Robust Secure Color Image Watermarking Using 4D Hyperchaotic System, DWT, HbD, and SVD Based on Improved FOA Algorithm

Hira Nazir, Imran Sarwar Bajwa, Muhammad Samiullah, Waheed Anwar, and Muhammad Moosa

Department of Computer Science & IT, The Islamia University of Bahawalpur, Bahawalpur, Pakistan

Correspondence should be addressed to Imran Sarwar Bajwa; imran.sarwar@iub.edu.pk

Received 25 December 2020; Revised 1 February 2021; Accepted 12 February 2021; Published 28 February 2021

1. Introduction

In the age of cloud computing and for computing, the security of data whether the sensor’s data or cloud’s data has become the dire need of today’s age to secure it from malicious attacks. Similarly, copyright infringement problems and illegal distribution and modification while disseminating information over the Internet may arise quite frequently [1, 2]. Therefore, watermarking coupled with hyperchaotic encryption can cope up with the emerging challenges of copyright infringement, watermarking attacks, and security issues. In watermarking, a logo or secret message is hidden in the host image on the transmitting side while this logo or secret message is extracted at receiving side in order to judge the digital ownership of the received data. With the advancement of computing such as DNA and quantum-based computing, the probability to breach currently highly secured watermarks may also increase. The techniques such as HbD, DWT, and SVD have been widely used by researchers in various watermarking methods to watermark the grayscale and color images. The trade-off between invisibility and robustness has always been a challenging issue in watermarking methods and it needs optimization.

Recently, several algorithms such as the firefly algorithm [3], artificial bee colony (ABC) [4], and particle swarm and fruit fly optimization algorithms [5, 6] are employed to optimize the watermarking technique. The problem of entrusting the watermarking to cloud service provider is addressed in [7], in which the authors made the following contributions: (1) modern public-key cryptosystems are employed to avoid the associated security hazards and implementation costs of key exchange are also considered, (2) reversible watermarking techniques compatible with homomorphic cryptosystems are studied, (3) storage efficiency is studied by encrypting a long sequence of bits, (4)
data preprocessing prior to encryption is not required, and (5) both offline and online content-adaptive predictors are developed for various operational requirements. The proposed schemes achieve a remarkable balance between fidelity and reversibility under the given capacity constraints. Moreover, it significantly reduces the size of the encrypted data and improves the space efficiency. Most of the existing watermarking techniques suffer from certain watermarking attacks, are not optimized, and are not coupled with hyperchaotic maps. A few studies have been published on watermarking followed by hyperchaotic encryption [8]. To this end, a novel watermarking technique by exploiting the interblock coefficient correlation for embedding the watermark is proposed by [9], in which chaos and Arnold transform is used for improving security. The modifications are done in such a way that image processing and geometric attacks are resisted. Furthermore, it is testified that watermarking based on DWT has certain advantages such as good compression and imperceptibility; however, DWT-based watermarking schemes are not too much robust against geometric attacks [10]. Therefore, in order to make the scheme more robust against image processing and geometric attacks, matrix decomposition such as SVD and Hbd is commonly used. The SVD-based schemes decompose the transformed host image into three vectors called U, S, and V.

The digital watermark can be embedded into U or S or V. The S matrix is mostly used for watermark embedding owing to its robust nature against attacks [11]. Additionally, a little change in singular values does not influence the visual quality of the host image. On another note, FPP arises when singular values are used for watermark insertion. The matrices U and V can be replaced by the attacker’s desired matrices for the extraction of a new watermark (that has never been inserted) to profess the false ownership. Computer science researchers have proposed the change in singular values with the help of scaling factor to control the strength of digital watermark to be embedded as shown in Sections 4.2 and 4.3 (Eq. (13) and Algorithm 3). The scaling factor can be further optimized by using different algorithms such as particle swarm and improved fruit fly optimization algorithms and bioinspired computing algorithms [5, 6, 12]. The FPP can be solved by encrypting the SVD components by using hyperchaotic systems or by using the one-way hash functions [13, 14]. Hyperchaotic encryption owing to excellent security results is the main source of strong security; i.e., the FPP can be solved. For example, the author in [15] verified the better confusion and diffusion by using the 5D hyperchaotic map to create secret keys for encryption and decryption. The initial parameters for 5D hyperchaotic are tuned by using the dual local search based multiobjective optimization, and the encryption architecture is based on two levels of permutation and diffusion. Similarly, the authors in [16–18] also used the hyperchaotic maps in a novel way for encrypting the images and obtained better results.

