Encryption Algorithms for Color Images: A Brief Review of Recent Trends

Anuja P Parameshwaran
Department of Computer Science,
Georgia State University (GSU)
Atlanta, Georgia, USA

Wen-Zhan Song
Department of Computer Science,
Georgia State University (GSU),
Atlanta, Georgia, USA

***Abstract*—The recent years have witnessed rapid developments in the field of image encryption algorithms for secure color image processing. Image encryption algorithms have been classified in different ways in the past. This paper reviews the different image encryption algorithms developed during the period 2007-2015, highlighting their contrasting features. At the same time a broad classification of the said algorithms into: (1) full encryption algorithm and (2) partial encryption algorithm, each further sub-classified with respect to their domain orientations (spatial, frequency and hybrid domains) have been attempted. Efforts have also been made to cover different algorithms useful for color images of various color spaces like Red-Green-Blue (RGB), Hue-Saturation-Intensity (HSI), Cyan-Magenta-Yellow (CMY) etc. Chaotic cryptosystems, various transforms like wavelets, affine transforms etc. and visual cryptography systems are being discussed in detail.**

Keywords*—RGB; HIS; CMY; Chaotic cryptosystem; wavelets; affine transforms and visual cryptography

I. INTRODUCTION

A. Background/Preliminary:

The field of encryption, which deals with information security, is an active research domain in modern times. Irrespective of the kind of data (multimedia or plain text data) dealt with, security is very crucial, attracting lots of researchers into this important domain. World-over, researchers are worried on security issues during: (1) transmission of data across a secure channel and (2) the storage of data, so that it is not attacked or retrieved by unauthorized parties. Encryption techniques in general help manage these two aspects related to security of any data.

The present review focuses on enlisting and exploring various encryption techniques related to image data. Image is an array or a matrix representation of square picture elements (pixels) that is systematically arranged in rows and columns. For example, a given 2D image I can be represented as MxN matrix where M and N represent the number of rows and columns respectively. Each pixel represented in the image matrix has an intensity value. In case of gray scale images the pixel can take intensity value between the ranges [0,255]. Figure 1 shows an instance of a gray scale image matrix with its intensity value representation.

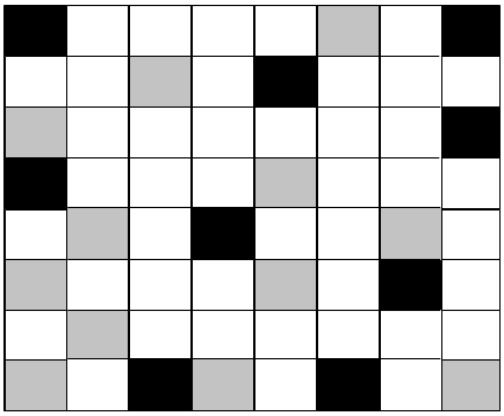

Fig. 1. Each pixel in the gray scale image matrix can take any of the 256 values between [0,255]

A given Image data will differ from the corresponding text data in several ways. For instance, an image data is inherently redundant with the pixels highly correlated (This is generally derived from the fact that an image has smooth texture). Some cryptosystems like the chaotic cryptosystem [1-2] make use of this inherent property of images for encrypting them. Moreover, traditional cryptosystems used for text is not used in case of images for the following two reasons: Firstly, the size of the image data and normal text data varies greatly, with the former being many times bulkier, going up to several Gigabytes in size. Due to the size variation between the two kinds of data at hand, it is obvious that traditional cryptosystems would take a lot more time encrypting image data than normal text data, which is not a very wise choice to go with. A second constraint while dealing with the text data arises out of the requirement that the decrypted data must be same as the original data (in terms of size and content). In contrast, small distortions in the decrypted image data are acceptable as it always depends on human perception which is different for different people. Moreover, lossy compression can be applied to image data before encryption, though this would lead to slight loss of redundant data in the decrypted image [3].

In image encryption, it is important to verify the following three parameters of transmitted digital images: integrity, confidentiality and authenticity [4]. Some of the applications in which image encryption algorithms come really handy are: internet communication, multimedia systems, medical imaging, telemedicine, military communication etc. [3].

Figure 2 shows an illustration of the encryption/decryption process of image data.

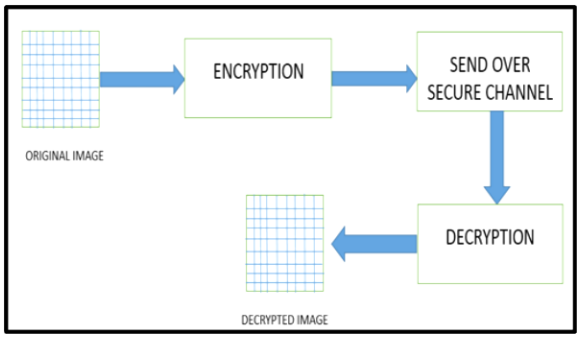

Fig. 2. Encryption and decryption process illustration of a single image matrix

Image compression is another interesting area of research that goes hand in hand with image encryption. Many encryption algorithms like to compress the initial image first before actually encrypting it so as to get rid of any redundant data in the beginning itself [5]. However, some researchers still prefer to encrypt data first and then compress it before transmitting [5] [6] [7] [8]. Compression before encryption works better for data that is highly redundant in nature [5] [6]. In addition to removing redundant data, compression also reduces the bandwidth requirement for transmission of the encrypted image along secure channels. However, it may be added that not all encryption algorithms make use of compression. Most of the algorithms which are frequency domain- oriented and those that involve usage of transforms like DCT, affine etc. have compression either of lossy or lossless form, inherently added to them.

B. *Types of Images:*

Images can be divided broadly into two categories: (1) Gray-scale images and (2) Color-images. A simple definition of gray scale images would be an image which has only shades of gray and is devoid of any other color. In simpler words gray scale images contain no color information and is mostly referred to as one-color images [29]. The difference between gray-scale images and colored images is that for gray scale images, only less information is required to represent a given pixel of the image. A given pixel of a gray scaled image can take up any of the 256 values ranging from [0-255], which is basically called the intensity of the pixel [9].

Color images can be picturized as a three-band monochrome image data, where each band of the image corresponds to different color [29]. For example: RGB images corresponds to red, green and blue bands if separated into their individual components. Figure-3 shows the famous picture of Lena as a binary image (consists of black and white color only), gray scale image (consists of shades of gray ranging from [0-255] i.e. range from black-shades of gray-white) and RGB colored image (consists of 3 individual components R, G and B) [9].

Fig. 3. Image of Lena as a binary, gray-scale and RGB color image (courtesy: ref [9])

RGB color images consist of three separate components of 8 bits each. Thus RGB has a color depth of 24 bits in total compared to the 8 bits of gray scaled images. There are various other color space representations of images which are more effective than the RGB representation of images. For instance: YIQ color space (luminance (Y), Chrominance1 (I) and Chrominance2 (Q)), HSI color space (Hue (H), Saturation (S) and Intensity (I)) etc. [9]. Recent experimental results of correlation indicate that YIQ color space algorithm is better than RGB algorithm [9].

