiring parameter control rather than the fixed and preset algorithmic dependent parameters used by many NIO algorithms, allowing automatic switching from exploration to exploitation to reach the optimal solution. Table 1 shows the basic parameters for the optimization of the hybrid G-BAT [28], and Figure 6 shows the flowchart representation for the hybrid G-BAT.

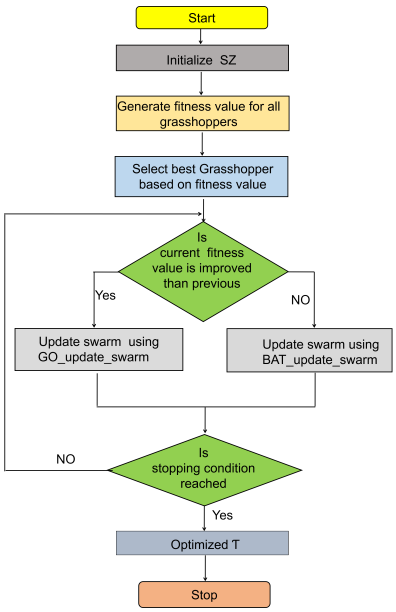

Figure 6. Flowchart of scaling factor (Γ) optimization using G-BAT.

Table 1. List of basic parameters for hybrid G-BAT.

Parameter	Initial Value
Maximum Number of iterations (MNI)	30
Swarm size (SZ)	25
Swarm_minval (min)	0.001
Swarm_maxval (max)	MNI
Swarm generation	SP = min + (max − min) × rand(SZ,1)
Termination condition	MNI
Loudness (l)	1
Pulse rate (r0)	1
Alpha	0.97
Gamma	0.1
Freq-min	0
Freq-max	2

The fitness function used for scaling factor (Γ) optimization, as shown in Equation (7).

$$\mathrm{FF} = (\mathrm{PSNR} + \mathrm{SSIM})/\gamma + \left(\sum_{i=1}^{K} \mathrm{NC}(c) + \sum_{i=1}^{K} \mathrm{BER}(c)\right)/\gamma \quad (7)$$

where PSNR and SSIM are the imperceptibility metrics; NC and BER are robustness metrics; and γ is the chosen population value. K is the number of attacks. In the proposed scheme, an average of $K = 6$ attacks are considered for FF evaluation. To start the hybrid G-BAT, the swarm (grasshopper) is randomly generated using the swarm generation formula, as

shown in Table 1. Then, using Equation (7), fitness values are calculated for all entries in the swarm, and the best grasshopper (minimum fitness value) is chosen for further processing. The position of a grasshopper is updated each time the fitness values of the current and previous grasshopper are compared. If the current fitness value improves over time, the proposed scheme updates the new swarm using GO optimization. BAT optimization, on the other hand, helps in upgrading new swarms. To achieve the best scaling factor (Γ), GO and BAT are executed in parallel to encourage swarm information sharing, and as a result, improve search efficiency.

4. Experimental Results and Discussion

This section is devoted to reporting about the performance of the proposed DIW method. Specifically, our proposal is evaluated according to terms of imperceptibility, robustness and security. The experiments were carried out in the MATLAB 2014b environment. A cover image of size 256 × 256 and a binary watermark of size 64 × 64 were considered for the generation of experimental results. Aerial RS and general cover test images and watermark were taken from the USC-SIPI [29] and Kaggle [30] dataset, as shown in Figures 7 and 8.

Figure 7. Gray-scale cover images (**1**–**6**) with corresponding watermark images (**a**–**f**) and extracted watermark images (**I**–**VI**). Color cover images (**7**–**12**) with corresponding watermark images (**g**–**l**) and extracted watermark images (**VII**–**XII**) under zero attacks.

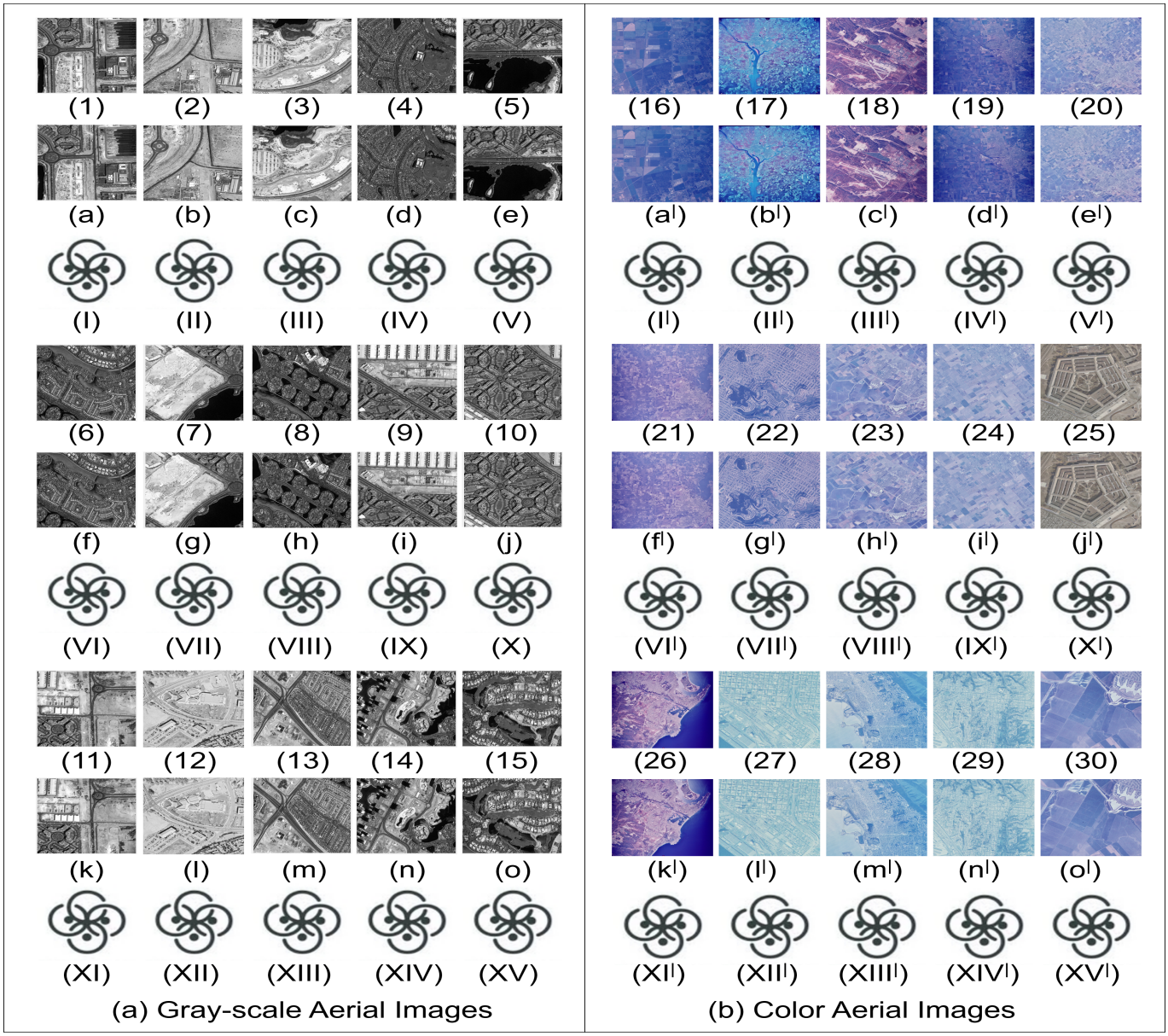

Figure 8. Aerial images (gray-scale images (**1**–**15**), color images (**16**–**30**)) with corresponding watermarked images (gray-scale watermarked images (**a**–**o**), color watermarked images (**a**^{1}–**o**^{1})) and extracted watermark images (**I**–**XV**,**I**^{1}–**XV**^{1}).

