A Critical Study on Encryption Based Compression Techniques

C. Sankara Narayanan[1*], S. Anna Durai[2]

[1] Research Scholar, Anna University, Principal of Government Polytechnic College, Chennai 600012, Tamil Nadu, India.

[2] Principal, Sri Krishna College of Engineering and Technology, Coimbatore, Tamil Nadu, India.

*Corresponding author. Tel.: +914426440844; email: sankaranarayananphd15@gmail.com

Manuscript submitted August 8, 2015; accepted November 25, 2015.
doi: 10.17706/jcp.11.5.380-399

Abstract: The main goal of this study is to present an overview on the various encryption based compression techniques. By surveying the various encryption, compression and embedding techniques, the optimal encryption based compression technique is estimated. In communication systems, the data from the source is first compressed and then encrypted before forwarding over a channel to the receiver. But, in multiple practical scenarios, the image encryption has to be performed prior to the image compression. This leads to the problem of how to formulate the combination of encryption and compression algorithms. Compressing the image after encryption is found to be more efficient. For the image encryption, the input images are initially transformed. The encrypted images are then compressed and reconstructed using standard algorithms. This paper presents a survey on the various image encryption algorithms such as Blowfish algorithm, RC4, Data Encryption Standard (DES), etc. Further, the various image embedding techniques such as Discrete Wavelet Transform (DWT), Undecimated Wavelet Transform (UWT), etc., and the various compression techniques such as Set Partitioning in Hierarchical Trees (SPIHT), Arithmetic Encoding (AE), etc., are surveyed. The analysis results show that the combination of the Blowfish, RC4, UWT and SPIHT can provide high quality reconstructed image.

Key words: Encryption based image compression, compression algorithms, image embedding, and image security.

1. Introduction

The field of encryption is vital in the present era. As the attacks create serious impact, the image security is one of the important concerns. The image encryption and decryption is implemented in the applications such as medical imaging, internet communication, military communication, multimedia systems, telemedicine, etc. The encoder receives the original image and converts it into a bit stream. The mapper accepts the encoded bit stream and converts it into the form of mapped image. If the total data quantity of the mapped bit stream is less than the total quantity of the original input image, then it is called as image compression. Most of the image content encryption algorithms have been proposed in the past few years. To make the image/data in a secure way, it is necessary to encrypt the data before it is stored or transmitted. Government, financial institutions, military and hospital deals with confidential images regarding the patient's images, geographical areas, product and enemy position in defense. Most of this fore mentioned information is gathered and stored on the electronic systems. Further, the information is transmitted across

the network to other systems, which are highly confidential. Hence, protecting the confidential images are a legal and an ethical requirement.

Many encryption algorithms are broadly available and are used for information security. Encryption algorithms can be classified into two types namely 1. symmetric (private) and 2. asymmetric (public) key encryption. The symmetric key encryption is also called as a secret key encryption. It uses only one key to encrypt and decrypt the image. The key must be distributed before the transmission among entities. The strength of the symmetric key encryption is based on the size of the key used. The longer key is harder to break. Some examples of the strong and weak keys of cryptographic algorithms are RC2, DES, 3 DES, AES, RC4, RC6 and Blowfish [1]-[3].

RC2 uses one 64 bits key, DES uses one 64 bits key, 3 DES uses three 64 bits keys, AES uses various bits keys (128,192,256), RC4 uses 40-2048 bits, RC6 uses various bits keys (128,192,256) and blowfish uses (32-448) bits keys. Asymmetric key encryption is utilized to solve the issues of key distribution. Generally, asymmetric keys use both the private and public keys. The public key is used for the encryption and private key is used for the decryption. The computationally intensive public key encryption is based on the mathematical formulations. Moreover, asymmetric encryption systems are almost 1000 times slower than the symmetric systems [2]. Because, the asymmetric system requires more processing power.

Moreover, Image embedding techniques are applied to the encrypted image to improve the security level of the confidential images. Image hiding techniques should be capable of embedding the image into the non-understandable format. Image compression is an application of data compression technique that encodes the original image with fewer bits. Because of the emerging demand for information security, image encryption, decryption has developed into a major research area and has broad applications. The objective of the image compression is to minimize the redundancy of the image and to save or transmit the data in an effective manner. Image compression coding is used to store the image into a bit stream as compressed as possible and it spectacles the decoded image in the monitor as exact as possible. In order to provide a high level security, the encrypted image is further integrated with another image and it is embedded based on the wavelet transforms. Then, the embedded image undergoes a compression algorithm to obtain the encoded bits. The process gets repeated to obtain the compressed original image in a secure manner. This paper presents various techniques and algorithms that are used to formulate an encrypted image compression paradigm.

The key objectives of this study are listed below,

- To analyze the various image encryption algorithms such as RSA, DES, AES, Triple DES, Blowfish Algorithm, RC4, Digital Signatures, and Chaos technique based image encryption.
- To study the different image embedding techniques such as DWT, UWT, FFT, and JST.
- To examine the popularly used image compression techniques such as SPHIT, RLE, DCT, AE, and Huffman Encoding. .
- To compare the advantage and disadvantage of each technique.
- To estimate the optimal encryption based image compression technique.

2. Encryption Based Image Compression Techniques

As shown in Fig. 1, the encryption based image compression techniques consist of four major stages,

1) Image encryption
2) Image embedding
3) Image compression and
4) Image decryption.

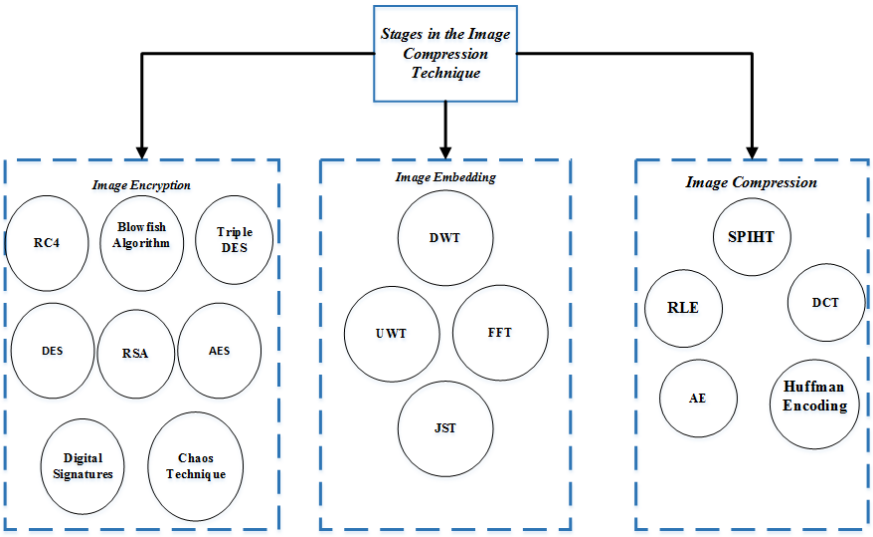

Fig. 1. Categorization of stages in the encryption based image compression techniques.

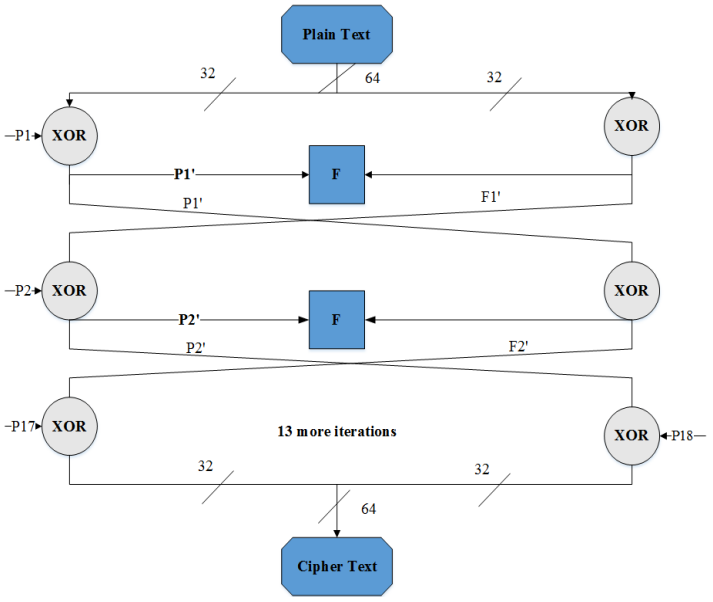

Fig. 2. Blowfish encryption.

