Ingénierie des Systèmes d'Information
Vol. 30, No. 3, March, 2025, pp. 593-606
Journal homepage: http://iieta.org/journals/isi

Enhancement on Secure Transmission of DICOM Images with Optimized Chaos Based Encryption and SPIHT-SVD with QIM Based Watermarking

Abirami Ramalingam*, Malathy Chidambaranathan

Department of Networking and Communications, School of Computing, SRM Institute of Science and Technology, Kattankulathur 603203, India

Corresponding Author Email: ar1160@srmist.edu.in

https://doi.org/10.18280/isi.300304

Received: 8 November 2024
Revised: 30 December 2024
Accepted: 14 January 2025
Available online: 31 March 2025

Keywords:
image security, watermarking, fruit fly optimization, chaotic, SPHIT, SVD, QIM

ABSTRACT

Medical images are essential visual data that are utilized in the field of healthcare for patient monitoring, diagnosis, and treatment. They include techniques that provide in-depth understanding of the human body, such as MRI, CT scans, and X-rays. In order to preserve patient privacy, stop illegal access, and guarantee the accuracy of key diagnostic data, medical image security is essential. It protects patient confidentiality and makes sure that private medical data is not compromised. Watermarking incorporates authentication information to confirm the integrity and ownership of the image, while encryption guarantees the privacy of the image data during transmission and storage. When combined, these methods aid in preserving the confidentiality, integrity, and security of vital medical data. In this work planned to propose a model for image encryption followed by watermarking for image security. An enhanced chaotic image encryption optimized with fruit fly optimization technique is used for image encryption and decryption purpose. A novel hybrid method combining SPHIT (Set Partitioning in Hierarchical Tree) with SVD (Singular Value Decomposition) used for watermarking where QIM (Quantization Index Module) is used in embedding. This work gives MSE of 1.52E-06, PSNR of 56.6366, SNR of 46.7524, NCC of 0.9985 and Q with 0.9990.

1. INTRODUCTION

Medical image communications, data storage, and transport have all grown in popularity during the last several decades. Network assaults, denial-of-service, phishing, and man-in-the-middle are among the cyber threats that data encounters during transmission over the internet. Data encryption systems need to be trustworthy and secure in order to guarantee the safety of information while it is being sent. The spatial or frequency domains are often the basis for image encryption (IE). The frequency domain is occupied by IEs like wavelets and Fourier transforms, whereas the spatial domain is occupied by IEs like chaotic maps, DNA encoding, cellular automata, and compressed sensing [1]. Some features of digital images include data redundancy, strong correlation between neighboring pixels, reduced sensitivity to changes in image quality caused by small changes in pixel attributes, and large data storage capacities, among other things. Since they need a great deal of processing power and time to decipher, standard ciphers such as IDEA, AES, DES, RSA, etc., are unfit for use in real-time image encryption. Crypts that are both fast and secure are the only ones that should be used for real-time image encryption. Even with a high level of security, real-time operations would be severely hindered by an inefficient encryption method [2]. Due to its great dynamism, complexity, and sensitivity to control parameters of the maps, chaos-based IE is the most often used system. Two processes, permutation and diffusion, are often used to run IE systems based on chaotic maps.

In order to address the growing need for secure real-time image transmission across wireless networks and the Internet, image encryption techniques have been extensively researched. When dealing with very big images, traditional image encryption algorithms like data encryption standard (DES) struggle due to their poor degree of performance. The challenge of providing quick and very secure image encryption has proven to be intractable, but chaos-based encryption has proposed a novel and effective solution. Image encryption research has increasingly relied on chaotic systems since 1989, when Matthews introduced the chaotic encryption technique. A chaotic encryption system has been the subject of several recent publications [3]. Traditional encryption algorithms and chaotic-based encryption systems are only two examples of the numerous available encryption approaches. But some of these older encryption techniques are complicated, sluggish, and difficult to learn, and unfit for use in real-time applications. However, the benefit of disorder resides in the fact that unapproved users see it as noise, which is a major deterrent. As a result, encryption systems based on chaos provide secure transmission networks that are both quick and simple to build [4].

Image processing, signal processing, electric power systems, optical assessment, neural networks, and many more scientific fields make use of chaos, a stochastic phenomenon that arises

in nonlinear and guaranteed systems with the properties of randomness, regularity, ergodicity, and sensitivity to the initial values. Combining chaos with optimization methods for issues improves the efficiency of image encryption [5]. This is because the qualities of chaos are similar to those of swarm intelligence. Chaotic systems, which are ergodic and very sensitive to the secret key—a combination of the system characteristics and beginning conditions—also possess the two crucial features of confusion and diffusion—and so are essential to any successful conventional encryption scheme. Analogue chaotic cryptosystems use discrete dynamical systems, whereas digital chaotic cryptosystems use continuous dynamical systems; these two types of methods are based on the categorization of chaotic systems. Also, the architecture of chaos-based encryption algorithms tends to be somewhat simple. Consequently, many cryptosystems based on chaos have been suggested recently [6].

Compressing an image implies reducing its file size in bytes without drastically lowering its quality. Eliminating duplication is the primary objective of compression. The number of meta-heuristic algorithms has increased in recent years due to the proliferation of optimization issues. These algorithms include genetic, ant colony, particle swarm, fish swarm, Virus colony search (VCS), fire-fly, and many more. One such technique that Professor Pan Wenchao introduced in 2011 called the fruit fly optimization algorithm (FOA) has a basic structure, is straightforward to select parameters for, and converges quickly. Managers, forecasters, planners, etc., in the economic sector rely heavily on FOA. However, since the fly step size was set in the original FOA, it may have sunk into local optima like other optimum evolutionary algorithms. Denoising performance is impacted by the fitness function, which is a critical issue. Evidence has been presented by a number of academics. But the vast majority of them include complete citations. Both the methodology and the reference images are impractical for real-world use [7].

The challenge of decreasing the data needed to depict a digital image is the focus of image compression. First, coding redundancy, which occurs when coding is suboptimal; second, inter-pixel redundancy, which arises from pixel correlations; and third, psycho-visual redundancies, which arise from data that the human visual system ignores, are the three primary types of data redundancies that must be removed in order to achieve compression [8]. By using SPIHT, the grey level image compression technique may be extended to compress colour images as well. The SPIHT method, in relation to lowering thresholds, produces an embedded bit stream of wavelet coefficients. Extremely low image quality is the outcome when an excessive amount of bits in the encoded bit stream indicate insignificance. A confirmation of better reconstructed image quality may be achieved by reordering the bits in the encoded bit stream so that important information is given priority. Because of this, the energy compression of the wavelet coefficients may be used more effectively [9].

When it comes to assuring the validity, integrity, and security of data in the healthcare industry, medical image watermarking is very necessary. Protecting sensitive patient information from being accessed or altered by unauthorised parties is accomplished by the incorporation of watermarks into medical images. This method also makes it possible to verify the authenticity of the image, since any modification would cause the watermark to become disrupted, which would indicate that the image may have been tampered with. In addition, watermarking lends assistance to copyright protection, which guarantees that the ownership of the medical data as well as its point of origin are maintained. In the grand scheme of things, it is an extremely important factor in ensuring the dependability and secrecy of medical images in clinical and research environments. Chaos-based image encryption is crucial in ensuring the security of digital images, because of its remarkable capability of generating intricate and unpredictable patterns. By utilizing the principles of chaos theory, this method guarantees that even the smallest alteration in the encryption key will produce a wholly distinct encrypted image, thereby greatly impeding unauthorized individuals from deciphering the encryption. Chaos-based systems possess a unique ability to effectively safeguard large and sensitive image data in a wide range of applications due to their inherent sensitivity and unpredictability.

When decomposing matrices, SVD and HD are the tools of choice. False positives are an issue with SVD processing, however they can be solved by encrypting the SVD components with chaotic systems. The objective evolution function (oefunc) aids FOA in finding an optimised scaling factor, which enhances efficiency and helps preserve the watermarking model's balance between invisibility and resilience. A methodology for first encrypting images and then watermarking them is worked in this paper. Using the fruit fly optimisation method, Chaotic has optimised its encryption. The Quantisation Index Module (SPHIT) with SVD will be used for embedding.

2. RELATED WORKS

One of the primary challenges in multimedia applications is effectively compressing colour images [10]. This led them to investigate how well the SPIHT method compressed colour images. To convert an RGB image to a YCbCr format, the SPIHT technique is used for the luminance (Y) and chrominance (Cb, Cr) components. Human eyesight and PSNR are used to verify the reconstructed image. To get even better compression, combine Huffman and arithmetic coding. Sending a compressed image between two computers and then checking the reconstructed image allowed them to test the channel behaviour. Utilising the Gaussian mutation process and chaotic local search, Zhang et al. [11] introduced a new FOA-based technique in their study. Immature convergence is reduced and diversity and intensification are brought into greater harmony in their algorithm. Starting from the starting point, all of the agents in the population undergo complete application of the Gaussian mutation mechanism in order to maintain their diversity and boost the algorithm's convergence trends. Additionally, in the following phases, the local searching ability is enhanced by using the chaotic local search method. Consequently, they saw a balanced interaction between the worldwide search and the local disruption. A technique for encrypting images is developed by Jiao et al. [12] using three different types of chaotic maps. In order to determine how many iterations are necessary to completely muddle the image, the algorithm use the Cat mapping technique. The key space is much improved when the initial key is generated using a random integer and the mutual mapping of Logistic and ChebyShev. After encryption, the correlation between neighbouring pixels in the image is diminished since the pixel disparities between them are higher and the correlation between pixel points is eliminated using the Logistic and ChebyShev double chaotic mapping

techniques. For each encrypted pixel, the R, G, and B pixels from the pixel before it is used to execute an exclusive-OR operation on the B, R, and G pixels, respectively. This ensures that even a small change to the plaintext causes the encrypted images to diverge substantially from the plaintext ones. As a result, it may significantly strengthen the image's defense against various threats. In order to make their algorithm more efficient and suitable for image encryption on Android-based smart mobile devices with high real-time performance, they need to decrease the number of iterations for Cat, Logistic, and ChebyShev while still guaranteeing security. This will decrease encryption time and improve encryption efficiency.