Specifically, in this paper, a novel digital watermarking method consisting of DWT, Hbd, and SVD based on hyperchaotic encryption, gauging function (GF), and improved evolution fruit fly algorithm (IEFOA) is proposed. Specifically, GF abets IEFOA to find the optimal scaling factor α, for balancing the trade-off between imperceptibility and robustness, while hyperchaotic encryption of watermark before the use of SVD and chaotic encryption of SVD components solves the FPP effectively at a less computational cost. The main contributions of this paper include the following: (1) scheme has shown a good balance of trade-off even with the multiple size watermarks, (2) robustness is improved by coefficient modification through Hbd, (3) encryption of color watermark by the 4D hyperchaotic system before SVD procedure and chaotic encryption of SVD components is also applied to make the scheme more secure, and (4) GF and IEFOA are employed to help in finding the optimal scaling factor.

The proposed work is organized as follows. Section 2 gives the related work, Section 3 highlights the preliminaries, Section 4 presents the proposed scheme, and Section 5 contains experimental results and analysis. Concluding remarks with future directions are given in Section 6.

2. Related Work

This section deals with the earlier research work done in designing color watermark embedding and extracting schemes. The list of abbreviations used in this study is shown in Table 1. Imperceptible and robust digital watermarking schemes can be a potential solution for the privacy and security of sensitive information such as Electronic Patient Records (EPRs). To this end, a combination of fast curvelet transform and SVD embeds watermark (EPR) after encoding into patient’s healthy and diseased optical coherence tomography (OCT) scans [19]; this scheme has shown a high level of imperceptibility, robustness, and security of EPRs as compared to existing watermarking schemes. A digital watermark protocol proposed by [20] solves the false-positive problem by using a chaotic Kbest gravitational search algorithm in two domains, i.e., SVD and DCT. An efficient watermarking scheme in terms of imperceptibility, security, and robustness proposed by [21] embeds the watermark by Fractional Moments of Charlier–Meixner. The proposed method by [22] achieves robustness against geometric and filtering attacks and shows a better trade-off among robustness and distortion than the state-of-the-art methods. The proposed watermarking scheme in [10] uses a double encryption method based on fractional Fourier transform and DCT in the hybrid wavelet domain. The author in this scheme used multiparameter particle swarm optimization (MP-PSO) for obtaining the optimized embedding factors and reveals high security and invisibility and is robust against geometrical attacks. A robust and secure watermarking scheme to improve the management of medical images is presented in [23]. In this scheme, the techniques of invisible and zero watermarking avoid the detachment between medical images and EPRs and provide authenticity for the identification of patients. Another digital watermarking scheme comprises six modules (level shifting, mixed modulation, sign correlation, orthogonal restoration, distortion compensation, and iterative regulation) that overwhelm the inadequacies of existing SVD-based watermarking schemes while improving
DIEHARD, and ENT test suites. To perform the pseudorandom number generator after testing it on NIST, a chaotic sequence produced by Henon Map can be used as a robustness [30]. The authors in this scheme suggested that the watermark has good imperceptibility and acceptable robustness to some extent. Combining IWT, DWT, contourlet transform, and 3D Henon Map in embedding and extracting watermark has good imperceptibility and acceptable robustness [30]. The authors in this scheme suggested that the chaotic sequence produced by Henon Map can be used as a pseudorandom number generator after testing it on NIST, DIEHARD, and ENT test suites. To perform the watermarking, the authors in [31] divided the algorithm into four phases called image scaling, block separation by DCT, feature vector computation, watermark spotting regions, message transformation, watermark embedding, IDCT, and message restoration followed by an optimized FCM clustering with Least Favorable Whale Optimization Algorithm based watermarking scheme and obtained the effective results in terms of robustness and invisibility. A substitution scheme for RGB images watermarking based on Fourier transform is proposed in [32]. In this approach, several variants of Fourier transforms are applied to R, G, and B components of an image separately, the watermark is embedded in medium frequency band based on the combined parity of coefficients, and the obtained results are satisfactory in terms of average PSNR greater than 40 decibels for integration into a variant of Fourier transform coefficients.