4.1. Imperceptibility Test

The visual quality of the original and embedded images is identical and is called imperceptibility. It is very important to maintain high imperceptibility for watermarked images. First, the imperceptibility performance of the proposed scheme for gray-scale cover images of different modalities is evaluated, as shown in Figure 7. Subjective analysis of the cover and the watermarked image in Figure 7 shows that both images are identical. Furthermore, the watermark extracted under zero attacks is clearly visible. In addition, the objective analysis of imperceptibility was performed using PSNR and SSIM. The relationships used for PSNR and SSIM are given in Equations (8) and (10), respectively.

$$\text{PSNR} = 10 \ \ log \ \ 10\left(\frac{255^2}{\text{MSE}}\right) \tag{8}$$

$$\text{MSE} = \frac{1}{X \times Y} \sum_{i=1}^{X} \sum_{j=1}^{Y} \left[C(i,j) - C'(i,j)\right]^2 \tag{9}$$

$$\text{SSIM} = \left[l(C,C').\ b(C,C').\ s(C,C')\right] \tag{10}$$

$$l(C,C') = \frac{2\mu_C \times \mu_{C'}}{\mu_{C^2} + \mu_{C'^2}}$$

$$b(C,C') = \frac{2\sigma_{C \times \sigma_{C'}}}{\sigma_{C^2} \times \sigma_{C'^2}}$$

$$s(C,C') = \frac{\sigma_{CC'^1}}{\sigma_C \times \sigma_{C'}}$$

where C and C' are cover and watermarked images; $X \times Y$ are the sizes of cover and watermarked images. μ_C and $\mu_{C'}$ are means of C and C': σ_{C^2}. $\sigma_{C'^2}$ is the variance of C and C'. $\sigma_{CC'^1}$ is the covariance of C and C'.

The imperceptibility performance of the proposed scheme was analyzed for general images with random scale factor (Γ = 0.9); the scale factor was optimized using GO, BAT and the proposed hybrid G-BAT, and the corresponding PSNR and SSIM values are shown in Table 2. In Table 2, it can be seen that for all gray-scale and color cover images, the PSNR $>$ exceeds the threshold value of 37 dB. Furthermore, SSIM is equal to the ideal value 1 using random Γ and optimized Γ using GO, BAT and hybrid G-BAT. The result presented in Table 2 indicates greater imperceptibility of the proposed scheme for general images.

Table 2. PSNR and SSIM for general images (gray-scale and color) with random $\Gamma = 0.9$ and optimized Γ using GO, BAT and G-BAT.

Gray-Scale Images								
Image	**With Random Γ = 0.9**		**With GO**		**With BAT**		**With G-BAT**	
	PSNR	**SSIM**	**PSNR**	**SSIM**	**PSNR**	**SSIM**	**PSNR**	**SSIM**
Baboon	89.03	1	80.87	1	79.64	1	59.74	1
Lena	89.35	1	75.67	1	77.92	1	59.61	1
Cameraman	89.66	1	81.47	1	81.03	1	60.82	1
Pirate	89.21	1	70.84	1	66.37	1	69.26	1
Living room	88.98	1	73.96	1	75.82	1	56.92	1
MRI Brain	89.42	1	74.29	1	72.71	1	62.85	1
Color Images								
Image	**With Random Γ = 0.9**		**With GO**		**With BAT**		**With G-BAT**	
	PSNR	**SSIM**	**PSNR**	**SSIM**	**PSNR**	**SSIM**	**PSNR**	**SSIM**
Baboon	Inf	1	88.43	1	85.92	1	64.27	1
Lighthouse	Inf	1	86.45	1	87.27	1	71.76	1
Peppers	Inf	1	84.23	1	79.26	1	59.32	1
Splash	Inf	1	81.59	1	80.75	1	65.28	1
Koala	Inf	1	89.65	1	88.62	1	74.24	1
Skin	Inf	1	79.27	1	76.29	1	58.29	1

In addition, the imperceptibility performance of our proposal was analyzed for 30 aerial images (gray-scale and color) with random Γ and Γ optimized by GO, BAT and G-BAT. The corresponding PSNR and SSIM values are presented in Table 3. In Table 3, it can be seen that PSNR and SSIM for the 30 aerial images using random Γ and with

GO, BAT and G-BAT are higher than the threshold values. With random Γ, the proposed scheme shows the ideal value of PSNR for all the color images. The proposed DIW method also shows the ideal SSIM value for all the images with random Γ and with optimized Γ. Therefore, our proposal shows higher imperceptibility even if it uses random Γ, and an ever higher value of PSNR for optimized Γ, while keeping the SSIM value at 1 for all the images with different Γ. Therefore, our method demonstrated high imperceptibility for both aerial RS and conventional images.

Table 3. PSNR, SSIM, NC and BER (under zero attack) with random $\Gamma = 0.9$ and Γ optimized using GO, BAT and G-BAT for 30 aerial RS (gray-scale and color) images taken from USC-SIPI [29] and Kaggle [30] datasets.