Based on the SPIHT technique, Chen [13] presented a novel approach for compressing medical images. Their algorithm has a stellar reputation for efficiency. Some adjustments are necessary to ensure the findings are suitable for use in the medical field. To achieve high compression rates while maintaining decent image quality, their modified SPIHT method takes the frequency domain correlation into consideration. Ensure accurate diagnosis with high-quality decoded images. Medical image management becomes more efficient with a strong compression ratio due to decreased bandwidth and storage needs. Degradative encryption was suggested by Xiang et al. [14] as a new paradigm for selective image encryption. In contrast to current confidential selective encryption, their goal is to reduce the original image quality while keeping a blurred preview, rather than making the full image unreadable. The versatility of progressive encryption makes it useful in many contexts. Following this, they introduced a degradative encryption scheme that complies with formats using SPIHT compression. According to their analysis of SPIHT's output, the BLIS-O-sgn coefficients—which make up only around 10% of the compressed bitstream—should be encrypted. A personalised tradeoff between efficiency and security may be supported by the scheme's progressive design.

Using 9/7-M IB-IWT, SPIHT, and a novel hyper-chaotic system, Zhang and Tong [15] presented a combined lossless image compression and encryption method. 9/7-M IB-IWT and SPIHT are responsible for compression. In encryption, confusion and diffusion are carried out using the novel hyper-chaotic system and the Cat map. It is used to improve the encryption effect since SHA-256 is quick and sensitive to input. The plaintext and the keystream are related concepts in encryption. Attack resistance is further enhanced by the use of a nonlinear inverse process. There are three layers of encryption applied to the image during the compression process. First, all of the wavelet coefficients are permuted at once, and then the crucial ones are diffused. Diffusion uses a key stream that is associated with the plaintext. Thereafter, SPIHT's sorting pass incorporates the encryption. As a last step in encrypting the final data, we suggest using code stream diffusion and permutation. There is enough room in the key area to withstand brute-force assaults. The findings demonstrate that the suggested encryption and compression technique is quite secure and performs well when it comes to lossless compression.

The Internet of Medical Things (IoMT) [16] offers several benefits to healthcare, such as telemedicine, remote patient monitoring, and the ability to store medical records at facilities and clinics. Two thorny problems with IoMT are data encryption and quality of service. The findings of their study point to a substantial and extensible approach to access control. In this study, they illustrate how to encrypt IoMT medical devices using a DSTEA. Quality of Service, data encryption, optimisation, and transmission are all sufficiently improved by the suggested approach, DS-TEA. Research on the suggested subject may go on to examine various security concerns that may arise when transmitting data.

An innovative HCM for promoting an image encryption model was introduced in the study [17]. Steps such as image pre-processing, key creation, image encryption with optimised HCM, and image decryption make up the built-in image encryption framework. Key creation, an essential step in chaotic-based image encryption, was carried out using the SHA-256 cryptographic hash method. In addition, the HCM that encrypts the image was developed by integrating 2DLCM and PWLCM. The performance was improved by performing the first piecewise points of HCM. The parameters were optimised using an updated meta-heuristic technique called CI-WOA.

A method for embedding and extracting digital image watermarks was suggested in Veni and Meyyappan [18]. They used the oppositional fruit fly algorithm to implement DWT-SVD, which stands for Discrete Wavelet Transform and Singular Value Decomposition. The technique that has been suggested is put into action using MATLAB. Quality measures like SSIM, PSNR, Normalised correlation, and MSE value are used to assess the performance of the suggested technique. After evaluating the planned work using each measure, the findings were further analysed using current methodologies. Therefore, in comparison to the current approaches, the suggested work has a superior PSNR and SSIM value. In comparison to the current technique, the suggested strategy achieves a high overall PSNR value of 47.49 dB and SSIM of 0.9865. A simulation of the wavelet-based progressive SPIHT method and the singular values/rank based SVD or rank reduction approach were both detailed by Rao et al. [19]. Using these two methods, MRI and CT scans of the brain are created. We evaluate the two methods' efficacy using the PSNR, MSE, CR, and BPP quality measures. It can be inferred from the data that the SPIHT approach outperforms the SVD technique in terms of PSNR, MSE, and CR. Using both methods, MRI images had superior PSNR and MSE values than CT images. As the PSNR value rises, CR lowers in SVD. However, with SPIHT, the Compression Ratio grows in tandem with the PSNR. The SPIHT method yields higher-quality images than SVD. Hence, when comparing the SPIHT method to the SVD method, the latter with fewer encoding loops achieves superior results in terms of PSNR, MSE, and CR. To include the hidden data into the cover image, Ingaleshwar and Dharwadkar [20] suggested a WCFOA-based Deep CNN, a method for efficient and effective image watermarking. The confidentiality of information is preserved in multimedia network communication technology by encoding secret messages in cover media. This prevents unauthenticated recipients from accessing the concealed data. During the griding step, the suggested technique divides the cover image into grid lines. When the grid lines reach the feature extraction phase, they are used to successfully extract features like neighborhood-based features and CNN features. The neighborhood-based features, for example, comprise area, edge information, and entropy of loop. Once the characteristics have been extracted, the best area to conceal the secret message is chosen by training the Deep CNN classifier with the suggested WCFOA.

3. PROPOSED METHOD

A combined compression and encryption strategy that executes compression and encryption in one step is a promising solution when taking into account the qualities of image data, image processing technology, security needs, and the rising amount of images. The length of the cipher image is made unknown by the interplay between the compression operation and the encryption procedure. The encryption and watermarking of proposed work are explained in Figure 1.

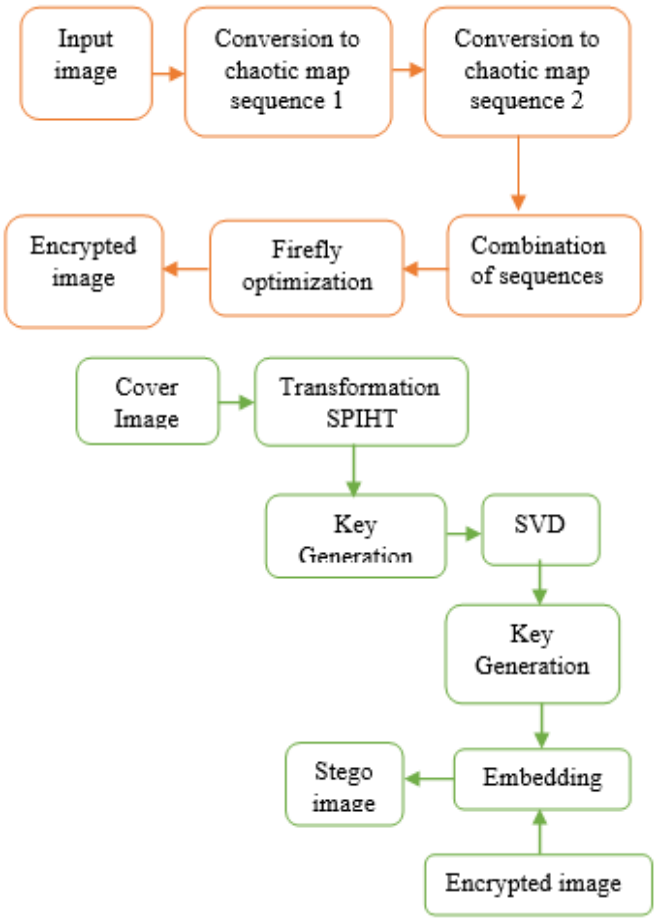


Figure 1. Encryption and watermarking of proposed method

The flowcharts illustrate the processes and procedures involved in image encryption, each of which makes use of a unique set of methods to improve safety. The first chart is an illustration of a technique that utilizes chaotic maps with firefly optimization. From the beginning, an input image is transformed into chaotic maps, which results in the production of sequences that are subsequently concatenated. An image that has been encrypted is the end product, and it is then subjected to further optimization employing firefly methods in order to guarantee its resilience and security. A procedure that combines the SPIHT algorithm with the SVD technique is shown in the second flowchart. The process starts with a cover image, which is then altered using SPIHT in order to ease the production of an effective key. After that, the procedure makes use of SVD for further augmentation, which results in the generation of a stego image that incorporates the information from the original image. After all is said and done, an encrypted image is created by using key generation and embedding procedures. This ensures that the data is hidden in a safe manner.