2.1. Encryption Algorithms

This section presents some of the key encryption algorithms used for the image encryption. Some of the algorithms are Blowfish algorithm, RC4, Data Encryption Standard (DES), Triple DES, Advanced Encryption Standard (AES), RSA, Digital Signatures, chaos technique.

2.1.1. Blowfish algorithm

Blowfish algorithm is a symmetric block cipher that can be used for encryption and safeguarding of data. It takes a variable length sized keys from 32 bits to 448 bits [4]. The block diagram of the blowfish algorithm is depicted in Fig. 2. Each encryption round has a key independent permutation and a key and data dependent substitution. The XOR and addition operations are performed on each 32 bit word. Blowfish is a Feistel network, and it iterates an encryption function for 16 times. If the block size is 64 bits, then length of the key can be any lengths up to 448 bits. Moreover, a complex initialization process is essential before the encryption process is carried out.

The salient features of the blowfish algorithm are as follows:

1) Uses very modest operations like addition and XOR operations.

2) 64 bit sized block.
3) Scalable key from 32 to 256 bits.
4) It employs data in large blocks.
5) It is compact and the execution is completed less than 5kb memory.
6) It is a fast and efficient, that this algorithm. But uses 32 bit microprocessor.

At a time, Blowfish algorithm can encrypt the block data of 64 bits. This algorithm is divided into two categories, 1. Key expansion and 2.Data encryption.

2.1.1.1. Key expansion

It converts a variable length key of utmost 448 bits (56 bytes) into various sub key arrays sum-up to 4186 bytes. Before the encryption and decryption process, the keys are generated. There is an array A and four 32 bit S boxes. The array includes 18 entry 32 sized bit sub keys and four S boxes. Each box holds 256 entries. The function generation is described below:

1) The 32 bits input can be divided into four subparts such as a, b, c and d. Each subparts is b bit length.
2) Each subpart arrive to the corresponding four S-boxes as 1, 2, 3 and 4.
3) The consistent 32 bit outputs are undergoing the XOR operation and addition modulo 2^{32}.
4) The final result is 32 bits key.

The key generation can be mathematically defined as:

$$G = \{(S1[a] + S2[b])XOR(S3[c] + S4[d]\} \quad (1)$$

The generated key is used to formulate 18 entry 32-bits sub keys and four 8×32 boxes includes 1024 bits entries (4186 bytes).

2.1.1.2. Data encryption

Each round of the algorithm includes a data dependent substitution and key dependent permutation. The entire process of encryption and the various types of encryption are defined as follows:

1) Divide the 64 bit blocks into two equal sized blocks having 32 bit size (LB and RB).
2) The left block (LB) is XORed with the first element of block K, the result obtained (K_1) is then fed into the function G.
3) Then, the substitution operation is carried out in the G function, where the 32 bit given input is converted into another 32 bit output.
4) The output from G is XORed with right half (RB) and the results are swapped.
5) After concluding each round successfully, the RB becomes the new LB or vice versa.
6) These steps can be repeated up to 16 rounds.
7) The final left and right halves are not swapped, but it is XORed with the 17^{th} and 18^{th} box elements.
8) Hence, the result is the cipher text and it is non understandable for attackers and outsiders.

Advantages

- The main advantage of the blowfish algorithm is its speed.
- The loss of information is prevented in both the encryption and decryption phases.

Disadvantage

- The encryption of the blowfish algorithm is not safe because, the algorithm is greatly dependent on the symmetric key. If the key is hacked, then the blowfish security will be completely destroyed [5].

2.1.2. RC4

RC4 stream cipher algorithm is the most desired stream cipher algorithm. In RC4 algorithm [6], there are two phases that are available during the encryption and decryption process. The algorithm can be categorized into two parts. One is Key Scheduling Algorithm (KSA) and the other one is a Pseudo Random

Generator Algorithm (PRGA). KSA as the initial step of an algorithm known as the initialization of S and PRGA is known as stream generation in the RC4 whole process. The working steps of the RC4 encryption algorithm is depicted in Fig. 3.

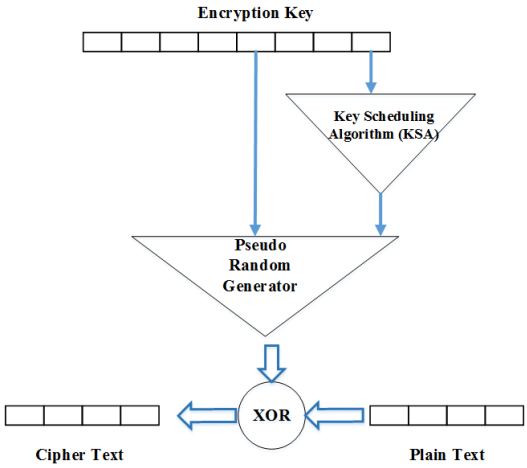

Fig. 3. RC4 encryption algorithm.

During the initial step of RC4 stream cipher on the bases of variable sized key from 1 to 256. A state vector of fixed length (256 bytes) is generated on the base of the state table. During the encryption and decryption, the stream cipher is generated based on XOR with the cipher text and the plain text during the encryption and decryption. The key stream is XORed with the plaintext during the encryption phase and the cipher text is XORed with the key, stream is then converted into the plain text during the decryption stage.

Steps for Key Scheduling Algorithm:

1) Input the variable length sized key from 1-256.
2) Initialize the key matrix based on the size of the input key.
3) Initialize the state table of stable sized bytes (256 bytes) from 0-255 in ascending order.
4) The permutation is done on the table based on the key matrix of variable size.
5) After shuffling process, the output of Key Derivation Algorithm (KDA) is obtained from the table S.

Based on the above steps, the state table (state matrix) of 256 bytes is generated.

Steps for Pseudo Random Generation Algorithm:

The steps of the algorithm consists in generating a key stream of the size of the message to encrypt. The algorithm enables to generate a key stream of any size.

1) Initialize two indexes to 0 (*i.e i=j=*0)
2) Compute new values of *i* and *j* as given below:

a. $i = (i + 1) mod\ 256$

b. $j = (i + s[i]) mod 256$

3) Swap $s[i]$ and $s[j]$ to have a dynamic state (it makes it obviously harder to crack them if the state is computed only once and used for the generation of the whole stream.
4) Retrieve the next byte of the key stream from the S array at the index $(s[i] + s[j]) mod\ 256$.

Advantages

- RC4 has variable key length
- Mostly preferred for secured communication.
- Energy Efficient
- It can be used only once
- 10 times faster than the DES encryption algorithm [7]

Disadvantages

- Vulnerable to analytic attacks
- The keys of RC4 are weak [7]

2.1.3. Data encryption standard (DES)

DES is a 64 bit block cipher underneath 56 bit key [8]. DES algorithm can process with an initial permutation, 16 rounds block cipher and final permutation. The application of the DES algorithm is very widespread in military, commercial and other domains in the last few decades. Even though, this algorithm is public and the design issues used are classified. It has some drawbacks particularly in the selection of 56 bit key algorithms as it can be vulnerable to brute force attacks. In order to improve this, 2DES and 3DES algorithms are developed. The single round of DES is depicted in Fig. 4.