USC-SIPI [29] Dataset Images																
Image	**With Random Γ**				**With GO**				**With BAT**				**With G-BAT**			
	PSNR	**SSIM**	**NC**	**BER**	**PSNR**	**SSIM**	**NC**	**BER**	**PSNR**	**SSIM**	**NC**	**BER**	**PSNR**	**SSIM**	**NC**	**BER**
Img1	87.55	1	0.9789	0.0332	81.63	1	0.9919	0.0327	75.28	1	0.9931	0.0089	66.37	1	**0.9996**	**0.0006**
Img2	87.24	1	0.9797	0.0371	80.29	1	0.9913	0.0298	76.16	1	0.9942	0.0094	68.50	1	**0.9999**	**0.0002**
Img3	87.32	1	0.9796	0.0374	82.82	1	0.9893	0.0301	77.13	1	0.9932	0.0085	64.35	1	**1**	**0.0002**
Img4	87.14	1	0.9798	0.0371	81.49	1	0.9903	0.0286	75.20	1	0.9949	0.0065	66.18	1	**0.9998**	**0.0004**
Img5	87.04	1	0.9797	0.0374	79.95	1	0.9911	0.0205	68.51	1	0.9941	0.0057	62.19	1	**0.9998**	**0.0004**
Img6	87.17	1	0.9798	0.0371	81.63	1	0.9853	0.0302	72.61	1	0.9942	0.0085	63.28	1	**0.9999**	**0.0002**
Img7	87.06	1	0.9799	0.0371	82.62	1	0.9868	0.0315	71.28	1	0.9921	0.0187	59.93	1	**0.9999**	**0.0002**
Img8	87.19	1	0.9797	0.0371	78.76	1	0.9903	0.0251	76.25	1	0.9938	0.0083	60.18	1	**0.9999**	**0.0002**
Img9	87.08	1	0.9799	0.0371	80.61	1	0.9862	0.0253	77.51	1	0.9917	0.0176	67.92	1	**0.9999**	**0.0002**
Img10	87.13	1	0.9798	0.0371	82.49	1	0.9827	0.0217	77.69	1	0.9929	0.0129	69.16	1	**0.9999**	**0.0002**
Img11	86.91	1	0.9800	0.0371	79.95	1	0.9918	0.0203	76.27	1	0.9903	0.0193	64.29	1	**0.9999**	**0**
Img12	87.09	1	0.9799	0.0371	80.62	1	0.9894	0.0296	74.18	1	0.9931	0.0153	58.93	1	**0.9998**	**0.0004**
Img13	87.07	1	0.9799	0.0371	81.59	1	0.9905	0.0229	72.12	1	0.9941	0.0136	63.59	1	**0.9999**	**0.0002**
Img14	87.14	1	0.9797	0.0371	78.27	1	0.9893	0.0301	74.18	1	0.9905	0.0241	61.73	1	**0.9997**	**0.0004**
Img15	87.09	1	0.9798	0.0371	79.27	1	0.9799	0.0298	74.14	1	0.9848	0.0253	60.37	1	**0.9995**	**0.0012**
Kaggle [30] Dataset Images																
Image	**With Random Γ**				**With GO**				**With BAT**				**With G-BAT**			
	PSNR	**SSIM**	**NC**	**BER**	**PSNR**	**SSIM**	**NC**	**BER**	**PSNR**	**SSIM**	**NC**	**BER**	**PSNR**	**SSIM**	**NC**	**BER**
2.1.08	Inf	1	0.9692	0.0503	89.28	1	0.9819	0.0341	79.17	1	0.9925	0.0194	72.18	1	**0.9997**	**0.0008**
2.1.09	Inf	1	0.9682	0.0503	87.18	1	0.9825	0.0361	81.28	1	0.9931	0.0162	78.92	1	**0.9995**	**0.0012**
2.1.10	Inf	1	0.9703	0.0503	88.21	1	0.9863	0.0314	83.21	1	0.9926	0.0123	74.19	1	**0.9996**	**0.0012**
2.1.11	Inf	1	0.9720	0.0503	89.71	1	0.9827	0.0316	82.84	1	0.9916	0.0113	70.17	1	**0.9995**	**0.0013**
2.1.12	Inf	1	0.9699	0.0503	88.61	1	0.9851	0.0381	82.81	1	0.9926	0.0202	75.72	1	**0.9993**	**0.0018**
2.2.01	Inf	1	0.9718	0.0503	87.19	1	0.9902	0.0302	81.27	1	0.9931	0.0167	72.16	1	**0.9994**	**0.0018**
2.2.02	Inf	1	0.9655	0.0503	84.81	1	0.9852	0.0395	79.74	1	0.9918	0.0206	68.28	1	**0.9992**	**0.0011**
2.2.03	Inf	1	0.9660	0.0503	89.71	1	0.9793	0.0426	82.65	1	0.9862	0.0184	76.13	1	**0.9992**	**0.0010**
2.2.04	Inf	1	0.9699	0.0503	88.01	1	0.9795	0.0397	81.54	1	0.9894	0.0183	78.63	1	**0.9993**	**0.0014**
2.2.05	Inf	1	0.9718	0.0503	88.92	1	0.9804	0.0399	83.18	1	0.9875	0.0267	75.73	1	**0.9993**	**0.0010**
2.2.06	Inf	1	0.9671	0.0503	89.90	1	0.9807	0.0403	84.28	1	0.9883	0.0204	74.82	1	**0.9992**	**0.0012**
2.2.07	Inf	1	0.9678	0.0503	85.10	1	0.9789	0.0294	79.27	1	0.9826	0.0196	71.33	1	**0.9993**	**0.0011**
2.2.08	Inf	1	0.9717	0.0503	84.82	1	0.9821	0.0391	75.19	1	0.9904	0.0271	69.37	1	**0.9994**	**0.0008**
2.2.09	Inf	1	0.9669	0.0503	82.64	1	0.9749	0.0396	77.26	1	0.9827	0.0292	68.65	1	**0.9993**	**0.0012**
2.2.10	Inf	1	0.9652	0.0503	81.62	1	0.9729	0.0392	79.86	1	0.9826	0.02892	72.48	1	**0.9992**	**0.0016**

4.2. Robustness Test

Robustness is another important requirement of the watermark scheme. Ideally, the original and extracted watermark should be identical. The robustness performance of the

proposed scheme was analyzed using NC and BER metrics. The relationship between NC and BER is shown in Equations (11) and (12).

$$\mathrm{NC} = \frac{\sum_{i=1}^{A}\sum_{j=1}^{B}[W(i,j) - W'(i,j)]^2}{\sqrt{\left[\sum_{i=1}^{A}\sum_{j=1}^{B}W(i,j)^2\right]} \times \sqrt{\left[\sum_{i=1}^{A}\sum_{j=1}^{B}W'(i,j)^2\right]}} \tag{11}$$

where W and W' are original and extracted watermarks.

$$\mathrm{BER} = \frac{\mathrm{EB}}{\mathrm{TB}} \tag{12}$$

$$\mathrm{EB} = \begin{cases} counter + 1 & if \sum_{i=1}^{A}\sum_{j=1}^{B}W(i,j) \neq W'(i,j) \\ 0 & otherwise \end{cases}$$

$$\mathrm{TB} = A \times B$$

where EB represents the number of incorrectly decoded bits in the extracted watermark, TB represents the total number of bits and the initial value of the counter = 0.

The robustness of our proposal was evaluated on 30 aerial RS images taken from the USC-SIPI [29] and Kaggle [30] datasets, as shown in Figure 8. The corresponding parameters NC, BER below zero random Γ attacks and optimized Γ (GA, BAT, G-BAT) are tabulated in Table 3. For the 30 aerial RS images, the proposed method shows a value of NC above the threshold and BER below the threshold with random Γ and optimized Γ. When compared to random Γ and optimized Γ, it can be observed that the robustness improves with optimized Γ. NC values are highest and BER is lowest with Γ optimized using the proposed G-BAT for all the aerial RS images. This observation demonstrates that greater robustness is achieved by using Γ optimized using G-BAT.