3.1 Preprocessing

Preprocessing steps in image processing and computer vision sometimes include resizing images. One way to filter out background noise while keeping edges intact is via the Alpha Trimmed Mean Filter technique. The Alpha Trimmed Mean Filter technique is useful for determining the neighborhood's lowest gray-level d/2 value and greatest g(s,t) d/2 value. After removing the lowest and greatest values by d/2, this filter substitutes the average grey level value in the subimage beneath the size m×n adjacency window for each pixel's value. Here is the equation that defines the Alpha Trimmed Mean Filter method:

$$f(x,y) = \frac{1}{mn-d}\sum(s,t)\in Sxyg(s,t) \quad (1)$$

The filter kernel size is denoted by mn, the input value ranges from 0 to 9, the pixel intensity value that will be replaced by the filtering result value is $g(s,t)$, and the number of pixel intensities influenced by the filtering process is $\sum(s,t) \in Sxyg(s,t)$.

3.2 Chaotic optimized with fruit fly optimization for encryption method

To make algorithms seem more random, statistic distributions like the uniform and Gaussian distributions are used. Given its unpredictability qualities, chaos is an excellent option for producing random data. Algorithms may be able to complete iterative search steps more quickly than traditional stochastic search using normal probability distributions due to the chaotic properties of ergodicity and mixing chaos [5]. With its straightforward operation and well-dynamic unpredictability, logistic mapping—introduced by May in 1976—is the most emblematic chaotic mapping system. One definition of logistic mapping is:

$$z(t+1) = \mu z(t)\big(1-z(t)\big) \quad z \in (0,1), 0 < \mu \leq 4 \quad (2)$$

The iteration number is represented by t and control parameter c decides whether the chaotic variable z stabilises at a constant value. You can't put a value of 0, 0.25, 0.75, 0.5, or 1 into the variable z. When the value of μ is 4, the logistic mapping sequence becomes chaotic. The value of 4 μ = is used in subsequent investigations [5]. Using a 256×256 key for a 256×256 image in chaotic cryptography guarantees that every pixel or block of the image is safely encrypted, providing enough key variety and defense against brute-force assaults. By matching the image dimensions, this key size maximises computing efficiency and security.

A novel global optimisation algorithm called FFOA was developed based on the entrapment behaviour of fruit flies. Using a model inspired by fruit flies, the FFOA approach finds the optimal solution to the optimisation issue. Sensitive olfactory organs let them identify a target's aroma, and then acute visual organs helped them find their way there. The original fruit fly optimisation approach (osphresis foraging, start, population assessment, and vision) [16] comprises these steps.

- Step 1: Let X_a, Y_a represent the positions of the fruit fly swarm. Initialize these variables randomly.
- The methodology has population size = 50, total

iteration = 500, with a step size of = 0.1.

- Step 2: Let $X_i = X_b + x_r$, $Y_i = Y_b + y_r$, where x_r and y_r are the variables for random values. And let the location coordinates be X_b and Y_b, where X_a, Y_a are the initial values of these location coordinates, gives information about distance and random direction of the individual fruit fly.
- Step 3: Let $D_i = \sqrt{X_i^2 + Y_i^2}$ is the distance measured from the origin. D_i is calculated first and then the reciprocal of D_i is calculated which is called as the smell concentration $Si = 1/D_i$.
- Step 4: At individual location of the fruit fly the smell concentration is given as $Smell_i$. Find $Smell_i$ which is a function of S_i, and it is denoted as $Smell_i$ =Function (S_i). Function (S_i) is the smell concentration judgment function
- Step 5: Find the maximal value of the smell concentration in the swarm of fruit flies, and it is given as [$bestSmell$, $bestIndex$] = max ($Smell_i$).
- Step 6: The best values of the smell concentration and the coordinate is recorded, then the swarm of the fruit flies will move towards the final location using vision $Smellbest = bestSmell$, $X_b = X(bestIndex)$, $Y_b = Y(bestIndex)$.
- Step 7: Perform iterative optimization and repeat Steps 2-5 if the iterative smell concentration is better than its previous value. Otherwise, return to Step 6.

Algorithm 1. Chaotic optimized with fruit fly optimization

1. Initialize the center location of fruit fly swarm (Xaxis, Yaxis);
2. for i=0 to M-1
3. Mv [i]=i+1
4. End i
5. for k=0 to r-1
6. for i=0 to M-1
7. uMvm[i]=Mv[i]x u[k]+r[k]+rc[k]
8. vMvm[i]= Mv[i]xv[k]
9. End i
10. population_size = 50;
11. iterations = 500;
12. [bestSmell, bestIndex] = Fittest(Smell);
13. Smellbest = bestSmell;
14. SmellbestPOS = S(bestIndex,:);
15. Xaxis = X(bestIndex,:);
16. Yaxis = Y(bestIndex,:);
17. [bestSmell, bestIndex] = Fittest(Smell);
18. If bestSmell is better than SmellbestSmellbest = bestSmell;
19. SmellbestPOS = S(bestIndex,:);
20. Xaxis = X(bestIndex,:);
21. Yaxis = Y(bestIndex,:);
22. End if
23. Perform chaotic mapping using Eq. (2) for SmellbestPOS, then Update SmellbestPOS if there is a better search agent;
24. End while
25. Return SmellbestPOS and Smellbest.

3.3 SPIHT

It is common practice to combine image compression algorithms, such as SPIHT, with watermarking methods to include watermarks into compressed domains. When SPIHT is used to watermarking, it becomes much more resistant and almost undetectable. Image compression is the main use of the SPIHT method. As more bits are received, the image quality increases, and it is very efficient. It also offers progressive image transmission. In contrast to JPEG 2000, the SPIHT method does not aim to minimise memory or bandwidth and is not optimised to examine areas of interest, yet it achieves a high compression rate when N is big. When it comes to lossy image compression, the SPIHT algorithm is often considered to be among the most effective methods [14]. In 1996, Said and Pearlman created the SPIHT image coding technique, which is an improved version of Shapiro's embedded zerotree wavelet (EZW) algorithm [10]. Following an image's application of the wavelet transform, the primary algorithm divides the resulting wavelet decomposed image into meaningful and irrelevant parts using the following function:

$$Sn(T) = \begin{pmatrix} 1, & \max_{(i,j)\in T}\{|c_{(i,j)}|\} \geq 2^n \\ 0, & \text{Otherwise} \end{pmatrix} \quad (3)$$

Algorithm 2. SPIHT algorithm

Initialization:n = floor(log2(max(|c_{i,j}|)))); LSP = ∅;
add all (i,j) in H to LIP; add (i,j) with descendants to LIS as type A.
Sorting Pass:For all (i,j) in LIP:
    output S_n(i,j);
    if S_n(i,j) = 1: Move (i,j) to LSP; output sign of c_{i,j}.
For all (i,j) in LIS:
    If type A, output S_n(D(i,j));
    if S_n(D(i,j)) = 1:
    For all (k,l) in O(i,j):
        Output S_n(k,l);
        if S_n(k,l) = 1: Move (k,l) to LSP;
        output sign; else, add to LIP.
        If L(i,j) ≠ empty: Move (i,j) to LIS as type B; else, remove from LIS.
If type B, output S_n(L(i,j));
    if S_n(L(i,j)) = 1: Add (k,l) in O(i,j) to LIS as type A; remove (i,j) from LIS.
Refinement Pass:Output next significant bit for all (i,j) in LSP (added before this pass).
Update Threshold:n = n - 1; repeat until n < 0 or desired bit rate achieved.

The significance of a set of coordinates T is denoted as Sn(T), and the value of the coefficient at coordinate (i,j) is ci,j. The sorting pass and the refining pass are the two stages of the algorithm [14, 15]. The SPIHT technique compresses images efficiently using redundancy. After converting the input image to grayscale, it uses a wavelet transform like the 2D Discrete Wavelet Transform to divide it into frequency bands. Starting with the List of Significant Pixels (LSP), List of Insignificant Sets (LIS), and List of Zero Trees (LZT), the method arranges these wavelet coefficients into a hierarchical tree structure. It then tests LSP coefficients for significance against a threshold and encodes their significance and sign. LIS coefficients with significant child coefficients are promoted to the LSP and zero trees are identified. The threshold is decreased by 0.5 for successive repetitions. Significance testing, encoding, and threshold modification are performed until the desired bit count or maximum distortion is reached. The compressed

bitstream encodes the image's important information effectively, enabling large compression ratios without sacrificing visual quality.

Advantages of SPIHT:

- SPIHT's great compression performance aids in keeping the watermarked image's quality, which is a major plus.
- Adding the watermark to the wavelet coefficients prior to SPIHT compression makes the image more resistant to typical assaults in image processing, which improves its resilience.
- Invisibility: If the watermark is embedded correctly in the wavelet domain, it will not be visible to the naked eye.

3.4 SVD

As an orthogonal matrix decomposition technique, SVD is dependable and strong. The signal processing domain is seeing SVD's rising popularity for conceptual and stability-related reasons. When it comes to processing images, SVD is a desirable algebraic transform. In imaging, SVD's characteristics stand out. More research and contributions are needed for certain SVD features, even if they are completely used in image processing for others. Its capacity to approximate matrices of a given rank and its connection to the rank of a matrix are important properties of SVD. It is possible to characterise digital images by adding up a limited number of eigen images as they are often represented by low rank matrices. The idea of treating the signal as two separate subspaces gives birth to this notion [8]. The SVD/Rank based reduction was first found for square matrices in 1873 and 1874. In 1930, it is improved for rectangular matrices. The following equation represents the SVD of a rectangular matrix P.

$$P = A\varepsilon B^T \tag{4}$$

Assuming that P is a rectangular matrix of size MXN.A and B are ortho-normal's rectangular matrices. The diagonal members of the diagonal matrix, denoted as ε, are singular values [19]. Since it is data dependent, SVD decomposition is unable to handle time-frequency domain data [21].