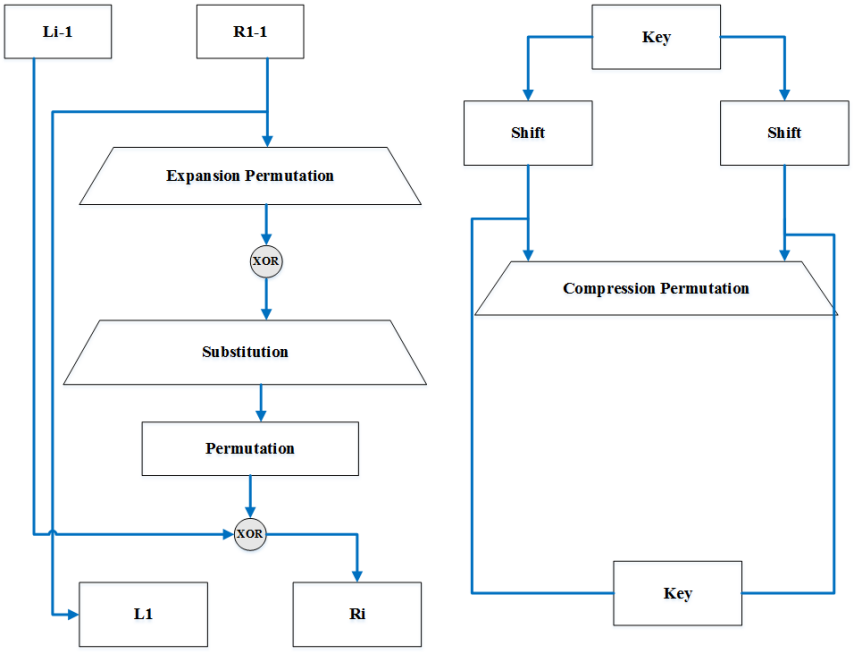

Fig. 4. Single round of DES.

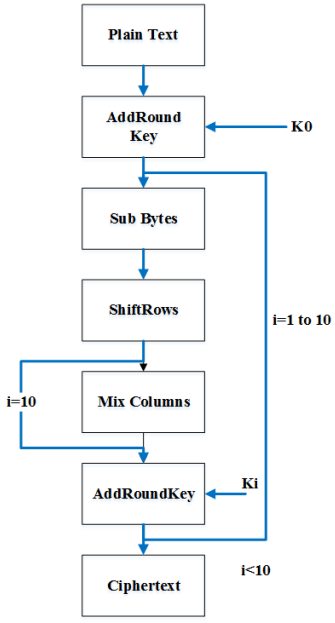

Fig. 5. Flowchart for the AES encryption algorithm.

2.1.4. Triple DES (3DES)

The 3DES algorithm is essential for the replacement of DES algorithm due to its improvement on key searching. It has three round message [8]. It provides strongest encryption algorithm, since it is harder to break the 2^{168} possible arrangements. It reduces the memory requirements among the keys. The major drawback of this algorithm is time consuming than the blowfish and RC4 encryption.

2.1.5. Advanced encryption standard (AES)

AES algorithm uses the Rijndael block cipher and the block lengths are 128,192 or 256 bits [8]. If the length of the block and length are 128 bits, then Rijndael performs 9 processing rounds. If the length of block and key is 192 bits, then 11 processing rounds are performed. If the length is 256 bits, then it performs 13 processing rounds. In [9], the encryption and decryption of the image is performed using the AES. The encryption process consumes 128 bit key size and a plain text. The word is converted into an 8 bit sequence. Various numbers, symbols and letters are used for the encryption process. The flowchart of the AES encryption algorithm is depicted in Fig. 5. In [10], the encryption performance is increased by adding the key stream generator to the AES.

The algorithm supports the data and key size that are 128, 192 and 256 bits. In [11], a 128 bit AES is used for the image encryption and decryption. The authenticated information is protected against the

unauthorized access. The number of clocks used for the image encryption and decryption is 84 and the number of clocks used for the AES encryption and decryption is 24 clocks. The AES encryption algorithm is fast, secure, simple and flexible, but the major demerits of this algorithm is that it requires more rounds of communication as compared with the Blowfish and RC4 algorithm. Also, AES needs more processing time, it is not much suitable for the real time applications. The Table 1 gives the information such as founder, year identified, size of the key in bits and block size in bits for different encryption algorithms.

2.1.6. Rivest-Shamir-Adleman (RSA) encryption algorithm

It is a secure public key encryption method [12]. With the encryption key (e, n), the RSA algorithm splits the larger messages into many small blocks. Each block is represented with a specified range of integer. The messages are encrypted to the eth power modulo n. This results in the cipher text message C. The cipher text C is decrypted by applying d modulo n. The encryption key (e, n) is made public whereas, the decryption key(d, n) is maintained private. The working principle of the RSA is depicted in Fig. 6. When compared to the DES and blowfish, the RSA is faster and the data is maintained more secure [13].

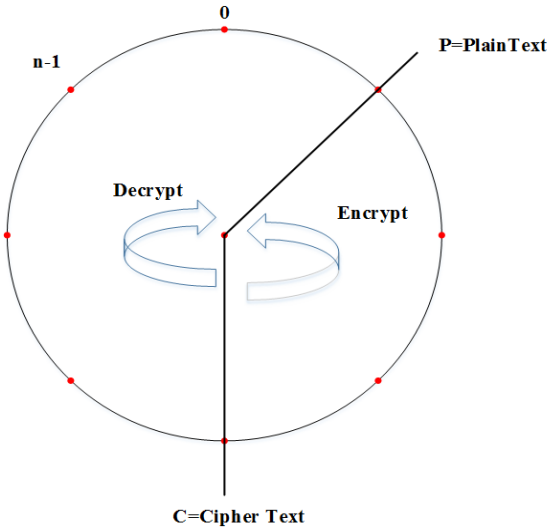

Fig. 6. Working of RSA algorithm.

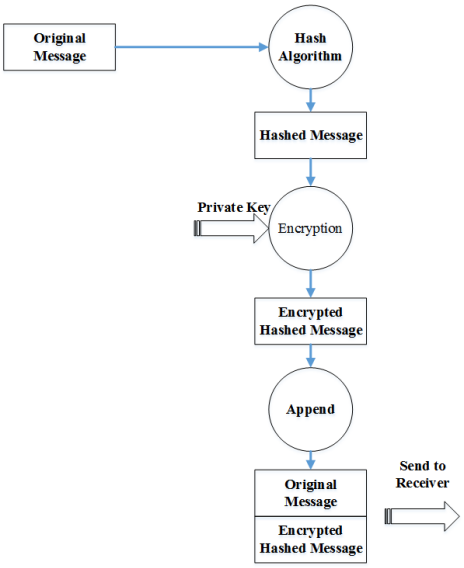

Fig. 7. Working of the digital signature encryption algorithm.

The RSA is more secure from multiple attacks, but the demerits of the RSA are low speed, demand for key deposit, not appropriate for global system.

2.1.7. Digital signatures

The digital signatures for an image is produced using the one-way hash function. The message of variable length is converted into a fixed sized message using standard digital image algorithms. The hash function for an image is unique, it is very difficult to duplicate it. Before transmitting the messages, it is combined with the public key encryption algorithm like the RSA. In [14], the Bose-Chaudhuri decryption Hochquenghem (BCH) is used to encode the image. At the receiving end, the digital signature is used to validate the authenticity of the image. The steps involved in the digital signature based encryption algorithm is depicted in Fig. 7.

The merits of the Digital Signature are authenticity for the source of the message, integrity for the messages of both the sender and the receiver, minimal processing time, reduced overhead, optimal time management, etc. The key disadvantages of the digital signature are non-repudiation, increased cost, difficult to frame standards, increased legal and security issues, etc.

2.1.8. Chaos technique

Applying the Chaotic technique for the image encryption results in higher efficiency. The key properties of the chaotic systems are sensitive dependence on the system parameters, pseudorandom property, etc. The steps involved in the chaos based image encryption is depicted in figure 8. As the figure shows, the chaos technique has two steps namely, chaotic confusion and pixel diffusion. The chaotic confusion is used to permute the plain input image into a 2D chaotic map. The pixel diffusion is used to alter the value of the pixels one by one. The parameters of the chaotic map is considered as the confusion key. The parameters of the diffusion process is considered as the diffusion key. In [15], the dynamic chaotic system is utilized to shuffle the image pixel positions. It efficiently handles the huge key space, smaller iterations and higher security analysis.