C. *Full Encryption Algorithms vs. Partial Encryption Algorithms:*

The image encryption algorithms can be broadly divided into two categories: full encryption algorithms and partial encryption algorithms (selective encryption algorithms). These algorithms will be discussed with respect to their domain orientation, *i.e.*, whether they are spatial domain specific, frequency domain specific or a mix of both (hybrid domain specific) as shown in Figure 4 [4].

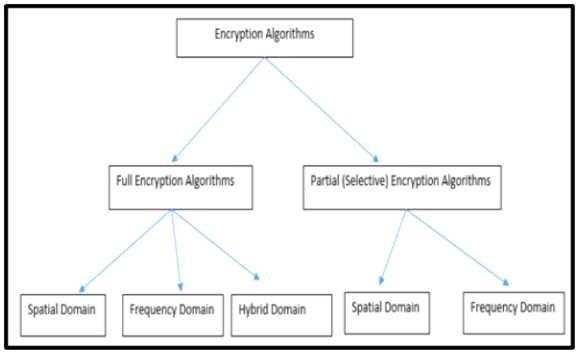

Fig. 4. Broad classification of Image encryption algorithms into full and partial encryption algorithms

Full encryption algorithms, as the name itself suggest, deals with the image as a whole and encrypts the whole image. Partial encryption algorithms encrypts only a part of the image rather than encrypting the whole image. Region of Interest (ROI) is used in selecting the part of image that needs to be encrypted in partial encryption algorithm [20]. Visual cryptography (VC) algorithms also fall into the category of partial encryption algorithms. The application areas of these two classes of encryption algorithms vary greatly. For the first comparison, we look at the computation requirements of both the algorithms. It is seen that in partial encryption algorithms, the computational requirements are greatly reduced as only the lowest portion of data is encrypted. In fact, the full encryption

algorithms have greater computational complexity than partial encryption algorithms. For the second comparison, the overall time required by both sections of the algorithms is looked at. Time can be further divided into two categories, (i) the encryption-decryption time (EDT) and (ii) the actual transmission time of the encrypted data over a secure channel. EDT for full encryption algorithms are bound to be larger as compared to partial encryption algorithms as the latter focuses on encrypting-decrypting only a small region of the image [4]. Figure 5 a) and b) shows the flow and stages of full encryption algorithms and partial encryption algorithms.

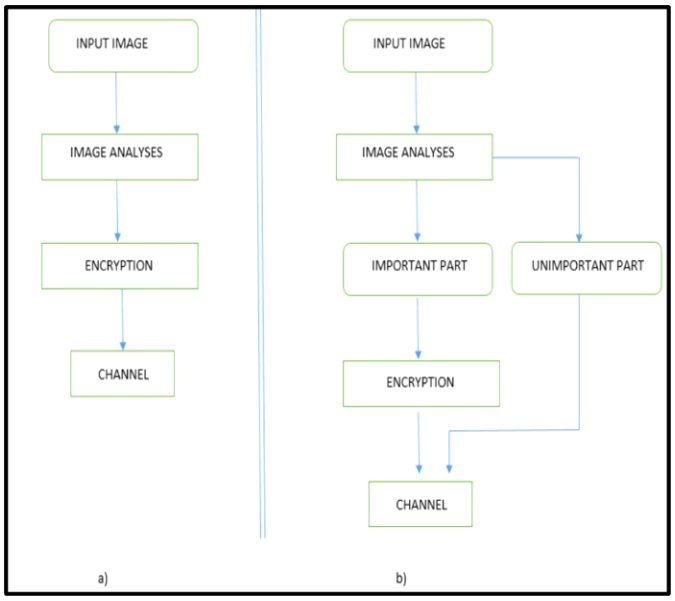

Fig. 5. a) Full encryption algorithm b) partial encryption algorithm

D. Evaluation and Comparison Criteria of Image Encryption Algorithms:

Most of the papers discussed under the category of full encryption based algorithms under spatial domain category [1-4] are evaluated based on the following four parameters:

1) Security - Security as an evaluation parameter indicates the confidentiality and robustness of the encryption scheme against various attacks like statistical and differential attacks [9].

2) Speed–A factor used to differentiate partial and full encryption algorithms. Less data to encrypt implies usage of less CPU time which further implies faster encryption and decryption. This is the advantage of partial encryption algorithms over full encryption algorithms [9].

3) Correlation – Main aim of encryption is to destroy the correlation between adjacent pixels of the encrypted image (which is it will be nearly zero). Image data inherently processes the feature of highly correlated pixels (correlation value close to one), encryption aims to destroy that [1-4] [9].

4) Key space analysis – The key space of an encryption space should be large enough so as to avert any brute force or exhaustive attacks from the intruder. Moreover the encryption scheme must also be sensitive to the key [9].

Correlation of adjacent pixels of an encrypted image has to be reduced. But many encryption algorithms on color images existing today are just an extension of the original encryption algorithm on a gray scale image. The correlation analysis does not work out well for such group of algorithms as they apply the encryption schemes on the R, G and B components separately. While doing so, such class of algorithms conveniently neglect the correlations of the R, G and B components together. Most of the papers discussed here [1] [2-15] are not an extension of the gray scale image encryption scheme, with the exception of [2]. The comparison criteria of most of the papers [1-9] are based on the following:

1) Number of Pixel Change Rate (NPCR)–This depicts that the more a particular cryptosystem is sensitive to changing of the original inputted data, the more effective that cryptosystem is to resist a statistical attack from an intruder. The higher the value of NPCR (close to 100%), better is the cryptosystem able to resist statistical attacks.

$$NPCR\ (R,G,B) = \frac{\sum i,j\ D\ (R,G,B)\ (i,j)\ x\ 100\%}{W\ x\ H}$$

Where, W and H represent the height and width of the image.

2) Correlation coefficient–the correlation coefficient between the pixels of the encrypted image should be low i.e. close to zero as it will help resist statistical attacks. The correlation coefficient between each pair of adjacent pixels of an image can be calculated as:

$$D(x) = \frac{1}{N}\sum_{i=1}^{N}(x\ i - E(x))^2$$

Where,

$$Cov\ (x,y) = \frac{1}{N}\sum_{i=1}^{N}(x\ i - E(x))(y\ i - E(y))$$

$$E(x) = \frac{1}{N}\sum_{i=1}^{N} x\ i$$

$$r\ \{x,y\} = \frac{Cov\ (x,y)}{\sqrt{D(x)}\sqrt{D(y)}}$$

3) Unified Average Changing Intensity (UACI) – UACI values are given so as greater is the possibility that the cryptosystem can avert a differential attack from an intruder.

$$UACI\ (R,G,B) = \frac{1}{W\ x\ H}\frac{\sum i,j\ |\ C\ (R,G,B)\{i,j) - C'\{R,G,B)\{i,j)|}{255}\ x\ 100\%$$

Here, W and H represent the width and height of the inputted image and C (R, G, B) and C' (R, G, B) are encrypted images before and after a pixel in the original inputted image has been changed.