In addition, the robustness performance of the proposed scheme (under zero attack) was analyzed for the general images shown in Figure 7 and the corresponding NC. The BER is tabulated in Table 4 with random Γ, and with optimized Γ using GO, BAT and hybrid G-BAT. In Table 4 it can be seen that, with random Γ = 0.9 NC for all gray-scale and color images, the threshold value is above 0.7 [31], and the BER is less than the threshold value 0.5. To improve the robustness of the proposed scheme, the optimized Γ was built using GO, BAT and NC. The BER values are tabulated in Table 4. With the use of GO, the robustness of BAT was improved over the random Γ. For all gray-scale and color images, NC and BER with hybrid G-BAT are almost equal to the ideal value(s) of 1.0. As can be seen in Table 4, the proposed hybrid G-BAT improved the robustness performance (under zero attack) for gray-scale and color images from 0.92 to 0.99 and from 0.91 to 0.99, respectively. As can be seen in Tables 2 and 3, using optimized Γ shows a lower value PSNR than using random Γ, and keeps PSNR higher than threshold value 37 dB and the corresponding NC. BER reached the ideal value (1.0), as shown in Tables 3 and 4. Numerical comparisons of NC and BER for the 10 test aerial RS images with random Γ and optimized Γ are shown in Figures 9 and 10, respectively. It can be seen that the hybrid G-BAT Γ demonstrates greater robustness than the other optimized Γ. Thus, as revealed in Tables 2–4 and Figures 9 and 10, the hybrid G-BAT approach balances the trade-offs in the characteristics of DIW (i.e., imperceptibility and robustness).

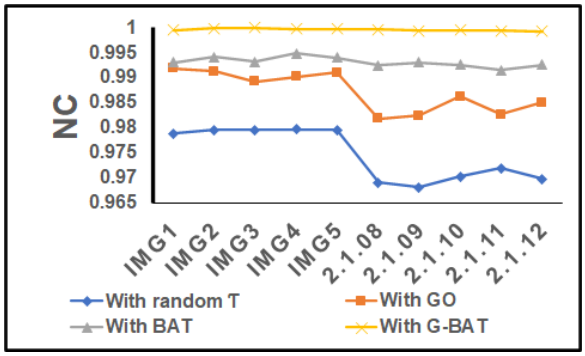

Figure 9. NC for the 10 test aerial cover images with random Γ, with GO, with BAT and with G-BAT under zero attacks.

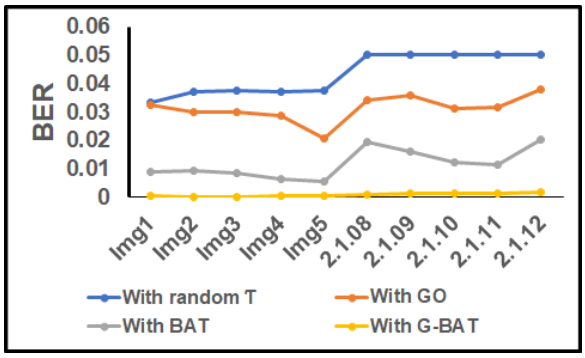

Figure 10. BER for the 10 test aerial cover images with random Γ, with Go, with BAT and with G-BAT under zero attacks.

Table 4. NC and BER for normal (gray-scale and color images) images with random $\Gamma = 0.9$ and with optimized Γ using GO, BAT and G-BAT under zero attacks.

Gray-Scale IMAGES								
Image	**With Random Γ = 0.9**		**With GO**		**With BAT**		**With G-BAT**	
	NC	**BER**	**NC**	**BER**	**NC**	**BER**	**NC**	**BER**
Baboon	0.9208	0.1084	0.9482	0.1062	0.9681	0.0089	**0.9993**	**0.0008**
Lena	0.9201	0.1116	0.9901	0.0197	0.9902	0.1092	**0.9995**	**0.0012**
Cameraman	0.9275	0.1076	0.9350	0.1051	0.9405	0.1062	**0.9894**	**0.0085**
Pirate	0.9205	0.1084	0.9968	0.1079	0.9974	0.0976	**0.9994**	**0.0012**
Living room	0.9208	0.1087	0.9797	0.1074	0.9683	0.1064	**0.9991**	**0.0010**
MRI Brain	0.9208	0.1081	0.9873	0.1076	0.9902	0.1062	**0.9993**	**0.0006**
Color Images								
Image	**With Random Γ = 0.9**		**With GO**		**With BAT**		**With G-BAT**	
	NC	**BER**	**NC**	**BER**	**NC**	**BER**	**NC**	**BER**
Baboon	0.9147	0.1182	0.9472	0.1103	0.9527	0.1095	**0.9990**	**0.0042**
Lighthouse	0.9147	0.1163	0.9381	0.1154	0.9294	0.1183	**0.9986**	**0.0088**
Peppers	0.9287	0.1194	0.9328	0.1119	0.9528	0.1104	**0.9993**	**0.0014**
Splash	0.9261	0.1173	0.9481	0.1103	0.9517	0.1100	**0.9992**	**0.0008**
Koala	0.9151	0.1160	0.9286	0.0953	0.9319	0.0915	**0.9988**	**0.0086**
Skin	0.9318	0.1121	0.9528	0.1101	0.9628	0.1086	**0.9997**	**0.0002**

The robustness performance of the proposed scheme under attacks was analyzed with random Γ and with optimized Γ using GO, BAT and G-BAT. The corresponding values of NC and BER are tabulated in Tables 5 and 6. In Tables 5 and 6 it can be seen that, under all attacks, G-BAT using optimized Γ shows a higher value of NC and BER than considering random Γ, or Γ optimized with GO or BAT.

In addition, the robustness of the proposed scheme was examined on three watermarked sampled aerial RS images against various image processing attacks, such as filtering attacks, sniffing attacks, compression and cut-off attacks with Γ optimized using G-BAT. Our proposal is robust against filtering, noise and some geometric attacks, since its NC and BER values (see Table 7) are higher than threshold values for all images (Img1, 2.1.01, 2.1.02). The proposed method revealed limited robustness performance on rotation, clipping and scaling attacks with random Γ. However, the robustness improves with Γ optimized using the proposed hybrid G-BAT. This observation demonstrates the effectiveness of using our proposal using the hybrid G-BAT to achieve higher robustness against most attacks, and higher stealth performance. Therefore, our design of G-BAT shows more imperceptibility and robustness.

Table 5. NC and BER for aerial RS images Img3 (taken from USC-SIPI [29]) under attacks with random Γ and with optimized Γ using GO, BAT and G-BAT.