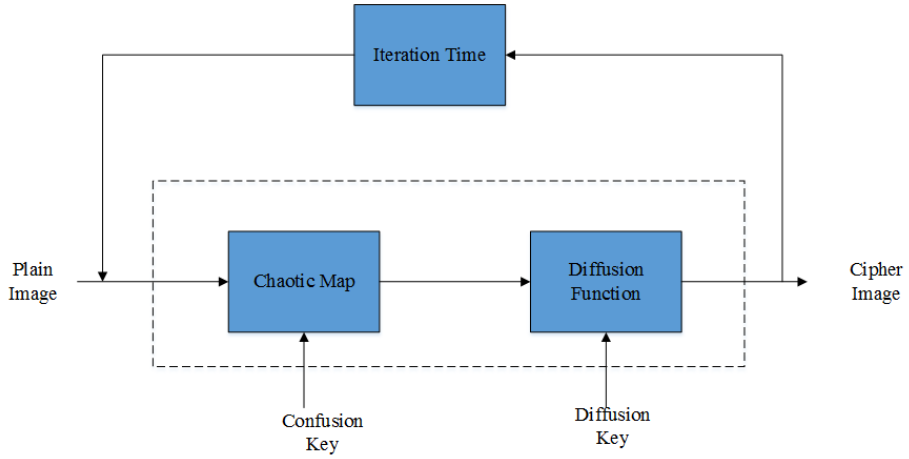

Fig. 8. Chaos based image encryption scheme.

The advantages of the chaos technique are increased efficiency and simplicity. The demerits are small key size, and less security.

2.2. Image Embedding Techniques

In order to provide high level security to the encrypted image, the encrypted image is further integrated into another image with the help of image embedding approaches. The embedded image is retrieved and tested to ensure that the image has not been tampered. In this embedding technique, a region of an image is copied and pasted into another region of the same image to hide the original content in the copied region. Whereas pasting, the copied portion may be scaled or rotated. Due to irregularity of the copied and pasted portions, the edges are habitually blurred to reduce the trace of forgery.

2.2.1. Discrete wavelet transform (DWT)

The wavelet transformation defines a multi resolution decomposition process in terms of extension of an image into a set of wavelet functions [16]. As shown in Fig. 9, the given input image is divided into 4 non overlapping multi resolution subbands by the following filters: LL1, LH1, HL1 and HH1. LL1 is processed to get the next coarser scale of wavelet coefficients until the final scale N is stretched. Initially, the HPF and the LPF is applied to each row image data. The output of the filters are down sampled by 2 to obtain the high frequency and the low frequency components. Then, the high and the low pass filters are applied again to the high and the low frequency components of the column. The results of the filters are again down sampled by 2. This process is repeated till the four sub band images such as, HH, HL, LH, LL are generated. Each sub band image has its own features. The low frequency information is placed in the LL band and the high frequency information is placed in the HH, HL and LL bands.

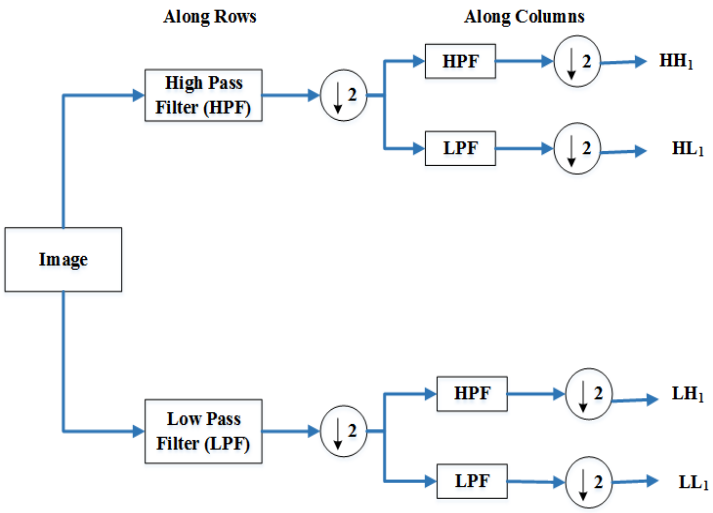

Fig. 9. DWT image decomposition.

The embedded secret image in the lower frequency LL subbands (LLX) may worsen the image significantly. Embedding the low frequency subbands might increase the robustness. In [17], the DWT and Singular Value Decomposition (SVD) are used in the watermarking schemes. Once the image is splitted into four bands, the SVD is substituted to each band and the image embedding is performed by changing the singular values. By changing all the frequencies, a security attack resistant watermarking scheme is developed. In [18], the DWT based watermarking scheme uses both the blind and non-blind algorithms. In addition to protecting the image, the watermarking scheme protects the image from misuse. The attacks are prevented by deploying the pseudo random generator in multiple stages of the algorithm. The DWT based technique uses randomness in selecting the location for embedding the watermark. The major limitation of DWT is, the cost of computing is higher when compared with the Undecimated Wavelet Transform (UWT). Then, the use of larger DWT basis functions produces blurring and ringing noise edge regions in images or video frames. It takes longer compression time and lower quality than JPEG and UWT techniques.

2.2.1.1. Undecimated wavelet transform (UWT)

Though the widely used DWT has been popular in image processing, its performance is far optimal in applications like pattern recognition, detection or more specific analysis of data [19]. This is due to bi-orthogonal wavelet transform is lacking the translation inverse property. To overcome the issues of DWT, translation invariant UWT is introduced. The UWT does not include any decimation. It is applied for both down sampling in the forward wavelet transform and up-sampling in the inverse wavelet transform. Here, the number of the wavelet coefficients does not shrink among the scales. This additional information can be important for better understanding and analysis for the signal characteristics.

Consider M be the input image, $l[k]$ and $h[k]$ are the low and high pass filters. The UWT of an image can be calculated based on the following *atrous algorithm*. Initiate at scale j=0 and M^0=M and calculate the scaling and wavelet coefficients at various scales j=1, 2,....J

$$a^{j+1}[n] = \sum_k l[k]a^j[n + 2^j k] \tag{2}$$

$$b^{j+1}[n] = \sum_k h[k]a^j[n + 2^j k] \tag{3}$$

Let $l^j[k]$ and $h^j[k]$ be the filters attained by injecting $(2^j - 1)$ zeros between the terms of $l[k]$ and $h[k]$.

Step 1: Initiate with M, assumed to be at scale zero, i.e. M^0=M.

Step 2: To attain the scaling and wavelet coefficients Q^j and W^j at scales j=1, 2,..., J

$$\text{Filter } M^{j-1} \text{ with } l^{j-1}[k] \tag{4}$$

$$\text{Filter } M^{j-1} \text{ with } h^{j-1}[k] \tag{5}$$

In UWT, there is no down sampling is involved. In the wavelet transform, I^j is called the low pass subband (L) and W^j are termed as high pass subbands (H). In case of two dimensional images, four subbands LL, LH, HL and HH are found at each scale of the decomposition. The size of each of these subbands is similar as the original image. The use of Undecimated Discrete Wavelet Transform (UDWT) for denoising the image provides optimal accuracy and smoothness than the traditional DWT [20]. But, the redundancy introduced by the UDWT is high than the dual tree complex wavelet transform [21].