Another blind image watermarking scheme in the transform domain, where there is no need for a watermark and host image for extracting the watermark, gives good imperceptibility and robustness with less computational cost [33]. In this scheme, the host image is split into nonoverlapping blocks each of size 8 x 8, and DCT coefficients of each block are computed; then, two datasets (d1 and d2) are created from the selected blocks, and DCT coefficients of d1 and d2 are compared with the prefixed threshold values (k1 and k2) as follows: if the watermark bit value is 1, then corresponding d1 and d2 coefficient values are modified with set α value; else, the corresponding d1 and d2 coefficient values are set to zero.

### 3. Preliminaries

Hessenberg decomposition (HbD) is a transformation of the square matrix A into the unitary matrix Q and Hessenberg matrix H such that A = QHQ^2, computed by household matrices, and aids in improving the watermark invisibility [34]. To this end, watermarking based on R level DWT, HbD, SVD, logistic map, and optimization based on FOA through objective evaluation function showed a good trade-off between robustness and invisibility [13]. This scheme can further be improved by using improved FOAs.

Although basic FOA [6] has advantages including fewer parameters and simple principles but has shortcomings such as local optimization, lack of robustness, and slow convergence that can be overcome by IEFOA [35]. The inclusion of two parameters called step control denoted by λ and evolution/elimination control (ec) in IEFOA makes it different and provides an advantage over basic FOA. In basic FOA, the number of iterations in which the algorithm needs to find an optimal solution is the main drawback. In the early stage of iterations with the vast domain, a small search radius (search step) makes basic FOA weak to approach the optimal solution. In the final stage of iterations when the swarm location is close to an optimal solution, a very small scope is a better option for fine-tuning solution vectors. Therefore, a search radius with the big to small (BS) feature may overcome this drawback. The (BS) feature means that a big search step in the early stage can refine the global search ability and a small search step in end stage can refine the local search.
ability by determining the scale of step for each fruit fly flexibly. Step control parameter (λ) provides the (BS) feature and can be expressed as

\[
\lambda = \lambda_{\text{max}} \times \exp\left[\log\left(\frac{\lambda_{\text{min}}}{\lambda_{\text{max}}}\right) \frac{\text{Iter}}{\text{Iter}_{\text{max}}}ight],
\]

(1)

where \(\lambda\) is the search radius in each iteration, while \(\lambda_{\text{min}}, \lambda_{\text{max}},\) and \(\text{Iter}\) are the minimum radius, maximum radius, and iteration number, respectively. The fruit fly gets a bigger search ability by determining the scale of step and hence eludes falling in local optimum value, while in the later iterations, \(\lambda\) decreases slower than linear decreasing.

The second parameter is called elimination parameter \(e_{v}\), in which the inferior swarm is eliminated and the dominant swarm is saved. The \(e_{c}\) can be expressed as

\[
e_{c} = 1 - e_{l},
\]

(2)

where \(e_{l}\) is the elimination coefficient and can be defined as

\[
e_{l} = e_{l_{\text{max}}} \times \exp\left[\log\left(\frac{e_{l_{\text{min}}}}{e_{l_{\text{max}}}}\right) \frac{\text{Iter}}{\text{Iter}_{\text{max}}}ight],
\]

(3)

where \(e_{l_{\text{min}}}, e_{l_{\text{max}}}, \text{Iter},\) and \(\text{Iter}_{\text{max}}\) are the minimum elimination coefficient, maximum elimination coefficient, iteration number, and maximum iteration number. Many bad performance swarms are removed as the search starts and the remaining advanced fly swarms will produce a new population. The repetitive process of swarm elimination will lead to the preservation of only a few swarms. The elimination procedure offers the advantage of letting IEFOA jump out of the local extremum (an extreme point having maximum or minimum value) to find a better global optimum. The beauty of IEFOA is the fact that it not only adopts \(\lambda\) but also segregates the inferior swarms by using \(e_{c}\).