Attack	**With Random Γ**		**With GO**		**With BAT**		**With G-BAT**	
	NC	**BER**	**NC**	**BER**	**NC**	**BER**	**NC**	**BER**
Sharpening	0.8895	0.2803	0.9217	0.1281	0.9728	0.01071	**0.9892**	**0.0794**
Gaussian filter (3 × 3)	0.9101	0.2163	0.9382	0.1286	0.9518	0.1082	**0.9993**	**0.0892**
Median filter (3 × 3)	0.9028	0.2518	0.9127	0.2107	0.9219	0.1128	**0.9509**	**0.0228**
Average filter (3 × 3)	0.8182	0.3286	0.8818	0.1729	0.9018	0.1384	**0.9493**	**0.0273**
Weiner filter (3 × 3)	0.7825	0.3918	0.8719	0.3017	0.8921	0.2061	**0.9785**	**0.0086**
Butterworth filter (G = 2, F = 20)	0.8828	0.2821	0.9156	0.2184	0.9418	0.0419	**0.9992**	**0.0012**
Salt & pepper (0.0002)	0.6692	0.4182	0.6985	0.3995	0.7928	0.2987	**0.9103**	**0.0698**
Gaussian noise	0.7382	0.2981	0.8129	0.1927	0.8528	0.1629	**0.9028**	**0.0518**
Speckle noise	0.7185	0.3276	0.7219	0.2916	0.7621	0.1986	**0.8828**	**0.0429**
Compression (60%)	0.8018	0.3281	0.8291	0.2281	0.8417	0.1192	**0.9105**	**0.0102**
Gamma correction (0.3)	0.8882	0.1719	0.9018	0.0917	0.8827	0.0281	**0.9945**	**0.0061**
Histogram equivalent	0.9592	0.2718	0.9401	0.2418	0.9702	0.1001	**0.9991**	**0.0064**
Shear	0.6519	0.3998	0.7019	0.3153	0.7663	0.1718	**0.8483**	**0.0615**
Row cut (10)	0.8716	0.3641	0.8941	0.3003	0.9142	0.1318	**0.9651**	**0.0063**
Column cut (10)	0.8852	0.2164	0.8931	0.1953	0.9172	0.0963	**0.9585**	**0.0084**
Rotation (10^0)	0.6318	0.4071	0.6528	0.3821	0.6629	0.3628	**0.7027**	**0.3017**
Scaling (0.5, 2)	0.6719	0.3715	0.6925	0.3514	0.7016	0.3217	**0.7182**	**0.3012**
Translate (0.25, 0.25)	0.8718	0.2614	0.8964	0.1915	0.9012	0.1216	**0.9126**	**0.1174**
Cropping	0.6056	0.5123	0.6515	0.3413	0.6414	0.3516	**0.7625**	**0.3016**

Table 6. NC and BER for Lena image under attacks with random Γ and with optimized Γ using GO, BAT and G-BAT.

Attack	With Random Γ		With GO		With BAT		With G-BAT	
	NC	BER	NC	BER	NC	BER	NC	BER
Sharpening	0.8829	0.2817	0.9161	0.1617	0.9672	0.0176	**0.9837**	**0.0881**
Gaussian filter (3 × 3)	0.9026	0.2185	0.9329	0.2086	0.9629	0.1718	**0.9991**	**0.0995**
Median filter (3 × 3)	0.8827	0.3016	0.9286	0.2894	0.9653	0.1286	**0.9931**	**0.0264**
Average filter (3 × 3)	0.8242	0.3172	0.8931	0.1842	0.9161	0.1281	**0.9519**	**0.0219**
Weiner filter (3 × 3)	0.7962	0.4271	0.8871	0.2715	0.9227	0.1852	**0.9728**	**0.0092**
Butterworth filter (G = 2, F = 20)	0.8731	0.2951	0.9042	0.2615	0.9318	0.0617	**0.9991**	**0.0012**
Salt & pepper (0.0002)	0.6318	0.5178	0.6935	0.5071	0.7418	0.4821	**0.9065**	**0.0762**
Gaussian noise	0.7194	0.3728	0.7418	0.2618	0.7726	0.1940	**0.8986**	**0.0721**
Speckle noise	0.7041	0.3718	0.7318	0.3015	0.7821	0.2518	**0.8731**	**0.0528**
Compression (60%)	0.7982	0.3517	0.8133	0.2185	0.8374	0.1027	**0.9082**	**0.0124**
Gamma correction (0.3)	0.8832	0.1842	0.8951	0.1372	0.9226	0.0264	**0.9921**	**0.0081**
Histogram equivalent	0.9521	0.2751	0.9427	0.2518	0.9671	0.1052	**0.9990**	**0.0071**
Shear	0.6417	0.4178	0.6682	0.3194	0.7327	0.2718	**0.8381**	**0.0861**
Row cut (10)	0.8618	0.3812	0.8817	0.3027	0.9026	0.1724	**0.9528**	**0.0075**
Column cut (10)	0.8863	0.2018	0.9021	0.1862	0.9261	0.0981	**0.9692**	**0.0029**
Rotation (10^0)	0.6281	0.4281	0.6381	0.4186	0.6528	0.3919	**0.7526**	**0.3672**
Scaling (0.5, 2)	0.6682	0.3819	0.6836	0.3729	0.6927	0.3317	**0.7091**	**0.3281**
Translate (0.25, 0.25)	0.8662	0.2718	0.8826	0.1829	0.8928	0.1286	**0.9071**	**0.1174**
Cropping	0.6219	0.4289	0.6487	0.3718	0.6518	0.2518	**0.7729**	**0.2926**

Table 7. NC and BER for aerial RS images (Img1, 2.1.01, and 2.1.02) taken from USC-SIPI [29] and Kaggle [30] datasets under attacks with optimized Γ using G-BAT.

Attacks	Img1		2.1.01		2.1.02	
	NC	BER	NC	BER	NC	BER
Sharpening	0.9867	0.1418	0.9842	0.0879	0.9921	0.0631
Gaussian filter (3 × 3)	0.9992	0.1078	0.9994	0.0085	0.9995	0.0064
Median filter (3 × 3)	0.9429	0.02819	0.9962	0.0221	0.9782	0.0185
Average filter (3 × 3)	0.9528	0.0221	0.9582	0.0221	0.9642	0.0201
Weiner filter (3 × 3)	0.9885	0.0065	0.9782	0.0069	0.9796	0.0059
Butterworth filter (G = 2, F = 20)	0.9991	0.0019	0.9994	0.0010	0.9995	0.0008
Salt & pepper (0.0002)	0.8907	0.08826	0.9105	0.0716	0.9108	0.0685
Gaussian noise	0.8927	0.0818	0.9021	0.0702	0.9281	0.0521
Speckle noise	0.8828	0.0796	0.8921	0.0491	0.8985	0.0384
Compression (60%)	0.9192	0.0119	0.9087	0.0131	0.9105	0.01301
Gamma correction (0.3)	0.9928	0.0069	0.9942	0.0086	0.9962	0.0076
Histogram equivalent	0.9987	0.0075	0.9986	0.0083	0.9990	0.0075
Shear	0.8164	0.0834	0.8291	0.0827	0.8392	0.0792
Row cut (10)	0.9692	0.0063	0.9631	0.0062	0.9684	0.0058
Column cut (10)	0.9719	0.0077	0.9728	0.0063	0.9785	0.0053
Rotation (10^0)	0.7072	0.2918	0.6951	0.3061	0.7051	0.2896
Scaling (0.5,2)	0.7132	0.3941	0.7095	0.3386	0.7196	0.3172
Translate (0.25,0.25)	0.9205	0.1062	0.9077	0.1173	0.9142	0.1023
Cropping	0.7562	0.3613	0.7718	0.3913	0.7821	0.3872