2.2.1.2. Fractional Fourier Transform (FFT)

The FFT is a generalized form of the standard Fourier Transform (FT) [22]. It is used to process the non-stationary signals such as linear chirps. FFT domain can be considered as a combination of time and frequency domains. The standard FT corresponds to a rotation angle 90^0 for the time frequency plane. It is a special case of all the FFT domains. The one dimensional FFT with the transformation angle θ is defined as [23], [24] follows:

$$D^{\theta}\{q(t)\}(v) = \int_{-\infty}^{\infty} q(t)K^{\theta}(t,v)dt \tag{6}$$

The inverse FFT is defined as follows:

$$q(t) = D^{-\theta}\left\{D^{\theta}\{q(t)\}\right\}(t) \tag{7}$$

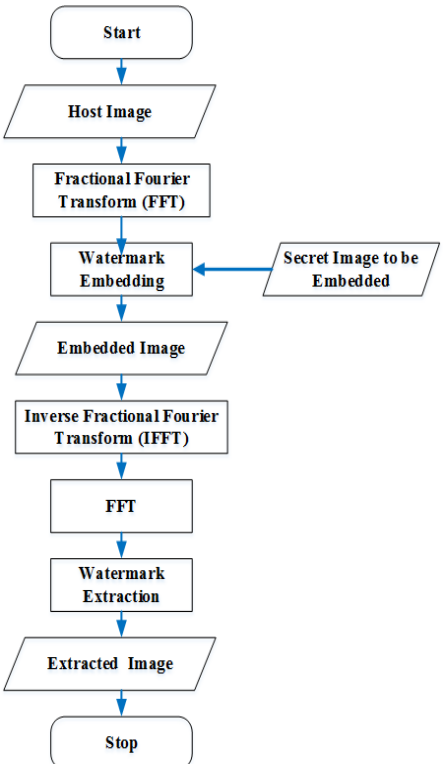

Fig. 10. State Flow diagram for FFT.

FFT cannot apply a filter directly to remove the noise, but with the help of the FFT the signal can be rotated which includes the preferred signal and noise. In [25], the FFT is used for the Digital Image

watermarking. When compared to the traditional information hiding techniques, the FFT based digital image watermarking provides better performance. In [26], the 2D discrete FFT of the image is computed, and then the watermark logo is exploited for transforming the real and imaginary component coefficients. The watermarked image is retrieved by applying the inverse transform. In [27], the double random fractional order fourier domain encoding technique is used to encrypt the 2D watermark image.

By exploiting the suitable fractional orders and random phase mask, the watermark image is recovered. By using the FFT in embedding the watermark, the security level is increased. An example of the working of FFT in the image embedding process is depicted in Fig. 10. The figure shows that the original images are transformed using the FFT and then a secret image is added to the transformed image. Applying the inverse FFT helps in obtaining the watermarking image. Using the extraction techniques, an embedded secret image is extracted [28]. The advantages of the FFT are, increased security level, less MSE than the other two dimensional transform.

2.2.1.3. Jig Saw transform (JST)

Jig Saw is a unitary transform, it rearranges the blocks for a complex image by means of random way [29]. There are some types of JST based on the number of dimensions such as 2 dimensional JST and 3 dimensional JST. It is used to rearrange the blocks for an image with the corresponding routine to apply the transform on an image. It does not inevitably act pixel-pixel, it might act on blocks with $n \times m$ pixels to form an image. The major drawback in this Jig Saw, it takes more time for transforming the image than the UW transform.

2.3. Image Compression Techniques

2.3.1. Set Partitioning In Hierarchical Trees (SPIHT)

The SPIHT algorithm is a more effective implementation of Embedded Zero Wavelet (EZW) algorithm [30] After applying the wavelet transform, the algorithm divides the decomposed wavelet into significant and insignificant partitions based on the function:

$$H_n(T) = \begin{cases} 1, max_{(i,j)\in T}\{|e_{i,j}|\} \geq 2^n \\ 0, \text{otherwise} \end{cases} \quad (8)$$

where, $H_n(T)$ denotes the significant set of coordinates T and $e_{i,j}$ denotes the coefficient value at (i, j) coordinate. The SPIHT algorithm has two passes: 1. Sorting pass and 2. Refinement pass.

The encoding process uses the following three lists:

1) List of Insignificant Pixels (LIP) – it includes the individual coefficients that have magnitudes smaller than the thresholds.
2) List of Insignificant Sets (LIS) – it includes a set of wavelet coefficients that are described by the type of tree structures. The magnitudes are smaller than the threshold.
3) List of Significant Pixels (LSP) – it includes a set of pixels obtained in magnitudes larger than the threshold (significant).

Merits of SPIHT

1) It provides higher PSNR than the EZW because of a special character that denotes the importance of the child nodes of a substantial parent and separation of child nodes from second generation descendants.
2) SPIHT uses the simple quantization algorithm. It can easily code the exact bit rate or distortion and provides optimized embedded coding. It results fast encoding and decoding.
3) It is very useful in transmitting the images over the internet. Because, the users with slower connection speeds can retrieve only a small portion of the file. It obtains much more advantageous results when compared to the other codec like progressive JPEG.

4) No additional entropy coding need to be applied.

2.3.2. Run length encoding (RLE)

RLE is used to minimize the number of symbols needed to be coded [31]. RLE must satisfy the two conditions as discussed below:

1) Four consecutive coefficients in the same stripe should be irrelevant.
2) All consecutive neighbors of the four coefficients should be irrelevant.

An example of the RLE is shown in Fig. 11. The steps involved in the RLE are as follows [32],

Step1: Input the string

Step 2: obtain a unique value from the first symbol

Step 3: Read the next character or string. If the character read is last, then exit.

(a) If the next symbol is same as the previous symbol, then the same unique value is given as the previous.

(b) If the next symbol is not same as the previous one, then a new value is assigned to the symbol.

Step 5: Repeat step 3 till, a matching symbol same as the previous symbol is obtained.

Step 6: The number of occurrences of the symbol is displayed as the output.

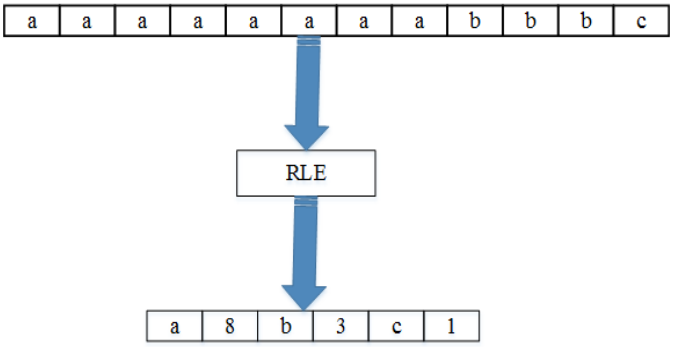

Fig. 11. Example of run length encoding.

One context is required when all the four samples are irrelevant. If any one of the samples becomes relevant, then more than one context is required to indicate the location of the relevant. The disadvantage of RLE is that every scan line is compressed discretely thus resulting in rather weak performance in terms of compression rate.

2.3.3. Huffman encoding

Huffman encoding is based on statistical coding, which refers the probability of a symbol has a direct link on the length of its representation [33]. It uses a particular method for selecting the representation of each symbol. It results in a specific code, which defines the most usual source symbols using shorter strings of bits than less common source symbols. The number of bits needed to denote each character depends on the number of characters to represent. Utilizing one bit, two characters are represented. i.e 0 denotes the first character and 1 denotes the second character. Then using two bits, four characters are represented and so on [34]. In [35], the image pixels are treated as the symbols. The most frequent symbols are assigned a smaller number of bits, whereas, the less frequent symbols are assigned a larger number of bits. Many image compression algorithms use lossy techniques in the initial stages of the compression and use the Huffman coding in the final step. In [36], the Huffman algorithm is divided into two ranges such as, static and adaptive. In static Huffman algorithm, the data is encoded in two passes. During pass 1, the frequency of each symbol is calculated. In the second pass, the Huffman tree is constructed. The adaptive Huffman algorithm constructs the Huffman tree in one pass. The disadvantage of the Huffman coding is, they are slow in reading and writing the files when compared to other techniques.

2.3.4. Arithmetic encoding (AE)

The context and decision data formulated from the context formation is coded in the arithmetic encoder [37]. AE used by JPEG 2000 standard is a binary arithmetic coder called MQ coder. The fundamental of the binary arithmetic encoding is a recursive probability subdivision procedure. There are only two sub intervals are used since it is a binary AE. For each decision, the present probability interval is split into two sub intervals.