3.5 Watermarking

Watermarking is a technique that allows one image to be embedded with specific data into another image. When it comes to protecting sensitive information and intellectual property, it is the most important subfield of data security. On the other hand, there are two main varieties of image watermarking: invisible and visible. As a result, video images use the visible watermark, whilst audio and static signals make extensive use of the invisible watermark. Image watermarking, on the other hand, provides encrypted digital data transfer by enclosing the data inside the cover image [20]. To create a digital image with a watermark, the watermarking algorithm and a watermarking key are used in the embedding process. Different image domains, such as the space domain, the frequency domain, or the wavelets, call for different embedding methods. The suggested watermarking procedure begins with converting the watermark image to watermark bits; next, the watermark bits are mixed with the gold sequence; and finally, the watermark is embedded in the blocks at the chosen spot using this mixture.

File compression using QIM (Quantisation Index Module) helps make storage and transmission more economical by lowering the amount of bits required to represent data. Digital material may have watermarks embedded into it using QIM with a step parameter of 255. Because it enables the quantisation process to cover the whole range of grayscale values (0-255), a step size of 255 is often employed in QIM for grayscale since we are using images. This guarantees that the watermark may be included into the image with enough variance to be undetectable. This method makes sure the watermark is unreadable and resistant to manipulation and assaults. The overall architecture of the proposed work is depicted in Figure 2. The first step in the procedure is to create a binary watermark, which is a logo. The frequency components of the original image may then be separated by using SPIHT methods to translate it into the frequency domain. The modified image's coefficients are then quantized using two levels. It is computationally feasible to handle a 2-level decomposition, particularly for DICOM images, which often need to be processed quickly for medical purposes. Because DICOM images are useful for diagnosis, it is crucial to maintain detail in key areas. For medical imaging requirements, a 2-level decomposition delivers sufficient compression while preserving image integrity. It offers sturdy areas where a watermark may be inserted without significantly deteriorating the image. The quantization indices are changed in accordance with the watermark bits in order to embed the watermark: for a '1' bit, the index is shifted up, and for a '0' bit, it is moved down. The watermarked image is created by transforming the altered coefficients back into the spatial domain once the watermark has been embedded. The watermarked image undergoes the same transformation and quantization processes as before, and the quantization indices are examined to confirm the watermark's existence. This technique is a good fit for secure watermarking applications as it is resistant to many types of attacks and maintains a high level of visual clarity.

The following steps explain the proposed work.

- Open the medical image and set its parameters.
- Use a chaotic map (such as a logistic map) to create chaotic sequences.
- Utilizing the chaotic sequences, jumble and alter the pixel values to encrypt the medical image (watermark image).
- Set up the required settings for the FOA.
- Specify the goal function that will be used to optimize the encryption procedure.
- To improve encryption security, use FOA to optimize the chaotic map settings.
- For the final image encryption, apply chaotic settings that are optimal.
- Apply Discrete Wavelet Transform (DWT) for two level on the image that has been encrypted.
- Apply the DWT coefficients' SPIHT encoding again for 2 level as given in the architecture shown in Figure 2. SVD should be applied to the SPIHT-encoded image.
- Utilizing Quantization Index Modulation, include the watermark into the SVD components (QIM).
- Reassemble the watermarked image by using the DWT and inverse SPIHT transformations for extraction and decrypt to get the watermark image.

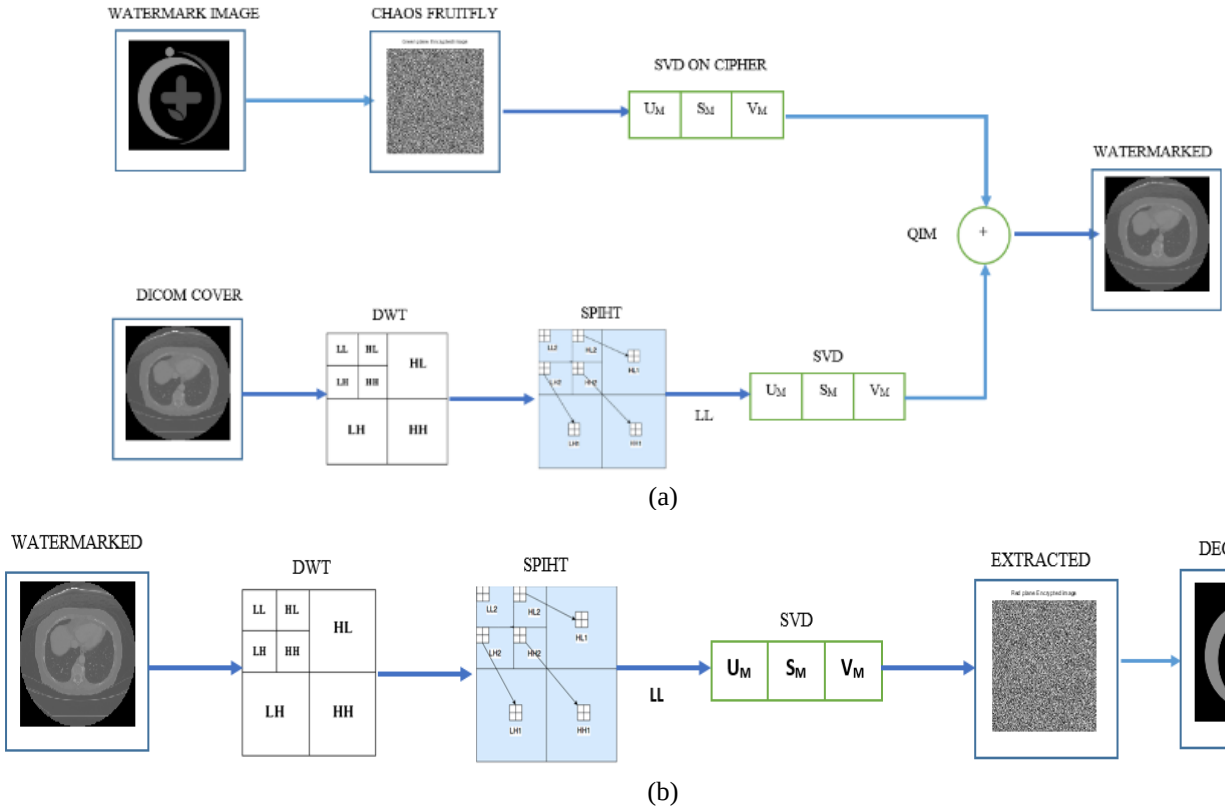


Figure 2. Architecture of (a) Encryption and watermarking process (b) Decryption and extraction process

Embedding process

The watermark may be embedded into the wavelet coefficients either before or during the SPIHT encoding process to integrate it into the SPIHT compression process.

- Apply a wavelet transform (such as DWT) to decompose the original image into multiple subbands (LL, LH, HL, HH). These subbands represent different frequency components of the image.
- Determine the amount of the chosen coefficients and tweak them to fit the watermark bit that has to be inserted. Several kinds of assaults may be thwarted by this strategy.
- Perform SPIHT encoding on the modified wavelet coefficients to compress the image.
- The SPIHT-encoded bitstream, now containing the embedded watermark, can be transmitted or stored efficiently.
- After SPIHT transformation the LL sub band is considered for SVD decomposition. While LH, HL, and HH include greater information, the LL sub band provides crude approximations. LL sub band embedding improves robustness towards compression.
- SVD applied to the encrypted watermark image and then the embedding process happens.

Extraction process

- Decode the SPIHT bitstream to reconstruct the wavelet coefficients of the image. Apply the inverse wavelet transform to obtain the watermarked image in the spatial domain.
- Extract the watermark from the wavelet coefficients using the same method and parameters used during embedding.
- Identifying variations in quantisation levels or coefficient magnitudes, is one embedding strategy that determines the precise extraction procedure.
- By integrating SPIHT with watermarking take use of effective image compression and implant strong, undetectable watermarks.

4. RESULTS

The results of proposed encryption and watermarking methods is shown in Table 1.

Two digital images may be statistically measured by their Mean Squared Error (MSE).

$$MSE = \frac{1}{m*n}\sum_{r=1}^{n}\sum_{c=1}^{m}(WM_{(r,c)} - EWM_{(r,c)})^2 \quad (5)$$

We multiply the original image size, n, by the reconstructed image size, EI. Peak Signal to Noise Ratio (PSNR) could be a better metric than Mean Squared Error (MSE) for statistical purposes. Dividing the signal intensity by the noise capacity or the image discrepancy yields the Signal to Noise Ratio (SNR).