4) Entropy–This parameter deals with the idea of self-information. It indicates how much of information has been lost in the decrypted image. The ideal value discussed in here

for [1-8] is 8. If an encryption scheme has a value closer to 8 means it has lost very little to negligible amount of information. If there are M messages as m is a symbol then entropy H(m) can be defined as:

$$H(m) = \sum_{i=0}^{M-1} p(m\,i) \log \frac{1}{P(m\,i)}$$

II. Full Encryption Algorithms

Full encryption algorithms are discussed in [10-17]. These papers are divided into spatial domain [10-13], frequency domain [14-15] or hybrid domain [16-17] based algorithms. [10], [11] and [13] are encryption algorithms that are based on chaotic cryptosystems. Chaotic cryptosystem is based on the mathematical theory of 'Chaos'. In theory, chaotic cryptosystems are those dynamic systems that are highly sensitive to system parameters and initial conditions. If the system parameters change by chance then the decrypted image at the receiving end would be totally different from the original inputted image [10]. Chaotic cryptographic systems exploit the inherent feature of bulk data capacity and high data redundancy of an image [30]. The reason chaotic cryptosystems is most useful is because the encryption algorithm destroys any original pattern existing in the reconstructed image, thus making it difficult for an intruder to reconstruct the image based on visual perception of the graphical information. Chaotic cryptosystems are tied to two properties of good ciphers i.e. confusion and diffusion [18]. Diffusion is based on the dependency of the output bits on the input bits whereas confusion is made possible by permuting the data sequence thus guaranteeing the relationship between the key and cipher text to be as complex as possible [18]. The secret key for chaotic cryptosystems described in [10] [11] [13] [16] [17] are the initial conditions and the system parameters defined.

The first step for designing a chaotic cryptosystem is selection of a chaotic map. The dynamics of the chaotic map is determined by control parameters [18]. A particular chaotic cryptosystem will choose the number of chaotic maps it needs. A single chaotic map has only a small key space making it easy for intruders to perform brute force attacks to break the system. Thus to avoid such a situation, multidimensional couple maps can be used together so as to improve the security. A chaotic map is generally used to produce a chaotic sequence after certain number of hops which is very important as this sequence controls the entire encryption process. Though the chaotic cryptosystems seem perfect in theory, in practice they also have a few limitations. First of all, the performance of a chaotic cryptosystem is relatively slow when compared to other cryptosystems (traditional cryptosystems). In addition, the fact that the chaotic cryptosystem is highly sensitive to system parameters and initial conditions of the system sounds great in theory. But in practice, it is difficult to get the correct decrypted image if the processors at the senders and receivers end are different! Because of these limitations, chaotic cryptosystems are very much limited in real world applications [18] [19]. Moreover most of the chaotic cryptosystems are already dysfunctional [31].

Rhouma *et al* [10] have made use of a piecewise linear chaotic map to build their chaotic cryptosystem. The main encryption procedure had three sub procedures which were interleaved with each other. While dealing with color images (in RGB color space) the first step of the encryption algorithm was to convert the image into its three individual vector components by scanning the original image matrix from left to right and top to bottom. The second step involved the division of the phase space of the skew tent map used to build the cryptosystem into 256 equal width intervals. A mapping function was also defined to help map the respective values. Finally, each of the color pixels belonging to R, G or B were individually encrypted by going through certain number of predetermined rounds. The results of [10] showed that the key space (key space = 1093) was not exhaustive enough (as in the case with most chaotic based cryptosystems) and it was vulnerable to brute force attacks. In order to prove that the encryption algorithm suggested by the authors was a secure one, they proved that the values of NPCR and UACI were high enough to avert any differential attacks. Information loss incurred by [10] was very small, almost negligible, as seen from entropy factor calculations (entropy = 7.9551, as against the ideal value of 8).

Ahmad and Alam *et al* [11] also talked about encryption and decryption of images by using a chaotic cryptosystem. The main difference between [10] and [11] is that in [11], the authors made use of three different chaotic maps namely, 2D cat map, 2D coupled logistics map and 1D logistics map. The original image was initially broken down to 8x8 sized blocks and were then shuffled by using the 2D cat map. The control parameters for the shuffling process were generated randomly as dictated by the 2D coupled logistics map and lastly, the shuffled blocks of the image was encrypted in accordance to the chaotic sequence that was generated by the 1D logistics map. Unlike the work described in [10], [11] surprisingly has a huge key space (about 10112) which was capable of preventing any sort of statistical or differential attack. The information loss as measured by calculating entropy (7.9992, ideal entropy value = 8) was low here too and the coefficient correlation was obtained to be very low as desired (that is, 0.0095, ideally if low correlation that has to be closer to 0). The work in [10] did not really calculate the coefficient correlation value, so was hard to tell whether the correlation of adjacent pixels of the encrypted image was low or high.

Chandel *et al* [12] made use of the color images found in the wang dataset. The security in [12] was increased because of two operations performed one after another: splitting procedure followed by the encryption procedure. Both the procedures have their own respective keys. The encryption algorithm used here is not really new but simple RSA encryption algorithm, which is a public key cryptography method. Initially, the color image from the database was split into many portions by making use of a splitting algorithm. RSA encryption algorithm was then used to encrypt each of the split pieces. On the receiving end, the inverse of RSA was successfully applied to decrypt each of the split pieces and finally all the split pieces were merged together to form the final decrypted image. The keys and number of split pieces were predetermined by the authorized sender in this

encryption methodology. This work made use of histograms only to measure the color bins of both the original and the encrypted images. The encryption scheme in [12] did not have an exhaustive analysis of the key space nor did it calculate the coefficient correlation between adjacent pixels of the encrypted image. The only criteria calculated in [12] was the entropy and it consequently proved that the information loss was almost negligible for each of the color images considered from the wang dataset. Since no other analysis was shown by the authors in [12] against any attacks it is difficult to say how effective this encryption methodology can really be. But on a positive note, the encryption scheme in [12] can be plainly noted for its simplicity.

Just like works in [10] and [11], Wang *et al* [13] was also based on chaotic cryptosystems. This paper made use of a single chaotic map that is the logistics map. The encryption algorithm was very similar to that discussed in [10]. Initially a RGB-color image was broken down to its independent R, G and B components. Then a permutation algorithm was performed on each of the independent component matrices. The permutation algorithm facilitated combined row and combined column scrambling which helped the R, G and B pixels to be mutually permuted. Lastly, a diffusion process was applied to the R, G and B components of the image to give the resultant cipher image. The encryption process discussed when applied in the reverse order gives back the original color image. This algorithm was effective against exhaustive attacks as it had a huge key space which could reach up to 10^{56}. Histogram analysis was also performed which showed that after encryption the R, G and B's components of the cipher image was fairly uniform proving it hard for an intruder to decrypt. Thus the methodology described in [13] was safe against statistical attacks. The correlation coefficient of the cipher image was also very small. NPCR and UACI values calculated by [13] were at a higher range which proved that it was secure against any differential attacks. The authors of [13] also proved that their methodology was safe against cipher text only attacks, known plaintext attacks, chosen plaintext and chosen cipher text attacks.