If the decision value is 1

More possible symbol

Otherwise, the decision value is 0

Less possible symbol

The basic operation of the AE is computing the new more possible symbol and least possible symbol based on the context and decision form context creation. The efficiency of AE is always better or less identical to a Huffman code. If there is a corrupt bit in the code word, then the whole message becomes corrupted the original image. These algorithms are slower than the SPIHT algorithm. It is fairly complicated to implement, and it doesn't provide the prefix codes. Moreover, random access is more difficult for AE than the SPIHT algorithm.

2.3.5. Discrete cosine transform (DCT)

It divides the original image into the small square blocks before to be transmitted [38]. Then, 2D DCT is applied to each block. The size of the DCT blocks are varied and the performance are evaluated. DWT gives a better compression ratio without losing the information about the image. But DWT needs more processing time. Whereas DCT needs only less processing time but it has blocks artifacts, i.e. it results loss of image information. But these problems are overcome by the SPIHT compression algorithm.

3. Results and Discussion

Table 1. Information about Different Encryption Algorithms

Algorithm	Founder	Year	Key size in bits	Block size in bits
Blow fish	Bruce Schneier	1993	32-448	64
RC4	Ron Rivest	1987	40-2048	248
DES	IBM	1975	56	64
3DES	IBM	1978	112 or 168	64
Rijndael	Joan Daemen& Vincent Rijmen	1998	256	128
RSA	R. Rivest, A. Shamir, and L. Adleman	1978	512	512
Digital Signature	NIST	1991	512-1024	256
Chaotic Technique	Edward Lorenz	1961	128	64

Various techniques for encryption based compression algorithms are depicted. A few experiments have been conducted and the results of the survey are shown in the Tables 2, 3 and 4. Table 2 illustrates the comparative study of different encryption algorithms, such as, RC4-256, AES-256, Blowfish-256, DES, RSA, chaos technique, and Digital Signature. The source of the Lena image and Cameraman image is cited in Table 2. The circuit board image is retrieved from the default images of the Matlab tool. Table 1 compares the experimental results such as encryption time and decryption time for the various encryption algorithms. Table 1 compares the various methods with experimental results such as encryption time and decryption time. From the survey, it is evident that the Blowfish and RC4 combination can result better encrypted image than the existing algorithms such as AES, DES, 3DES, RSA, chaos technique, and Digital Signature. To validate the results obtained from the survey, few experiments are conducted to measure the encryption and Decryption speeds of each algorithm for different images of size 256×256 and 512×512. The

encryption time can be used to calculate throughput of an encryption process. The throughput of the encryption scheme is calculated by dividing the total bytes encrypted on the total encryption time for each of the algorithms. The results show the superiority of the Blowfish algorithm over other algorithms in terms of processing time.

Table 2. Comparative Study of Different Encryption Algorithms

Techniques	Image	Encryption time (ms)	Decryption time (ms)	Total time (ms)	Average throughput Megabyte/sec
RC-4-256	Lena [39] (256×256)	56	62	118	4.04
	Camera man [40] (256×256)	61	67	128	
	Circuit board (256×256)	92	143	265	
AES-256	Lena 256×256	42.2	64.42	106.62	4.23
	Camera man (256×256)	41.8	66.22	108.02	
	Circuit board (256×256)	118.8	80.65	199.45	
BLOWFISH-256	Lena (256×256)	40.1	28.8	68.9	23.9
	Camera man (256×256)	40.6	29.5	70.1	
	Circuit board (256×256)	47.75	70.04	117.79	
DES	Lena (256×256)	36.6	46.64	83.24	4.05
	Camera man (256×256)	36.8	46.68	83.48	
	Circuit board (256×256)	49.8	76.72	126.52	
RSA	Lena (256×256)	43.4	52.3	84.2	4.25
	Camera man (256×256)	40.1	51.6	89.2	
	Circuit board (256×256)	45.6	58.6	111.1	
Digital signatures	Lena (256×256)	44.6	62.3	85.6	4.32
	Camera man (256×256)	42.3	60.1	85.5	
	Circuit board (256×256)	48.7	60.3	121.1	
Chaotic Technique	Lena (256×256)	49.5	58.6	86.7	4.56
	Camera man (256×256)	51.2	56.9	87.7	
	Circuit board (256×256)	55.2	65.2	131.3	

Table 3. Comparison of Image Embedding Algorithms

Technique	Authors & year	Images used	Experimentation	Observations
Undecimated Wavelet Transform (UWT)	Muhammad and Hussain [19]	1. Leaf (256×256)	• The image is transformed using UWT. • The lower part of the leaf of size 64×64 has been copied and pasted in 3 different places.	• The average time taken to transform and copy the image of size 64×64 is 0.378secs • Reconstruction accuracy is 100% • Copy move blocks detection Accuracy is 98.26 • False positive is 3.34%
	Ghulam et al. [41]	1. Tree 200×200 2. Flower 200×200 3. Leaf 200×200	Blind copy move image forgery detection is tested by using LL1 and HH1 sub-bands.	• Part of tree or flower or leaf are copied and moved. • Similarities and dissimilarities between blocks are computed to detect the copy move forgery and the results are as follows: • Accuracy in detecting the copy move blocks is 95.9%. • False positive rate is 4.54% • False negative rate is 6.67%
Fractional Fourier Transform (FFT)	*Olcay et al.* [22]	1. Lena 512×512	The Lena image is transformed by FFT and embedded in another part of the same image as a water mark.	• The time taken to compute FFT of the Lena image and to paste 128×128 sized image in a different part of the image is 0.411 sec. • The reconstruction accuracy is 99.9%

	Ran et al. [42]	1. Lena 256×256	The binary text image of size 128×128 is embedded in the Lena image and transformed using FFT.	• The time taken to embed the text image of size 128×128i in the Lena image and transform the combination to FFT is 0.621sec. • The signal to noise ratio computed is 29db. Mean square error between original and restored Image = 2.04%
Iterative FFT	*Zhengjun et al.* [43]	1. Lena 256×256 2. Rice 256×256 3. Blood cells 256×256 4. Woman 256×256	An image encryption algorithm is applied simultaneously to encrypt two images into a single one	• The Lena and rice images are encrypted to form a single image. Computed Encryption time =0.74 sec, • Mean square error of the reconstructed images are .011 and 0.012 respectively • Encrypted image of blood and woman • Computed Encryption time =0.756 sec, • Mean square error of the reconstructed images are .013 and 0.01 respectively.
Discrete Wavelet Transform (DWT)	*Ali et al.* [44]	1. Lena 256×256 2. Text image 64×64 as water mark	• Two experiments are conducted 1. • With two-level DWT only used to form water marked image and 2. • The combined DWT and DCT are used to form a watermarked image (Lena image as host and text image as water marking image).	**First Experiment results using DWT only**: PSNR of the reconstructed host image is 80.19db, Compression achieved for DWT approach is 0.65 Correlation factor achieved is 0.736 Extracted watermark image PSNR is 40db Robustness performance is not acceptable **Second Experiment Results using DWT-DCT:** PSNR of the reconstructed host image is 97.072db Compression achieved for DWT-DCT approach is 0.5246 Correlation factor achieved is 0.974 Extracted watermark image PSNR is 60db Robustness performance is high and acceptable.
	Jayanta et al. [45]	DWT combine the Vector Quantization (VQ) for image compression. Also, Huffman coding is used to check the image quality.	1. Lena 512×512 2. Baboon 512×512 3. Couple 512×512 4. Frog 512×512 5. Plane 512×512 6. Woman 512×512 7. Boat 512×512 8. Peppers 512×512	• PSNR for Pepper is 30.7, Boat is 29.87, Plane is 28.8 and Woman is 36.41 • CR for Pepper is 38.94, Boat is 36.97, Plane is 41.28 and Woman is 46.91 • Original image • Reconstructed image
Least Significant Bit (LSB)	*Mehmet et al.* [46]	Gray scale images Barbara512×512 Lena 512×512 Mandrill 512×512	It offers a secure, flexible, computationally efficient lossless image authentication watermark with tamper localization ability, low embedding distortion and public/private key support	• The water marking verification and recovery algorithm is tested with 3 different images namely Barbara, Mandrill and Lena. • The experimental results show that the average PSNR of the reconstructed images is 52.09db. • The average increase in pay load size is 3.5%. • The net increase in the file size is 1.5% • The original images are reconstructed without any loss.
	LuteKamstra and Heijmans [47]	1.Barbara 512×512 2.Lena 512×512 3.Mandrill 512×512	Reversible data embedding approaches like LSB prediction and improved Tians are studied	• Both methods allows to embed the large amount of data image • Experimental results show that embedding capacity from 0.5 to 1.0 bpp can be achieved using LSB method at 45db to 55db of PSNR. • Improved Tians method gives 5db better PSNR than LSB. • The two methods can control embedding capacity very well.