It's formulated as

$$SNR = \frac{\sum_{r=1}^{n}\sum_{c=1}^{m}(EWM_{(r,c)})^2}{\sum_{r=1}^{n}\sum_{c=1}^{m}(WM_{(r,c)} - EWM_{(r,c)})^2} \quad (6)$$

$$PSNR = 10.log_{10}(\frac{\mathrm{MAXi}^2}{MSE}) \quad (7)$$

Table 1. Results of proposed encryption and watermarking methods

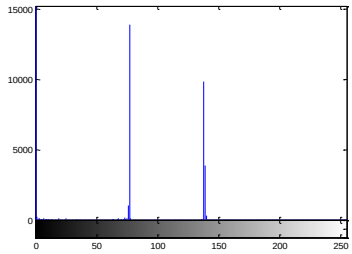
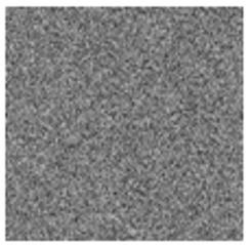
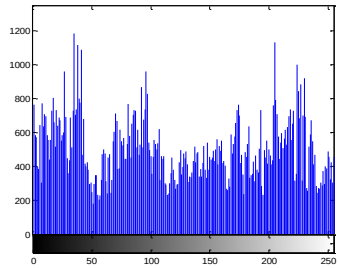
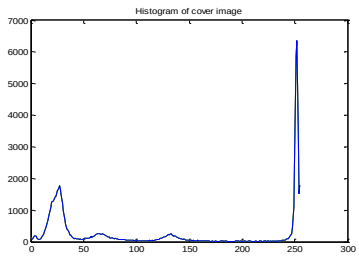
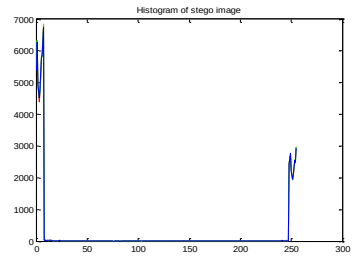
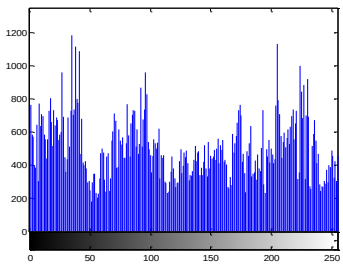
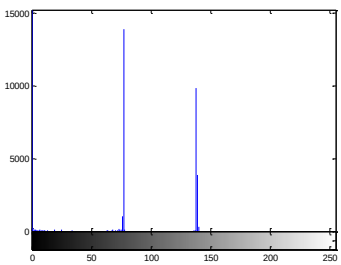
Image Title	Images	Histogram
Message/Watermark Image		
Encrypted Watermark Image		
Cover Image		
Watermark Image		
Extracted Watermark Image		
Decrypted Watermark Image		

Table 2. Quantitative analysis of different chaotic methods

Image	MSE	SNR	PSNR	NCC	Q
Sine	1.96E-05	38.2665	42.0358	0.9856	0.9812
Tent	1.98E-05	36.5421	44.2158	0.9912	0.9944
Henon	2.56E-06	31.2456	45.7689	0.9912	0.9956
Logistic	1.77E-05	33.8546	47.2546	0.9947	0.9979
Logistic+Fruitfly	1.82E-05	37.2456	49.6524	0.9985	0.9988

Table 3. Impact of different watermark size on the proposed method

Image	MSE	SNR	PSNR	NCC	Q
256×256	1.52E-06	46.7524	56.6366	0.9985	0.9990
128×128	1.66E-07	48.5421	58.2158	0.9992	0.9992
64×64	1.31E-08	51.2456	58.9689	0.9996	0.9992

Table 4. Comparison of proposed and existing watermarking methods

Image Transform Method	MSE	SNR	PSNR	NCC	Q
DCT [13]	2.12E-02	24.3356	34.2658	0.9212	0.9332
DWT [18]	1.63E-03	27.4425	38.5524	0.9411	0.9214
IWT [15]	1.77E-04	28.2561	41.3564	0.9542	0.9622
SWT [21]	1.81E-04	33.1214	42.8695	0.9615	0.9699
SWT-SVD [22]	1.82E-05	37.2456	47.6524	0.9985	0.9988
SPIHT-SVD-QIM	**1.52E-06**	**46.7524**	**56.6366**	**0.9985**	**0.9990**

Normalised Cross Correlation (NCC) measures how indistinguishable something is to the naked eye. Only a strong correlation coefficient between the extracted watermark and the original watermark should be considered significant. The formula for the metric is as follows:

$$NCC = \frac{\sum_r^3 \sum_c^n \sum_r^m \left(WM_{(r,c,p)} * EWM_{(r,c,p)}\right)}{\sqrt{\sum_r^3 \sum_c^n \sum_r^m \left(WM_{(r,c,p)}\right)^2} \sqrt{\sum_r^3 \sum_c^n \sum_r^m \left(EWM_{(r,c,p)}\right)^2}} \quad (8)$$

Various measures of image quality employed worldwide there is an explicit definition for each of the three quantities that make up Q in Terms.

$$Q = \frac{4\sigma_{xy}\bar{x}\bar{y}}{\left(\sigma_x^2 + \sigma_y^2\right)\left((\bar{x})^2 + (\bar{y})^2\right)} \quad (9)$$

where,

$$\bar{x} = \frac{1}{MN}\sum_{c=1}^{n}\sum_{r=1}^{m} WM_{r,c}$$
$$\bar{y} = \frac{1}{MN}\sum_{c=1}^{n}\sum_{r=1}^{m} EWM_{r,c} \quad (10)$$

$$\sigma_x^2 = \frac{1}{M * N - 1}\sum_{c=1}^{n}\sum_{r=1}^{m}\left(WM_{r,c} - \bar{x}\right)^2 \quad (11)$$

$$\sigma_y^2 = \frac{1}{M * N - 1}\sum_{c=1}^{n}\sum_{r=1}^{m}\left(EWM_{r,c} - \bar{y}\right)^2 \quad (12)$$

$$\sigma_{xy}^2 = \frac{1}{M*N-1}\sum_{c=1}^{n}\sum_{r=1}^{m}\left(WM_{r,c} - \bar{x}\right)\left(EWM_{r,c} - \bar{y}\right) \quad (13)$$

The performance measurements for several chaos-based encryption techniques applied to images [23] are shown in the Table 2. The difference between the original and encrypted images is shown by the Mean Squared Error (MSE), where lower values imply higher encryption quality. Higher values of SNR (signal-to-noise ratio) and PSNR (peak signal-to-noise ratio) indicate less distortion in the image. Higher values of NCC (Normalised Cross-Correlation), which measures how similar the original and encrypted images are, indicate greater recovery. Higher Q values indicate more effective encryption. Q is a quality statistic.

The performance metrics (MSE, SNR, PSNR, NCC, Q) for images of varying sizes (256×256, 128×128 and 64×64) are shown in the Table 3. SNR and PSNR both rise and the MSE falls with decreasing image size, suggesting better image quality. The NCC and Q values stay around 1, indicating that the compressed images have little distortion and are of good quality.

The comparison of proposed and existing watermarking methods mentioned in Table 4. The comparison of Mean Square Error (MSE), Signal to Noise Ratio (SNR), PSNR of proposed and existing methods are shown in Figures 3-5. The Figures 6 and 7 explain Normalized Correlation (NC) and Q values of proposed and existing methods.

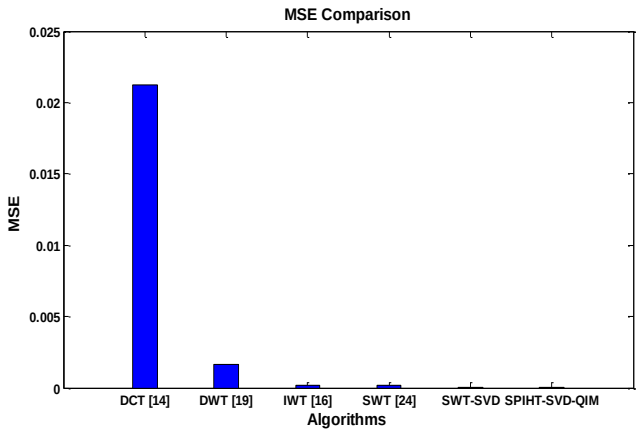


Figure 3. Comparison of MSE of proposed and existing methods

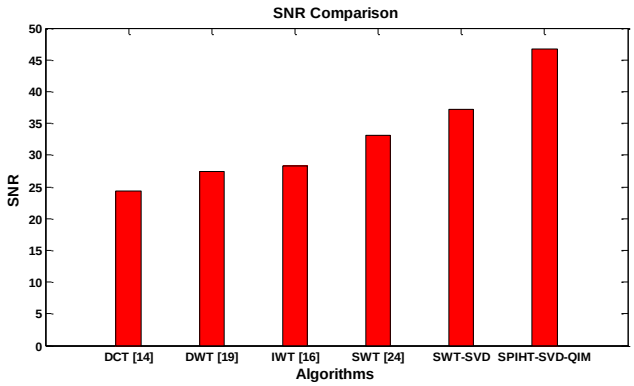


Figure 4. Comparison of SNR of proposed and existing methods

Table 5 illustrates how the alpha-controlled watermark size affects the suggested approach. Better image quality with lower watermark sizes is shown by decreasing alpha,

decreasing MSE, and increasing SNR and PSNR. The watermarked images show no distortion and excellent clarity since the NCC and Q values remain around 1.