With Chen *et al* [14] deals with a new domain, *viz.*, frequency domain. This paper made use of two transforms: affine transforms and the gyrator transforms. The affine transform was used twice in the encryption process. The parameters for these transforms served as the secret key in [14]. Initially the RGB image was broken down into its 3 independent components. A function of the affine transform was then used to mix these R, G and B components. Basically, the components were converted to a complex function (real and imaginary parts) with the help of the affine transform. The scrambled pieces of the images was then combined using the gyrator transform. In other words the gyrator function helped to encode and transform the complex function obtained in the first step of using the affine transform. And lastly, the encrypted image came into being by using the final function of the affine transform. The reason the affine transform was performed twice for this encryption scheme was in order to enhance the security of the algorithm. The work in [14] has also demonstrated some numerical simulations to prove that the methodology is valid, secure and robust in nature. Figure 6 shows the encryption scheme as discussed in [14].

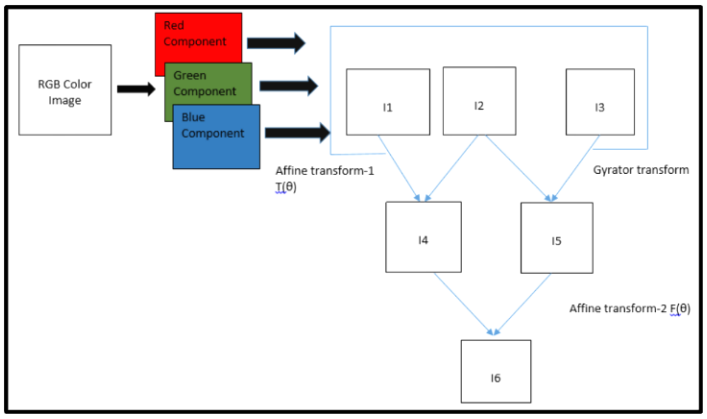

Fig. 6. Encryption algorithm as demonstrated in [14]

Samson *et al* [15] bought in compression before the encryption process. The compression achieved by the use of wavelets in [15] was however lossless in nature. In order to achieve additional compression, [15] made use of lossless predictive coding alongside wavelets. The encryption scheme in [15] flowed as following: firstly, the input color image was compressed using a wavelet of sender's choice, the level of decomposition was also set according to the sender's choice. Then additional compression was achieved using lossless predictive coding. The second step was the encryption process which was achieved using the regular secure advanced hill cipher. This in turn involved a pair of involutory matrices (that is the inverse of a matrix A is equal to the matrix A itself), a mix() function and an XOR operator. The decryption process involved firstly the decryption algorithm followed by the decompression algorithms in order. The wavelet used in [15] was the simple haar wavelet. One of the drawbacks of the methodology in [15] was that there was no analysis of the key space or the nature of attacks possible described in it. The only form of validity [15] has is that the decrypted image obtained was the same as the original inputted image. Figure 7 depicts the flow of stages in [15].

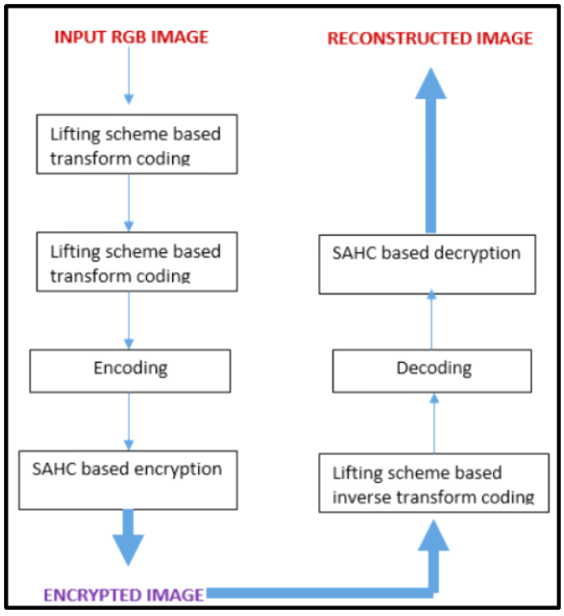

Fig. 7. Flowchart of the encryption scheme followed in [15]

Yu *et al* [16] and Zhou *et al* [17] discussed their methodology in the hybrid domain. Yu *et al* [16] made use of both: the chaotic cryptosystem and the wavelet transform to form its own encryption scheme. This paper made use of the wavelet decomposition and reconstruction processes twice in its encryption scheme. Initially the original color image underwent a 1-level wavelet decomposition. Only the low frequency coefficients were considered of importance at this stage and they were encrypted using a chaotic based encryption algorithm. There were four chaotic maps used for this purpose alone namely: Logistic maps, Chebyshev map, 2D Arnold map and 3D Arnold map. The next step in encryption involved that these encrypted low frequency coefficients act as a key stream and be XORed with the unencrypted high frequency coefficients. This step rendered all the image information held by the high frequency coefficients as hidden. The third step in the encryption process was the wavelet reconstruction which rendered all the encrypted low frequency portions to be equally distributed among the whole image. In order to have effective confidentiality in the encryption scheme of [16], the authors decided to make use of chaotic scrambling for the reconstructed image. Arnold scrambling was used in order to produce a smooth image as the final decrypted image. The wavelet decomposition and reconstruction processes were performed again (level-2 wavelet decomposition). The key space was large enough to resist brute force attacks (2^{128}). The encryption algorithm proved efficient enough against statistical and exhaustive attacks. Overall, methodology discussed in [16] only gave us reasonable performance, but the encryption time taken by the encryption process was determined to be 0.266 seconds (which is still long).

The works from [10], [11], [12], [13], [14], [15] and [16] dealt with image from the RGB color space. Zhou *et al* [17] instead dealt with the image from HSI color space only. The first step in the encryption scheme of [17] was converting the image from any other color space to the HSI color space. New phase plates were generated in this encryption scheme using the fractional fourier transform. The S component obtained a new random phase with the help of random phase encoding which was based on fractional fourier transform. The I component too was transformed into two new phase plates with the help of double random phase encoding which was again based on fractional fourier transform and it made use of the H component and the new random phase component. The final step to the encryption scheme in [17] was chaos scrambling which essentially encrypted the image. The difference between the cipher/encrypted image in [17] compared to all the other cipher/encrypted images in [10-16] was that it was non-linear in nature and was a combination of gray image and a phase matrix. Numerical simulations had been performed which enabled to illustrate the effectiveness and the level of security obtained by the proposed encryption scheme in [17]. Table-1 summarizes the full encryption algorithms discussed in this section.