4.3. Security Test

The security of the proposed encryption approach was tested using the correlation coefficient (CC) and entropy metrics [32] on five binary test images, and the experimental results are shown in Table 8. The value of CC between any two images is in the range from −1 to 1. If the value of CC is 1, it means that both images are identical, and if the value of CC is 0, it stands that both images are irrelevant. Otherwise, both images are negatively or positively correlated. In Table 8 it can be seen that, for all test images, the values of CC between the original encrypted images in the horizontal, vertical and diagonal directions are less than −0.6. The latter indicates that both the original and encrypted images are negatively correlated. When the value of CC between original and decrypted images is 1, it indicates that they are identical. Additionally, security performance is analyzed by entropy, as shown in Table 8. For all test images, the entropy value of the encrypted image is higher than that of the original image. This also indicates that the proposed method generates an encrypted image with different entropies. From the above discussion, it can be stated that the proposed 3-level encryption scheme generates stronger encryption images and decrypts successfully as well.

4.4. Computational Time

Computational time is the amount of time required to complete an embedding and extraction process. Computational times for the embedding and extraction in seconds for different test images are tabulated in Table 9. In Table 9, it can be seen that the embedding and extraction times for all gray-scale images were less than 0.16 and 0.07 s, and for the color images they were less than 0.26 and 0.09 s, respectively. The variation in gray-scale and color images was minimal. Therefore, the computational time of our DIW method can be considered minimal, since the watermark was embedded and extracted in less than 0.3 and 0.09 s, respectively.

Table 8. CC values between original watermark image (W) and encrypted images (EW). CC values between original image and decrypted images (DW) and entropies of original image and encrypted images (H—horizontal, V—vertical, D—diagonal).

Image	CC between W, EW			CC between W, DW			Entropy of	Entropy of
	H	V	D	H	V	D	W	EW
Watermark	−0.2838	−0.2665	−0.2715	1	1	1	0.8930	1.0723
Cameraman	−0.5248	−0.4789	−0.4750	1	1	1	0.9880	0.9985
Lena	−0.3430	−0.2804	−0.2542	1	1	1	0.7194	0.9671
Baboon	−0.5231	−0.5669	−0.5494	1	1	1	0.9960	1.8954
MRI Brain	−0.6319	−0.5724	−0.5633	1	1	1	0.9979	1.0276

Table 9. Embedding and extraction time (in seconds) of different test cover images.

Gray-Scale Image	Embedding Time	Extraction Time	Color Image	Embedding Time	Extraction Time
Lena	0.150429	0.055273	Baboon	0.269723	0.095462
Cameraman	0.150136	0.065511	Peppers	0.240605	0.077382
Img1	0.156776	0.057896	2.1.08	0.250033	0.087393
Img2	0.151721	0.056821	2.1.09	0.239758	0.075609
Img3	0.150708	0.055821	2.1.10	0.241862	0.082617

4.5. Comparative Study

In this section, the performance of the proposed DIW method is compared against those considered state-of-the-art in terms of imperceptibility, robustness, embeddability and security: Ali and Nasab [14], Preethi and Kishore [15], Zhu et al. [16], Ali [17] and

Singh et al. [18]. Ali and Nasab [14] proposed a SWT and SURF transformation method for robust image transmission. The Arnold map was suggested for security and BAT optimization for scaling factor optimization. An ABC optimization method was proposed by Preethi and Kishore [15] in the DWT+DCT domain for robust image transmission. The IWT+SVD hybrid was suggested by Zhu et al. [16]. In particular, GA is used to optimize the scaling factor and affine transformation for security. Ali [17] proposed the DWT+SVD DIW scheme for robust image transmission. Recently, Singh et al. [18] contributed a IWT+SVD method in the hybrid domain for robust image transmission. This latter scheme uses a pseudo-random key for encryption. A comparative overview against other state-of-the-art methods is explained in Table 10. In Table 10 it can be seen that the DIW scheme proposed in [15,17] has overlooked watermark security, and the schemes proposed in [14,16] uses the Arnold map and Affine transformations for safety. The Arnold map and affine transformations provide less security and can be easily cracked. The scheme proposed in [18] uses a pseudo-random key approach for encryption, and the key is generated with high computational cost. Our proposed scheme guarantees security at three levels, shuffling, substitution and partitioning, at a low computational cost. Compared to the state-of-the-art methods proposed in [14–17], our proposal of encryption scheme shows better watermark security. Compared to the scheme in [18], our scheme provides high security at a low computational cost. Moreover, the method proposed in this paper provides a higher incorporation capacity than all other next-generation schemes ([14–18]), as analyzed in Table 10.

Furthermore, the imperceptibility performance of our DIW method was tested with the comparison schemes for Lena and Baboon, as shown in Figure 11. The PSNR of the Lena and the Baboon images treated with the proposed method are higher than those of all tested schemes [14–18]. The robustness performance under zero attacks is shown in Figure 12. In Figure 12, it can be seen that the our DIW scheme has a higher NC than the scheme presented in [16], and it has an almost equal NC to the schemes of [14–16,18]. The additional robustness of our proposal was tested under different image processing attacks, such as Gaussian filter (GF), median filter (MF), sharpness (SHARP), salt and pepper noise (SP), histogram equivalent (HQ), gamma correction (GC) and JPEG compression (JPEG), as shown in Figure 13.

In Figure 13, it can be seen that the proposed method shows greater robustness when considering filter attacks (GF, MF, and SHARP) than the schemes in [16,17]. Additionally, it shows robustness performance almost equal to those of the schemes proposed in [14,15,18]. For JPEG attack, the proposed scheme had better performance than the schemes of [14], but lagged in comparison to the schemes presented in [15–17]. However, the proposed scheme had an NC ≈ 0.92, indicating higher robustness against JPEG attack. For noise attacks, the proposed scheme showed greater robustness than the schemes proposed in [16,18] and was almost equally robust as the schemes proposed in [14,15,17]. The proposed scheme showed limited performance against noise attacks, which can be seen as future work. For HQ and GC attacks, the proposed scheme is superior to the schemes in [14–16,18]. The comparative analysis revealed that the proposed scheme reports higher than or the same robustness as the other schemes. Furthermore, it revealed high imperceptibility, embedding ability and security compared to the other schemes ([14–17]).