Table 6 shows how various compression rates (CR) affect the suggested approach. SNR and PSNR both drop as the compression rate rises, and MSE rises as well, suggesting that image quality deteriorates with increasing compression. At larger compression rates, the NCC and Q values similarly drop, indicating more distortion and less quality in the watermarked images. Table 7 mentions the computational complexity of existing and proposed technique and compared in Figure 8.

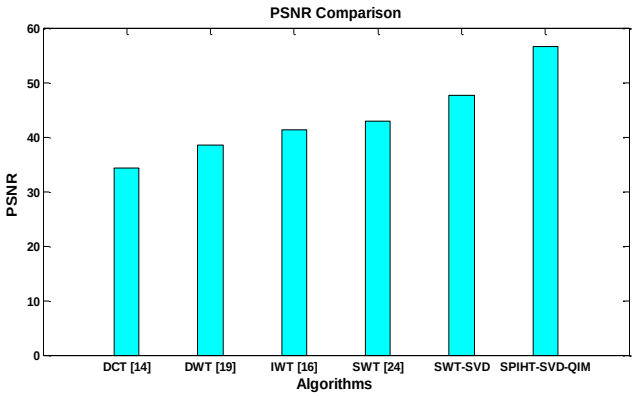


Figure 5. Comparison of PSNR of proposed and existing methods

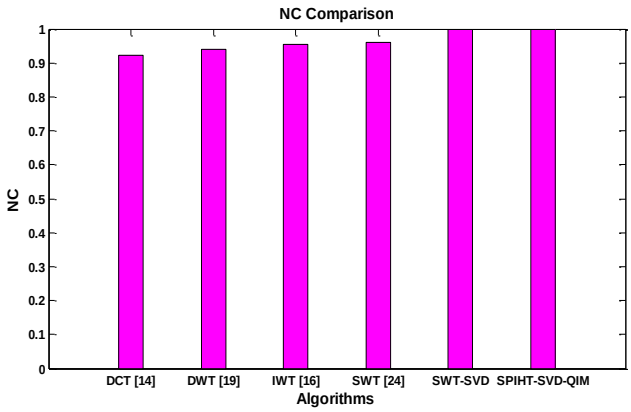


Figure 6. Comparison of NC of proposed and existing methods

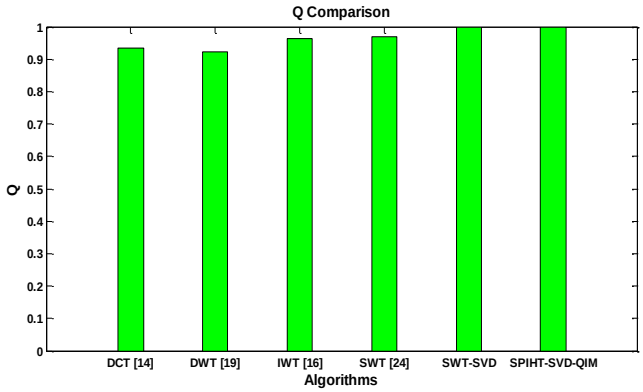


Figure 7. Comparison of Q of proposed and existing methods

Table 5. Impact of watermark size on proposed method

Alpha	MSE	SNR	PSNR	NCC	Q
0.075	1.52E-06	46.7524	56.6366	0.9985	0.9990
0.05	1.215E-06	48.2421	57.1258	0.9989	0.9991
0.025	1.021E-06	49.1214	58.2642	0.9991	0.9992

Table 6. Impact of different compression rate on proposed method

CR	MSE	SNR	PSNR	NCC	Q
5.642	1.52E-06	46.7524	56.6366	0.9985	0.9990
7.1258	1.845E-06	41.2521	51.2468	0.9976	0.9968
10.1218	1.945E-06	39.5846	48.12454	0.9954	0.9948

Table 7. Computational complexity of existing and proposed method

Time Consumption	Total Time (s)	Self-Time (s)	Total Time (s)	Self-Time (s)
	SWT-SVD [21]		SPIHT-SVD-QIM	
Overall time consumed	7.834	1.568	8.234	2.146
Encryption time	4.068	0.835	4.084	2.211
Decryption time	1.014	0.321	1.048	1.586
Embedding time	3.853	0.372	4.662	1.221
Extraction time	0.725	0.299	1.212	1.114

The values of qualitative analysis of watermarked images with different attacks for different set of images are mentioned in Table 8, and the results are shown in Figure 9.

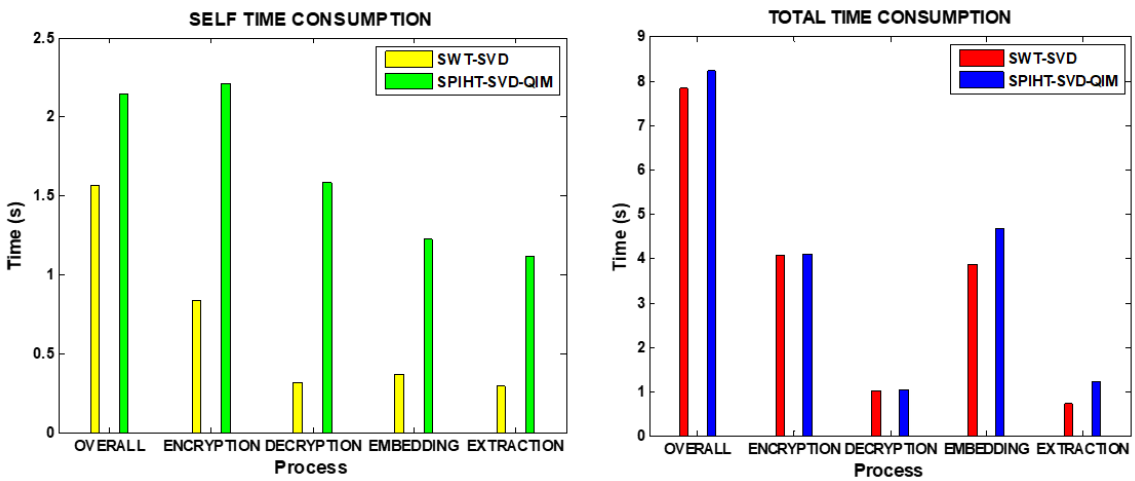
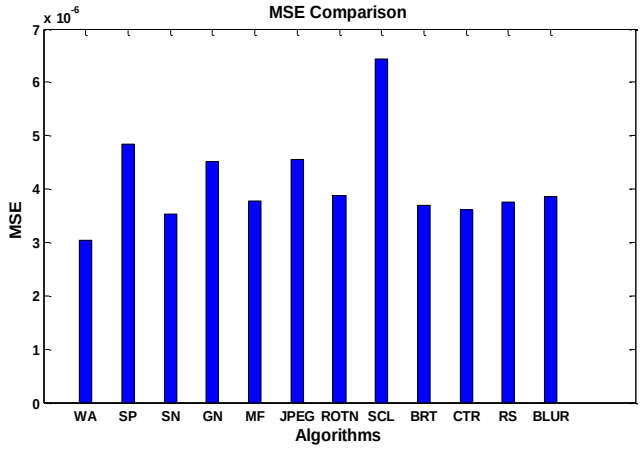


Figure 8. Comparison of self-time and overall time consumed by the proposed and existing method

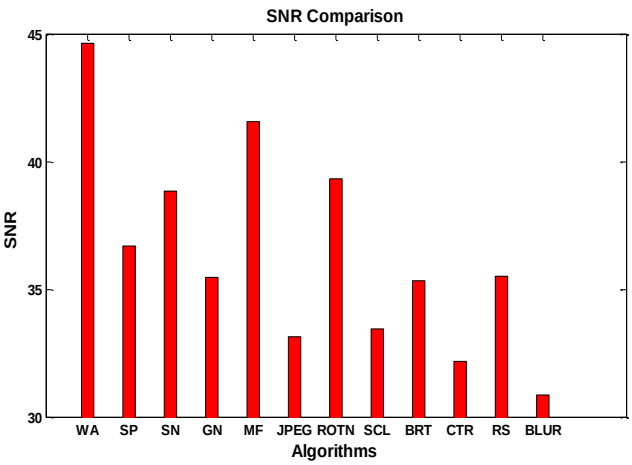
Table 8. Qualitative analysis of watermarked images with different attacks for different set of images