TABLE I. SUMMARY OF FULL ENCRYPTION ALGORITHMS. THE WORK DISCUSSED IN [14] & [15] BANKS ITS VALIDITY ON YHE FACT YHAT YHE FINAL DECRYPTED IMAGE SHOULD BE YHE SAME AS YHE ORIGINAL INPUTTED IMAGE (THIS IS WRT YHE VISUAL PERCEPTION ALONE)

Ref No	Category	Domain	Methods used	Number of chaotic maps used	Color space of inputted image	Comparison criteria used wrt security achieved
[10]	Full encryption algorithms	Spatial domain	Purely chaotic cryptosystem	1	RGB color space	-NPCR -UACI -Entropy
[11]	Full encryption algorithms	Spatial domain	Purely chaotic cryptosystem	3	RGB color space	-Larger key space -Entropy -Coefficient correlation
[12]	Full encryption algorithms	Spatial domain	Splitting procedure + RSA encryption algorithm	-	RGB color space	-Entropy
[13]	Full encryption algorithms	Spatial domain	Purely chaotic cryptosystem	1	RGB color space (Made use of color images from wang dataset)	-Huge key space -NPCR -UACI
[14]	Full encryption algorithms	Frequency domain	Affine transforms (used twice) + Gyrator transforms	-	RGB color space	-Security, validity and robustness of methodol

						ogy illustrated through numerical simulations
[15]	Full encryption algorithms	Frequency domain	Wavelet transforms used for compression prior to encryption	-	RGB color space	-
[16]	Full encryption algorithms	Hybrid domain (includes both spatial + frequency domains)	Chaotic cryptosystem + wavelet transform [partially chaotic cryptosystems]	4	RGB color space	-Huge key space
[17]	Full encryption algorithms	Hybrid domain (includes both spatial + frequency domains)	Chaotic cryptosystem + Fractional Fourier Transform [partially chaotic cryptosystems]	1	HSI color space	-Security, validity and robustness of methodology illustrated through numerical simulations

III. Partial Encryption Algorithms

Partial encryption algorithms are also known as selective encrypted algorithms. The fundamental rule of these algorithms is that there must be independence of the encrypted regions from the unencrypted regions of the image. Since they deal with encrypting only a fraction of the whole image, the computation requirements are greatly lessened by this class of algorithms. Another important point to take into consideration is that this class of algorithms play a very important role in real-time applications. These algorithms play an important role specifically in medical applications. They generally help separate out information into sensitive and insensitive data only based on perception. [20-26] are papers that fall into the category of partial encryption algorithms. [20-26] are further are divided into spatial domain [20-24] and frequency domain [25-26] based algorithms.

Along with ROI based algorithms VC algorithms also forms a part of the partial encryption algorithms class. Visual cryptography is a class of encryption algorithms that does not make use of any key. It makes use of a technique where a secret image is hidden inside an image into multiple layers. So each layer of the image essentially holds some secret information. At the receivers end all one has to do is align the layers and the secret information in a proper way and the original image is revealed to the receiver plainly by human perception. The advantage of VC algorithms is that no complex computation is required to be performed in order to get the decrypted image. And since VC algorithms do not use keys, there is no key management needed for these class of algorithms [21].

The methodology of how a VC algorithm work is as following: a secret picture needs to be shared among n participants. So in order to divide this secret image among the n participants, a splitting algorithm is very important. The secret picture is divided into n transparencies/shares based on the splitting algorithm in such a way that if any m shares (as determined by the sender) are placed together then the original image become visible to the authorised receiver. But, if fewer than m shares are placed together than the original image is not revealed to the authorised receiver. The superimposition of the pixels of the various shares is achieved by using a simple logical OR operator. Thus the computational complexity of the VC class of algorithms is not much [21]. The only disadvantage of the VC class of algorithms is that the quality of reconstructed image could be very poor for certain classes of inputted images.

Wong *et al* [20] proposed a multi-level ROI image encryption universal architecture which dealt with biometric data. It made use of the multi-level encryption in addition with the stream cipher RC4 for encryption purposes. The encryption scheme in [20] made use of multiple ROIs to select the regions of interest for specific users (the authorized users in the receiving end). This implied that a particular user could only see a part of the decrypted image file which it was intended to see and not the other parts of the file which it had no authorized access to. Multiple level ROIs selected for each image and they were encrypted using three levels of authority using RC4 and biometric fingerprinting matching algorithms. If the sender had to send the image to two users at the receiving end, it first requested the authenticated server (AS) for encryption keys for the receivers. The AS generated the keys and send them to the original sender. Using all the keys obtained from AS, the sender then performed multi-level ROI encryption on the main image (the encrypted image is the same but the receivers can only decrypt that part that is sent for it and not the whole image) to form an encrypted image. This encrypted image was then send to the two receivers by the sender. The users at the receiving end send their biometric information to the AS and requested their specific keys from the AS. AS who kept a copy of the keys and the level of authority verifies the same and send the keys to the receivers if the biometric information matched the template already stored in the AS biometric database. Once the receivers received their respective keys from the AS, they can decrypt the image file and read out the portion of the image file meant for them. The encryption scheme implemented by the authors involved implementing two ROIs and also made use of two levels of authority. The work described in [20] made it less susceptible to any kind of thefts as it made use of biometric authentication. The encryption scheme in [20] was thus proved to be one of the most secure encryption schemes discussed in this review paper.

Abdulla and Sozan [21] worked on the subtractive color model CMY (Cyan (C), Magenta (M) and Yellow (Y)). There was no key involved in the encryption scheme of [21] and it was purely based on visual cryptography. The original image was decomposed to its three constituent components based on the CMY color model. Now, if the C primitive matrix was considered then every pixel $P_{i,j}$ in the matrix had $1/4^{th}$ of its

value (1/4 $P_{i,j}$) stored in another matrix which represented a new image. Since the original image was decomposed to its C, M and Y components, which resulted in three cover images existing. The encryption scheme in [21] was very simple and it was based on superimposing (made use of simple OR logical operator) the secret image (size was equal to or smaller than that of the cover image) onto the cover image, thus encrypting the cover image. Three shares were generated such that each share contained a part of the secret image. In this paper, the number of pixels in the decrypted image were the same as the number of pixels in the original image, which was a rule normally followed by text data. The proposed method in [21] was not a huge success when dark pictures with high contrast were taken into consideration as the decrypted images appeared corrupted with large amounts of noise. This distortion basically happened because the secret image generated was not distinctive. The encryption technique in [21] thus needs to work on how to get a distinct sharp secret image when dark high contrast images are involved. The security of the encryption scheme in [21] depended on the color composition and the color distribution of the secret image. This paper did not mention any detailed analysis of attacks.