Comparative results of the image embedding algorithms are shown in Table 3. The measures such as false positive, false negative, accuracy and MSE values of each image embedding algorithm is compared. From the

comparison, it is evident that the Undecimated Wavelet Transform (UWT) can efficiently integrate the source image with the other image into a non-understandable format than the existing algorithms such as Fractional Fourier Transform (FFT), Iterative FFT, Discrete Wavelet Transform (DWT), Jig Saw Transform (JST) and Least Significant Bit (LSB).

Table 4. Comparison of Image Compression Algorithms

Technique	Authors & Year	Images used	Experimentation	Observations
SPIHT	*Swetha et al.* [30]	SPIHT is used for the replacement for wavelet compression methods.	1.Lena	1.Original image 2.Reconstruction using DCT 3.Reconstruction using SPIHT 4.PSNR for DCT is 30.81 and SPIHT is 39.85 5.MSE for DCT is 54 and SPIHT is 6.72
Run Length Encoding (RLE)	RuifengXu, *et al.* [31]	This method extends JPEG 2000 to compress the HDR images.	1.High Dynamic Range (HDR) 512×768	1.Compressed image (RLE) (248Kb) 2.Reference image (2683 Kb) 3.Root mean square error 4.Compression time is 1.7 sec and decompression time is 0.65s
Huffman Encoding	Srikanth and Meher [33]	It uses the different embedded wavelet based image coding with the Huffman encoder for further compression	1.Boat	1.PSNR for different wavelet families 2.Original image 3.Reconstructed image
DCT	Ito Masanori, *et al.* [38]	This compression technique is combined with the image encryption technique called Independent Component Analysis (ICA).	1.Lena 256×256 2.Mandrill 256×256	1.RMSE between original source image and reconstructed images, 2.RMSE between the compressed source images and reconstructed images 3.Original image 4.Encrypted image 5.Reconstructed image
JPEG 2000	*Subramanyan, et al* [48]	JPEG 2000 is used to encrypt and compress the image	1.Gray scale image 512×512	1.Original image 2.Watermarked image 3.PSNR for encrypted data is 35 dB 4.Average payload capacity is 2.5 5.Unwatermarked decompressed image 6.Encrypted image 7. Watermarked image
Arithmetic Encoding	Grangetto.M, *et al.* [37]	AES based arithmetic encoding approach for selective encryption. It applied to any multimedia coder employing arithmetic coding	1.Lena 256×256 2.Gold hill image 256×256	1.Image produced by a standard JPEG 2000 decoder in case of total encryption 2.PSNR is 40dB 3. Thumbnail generation for the Goldhill image 3.Rate overhead due to zero padding for AES is 0.54% 4. Running time is 0.22 sec for 0 level and 0.33 sec for all levels
	Zhou Jiantao, et al. [49]	Arithmetic encoding is used to compress the encrypted image. It is conducted over the prediction error domain.	1. Lena 256×256 2. Barbara 256×256 3. Man 256×256 4. Boat 256×256 5. Harbor 256×256 6. Airplane 256×256 7. Liver 256×256	1.Original image 2.Encrypted image 3.Compression performance for Lena is 4.096bpp, Barbara is 4.589bpp, Man is 4.345bpp, Boat is 4.112bpp, Harbor is 4.9bpp, Airplane is 3.704bpp and Liver is 2.486bpp
Resolution Progressive	*Wei Liu, et al.* [50]	It used to compress an encrypted image progressively in resolution.	1.Baboon 2.Lena 3.Peppers 4.Boats 5.Goldhill	1.Correlation between neighboring pixels for Baboon is 0.81, Lena is 0.97, Peppers is 0.98, Boats is 0.97 and Goldhill is 0.99 2. Average Residual entropy is4.98 3. Average Compression performance is 5.35

Comparative results of the image compression algorithms are shown in Table 4. The compression algorithms compare the compression performance and PSNR values of the reconstructed image. From the table it is evident that the encryption with the compression improves the image quality and authenticity. Moreover, the surveyed result evidently proves that the SPIHT algorithm can compress the encrypted image with better image quality.

4. Proposed Work

Based on survey results, it is suggested that, the use of Blowfish and RC4 algorithms for the encryption process will optimize the performance. The UWT can be applied on the encrypted image with the other image to hide the original image. In order to compress the embedded image, SPIHT algorithm can be applied. The encoded bits are decoded based on the reverse process of the UWT, Blowfish and RC4 algorithms. Application of these algorithms will retrieve the original image from the encrypted image. The resultant image will yield better compression ratio, PSNR, with lesser MSE and memory capacity.

5. Conclusion

In this paper, an overview of various encryption based compression techniques is presented. From the survey, it is found out that Blowfish and RC4 are the very powerful encryption techniques to encrypt the input source image. When combined with the SPIHT compression techniques, it results more improved quality including reconstructed image. An efficient encryption based image compression approach can be formulated based on the Blowfish, RC4, UWT and SPIHT algorithms to attain the best quality reconstructed image.

References

[1] Elminaam, D. S. A., Mohamed, M., *et al.* (2010). Evaluating the performance of symmetric encryption algorithms. *IJ Network Security, 10*, 216-222.

[2] Hardjono, T., D., & Lakshminath, R. (2005). *Security in Wireless LANS and MANS (Artech House Computer Security)*, Artech House, Inc.

[3] Nie, T. Y., & Zhang, T. (2009). A study of DES and Blowfish encryption algorithm. *Proceedings of TENCON 2009-2009 IEEE Region 10 Conference* (pp. 1-4).

[4] Abdul, J. J., & Jisha, M. T., (2013). Guarding Images using a symmetric key cryptographic technique: Blowfish algorithm. *International Journal of Engineering and Innovative Technology, 3*, 196-201.

[5] Tahseen, I., *et al*. (2012). Proposal new approach for blowfish algorithm by using random key generator. *Journal of Madent Alelem College, 4(1).*

[6] Kumar, P., & Pateriya, P. K. (2013). RC4 enrichment algorithm approach for selective image encryption. *IJCSNS, 13*, 95.

[7] Stošić, L., & Bogdanović, M. (2012). RC4 stream cipher and possible attacks on WEP. *Editorial Preface, 3*.

[8] Pia, S., & Karamjeet, S. (2013). Image encryption and decryption using blowfish algorithm in Matlab. *International Journal of Scientific & Engineering Research, 4.*

[9] Kundankumar, R. S. V. P. J., & Amit, K. M. (2014). Text and Image Encryption Decryption Using Advanced Encryption Standard. *International Journal of Emerging Trends & Technology in Computer Science, 3*, 118-126.

[10] Zeghid, M., Khriji, B., & Tourki (2007). A modified AES based algorithm for image encryption. *International Journal of Computer, Electrical, Automation, Control and Information Engineering, 1*, 11-16.

[11] Manjula, N. H., & Manoj, B. (2012). Image encryption and decryption using AES. *International Journal of Engineering and Advanced Technology, 1*, 290-294.