The main process of IEFOA can be illustrated as follows:

**Step 1.** Randomly generate multiple swarms’ center locations.

**Step 2.** Generate N new swarms; PSF in each swarm represents the population size according to the update rule of the Osphrines foraging stage.

**Step 3.** The optimal fruit fly is selected in each swarm as a new center location by vision foraging phase according to the fitness function value (fval).

**Step 4.** Center locations of all the new swarms are sorted in ascending order according to their fval.

**Step 5.** A certain number of inferior swarms are eliminated; the remaining dominant swarms become the next iteration swarm center locations according to the coefficient of \(e_{l}\) and the number of swarm locations at present.

**Step 6.** Repeat Steps 2 to 5 till the satisfaction of termination condition. The global optimum is only obtained when the optimized process is terminated.

### 4. Proposed Scheme

The watermark encryption algorithm is introduced in Section 4.1 and the embedding algorithm is introduced in Section 4.2, while the extraction and decryption algorithm is introduced in Sections 4.3 and 4.4. Optimization of the proposed watermarking method to achieve the trade-off between invisibility and robustness is given in Section 4.5. The flowchart of the proposed scheme is given in Figure 1.

#### 4.1. Watermark Encryption.

A color watermark of multiple sizes \((N \times N)\), where \(N = 2, 4, 8, 16, 32, 128, 256, 512\) is input to the watermark encryption algorithm. Initial conditions based on the DNA sequence taken from the NCBI dataset are calculated. External key \(x_{K}\) is extracted from the DNA sequence taken from the NCBI dataset. For example, we downloaded a DNA sequence of some animals having a length of 183015. The mean intensity value of the watermark image is used as a starting index to cut the DNA sequence from this location having a length of 128. After cutting the DNA sequence of length 128, each nucleotide base is converted into a two-bit binary equivalent according to the DNA mapping rules [36], shown in Table 2, which meet the Watson–Crick complement rule. In this way, a certain number of inferior swarms are eliminated; the remaining dominant swarms become the next iteration swarm center locations according to the coefficient of \(e_{l}\) and the number of swarm locations at present.

The initial conditions using \(x_{K}\) are computed as follows:

\[
\text{bin}_{K} = \{g_1, g_2, \ldots, g_{32}\}.
\]

(4)

Now, the initial conditions using \(\text{bin}_{K}\) are computed as follows:

\[
\begin{align*}
\text{x}(0) &= \frac{(g_1 \oplus g_2 \oplus g_3 \oplus g_4 \oplus g_5 \oplus g_6 \oplus g_7 \oplus g_8)}{256} \\
y(0) &= \frac{(g_9 \oplus g_{10} \oplus g_{11} \oplus g_{12} \oplus g_{13} \oplus g_{14} \oplus g_{15} \oplus g_{16})}{256} \\
z(0) &= \frac{(g_{17} \oplus g_{18} \oplus g_{19} \oplus g_{20} \oplus g_{21} \oplus g_{22} \oplus g_{23} \oplus g_{24})}{256} \\
u(0) &= \frac{(g_{25} \oplus g_{26} \oplus g_{27} \oplus g_{28} \oplus g_{29} \oplus g_{30} \oplus g_{31} \oplus g_{32})}{256}.
\end{align*}
\]

(5)

Initial conditions with control parameters \((a, b, c, d, e)\) are input to the 4D hyperchaotic system (Equation (1)). The 4D hyperchaotic at any given initial conditions with control parameters

\[(a = 27.5, b = 3, c = 19.3, d = 2.9, e = 3)\]
behaves hyperchaotic and generates a hyperchaotic key called hyp-K which is used to encrypt the watermark.

\[
\begin{align*}
\dot{x}_1 &= a(x_2 - x_1) \\
\dot{x}_2 &= bx_1 + cx_2 - x_1x_3 + x_4 \\
\dot{x}_3 &= x_2^2 - dx_3 \\
\dot{x}_4 &= -cx_1
\end{align*}
\]  

Encryption steps based on hyp-K to encrypt the watermark image are as follows (Algorithm 1).

In Algorithm 1, \( C_o \) is a constant number ranging from 0 to 255 and \( \text{mEW} \) is the mean intensity value of EW produced in Step 3.