Like [21], Patil *et al* [22] was also an encryption technique that was based purely on visual cryptography. The work in [22] also functioned devoid of any key and the encryption process was very simple to execute. The encryption process basically consisted of three main operations namely, sieving (logical XOR operator implemented), division and shuffling. Sieving meant breaking the original RGB image to its individual components. Division meant breaking each of the R, G and B components to numerous number of shares. For example, R could be broken to shares from R_A to R_Z. Shuffling was the last step in the encryption process which dealt with shuffling of the divided shares (from the division step) within itself. These shuffled shares were later combined to generate random shares, for instance in Figure 8 there are two random shares generated. Since majority of the operations in [22] were achieved by using logical operators like OR, XOR and the Mod operator, the computation cost incurred was very less and the implementation in [22] was written purely in java ([10-21] [23-26] implementations were mainly in MATLAB). Since encryption schemes in [21] and [22] do not have keys involved, they do not have to bother about key management or brute force attacks. But the main drawback of the encryption schemes in [21] and [22] was that, generally the quality of the decrypted or recovered image is poor. Figure 8 illustrates the encryption scheme as depicted in [22].

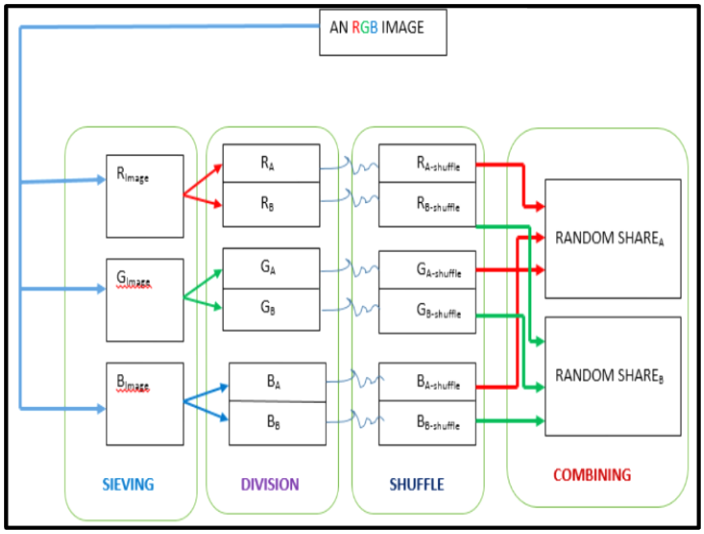

Fig. 8. Encryption scheme as illustrated in [22]

Oh *et al* [23] proposed a selective encryption algorithm (SEA), which worked on similar lines to the Advanced Encryption standard (AES) algorithm. Like AES, the SEA algorithm also dealt with block cipher. Here the core computation went through several rounds of iteration depending on the key size chosen. In short, the number of iterations required was directly dependent on the key size chosen at the beginning of the algorithm. AES was generally not considered suitable for visual data (included audio, video, image, text etc.) as it could have involved really long computation processes. This work introduced a selector component right at the beginning of the encryption process. The selector component made the selection of ROI from the inputted image and then it performed compression of the ROI based on Huffman coding. The remaining rounds of encryption was very similar to the normal AES algorithm. Figure 9 and Figure 10 depicts the similarity and differences between the SEA algorithm and AES algorithm with the SEA algorithm maintaining the rounds of iteration (computation) same as that as AES.

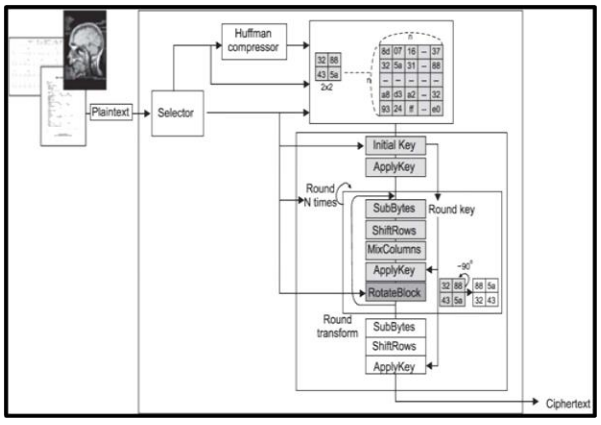

Fig. 9. SEA encryption scheme which is likely to AES (courtesy: ref [23])

SEA algorithm had the same security level that the AES standard algorithm possessed. The only difference between the

two seemed to be that the SEA encryption scheme had a compressor component and a selector component to its architecture, something that a normal AES algorithm did not have.

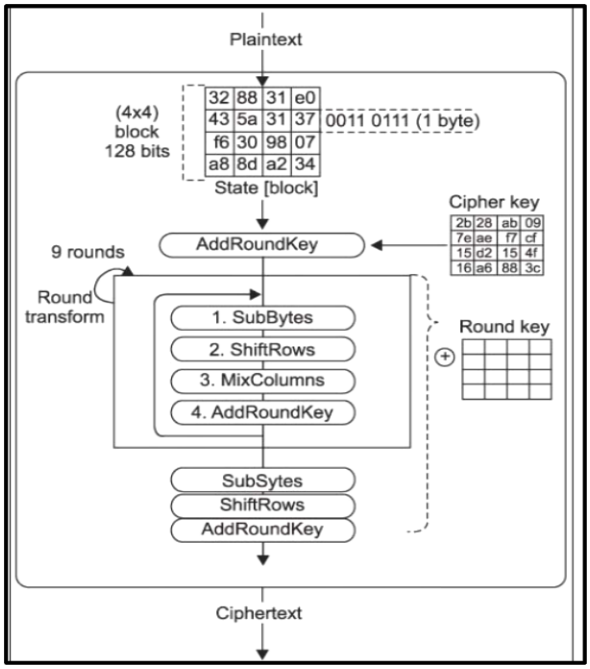

Fig. 10. Encryption scheme of regular AES (courtesy: ref [23])

The encryption technique in Parameshachari *et al* [24] was a very simple technique. The key was generated by a random key generator here. The inputted color image was initially broken to sub blocks. The selection of the blocks for encryption happened randomly using the XOR and Mod operators. The randomly selected block along with the random key generator gave the encrypted block. The encrypted block was later combined with all the unencrypted blocks which in turn produced the partially encrypted image for the encryption scheme in [24]. The additional feature [24] had was that it made use of Self Monitoring Analysis & Reporting Technology copyback system (SMART) to store long term the encrypted images generated. Figure 11 explains the flow of the encryption process at [24].