Image	Attack	MSE	SNR	PSNR	NC	Q
Image 1	Without attack	3.73E-06	44.6076	44.9656	0.99862	0.99892
	Salt & Pepper	4.84E-06	36.6945	37.1389	0.9647	0.9685
	Speckle noise	3.53E-06	38.8387	39.2523	0.9611	0.9619
	Gaussian noise	4.51E-06	35.4451	36.0065	0.956	0.9579
	Median filter	3.78E-06	41.5624	41.2285	0.9898	0.9876
	JPEG Compression	4.56E-06	33.1401	30.3734	0.9581	0.9563
	Rotation	3.18E-06	39.3121	40.1145	0.9853	0.9818
	Scaling	6.44E-06	33.4589	34.3232	0.9701	0.9814
	Brightness	3.69E-06	35.309	36.5076	0.9591	0.97
	Contrast	3.12E-06	32.1834	33.4921	0.9593	0.9746
	Rotation & Scale	3.76E-06	35.4956	36.2734	0.9549	0.9754
	Blurring	3.85E-06	30.8542	31.8542	0.9579	0.9296
Image 2	Without attack	4.22E-06	43.6086	44.0656	0.99642	0.98862
	Salt & Pepper	6.21E-06	41.9656	35.5834	0.9665	0.9683
	Speckle noise	5.58E-06	35.4568	37.8936	0.9616	0.9568
	Gaussian noise	6.23E-06	37.9912	34.2724	0.9588	0.9534
	Median filter	4.56E-06	41.0824	42.2010	0.99001	0.9811
	JPEG Compression	5.87E-06	34.9212	28.9289	0.955	0.9762
	Rotation	6.50E-06	28.6821	37.1145	0.9404	0.9777
	Scaling	7.21E-06	36.669	31.3232	0.9699	0.9513
	Brightness	4.00E-06	34.569	34.9521	0.969	0.9598
	Contrast	5.39E-06	34.5076	31.6032	0.9333	0.9445
	Rotation & Scale	9.65E-06	29.9612	31.8289	0.9739	0.9554
	Blurring	8.17E-06	33.0512	28.6319	0.9281	0.9399
Image 3	Without attack	3.80E-06	44.4965	44.8545	0.99442	0.99452
	Salt & Pepper	4.57E-06	36.6945	36.3612	0.9788	0.982
	Speckle noise	4.14E-06	39.0388	37.3946	0.9712	0.9736
	Gaussian noise	3.98E-06	35.8825	34.8897	0.967	0.9716
	Median filter	3.98E-06	41.2502	41.1402	0.9882	0.9865
	JPEG Compression	4.56E-06	30.1401	29.9289	0.9651	0.971
	Rotation	5.31E-06	37.7812	37.4478	0.9505	0.968
	Scaling	6.41E-06	32.3478	32.0145	0.96	0.9614
	Brightness	2.97E-06	35.309	34.9521	0.9496	0.97
	Contrast	4.94E-06	32.0476	31.6032	0.9735	0.9746
	Rotation & Scale	8.44E-06	32.0512	31.8289	0.9538	0.9539
	Blurring	7.77E-06	28.8542	28.6319	0.9179	0.9376
Image 4	Without attack	4.42E-06	43.1632	43.7434	0.98402	0.98932
	Salt & Pepper	5.82E-06	42.5101	35.8921	0.9725	0.9763
	Speckle noise	5.32E-06	36.2846	38.4765	0.9652	0.9722
	Gaussian noise	5.76E-06	38.1267	35.1414	0.963	0.9704
	Median filter	4.63E-06	40.1105	40.2154	0.9786	0.9701
	JPEG Compression	5.40E-06	35.4521	29.3487	0.9647	0.976
	Rotation	5.41E-06	29.039	37.2256	0.9601	0.9716
	Scaling	4.07E-06	38.7812	33.3478	0.9597	0.9711
	Brightness	2.67E-06	34.4589	34.7545	0.9489	0.9797
	Contrast	3.62E-06	34.309	31.4056	0.9334	0.9645
	Rotation & Scale	7.71E-06	30.9612	32.7178	0.934	0.9653
	Blurring	5.61E-06	33.8289	29.2986	0.918	0.9365
Image 5	Without attack	4.04E-06	44.2989	44.7434	0.98752	0.98942
	Salt & Pepper	5.06E-06	36.5834	36.249	0.9666	0.9784
	Speckle noise	3.64E-06	39.1886	38.8646	0.9568	0.9735
	Gaussian noise	2.90E-06	35.9887	35.6768	0.954	0.971
	Median filter	4.35E-06	41.2354	41.1105	0.9785	0.9766
	JPEG Compression	4.50E-06	29.9289	29.5956	0.965	0.9762
	Rotation	5.02E-06	37.9787	37.5589	0.98	0.9715
	Scaling	6.38E-06	32.569	32.2121	0.9596	0.971
	Brightness	2.62E-06	35.4212	35.0632	0.9691	0.9647
	Contrast	4.35E-06	31.9612	31.6278	0.9435	0.963
	Rotation & Scale	8.10E-06	32.8289	32.4956	0.9341	0.9689
	Blurring	6.98E-06	29.4962	29.1875	0.9211	0.9376
Image 6	Without attack	4.84E-06	44.9656	45.7434	0.99202	0.99842
	Salt & Pepper	5.98E-06	43.1878	36.6945	0.9665	0.9783
	Speckle noise	3.94E-06	36.8198	39.2436	0.9615	0.9726
	Gaussian noise	3.61E-06	39.1268	36.0069	0.96	0.9704
	Median filter	5.1486	41.2587	42.0105	0.9802	0.9833
	JPEG Compression	5.81E-06	35.9212	29.7932	0.955	0.9769
	Rotation	6.03E-06	29.1401	37.3121	0.9501	0.9781

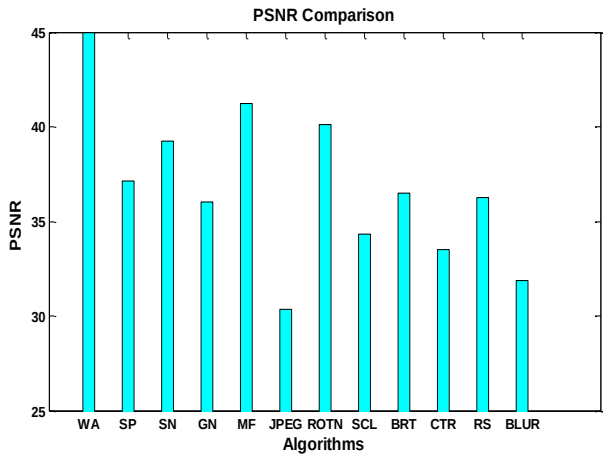
	Scaling	8.66E-06	37.1145	31.6812	0.9597	0.9761
	Brightness	4.00E-06	34.9034	35.309	0.9589	0.9777
	Contrast	7.16E-06	34.4101	31.4921	0.9633	0.972
	Rotation & Scale	1.00E-05	29.7389	31.4956	0.9457	0.9679
	Blurring	7.87E-06	32.939	28.5208	0.9281	0.9447
Image 7	Without attack	5.48E-06	44.0767	44.7434	0.99842	0.99612
	Salt & Pepper	6.94E-06	36.5834	36.249	0.9765	0.9582
	Speckle noise	4.88E-06	39.1368	38.7658	0.97	0.9532
	Gaussian noise	4.36E-06	35.8825	35.4458	0.9654	0.9504
	Median filter	5.86E-06	41.2539	41.5142	0.9833	0.9891
	JPEG Compression	6.30E-06	29.7932	29.4845	0.9748	0.966
	Rotation	7.26E-06	37.669	37.2256	0.9703	0.9817
	Scaling	9.17E-06	31.9034	31.569	0.9598	0.9811
	Brightness	4.97E-06	35.1989	34.8656	0.969	0.9799
	Contrast	8.26E-06	31.849	31.4921	0.9434	0.959
	Rotation & Scale	4.02E-05	31.8289	31.3845	0.9337	0.9709
	Blurring	8.91E-06	28.5208	28.2986	0.928	0.9437



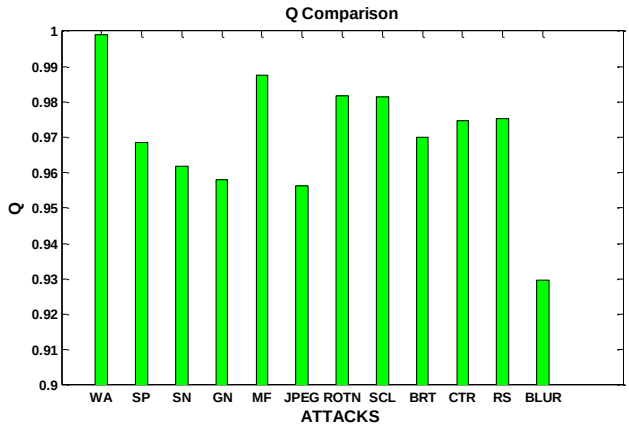
(a)



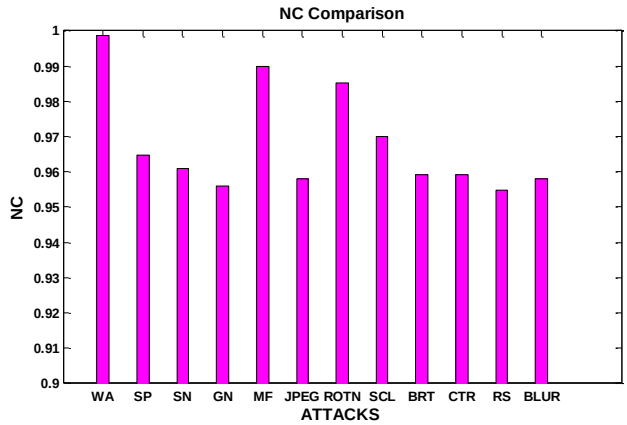
(b)



(c)



(d)



(e)

Figure 9. Qualitative analysis of watermarked images with different attacks for different set of images (a) MSE, (b) SNR, (c) PSNR, (d) NC and (e) Q value

5. DISCUSSIONS AND FINDINGS

Image encryption and watermarking are critical in safeguarding digital content, especially with the rise of unauthorized access and distribution. Enhanced chaos-based encryption techniques have been widely adopted for their ability to create highly secure and unpredictable encryption keys. Chaos theory, which leverages the sensitivity to initial conditions, enables the generation of complex patterns that are nearly impossible to predict or replicate. In the context of image encryption, this enhanced chaos can be applied to pixel

scrambling or key generation, resulting in encrypted images that resist various cryptographic attacks. The security provided by such methods makes them particularly suitable for high-stakes applications where data integrity and confidentiality are paramount. For tasks that need both safe copyright protection and high-quality image transmission, our hybrid solution shines.