[12] Rivest, R. L., Shamir, A., & Adleman, L. (1983). A method for obtaining digital signatures and public-key

cryptosystems. *Communications of the ACM, 26*, 96-99.

[13] Sameh, N. G., *et al*. (2014). Digital image encryption based on RSA Algorithm. *IOSR Journal of Electronics and Communication Engineering, 9*, 69-73.

[14] Sinha, A., & Singh, K., (2003). A technique for image encryption using digital signature. *Optics Communications, 218*, 229-234.

[15] Sakthidasan, K., & Santhosh, K. B. V. (2011). A new chaotic algorithm for image encryption and decryption of digital color images. *International Journal of Information and Education Technology, 1*, 137-141.

[16] Dey, N., Anamitra, B. R., & Sayantan, D. (2012). A novel approach of color image hiding using RGB color planes and DWT. *arXiv preprint arXiv:1208.0803*.

[17] Ganic, E., & Eskicioglu, A. M. (2004). Robust DWT-SVD domain image watermarking: Embedding data in all frequencies. *Proceedings of the 2004 workshop on Multimedia and Security*, Magdeburg, Germany.

[18] Amandeep, K., Navneet, S. B. K., Sukhdeep, S., & Parneet, K. (2011). Embedding Image in mid frequency band using DWT technique. *International Journal of Advances in Engineering & Technology, 1*, 12-17.

[19] Muhammad, G., & Hussain, M. (2013). Passive detection of copy-move image forgery using undecimated wavelets and Zernike moments. *International Information Institute, 16*, 2957-2964.

[20] Raj, V. N. P., & Venkateswarlu, T. (2011). Denoising of medical images using undecimated wavelet transform. *Proceedings IEEE Conference in Recent Advances in Intelligent Computational Systems* (pp. 483-488).

[21] Raj, V. N. P., & Venkateswarlu, T. (2012). Denoising of medical images using dual tree complex wavelet transform. *Procedia Technology, 4*, 238-244.

[22] Olcay, D., & Akay, O. (2011). A new method of wavelet domain watermark embedding and extracting using fractional fourier tranform. *Proceedings of 7th International Conference on Electrical and Electronics Engineering*.

[23] Saxena, R., & Kulbir, S. (2013). Fractional Fourier transform: A novel tool for signal processing. *Journal of the Indian Institute of Science, 85*, 11.

[24] Sejdic, E., Djurovic, I., & LJubisa, S. (2011). Fractional Fourier transform As a signal processing tool: An overview of recent developments. *Signal Processing, 91*,1351-1369.

[25] Kumar, M., Rewani, R., *et al*. (2013). Digital image watermarking using fractional fourier transform via image compression. *Proceedings of IEEE International Conference on Computational Intelligence and Computing Research* (pp. 1-4).

[26] Taba, M. T. (2013). The fractional fourier transform and its application to digital watermarking. *Proceedings of 8th International Workshop on Systems, Signal Processing and their Applications* (pp. 262-266).

[27] Nishchal, N. K., & Naughton, T. J. (2009). Three-dimensional image watermarking using fractional Fourier transform. *Proceedings of the International Conference on Optics and Photonics*, Chandigarh, India.

[28] Kumar, M. N. S., & Kumar, M. B. M. (2013). Improved image watermarking with curvelet wavelet. *International Journal of Computer Science and Mobile Computing, 2*, 363-368.

[29] Giraldo, L. M., & Villegas, E. Y. (2012). Optical encryption with jigsaw transform using Matlab. *arXiv preprint arXiv:1205.3445*.

[30] Swetha, D., David, S. R. Y., & MuraliMohan, K. V. (2013). Image compression using wavelet and SPIHT encoding scheme.

[31] Hughes, E. C., *et al.* (2005). High-dynamic-range still-image encoding in JPEG 2000. *Computer Graphics and Applications, 25,* 57-64.

[32] Pratishtha, G. G. N. P., & Varsha, B. (2014). A survey on image compression techniques. *International Journal of Advanced Research in Computer and Communication Engineering, 3*, 7762-7768.

[33] Srikanth, S., & Meher, S. (2013). Compression efficiency for combining different embedded image compression techniques with Huffman encoding. *Proceedings of 2013 International Conference on Communications and Signal Processing* (pp. 816-820).

[34] Tripatjot, S., Sanjeev, C., Harmanpreet, K., & Amandeep, K., (2010). Image Compression using wavelet and wavelet packet transformation. *IJCST, 1*.

[35] Asadollah, S. R. B., Mobin, S. R., Mostafa, A. M. (2011). Evaluation of huffman and arithmetic algorithms for multimedia compression standards.

[36] (1997). *Recent Development of Images Compression Technique*.

[37] Grangetto, M., Magli, E., & Olmo, G., (2006). Multimedia selective encryption by means of randomized arithmetic coding. *IEEE Transactions on Multimedia, 8,* 905-917.

[38] Ito, M., Ohnishi, N., Alfaluo, A., & Ali, M. (2008). New image encryption and compression method based on independent component analysis. *Information and Communication Technologies: From Theory to Applications*.

[39] *The USC-SIPI Image Database*. From: http://sipi.usc.edu/database/database.php?volume=misc&image=12

[40] *EE398B-Image Communication II*. (2007). From: http://web.stanford.edu/class/ee398b/samples.htm

[41] Ghulam, M., Muhammad, H., & George, B., (2012). Passive copy move image forgery detection using undecimated dyadic wavelet transform. *Digital Investigation, 9*, 49-57.

[42] Ran, T., Yi, X., & Yue, W. (2007). Double image encryption based on random phase encoding in the fractional Fourier domain. *Optics Express, 15*, 16067-16079.

[43] Liu, Z. J., & Liu, S. (2007). Double image encryption based on iterative fractional Fourier transform. *Optics Communications, 275*, 324-329.

[44] Al-Haj, & Ali, (2007). Combined DWT-DCT digital image watermarking. *Journal of Computer Science, 3*, 740.

[45] Jayanta, K. D., Newaz, M. S. R., & Fung, W.-K. (2008). A modified vector quantization based image compression technique using wavelet transform.

[46] Mehmet, U., Gaurav, S., & Murat, T. A. (2006). Lossless watermarking for image authentication: A new framework and an implementation. *IEEE Transactions on Image Processing, 15*, 1042-1049.

[47] Lute, K., & Heijmans, H. J. A. M. (2005). Reversible data embedding into images using wavelet techniques and sorting. *IEEE Transactions on Image Processing, 14,* 2082-2090.

[48] Subramanyam, A. V., Sabu, E., & Kankanhalli, S. M. (2012). Robust watermarking of compressed and encrypted JPEG2000 images. *IEEE Transactions on Multimedia, 14*, 703-716.

[49] Zhou, J. T., Liu, X. M., *et al.* (2013). On the design of an efficient encryption-then-compression system. *Proceedings of 2013 IEEE International Conference on Acoustics, Speech and Signal Processing* (pp. 2872-2876).

[50] Wei, L., Zeng, W. J., Lina, D., & Yao, Q. M. (2010). Efficient compression of encrypted grayscale images. *IEEE Transactions on Image Processing, 19*, 1097-1102.

C. Sankaranarayanan is the principal of Government Polytechnic College, Chennai, Tamil Nadu, India. Currently he is doing PhD. He has 30 years of experience in teaching. His research interests include image processing and soft computing.

S. Annadurai has been an academician for 33 years, who rose to the level of Additional Director of Technical Education heads as a principal since 2011. He is a good learner and up to date on many technical areas such as computer science, robotics, and management etc. Being a Ph.D degree holder in computer science and engineering, he has made significant contributions to the research activities. He had successfully guided fourteen research scholars to receive their Ph.D degrees and three scholars to receive MS (By research) degrees. He has published fifty-one research papers in international and national journals and two hundred and fifty nine research papers presented in national and international conferences. His areas of interest are digital image processing, parallel computing, neural networks, advanced computer architecture, computer networks and genetic algorithms. He was preceded by Dr. S. Subramanian, who served as a principal during 1998 to 2011.