4.2. Watermark Embedding. The inputs to the watermarking embedding algorithm are the EW of size \((N \times N)\) and the host image HI of size \((M \times N)\). And the output is watermarked host image WHI of size \((M \times N)\). The embedding steps are as follows (Algorithm 2).

4.3. Watermark Extraction. Watermark extraction takes WHI as input and the output is XW, similar to the original color watermark. The size of WHI is \(M \times N\) and the size of XW is \(N \times N\). The extraction steps are as follows (Algorithm 3).

4.4. Watermark Decryption. Watermark decryption is shown in Algorithm 4.

4.5. Algorithm Optimization Using IEFOA. In this section, an improved evolution fruit fly optimization algorithm (IEFOA) discussed in Section 3 is used to find the optimal scaling factor to solve the trade-off problem between invisibility and robustness. The flowchart to find the optimal scaling factor is shown in Figure 2. Invisibility is measured by PSNR and SSIM while robustness is measured by Normalized Correlation (NC). The steps to find the optimal scaling factor are given as follows.

---

**Table 2: DNA mapping rules.**

<table>
<thead>
<tr>
<th>R1</th>
<th>R2</th>
<th>R3</th>
<th>R4</th>
<th>R5</th>
<th>R6</th>
<th>R7</th>
<th>R8</th>
</tr>
</thead>
<tbody>
<tr>
<td>00</td>
<td>A</td>
<td>A</td>
<td>T</td>
<td>T</td>
<td>C</td>
<td>C</td>
<td>G</td>
</tr>
<tr>
<td>01</td>
<td>C</td>
<td>G</td>
<td>C</td>
<td>G</td>
<td>A</td>
<td>T</td>
<td>A</td>
</tr>
<tr>
<td>10</td>
<td>G</td>
<td>C</td>
<td>G</td>
<td>C</td>
<td>T</td>
<td>A</td>
<td>T</td>
</tr>
<tr>
<td>11</td>
<td>T</td>
<td>T</td>
<td>A</td>
<td>A</td>
<td>G</td>
<td>G</td>
<td>C</td>
</tr>
</tbody>
</table>
Step 1. Initialize the parameters $S1 = \beta, \omega_i$ and $S2 = NS, PS, \lambda_{\text{max}}, \lambda_{\text{min}}, \text{Iter}_{\text{max}}, \text{elc}_{\text{max}}, \text{elc}_{\text{min}}$. The parameters in $S1$ such as $\beta$ are the weight factor and $\omega_i$ ($i = 1, 2, 3$) are the quantization coefficients that directly reflect the proportion of invisibility or robustness. The parameters in $S2$ such as $NS, PS, \lambda_{\text{max}}, \lambda_{\text{min}}, \text{Iter}_{\text{max}}, \text{elc}_{\text{max}}, \text{elc}_{\text{min}}$ represent the number of swarms, the population size of the fruit fly, maximum search radius, minimum search radius, maximum iteration number, maximum elimination coefficient, and minimum elimination coefficient, respectively. The set $S1$ with different scaling factors will be used in the gauging function (GF) which is based on the objective evaluation function (OEF) [13] and is given by

$$
\text{GF}(\beta, \omega) = \omega_1 \frac{1}{\beta} \text{PSNR}(\text{HI}, \text{WHI}) + \omega_2 \text{SSIM}(\text{HI}, \text{WHI})
$$

$$
+ \omega_3 \left( \frac{\sum_{i=1}^{K} \text{NC}(W, DW_i)}{K} \right)
$$

where $DW_i$ is the decrypted watermark, i.e., decrypted from extracted watermark $EW_i$ under $i_{th}$ attack.

The scaling factor array is denoted by $\alpha_i (i = 1, 2, \ldots, n)$, where $n$ is the max number index. The scaling factors are used in computing PSNR, SSIM, and NC. For example, the scaling factor array $\alpha_i$ is used to embed the watermark to produce the watermarked image, and $i_{th}$ attack is applied on the watermarked image to produce the attacked watermarked image. After that, the PSNR and SSIM between the cover and attacked watermarked images is calculated. Similarly, NC between original and decrypted watermarks is computed. $S2$ will be used in IEFOA mentioned in the related work section.

Step 2. The GF values of each location for smell judgment are calculated according to Equation (7).