Watermarking using SPIHT decomposition is a robust approach for embedding watermarks within an image. SPIHT is a wavelet-based compression technique that preserves the hierarchical structure of image data, allowing for efficient compression and progressive image transmission. By integrating watermarking within the SPIHT framework, the watermark can be embedded in the most significant wavelet coefficients, ensuring that it remains intact even after compression. This method enhances the imperceptibility of the watermark while maintaining the quality of the original image. Moreover, the hierarchical nature of SPIHT allows for multi-level watermarking, where different levels of wavelet decomposition can carry varying strengths of watermark signals, balancing robustness and transparency. Singular Value Decomposition (SVD) combined with Quantization Index Modulation (QIM) embedding further strengthens the watermarking process. SVD is a powerful mathematical tool used to decompose an image into its singular values, which can then be modified to embed a watermark. QIM is a robust watermark embedding technique that adjusts the quantization of the singular values to encode the watermark. The synergy between SVD and QIM ensures that the watermark is resilient to common attacks like noise addition, compression, and geometric transformations. Additionally, the SVD-based approach benefits from the stability of singular values, meaning that minor modifications do not significantly alter the visual quality of the image. This combined technique is highly effective in maintaining the balance between watermark imperceptibility and robustness, making it ideal for applications that require secure image transmission and authentication.

6. CONCLUSION

Encrypting and watermarking medical images is essential for protecting private patient data from unwanted access and adhering to privacy laws such as HIPAA. While watermarking provides an extra degree of protection by directly embedding ownership or authentication information into the image, encryption secures the data during transmission and storage. These methods work together to protect data integrity which makes them essential for maintaining confidence in digital healthcare systems. A strong and secure framework for medical image protection is provided by the combination of watermarking utilizing SPIHT-SVD and QIM embedding, with chaos-based encryption optimized using the FOA. Because the encryption process is very resistant to unwanted access, the chaos-based technique guarantees great sensitivity to beginning circumstances. By adjusting chaotic settings, FOA optimization improves encryption even further and guarantees maximum security. When SPIHT-SVD and QIM embedding are used together for watermarking, image quality is preserved and the watermark is safely and invisibly implanted. This integrated technique is ideal for situations where data security and integrity are critical since it provides a complete solution for protecting sensitive medical images. Future work could explore the integration of advanced chaotic maps and deep learning for watermark. Additionally, testing the method's robustness against emerging threats like quantum attacks could be valuable. Finally, real-time implementation and performance evaluation in large-scale healthcare systems would be an important step toward practical deployment.

In order to assess the algorithm's resilience, speed, and compatibility with current systems, future research will concentrate on testing it in actual medical settings, such as clinics and hospitals. We'll also evaluate user acceptability and how it affects clinical processes. This will guarantee the algorithm's real-world usability and improvement.

REFERENCES

[1] Toktas, A., Erkan, U., Toktas, F., Yetgın, Z. (2021). Chaotic map optimization for image encryption using triple objective differential evolution algorithm. IEEE Access, 9: 127814-127832. https://doi.org/10.1109/ACCESS.2021.3111691

[2] Pareek, N.K., Patidar, V., Sud, K.K. (2006). Image encryption using chaotic logistic map. Image and Vision Computing, 24(9): 926-934. https://doi.org/10.1016/j.imavis.2006.02.021

[3] Gao, H., Zhang, Y., Liang, S., Li, D. (2006). A new chaotic algorithm for image encryption. Chaos, Solitons & Fractals, 29(2): 393-399. https://doi.org/10.1016/j.chaos.2005.08.110

[4] Darwish, S.M., Elmasry, A., Ibrahim, A.H. (2020). Parameter estimation for chaotic systems using the fruit fly optimization algorithm. In the International Conference on Advanced Machine Learning Technologies and Applications (AMLTA2019), pp. 80-90. https://doi.org/10.1007/978-3-030-14118-9_9

[5] Lei, X., Du, M., Xu, J., Tan, Y. (2014). Chaotic fruit fly optimization algorithm. In Advances in Swarm Intelligence: 5th International Conference, ICSI 2014, Hefei, China, pp. 74-85. https://doi.org/10.1007/978-3-319-11857-4_9

[6] El Assad, S., Farajallah, M. (2016). A new chaos-based image encryption system. Signal Processing: Image Communication, 41: 144-157. https://doi.org/10.1016/j.image.2015.10.004

[7] Liu, Y., Wang, Z., Si, L., Zhang, L., Tan, C., Xu, J. (2017). A non-reference image denoising method for infrared thermal image based on enhanced dual-tree complex wavelet optimized by fruit fly algorithm and bilateral filter. Applied Sciences, 7(11): 1190. https://doi.org/10.3390/app7111190

[8] El Abbadi, N.K., Al Rammahi, A., Redha, D.S., AbdulHameed, M. (2014). Image compression based on SVD and MPQ-BTC. Journal of Computer Science, 10(10): 2095-2104. https://doi.org/10.3844/jcssp.2014.2095.2104

[9] Rema, N.R., Oommen, B.A., Mythili, P. (2015). Image compression using SPIHT with modified spatial orientation trees. Procedia Computer Science, 46: 1732-1738. https://doi.org/10.1016/j.procs.2015.02.121

[10] Jayakar, M., Babu, K.A., Srinivas, D.K. (2011). Color image compression using SPIHT algorithm. International Journal of Computer Applications, 16(7): 34-42. https://doi.org/10.5120/2023-2728

[11] Zhang, X., Xu, Y., Yu, C., Heidari, A.A., Li, S., Chen,

H., Li, C. (2020). Gaussian mutational chaotic fruit fly-built optimization and feature selection. Expert Systems with Applications, 141: 112976. https://doi.org/10.1016/j.eswa.2019.112976

[12] Jiao, G., Li, L., Zou, Y. (2019). Improved security for android system based on multi-chaotic maps using a novel image encryption algorithm. International Journal of Performability Engineering, 15(6): 1692. https://doi.org/10.23940/ijpe.19.06.p20.16921701

[13] Chen, Y.Y. (2007). Medical image compression using DCT-based subband decomposition and modified SPIHT data organization. International Journal of Medical Informatics, 76(10): 717-725. https://doi.org/10.1016/j.ijmedinf.2006.07.002

[14] Xiang, T., Qu, J., Yu, C., Fu, X. (2012). Degradative encryption: An efficient way to protect SPIHT compressed images. Optics Communications, 285(24): 4891-4900. https://doi.org/10.1016/j.optcom.2012.06.097

[15] Zhang, M., Tong, X. (2017). Joint image encryption and compression scheme based on IWT and SPIHT. Optics and Lasers in Engineering, 90: 254-274. https://doi.org/10.1016/j.optlaseng.2016.10.025

[16] Sutradhar, S., Karforma, S., Bose, R., Roy, S. (2023). A dynamic step-wise tiny encryption algorithm with fruit fly optimization for quality of service improvement in healthcare. Healthcare Analytics, 3: 100177. https://doi.org/10.1016/j.health.2023.100177

[17] Saravanan, S., Sivabalakrishnan, M. (2021). A hybrid chaotic map with coefficient improved whale optimization-based parameter tuning for enhanced image encryption. Soft Computing, 25(7): 5299-5322. https://doi.org/10.1007/s00500-020-05528-w

[18] Veni, M., Meyyappan, T. (2019). Digital image watermark embedding and extraction using oppositional fruit fly algorithm. Multimedia Tools and Applications, 78: 27491-27510. https://doi.org/10.1007/s11042-019-7650-0

[19] Rao, G.S., Muddada, L.P.R., Rao, B.P. (2019). Comparative analysis of SVD and progressive SPIHT techniques for compression of MRI and CT Images. In Proceedings of International Conference on Sustainable Computing in Science, Technology and Management (SUSCOM), Amity University Rajasthan, Jaipur-India. https://doi.org/10.2139/ssrn.3352392

[20] Ingaleshwar, S., Dharwadkar, N.V. (2023). Water chaotic fruit fly optimization-based deep convolutional neural network for image watermarking using wavelet transform. Multimedia Tools and Applications, 82(14): 21957-21981. https://doi.org/10.1007/s11042-020-10498-0

[21] Tripathi, D., Sharma, S. (2016). A robust 3-SWT multiple image steganography and contrast enhancement technique. In 2016 International Conference on Inventive Computation Technologies (ICICT), Coimbatore, India, pp. 1-6. https://doi.org/10.1109/INVENTIVE.2016.7823256

[22] Abirami, R., Malathy, C. (2024). Medical image security by crypto watermarking using enhanced chaos and fruit fly optimization algorithm with SWT and SVD. Multimedia Tools and Applications, 83(27): 70451-70476. https://doi.org/10.1007/s11042-024-19019-9

[23] Kaggle. Dicom images https://www.kaggle.com/code/adkarhe/dicom-images/input?select=stage_2_images.