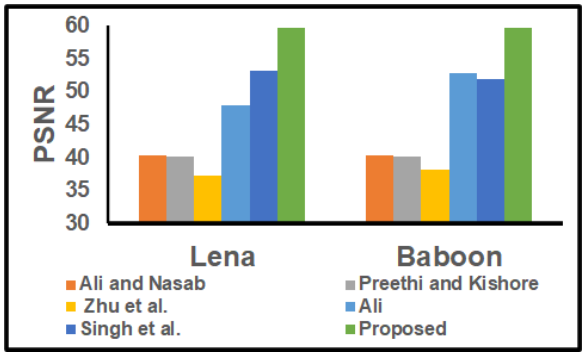

Figure 11. PSNR comparison of our method against state-of-the-art methods for Lena and Baboon images.

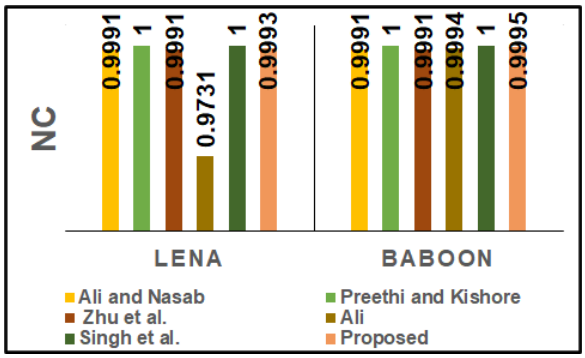

Figure 12. NC comparison of our method against state-of-the-art methods for Lena and Baboon images under zero attacks.

The scheme of Singh et al. [18] showed higher imperceptibility, robustness and embedding capacity with an adaptively generated ISF scaling factor. Moreover, the authors suggested using NIO algorithms (GA, ABC and FO) to improve watermarking characteristics. The GA approach showed higher imperceptibility and robustness. For a fair comparison, in this paper, GA generated PSNR and NC values according to those in Singh et al. [18]. As shown in Figure 11 and Table 10, the PSNR and the embedding capacity of our proposed DIW method are higher than those of the scheme [18]. The computational times for the embedding and extraction process of Singh et al. [18] and our proposal for different images (Lena, Baboon and Koala) are analyzed next. The scheme of Singh et al. [18] achieved embedding times of (in seconds) 1.919606, 1.465614 and 1.378239 s for Lena, Baboon and Koala images, respectively, whereas our method took 0.150429, 0.269723, 0.240605 s. Likewise, the computational times by the method of Singh et al. [18] regarding extraction were 1.056093, 0.915152 and 0.98756 s, whereas our proposal achieved 0.055273, 0.095462 and 0.077382 s for the same images. From this, it can be seen that our proposal takes less time to embed and extract images for all those images. Therefore, the computational time reported by our hybrid G-BAT DIW method is lower than that of the GA-based method in [18]. Moreover, our method brings NC closer to the ideal value. From the above observations and discussions, it can be stated that the proposed scheme outperformed all the state-of-the-art methods in [14–18].

Table 10. Comparison analysis of our proposal against those considered the state-of-the-art.

Parameter	Ali and Nasab [14]	Preeti and Kishore [15]	Zhu et al. [16]	Ali [17]	Singh et al. [18]	Proposed
Transformation scheme	SWT+SURF	DWT+DCT	IWT+SVD	DWT+SVD	IWT_ SVD	RDWT+SVD
Cover image size	512 × 512	512 × 512	512 × 512	512 × 512	512 × 512	256 × 256
Watermark size	32 × 32	64 × 64	32 × 32	256 × 256	64 × 64	64 × 64
Optimization algorithm	BAT	ABC	GA	NO	GA, ABC, FO	Hybrid G-BAT
Security technique	Arnold map	No	Affine transform	NO	Pseudo random key	3 level security
Embedding capacity	0.0039	0.01562	0.0039	0.25	0.01562	**0.0625**

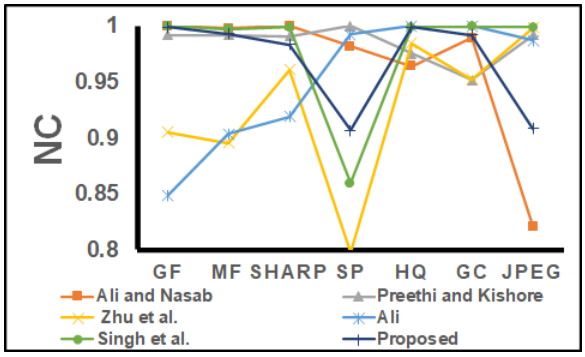

Figure 13. Robustness comparison of our method against those in the state-of-the-art for the Lena image under attacks.

5. Conclusions and Future Work

In this article, a RDWT-SVD-based DIW scheme was proposed for secure transmission of aerial remote sensing images on the Internet. A hybrid G-BAT optimization algorithm was proposed for optimizing the scaling factor (Γ). An optimized scaling factor was used as an embedding advantage for watermark embedding to balance the watermarking characteristics trade-off. Further, to ensure high watermark security at with little computation, a 3-level encryption approach was proposed. The performance of the proposed DIW scheme was analyzed with random Γ, optimized Γ (using NIO algorithms, i.e., GO and BAT) and optimized Γ using the proposed hybrid G-BAT. Experimental results show that the proposed DIW scheme has superior imperceptibility and robustness with the Γ obtained by G-BAT optimization. Thus, the proposed scheme effectively balances watermarking characteristics. It was observed that original and encrypted watermarks were highly uncorrelated, whereas original and decrypted watermarks were highly correlated, indicating higher watermark security. Furthermore, the performance of the proposed scheme has been compared with those of recent state-of-the-art DIW schemes. The proposed DIW scheme reported comparatively higher performance in terms of imperceptibility, robustness, security and embedding capacity. Experimental results and a comparative study validated the effectiveness of the proposed DIW scheme. It can be successfully used for for copyright protection, ownership verification, image authentication and image security when remote sensing images are transferred over the Internet. The proposed scheme has limited robustness against geometric attacks such as rotation and clipping attacks. Improving the robustness of the proposed scheme against geometric attacks could be seen as future research work.