Step 3. In order to get the optimal scaling factor, apply IEFOA discussed in Section 3. The only modification that will be in the IEFOA is to use GF in Step 3 of IEFOA, and repeat Steps 2 to 5 of IEFOA for updating the fruit fly population location when the iterative smell concentration is superior to the previous smell concentration.

5. Experimental Results and Analysis

The invisibility and robustness of the proposed scheme are analyzed in this section. The optimal scaling factor is computed in Section 5.1, invisibility and robustness analysis is carried out in Section 5.2, false-positive problem is done in Section 5.3, and comparison with related works whenever the data is available is done in Section 5.4. Intel(R) core i3 4010 CPU@1.7 GHz with 4.0 GB RAM and MATLAB version R2015a installed on Windows 7, a 64-bit operating system, is used for experimental purposes. Except for the other images, the standard color host images Lena and Pepper each of size $512 \times 512$ and color watermark images with sizes of $256 \times 256$, $128 \times 128$, and $64 \times 64$ shown in Figure 3 are used in the experiments. The initial population size of 50 and the maximum number of iterations of 200 are empirically selected in the experiments. Aside from the above parameters, the other parameters are set according to the improved fruit fly optimization algorithm (IFFOA) [35, 37]; i.e., $\lambda_{\text{max}} = (UB - LB)/2, \lambda_{\text{min}} = 10^{-14} \text{elc}_{\text{max}} = 0.1$, and $\text{elc}_{\text{min}} = 0.05$.

5.1. Finding Optimal Scaling Factor. Optimal state performance is characterized by an optimal scaling factor. According to Section 4.5, an optimal $n$ is decided and is input to gauging function (Equation (7)) to find the optimal scaling factor. The Normalized Correlation (NC) is normally used to evaluate the robustness of the watermarking algorithm and is defined by [13]

$$
\text{NC} = \frac{\sum_{i=1}^{N} \sum_{j=1}^{N} W_{i,j} \text{DW}_{i,j}}{\sqrt{\sum_{i=1}^{N} \sum_{j=1}^{N} W_{i,j} \sum_{i=1}^{N} \sum_{j=1}^{N} \text{DW}_{i,j}}}
$$

The NCs between original watermark (W) and extracted-decrypted DW watermark under various attacks and scaling factors are shown in Figure 4. The attacks used in the simulations are shown in Table 3. NC values vary in the range of $[0: 0.06]$ and get stabilized to large extent in the range of $[0.09: 0.2]$; therefore, the starting value can be set as $n_1 = 0.09$. Similarly, the curves for PSNR and SSIM are also shown in Figures 5 and 6. Similarly, the starting value for PSNR can be set as $n_2 = [0: 0.02]$ as values of PSNR have negative correlations with $\alpha_i$ within the range of $[0.009: 0.2]$, and for SSIM, it can be set as $n_3 = [0: 0.2]$ as SSIM values are almost constant within this range. And $n$ can be calculated as $n = (n_{\text{max}} - n_{\text{min}})/M_i$, where $n_{\text{max}} = 0.2, n_{\text{min}}$ is a set containing all elements of $n_1$ that also belong to $n_2$ and $n_3$, and $M_i$ is the minimum interval. The value of $m$ is then used in GF for obtaining the optimal scaling factor. Table 4 shows the better NCs under certain attacks at the scaling factor $\alpha = 0.115$.

5.2. Invisibility and Robustness Analysis. For invisibility performance, we used color images of Lena and Peppers as host images and colorful logos of the Islamia University of Bahawalpur, Pakistan, as watermarks with different dimensions. Except for visual representation, we also used three metrics, PSNR, SSIM, and NC, to quantify the invisibility. The invisibility performance of the proposed algorithm under no attacks, shown in Figure 7, reflects excellent invisibility. Robustness needs to be assessed when the invisibility is acceptable. In robustness, the quality of extracted watermarks is checked under certain attacks. Several cases of attacks on Lena color image (512 × 512) embedded with watermark (128 × 128) are shown in Figure 8. Watermarks are extracted from attacked images by the extraction algorithm and are decrypted by the decryption algorithm. The corresponding NC values of extracted-decremented watermarks are shown in Figure 9. The NC values
also shown in Figure 10.