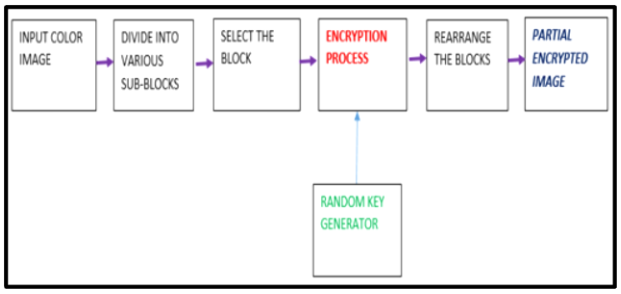

Fig. 11. The flow of the encryption scheme in [24](courtesy: ref [24])

Fan *et al* [25] and Flayh *et al* [26] were partial encryption algorithms which were implemented in the frequency domain. The encoding scheme for [25] was very much similar to the JPEG compression encoding. The only difference between the encoding scheme in [25] and that of the JPEG compression encoding was that, [25] made use of quaternion discrete cosine transforms (QDCT) instead of discrete cosine transforms used in the JPEG encoding scheme. Since the quantization phase was involved in the encryption scheme in [25], the encryption process was rendered to be lossy in nature. Just like the JPEG compression, the encryption scheme in [25] initially divided the original color image into 8x8 sized blocks. The QDCT is performed instead of normal DCT in [25]. The remaining phases remain the same as JPEG compression algorithm. That is, the next subsequent phase was quantization, followed by the zigzag scanning (exploited redundancy and organized coefficients in an array in an order of lower frequency coefficients followed by higher frequency coefficients), sorting of the coefficients and lastly the entropy coding and encryption. Figure 12 and Figure 13 attempt to draw a comparison between the encryption scheme as discussed in [25] and the JPEG compression encoding scheme.

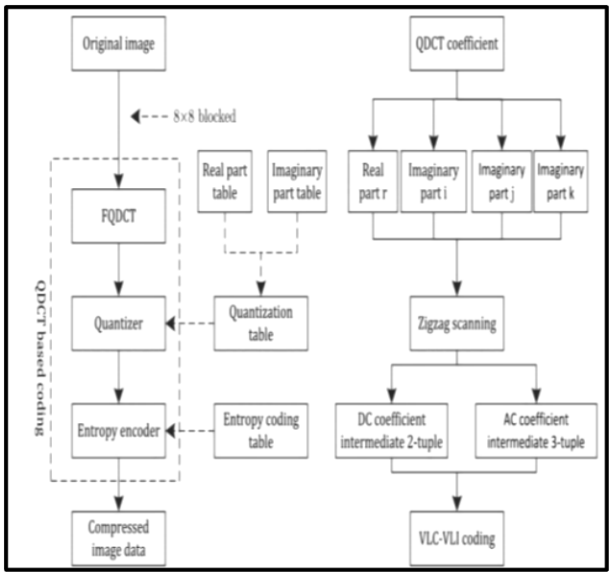

Fig. 12. Encryption scheme in [25] (courtesy: ref [25])

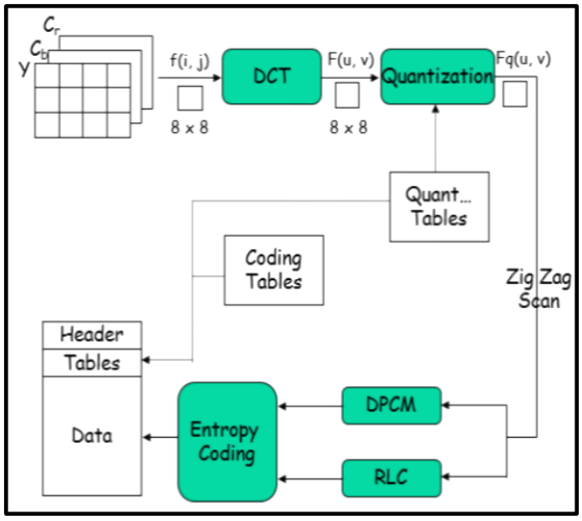

Fig. 13. JPEG compression encoding scheme (courtesy: ref [28])

Flayh *et al* [26] proposed a wavelet based partial encryption algorithm. The first step was to select a key which would be useful in the encryption process. The subsequent step was to select the wavelet filter that one would like to use in the encryption process. The encryption scheme in [26] made use of AES or stream cipher to partially encrypt the inputted color image. The whole idea behind using wavelets was to encrypt only the lower frequency coefficients as opposed to the whole image (consists of lower as well as higher frequencies). The higher frequency coefficients could be discarded or could be coded at a lower bit rate and the lower frequency coefficients could be fully encoded using AES or stream ciphers thus making it a partially encrypted algorithm. If there was a necessity to hide any further details in the cipher/encrypted image then, one could just perform a simple operation like smoothing over the image. The wavelet used for the encryption purpose in [26] was the haar wavelet. The coefficient correlation of the pixels in the cipher image was nearly equal to zero which was desirable in an encryption scheme. The main problem faced in the encryption scheme in [26] was that, as the amount of encrypted part of the image decreased, the execution time decreased too, which was a good aspect but the coefficient correlation of the pixels in the encrypted image had increased, which was not very desirable in the encryption scheme discussed in [26]. Table-2 summarizes the partial encryption algorithms discussed in this section.

TABLE II. SUMMARY OF PARTIAL ENCRYPTION ALGORITHMS. THE ALGORITHMS MENTIONED IN THE TABLE ARE CAPABLE OF DEALING WITH ANY COLOR SPACE

Ref No	Category	Domain	Class/Method	Type of dataset/Color space	Compression used??
[20]	Partial encryption algorithms	Spatial domain	ROI based algorithms	Biometric data	No
[21]	Partial encryption	Spatial domain	VC based algorithms	CMY color model	No
	algorithms				
[22]	Partial encryption algorithms	Spatial domain	ROI based algorithms	RGB color model	No
[23]	Partial encryption algorithms	Spatial domain	(VC + ROI) based algorithms	Medical/Dicom data	No
[24]	Partial encryption algorithms	Spatial domain	(VC + ROI) based algorithms	Any color space (mostly RGB color models dealt with)	No
[25]	Partial encryption algorithms	Frequency domain	Uses QDCT and very similar to JPEG compression algorithm	Any color space	Yes. Lossy compression by use of QDCT
[26]	Partial encryption algorithms	Frequency domain	Uses Haar wavelets	Any color space	Yes. Mostly lossless compression achieved because of Haar wavelets

IV. FUTURE WORK

The topic of 'Image Encryption Algorithms for Color Images' is pretty huge to cover in a single survey paper. This is a topic that has a lot of scope and is still evolving. There is a future and possibility for the partially encryption algorithms to evolve. Chaotic cryptosystem, though very new to explore, having been evolved only during the the last decade, still is not a strong enough system and can easily be broken. Many other transforms in the frequency domain of the various classes of encryption schemes could be explored and their results could be noted. Visual cryptography (VC) has a lot of scope of improvement in the near future. VC based algorithms generally render poor quality of decrypted image, so a lot of research is going on as to how to handle certain classes of images in a better manner (especially dark high contrasted images). It is difficult to say that any particular cryptosystem or encryption algorithm is the safest as most of the cryptosystems have easily been broken into. This is what makes encryption an ever evolving topic which has a lot of scope to expand in the future.