Author Contributions: Conceptualization and methodology, K.J.D., P.S., H.K.T., J.S. and J.K.D.; software, K.J.D. and P.S.; validation and formal analysis, H.K.T., J.S., J.K.D. and A.R.-M.; investigation and resources, P.S., K.J.D. and M.V.J.K.; writing original draft preparation, P.S. and K.J.D.; writing review and editing, J.S., H.K.T., J.K.D. and A.R-M.; supervision, P.S., H.K.T., J.S. and M.V.J.K.; project administration, P.S., K.J.D. and H.K.T. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Jiang, H.; Peng, M.; Zhong, Y.; Xie, H.; Hao, Z.; Lin, J.; Ma, X.; Hu, X. A Survey on Deep Learning-Based Change Detection from High-Resolution Remote Sensing Images. *Remote Sens.* **2022**, *14*, 1552. [CrossRef]
2. Evsutin, O.; Dzhanashia, K. Watermarking schemes for digital images: Robustness overview. *Signal Process. Image Commun.* **2022**, *100*, 116523. [CrossRef]
3. Zainol, Z.; Teh, J.S.; Alawida, M. Alabdulatif A Hybrid SVD-based image watermarking schemes: A review. *IEEE Access* **2021**, *9*, 32931–32968.
4. Gendreau, M.; Potvin, J.Y. (Eds.) *Handbook of Metaheuristics*; Springer: New York, NY, USA, 2010; Volume 2.9.
5. Singh, O.P.; Singh, A.K.; Srivastava, G.; Kumar, N. Image watermarking using soft computing techniques: A comprehensive survey. *Multimed. Tools Appl.* **2021**, *80*, 30367–30398. [CrossRef]
6. Khanduja, N.; Bhushan, B. Recent advances and application of metaheuristic algorithms: A survey (2014–2020). In *Metaheuristic and Evolutionary Computation: Algorithms and Applications*; Springer: Singapore, 2021; pp. 207–228.
7. Shankar, R.; Vara Prasad, R.U.; Adiraju, R.V.; Krishna, R.V.; Nandan, D. A Review Paper Based on Image Security Using Watermarking. In Proceedings of the International Conference on Recent Trends in Machine Learning, IoT, Smart Cities and Applications, Hyderabad, India, 28–29 March 2021; Springer: Singapore, 2021; pp. 697–706.
8. Su, Q.; Liu, D.; Yuan, Z.; Wang, G.; Zhang, X.; Chen, B.; Yao, T. New rapid and robust color image watermarking technique in spatial domain. *IEEE Access* **2019**, *7*, 30398–30409. [CrossRef]
9. Abraham, J.; Paul, V. An imperceptible spatial domain color image watermarking scheme. *J. King Saud. Univ.-Comput. Inf. Sci.* **2019**, *31*, 125–133. [CrossRef]
10. Kunhu, A.; Mansoori, S.A.; Al-Ahmad, H. A Novel Reversible Watermarking Scheme Based on SHA3 for Copyright Protection and Integrity of Satellite Imagery. *Int. J. Comput. Sci. Netw. Secur.* **2019**, *19*, 92–102.
11. Mahmoud, K.; Datta, S.; Flint, J. Frequency Domain Watermarking: An Overview. *Int. Arab J. Inf. Technol.* **2005**, *2*, 33–47.
12. Kang, X.; Chen, Y.; Zhao, F.; Lin, G. Multi-dimensional particle swarm optimization for robust blind image watermarking using intertwining logistic map and hybrid domain. *Soft Comput.* **2020**, *24*, 10561–10584. [CrossRef]
13. Tewari, T.K.; Saxena, V. An improved and robust DCT based digital image watermarking scheme. *Int. J. Comput. Appl.* **2020**, *3*, 28–32. [CrossRef]
14. Pourhadi, A.; Mahdavi-Nasab, H. A robust digital image watermarking scheme based on bat algorithm optimization and SURF detector in SWT domain. *Multimed. Tools Appl.* **2020**, *79*, 21653–21677. [CrossRef]
15. Garg, P.; Kishore, R.R. An efficient and secured blind image watermarking using ABC optimization in DWT and DCT domain. *Multimed. Tools Appl.* **2021**, 1–18.
16. Zhu, T.; Qu, W.; Cao, W. An optimized image watermarking algorithm based on SVD and IWT. *J. Supercomput.* **2021**, *78*, 222–237. [CrossRef]
17. Alzahrani, A. Enhanced Invisibility and Robustness of Digital Image Watermarking Based on DWT-SVD. *Appl. Bionics Biomech.* **2022**, *2022*, 5271600. [CrossRef] [PubMed]
18. Singh, P.; Devi, K.J.; Thakkar, H.K.; Santamaría, J. Blind and Secured Adaptive Digital Image Watermarking Approach for High Imperceptibility and Robustness. *Entropy* **2021**, *23*, 1650. [CrossRef]
19. Zhu, P.; Jiang, Z.; Zhang, J.; Zhang, Y.; Peng, W. Remote sensing image watermarking based on motion blur degeneration and restoration model. *Optik* **2021**, *248*, 168018. [CrossRef]
20. Yuan, G.; Hao, Q. Digital watermarking secure scheme for remote sensing image protection. *China Commun.* **2020**, *17*, 88–98. [CrossRef]
21. Tong, D.; Ren, N.; Zhu, C. Secure and robust watermarking algorithm for remote sensing images based on compressive sensing. *Multimed. Tools Appl.* **2019**, *78*, 16053–16076. [CrossRef]
22. Mohan, A.; Anand, A.; Singh, A.K.; Dwivedi, R.; Kumar, B. Selective encryption and optimization based watermarking for robust transmission of landslide images. *Comput. Electr. Eng.* **2021**, *95*, 107385. [CrossRef]

23. Hsu, P.H.; Chen, C.C. A robust digital watermarking algorithm for copyright protection of aerial photogrammetric images. *Photogramm. Rec.* **2016**, *31*, 51–70. [CrossRef]
24. Mascagni, M.; Srinivasan, A. Algorithm 806: SPRNG: A scalable library for pseudorandom number generation. *ACM Trans. Math. Softw. (TOMS)* **2000**, *26*, 436–461. [CrossRef]
25. Saremi, S.; Mirjalili, S.; Lewis, A. Grasshopper optimisation algorithm: Theory and application. *Adv. Eng. Softw.* **2017**, *105*, 30–47. [CrossRef]
26. Ewees, A.A.; Elaziz, M.A.; Houssein, E.H. Improved grasshopper optimization algorithm using opposition-based learning. *Expert Syst. Appl.* **2018**, *112*, 156–172. [CrossRef]
27. Yang, X.-S.; He, X. Bat algorithm: Literature review and applications. *Int. J. Bio-Inspired Comput.* **2013**, *5*, 141–149. [CrossRef]
28. Yang, X.S. A New Metaheuristic Bat-Inspired Algorithm. In *Nature-Inspired Cooperative Strategies for Optimization (NICSO 2010)*; Studies in Computational Intelligence; Springer: Berlin/Heidelberg, Germany, 2010; pp. 65–74.
29. The USC-SIPI Image Database. Available online: https:/sipi.usc.edu/database/ (accessed on 20 January 2022).
30. The Kaggle Image Database. Available online: https://www.kaggle.com/datasets (accessed on 20 January 2022).
31. Rao, Y.R.; Prathapani, N.; Nagabhooshanam, E. Application of normalized cross correlation to image registration. *Int. J. Res. Eng. Technol.* **2014**, *3*, 12–16.
32. Balaska, N.; Ahmida, Z.; Belmeguenai, A.; Boumerdassi, S. Image encryption using a combination of Grain-128a algorithm and Zaslavsky chaotic map. *IET Image Process.* **2020**, *14*, 1120–1131. [CrossRef]