5.3. False-Positive Problem Analysis. Digital watermark ownership protection and authentication is a vital application of watermarking schemes; i.e., only the actual owner should be able to extract the embedded digital watermark from the images correctly. FPP problems are very common

(Figure 9) are acceptable for the median, Gaussian noise, salt and pepper, speckle noise, and JPEG compression. Moreover, NC values of extracted-encrypted watermarks under different parameters suffering from numerous attacks are also shown in Figure 10.
Input: XW.
Output: decrypted watermark DW.
Steps: inverse steps of Algorithm 1 are carried out in the reverse order.

**Algorithm 4: Watermark decryption.**

- Find the optimal scaling factor
- Initialize parameters of GF and IEFOA
  - Scaling factor array
  - Weight factor
  - Quantization coefficients
  - Number of swarms
  - Population size
  - Max radius
  - Min radius
  - Max elimination coefficient
  - Min elimination coefficient
- Calculate GF values
- Host image
- Watermark
- Implement hyperchaotic encryption
- Implement embedding procedure
- Watermarked image
- Attack 1
- Attack 2
- Attack K
- Implement extraction and decryption procedure
- Extracted watermark
- Compute PSNR, SSIM, NC
- Compute GF values
- Optimal scaling factor

**Figure 2: Scaling factor optimization.**
Figure 3: (a-b) Host images of size 512 × 512. (c–e) Watermarks of size 256 × 256, 128 × 128, and 64 × 64, respectively.

Figure 4: NC values under various scaling factors and attacks.
and become a challenging issue in digital watermarking schemes, where an attacker claims false ownership of the watermark by embedding and extracting the forged watermarks. This state is a serious security matter that creates a barrier in confirming the real ownership of digital media [25]. There are two approaches to embed the watermark in the SVD domain: (i) computing the singular values of watermark and cover images and then embedding the singular values of the watermark into the singular values of the cover image or (ii) by directly embedding the watermark bits into the singular values of the cover image. Generally, SVD-based watermarking schemes satisfy the criteria of invisibility and robustness but may be exposed to the increased probability of FPP.

To solve the FPP problem, we have implemented two solutions in our study. First, we have performed encryption on $U$ and $V^T$ components by using the logistic map. Secondly, a 4D hyperchaotic system is used to encrypt the

<table>
<thead>
<tr>
<th>Table 3: Attacks used for experimental purpose.</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Attack</strong></td>
</tr>
<tr>
<td>Filter attack</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td>Noise attack</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td>Cropping attack</td>
</tr>
<tr>
<td>JPEG compression</td>
</tr>
<tr>
<td>Motion blur</td>
</tr>
<tr>
<td>Sharpening</td>
</tr>
<tr>
<td>Rotation</td>
</tr>
</tbody>
</table>

Figure 5: PSNR values under various scaling factors and attacks.
watermark before embedding it into the cover image. This gives an additional layer of security against FPP. Therefore, it will be mandatory to decrypt again the watermark after extraction. In the experimental setup of FPP, a watermark (64 × 64) is chosen as shown in Figure 11(a). A decrypted watermark with correct parameters having NC = 1.0000 is shown in Figure 11(b), while Figure 11(c) is the extracted watermark (NC = 0.62) with incorrect parameters which is not recognizable.

5.4. Performance Comparison. In this section, the proposed watermarking scheme is compared with some recently published schemes. The robustness comparison based on NC values after applying some attacks is shown in Table 5. It is obvious that, under some attacks, our results are better when compared with the recently published schemes. The improved results are written in bold format. The imperceptibility comparisons listed in Table 6 are based on the average NC, PSNR, and SSIM between the cover and watermarked images. It is clear that imperceptibility results are better than some recently published works when compared in most cases. Computational time consisting of watermark embedding time, watermark extraction time, watermark encryption, and decryption time is given in Table 7. The computational time is verified by using five test host images having a dimension of 512 × 512 taken from the USC-SIPI image database while the three RGB images (Figures 2(c)–2(e)) having dimensions of 256 × 256, 128 × 128, and 64 × 64 are used as watermarks. The improved results such as watermark embedding and extraction time are written in bold format.