V. CONCLUSION

Few significant papers on image encryption algorithms for color images were discussed during the period 2007-2015 and the algorithms were classified as full encryption algorithms or partial encryption algorithms. Traditional cryptosystems fail to work for bulky and voluminous data like image data and they fail to produce desirable results for real-time applications. The algorithms in either of the classes were further divided based on whether they were spatial domain, frequency domain or hybrid domain based algorithms. Based on the review one can safely conclude that full encryption based encryption schemes were more suitable for higher security applications while partial encryption based encryption schemes were a lot

suitable for real-time applications as they took lesser EDT time and the computational requirements and complexity were smaller.

REFERENCES

[1] Arroyo, David, Rhouma Rhouma, Gonzalo Alvarez, Veronica Fernandez, and Safya Belghith. "On the skew tent map as base of a new image chaos-based encryption scheme." In Second Workshop on Mathematical Cryptology, pp. 113-117. 2008.

[2] Yin, Ruming, Jian Yuan, Qiuhua Yang, Xiuming Shan, and Xiqin Wang. "Gemstone: a new stream cipher using coupled map lattice." In Information Security and Cryptology, pp. 198-214. Springer Berlin Heidelberg, 2010.

[3] Öztürk, İsmet, and İbrahim Soğukpınar. "Analysis and comparison of image encryption algorithms." *International Journal of Information Technology* 1, no. 2 (2004): 108-111.

[4] Jawad, Lahieb Mohammed, and Ghazali Bin Sulong. "A review of color image encryption techniques." *International Journal of Computer Science Issues* 10, no. 6 (2013): 266-275.

[5] MULUALEM, GETACHEW MEHABIE. "COMPRESSION AND ENCRYPTION FOR SATELLITE IMAGES: A COMPARISON BETWEEN SQUEEZE CIPHER AND SPATIAL SIMULATIONS." (2015).

[6] Johnson, Mark, Prakash Ishwar, Vinod Prabhakaran, Daniel Schonberg, and Kannan Ramchandran. "On compressing encrypted data." *Signal Processing, IEEE Transactions on* 52, no. 10 (2004): 2992-3006.

[7] Klinc, Demijan, Carmit Hazay, Ashish Jagmohan, Hugo Krawczyk, and Tal Rabin. "On compression of data encrypted with block ciphers." *Information Theory, IEEE Transactions on* 58, no. 11 (2012): 6989-7001.

[8] Zhou, Jiantao, et al. "Designing an efficient image encryption-then-compression system via prediction error clustering and random permutation." *Information Forensics and Security, IEEE Transactions on* 9.1 (2014): 39-50.

[9] Ali A.Yassin. "Design New Algorithm for Partial Image Encryption Based colors Space." Journal of Babylon University, Pure and Applied Sciences, No.(2), Vol.(20): 2012

[10] R. Rhouma, D. Arroyo, and S. Belghith, "A new color image cryptosystem based on a piecewise linear chaotic map", in 6th International Multi-Conference on Systems, Signals and Devices, Mar. 2009, pp. 1–6.

[11] Musheer Ahmad, and M. Alam, "A New Algorithm of Encryption and Decryption of Images Using Chaotic Mapping", International Journal on Computer Science and Engineering, vol. 2, no. 1, 2009, pp. 46–50.

[12] Chandel, Gajendra Singh, and Pragna Patel. "Image Encryption with RSA and RGB randomized Histograms." *Image* 3, no. 5 (2014).

[13] Wang, Xingyuan, Lin Teng, and Xue Qin. "A novel colour image encryption algorithm based on chaos." *Signal Processing* 92, no. 4 (2012): 1101-1108.

[14] H. Chen, X. Du, Z. Liu, and C. Yang, "Color image encryption based on the affine transform and gyrator transform", Optics and Lasers in Engineering, vol. 51, no. 6, Jun. 2013, pp. 768–775.

[15] Samson, Ch, and V. U. K. Sastry. "An RGB Image Encryption Supported by Wavelet-based Lossless Compression." *International Journal of Advanced Computer Science and Applications* 3, no. 9 (2012).

[16] Z. Yu, Z. Zhe, Y. Haibing, P. Wenjie, and Z. Yunpeng, "A chaos-based image encryption algorithm using wavelet transform", in 2nd International Conference on Advanced Computer Control, March 2010, Vol. 2, no. 4, pp. 217–222.

[17] Zhou, Nanrun, Yixian Wang, Lihua Gong, Hong He, and Jianhua Wu. "Novel single-channel color image encryption algorithm based on chaos and fractional Fourier transform." *Optics Communications* 284, no. 12 (2011): 2789-2796.

[18] Arroyo, David, Rhouma Rhouma, Gonzalo Alvarez, Veronica Fernandez, and Safya Belghith. "On the skew tent map as base of a new image chaos-based encryption scheme." In *Second Workshop on Mathematical Cryptology*, pp. 113-117. 2008.

[19] Kocarev, Ljupčo. "Chaos-based cryptography: a brief overview." *Circuits and Systems Magazine, IEEE* 1, no. 3 (2001): 6-21.

[20] A. Wong and W. Bishop, "Backwards Compatible, Multi-Level Region-of-Interest (ROI) Image Encryption Architecture with Biometric Authentication", International Conference on Signal Processing and Multimedia Applications, July 2007, pp. 324 – 329.

[21] Abdulla, Sozan. "New Visual Cryptography Algorithm For Colored Image." *arXiv preprint arXiv:1004.4445* (2010).

[22] Patil, Shobha, and V. R. Udupi. "A Secure Approach to Image Encryption of color image without using key." *Received from: http://inpressco. com/category/ijcet* (2013).

[23] Oh, Ju-Young, Dong-Il Yang, and Ki-Hwan Chon. "A selective encryption algorithm based on AES for medical information." *Healthcare informatics research* 16, no. 1 (2010): 22-29.

[24] Parameshachari, B. D., KM Sunjiv Soyjaudah, and Sumithra Devi KA. "Partial Image Encryption Algorithm Using Pixel Position Manipulation Technique: The SMART Copyback System."

[25] Fan, Jing, Jinwei Wang, Xingming Sun, and Ting Li. "Partial Encryption of Color Image Using Quaternion Discrete Cosine Transform." *structure* 8, no. 10 (2015).

[26] Flayh, Nahla A., Rafat Parveen, and Syed I. Ahson. "Wavelet based partial image encryption." In *2009 International Multimedia, Signal Processing and Communication Technologies*. 2009.

[27] Introduction to image processing - https://www.spacetelescope.org/static/projects/fits_liberator/image_processing.pdf

[28] Image compression: JPEG, Multimedia Systems (module 4, lesson 1)

[29] Umbaugh, Scott E. *Digital image processing and analysis: human and computer vision applications with CVIPtools*. CRC press, 2010.

[30] Debbarma, Nikhil, Lalita Kumari, and Jagdish Lal Raheja. "2D Chaos Based Color Image Encryption Using Pseudorandom Key Generation." *Chaos* 2, no. 4 (2013).

[31] Pisarchik, Alexander N., and Massimiliano Zanin. "Chaotic map cryptography and security." *International Journal of Computer Research* 19, no. 1 (2012): 49.