<table>
<thead>
<tr>
<th>Attacks</th>
<th>NCs at α = 0.115</th>
</tr>
</thead>
<tbody>
<tr>
<td>No attack</td>
<td>1.0</td>
</tr>
<tr>
<td>Gaussian low-pass filter</td>
<td>0.827244</td>
</tr>
<tr>
<td>Median filtering (5, 1)</td>
<td>0.887447</td>
</tr>
<tr>
<td>Gaussian noise</td>
<td>0.908452</td>
</tr>
<tr>
<td>Salt and pepper noise</td>
<td>0.944562</td>
</tr>
<tr>
<td>Speckle noise</td>
<td>0.951758</td>
</tr>
<tr>
<td>JPEG compression</td>
<td>0.900064</td>
</tr>
<tr>
<td>JPEG2000 compression</td>
<td>0.880474</td>
</tr>
<tr>
<td>Sharpening attack</td>
<td>0.99612</td>
</tr>
<tr>
<td>Histogram equalization</td>
<td>0.826685</td>
</tr>
<tr>
<td>Average filter</td>
<td>0.826939</td>
</tr>
<tr>
<td>Motion blur</td>
<td>0.830194</td>
</tr>
</tbody>
</table>

Figure 6: SSIM values under various scaling factors and attacks.
Figure 7: Invisibility test results at the scaling factor of 0.115. (a) Watermark. (b) Host image 1024 × 1024. (c) Watermarked images. (d) PSNR (db). (e) SSIM. (f) Extracted watermark. (g) NC (without attack).

Figure 8: Continued.
Figure 8: Various attacks on watermarked images. (a) Gaussian low-pass filter, (b) median, (c) Gaussian noise, (d) salt and pepper noise, (e) speckle noise, (f) JPEG compression, (g) JPEG2000 compression, (h) sharpening attack, (i) histogram equalization, (j) average filter, (k) motion blur.

Figure 9: Continued.
Figure 9: Watermarks extracted from attacked watermarked images given in Figure 8. (a) Gaussian low-pass filter: NC = 0.79286. (b) Median: NC = 0.94175. (c) Gaussian noise: NC = 0.9124. (d) Salt and pepper noise: NC = 0.93241. (e) Speckle noise: NC = 0.85761. (f) JPEG compression: NC = 0.88661. (g) JPEG2000 compression: NC = 0.85736. (h) Sharpening attack: NC = 0.82034. (i) Histogram equalization: NC = 0.76479. (j) Average filter: NC = 0.79288. (k) Motion blur: NC = 0.76144.

Figure 10: NC results under different attacks and parameters. (a) JPEG compression, (b) JPEG2000 compression, (c) Gaussian low-pass filter, (d) median filter, (e) Gaussian noise, and (f) sharpening attack.

Figure 11: FPP results with correct and incorrect parameters. (a) Original watermark, (b) the decrypted watermark with correct parameters, and (c) the decrypted watermark with incorrect parameters.
6. Conclusions and Future Directions

This paper is an attempt toward developing an imperceptible, secure, and robust watermarking framework with the procedure of scaling factor optimization based on IEFOA to solve the issues of authentication, integrity, and FPP. Host images can be embedded with color watermarks of multiple dimensions efficiently. Prior to the embedding procedure, the color watermark is encrypted by using a hyperchaotic system whose initial parameters are found from a DNA sequence taken from the NCBI dataset. After encrypting the RGB components of the watermark image, the embedding procedure consisting of logarithmic-based DWT, HbD, and SVD is utilized to obtain the watermarked image. Host images embedded with watermarks have shown an average PSNR greater than 35 which is considered acceptable and makes watermark invisible to the human visual system. This scheme also accomplishes excellent imperceptibility but with comparable robustness results. Moreover, the double encryption (before SVD and after SVD) makes it more secure to cope up with the security issues. A slight modification in the SVD parameters or hyperchaotic key makes the extracted watermark completely unrecognizable.

In the future, we intend to extend the proposed scheme to DICOM imaging such as ultrasound, X-rays, and magnetic resonance imaging. We also intend to make it more robust against attacks in which it is not robust. Moreover, we intend to adapt this scheme with other frequency transforms by combining it with higher-dimensional hyperchaotic systems to achieve high-efficiency batch processing.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

References