Performance of Fractional Transforms in Image and Video Processing

A thesis submitted
in fulfillment of the requirement for the award of degree
of
DOCTOR OF PHILOSOPHY

Submitted by

Neeru Jindal

Supervisor

Dr. Kulbir Singh

Associate Professor, ECED

Department of Electronics and Communication Engineering

THAPAR UNIVERSITY, PATIALA – 147004, Punjab (India)

Certificate

Certified that the thesis entitled **"Performance of Fractional Transforms in Image and Video Processing"** being submitted by **Ms. Neeru Jindal** to **the Department of Electronics and Communication Engineering, Thapar University, Patiala (Punjab), India** in fulfillment of the requirements for the award of degree of **Doctor of Philosophy** is a record of bonafide research work carried out by her. She has worked under my guidance and supervision and fulfilled the requirements for the submission of this thesis which has reached the requisite standard. The matter presented in this thesis does not incorporate any material previously published or written by any other person except where due reference is made in the text.

The results obtained in this thesis have not been submitted in part or full to any other institute or university for the award of degree or diploma.

(**Dr. Kulbir Singh**)
Associate Professor,
Department of ECE,
Thapar University,
Patiala (Punjab)-147001,
India.

ACKNOWLEDGEMENT

First of all the author is highly indebted to almighty GOD, the most Gracious, Merciful, who blessed me spiritual support to carry out this research work. All the praise and thanks to GOD, who gave me patience, strength and ability to complete this work.

I feel pride to express immense sense of gratitude to **Dr. Kulbir Singh,** Associate Professor, Department of Electronics and Communication Engineering, Thapar University, Patiala, for his untiring efforts, who always as a mentor encourages me throughout the tenure of this work. His healthy criticism, painstaking efforts made the author capable to compile the thesis in the present form. It is my great pleasure to work under his supervision.

I am highly obliged and wish to owe my sincere gratitude to Dr. Rajesh Khanna, Professor and Head, Department of Electronics and Communication Engineering, Thapar University, Patiala. His diligence and passion on scientific research really impressed me. The author is also thankful to Dr. Anil Verma, Associate Professor, Department of Computer Science and Engineering, Thapar University, Patiala, for his support to present and compile the data.

I express my reverence and great admiration to my parents, elder brother, bhabhi who always stood behind me with blessings and selfless support. I would like to express my deepest gratitude to my husband Sh. M.K. Singla, who acted as my backbone during the whole tenure of my research work. He always motivated me and helped me a lot to complete the research work. The acknowledgement is absolutely incomplete without mentioning sacrifice of my two little daughters Tanushri and Vani, who missed upon my love and care during the work.

Neeru

(Neeru Jindal)

ABSTRACT

Fourier transform is an important mathematical tool which is used in signal processing. The generalization of Fourier transforms are fractional Fourier transforms. Similarly, the other transforms available in mathematics can be fractionalized. The additional degree of freedom provided by fractional orders of fractional transforms has encouraged the researchers for its use in many applications. The fractional transforms are used in many applications of optics and signal processing area. The aim of this work is to explore the utilization of fractional transforms in image and video applications. The main bottleneck of image and video signals is privacy and saving of memory space in internet applications. Compression and encryption are the solutions for efficient transmission of image and video signals. So, the main contribution of research done in thesis is to develop better compression and encryption algorithms for image and video signals.

The images have been compressed using fractional Fourier transforms, fractional Cosine transforms and fractional Hartley transforms. When image is compressed in the transform domain, they are generally divided into sub blocks. So, the block size is varied for these fractional transforms by dividing the image into N × N blocks, where N is 4, 8, 16 and 32. It has been observed that 8 × 8 block size for fractional Fourier transforms, 32 × 32 block size for fractional Cosine transforms and 4×4 block size for fractional Hartley transforms is better by simulation approach. The comparison of these transforms has determined the superiority of fractional Fourier transforms in image compression applications. Then image compression at various compression percentages is performed. The peak signal to noise ratio and mean square error are taken as quality parameters of reconstructed images. Also, the fractional Fourier transforms and fractional Cosine transforms are compared with existing Joint Photographic Expert Group method and observed as better.

The block based fractional transforms are used due to its energy compacting property and relative easy implementation. However, annoying blocking artifacts are noted at high compression percentages as each block is transformed and quantized independently. These artifacts degrade the reconstructed image quality. These artifacts have been analysed for fractional Fourier transform and fractional Cosine transforms and has given more affect in fractional Cosine transforms. The compressed images also need security while being transferred. There are many algorithms

available in literature to provide security to compressed images. So, an algorithm to obtain the security of compressed images using scrambling is suggested in this work. It has been established that the improved scrambling degree from existing method is attained. The fractional keys used for compression also provide the security to compressed image.

Next to compression, the fractional Fourier transforms and fractional Cosine transforms are also used in image encryption algorithms. It is shown that peak signal to noise ratio of fractional Cosine transforms is superior to fractional Fourier transforms in image encryption. The variation of fractional keys has been done in this work and it is observed that security is enhanced. The images have been encrypted using two, three and four fractional keys and taken the advantage of extra keys of fractional transforms. The sensitivity of fractional keys in decryption is also shown and observed that for proper decryption all the fractional keys and random phase masks should be correct. The popularity of digital library applications and internet commerce has increased the demand of security in the mind of content providers. The algorithm for image encryption and scrambling is suggested in this work. In scrambling, the positions of pixels are shuffled and the scrambled image becomes unrecognizable. So, scrambling enhance the security as the invader has to crack the fractional keys as well as descrambling algorithm.

A joint algorithm is suggested based on the preceding observations. The images are compressed using fractional Fourier transforms and encrypted using fractional Cosine transforms. There are two approaches for joint algorithm compression-encryption and encryption-compression. It is established by comparing these two techniques that the former is better, because the data to be encrypted is compressed earlier.

The work has been extended from still images to video. The video is motion of images including the time parameter. The video signal is encrypted using fractional transforms. The noise effects have been also analyzed in video encryption. Video is also compressed-encrypted using fractional transforms. Frames are extracted from video. The difference frame is compressed and encrypted. At various compression percentages, the proposed algorithm is observed better from SCAN method based on quality parameters.

Finally, the proposed image and video processing techniques has proven the efficacy of fractional transforms and motivated to develop more application in future.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

ABBREVIATIONS AND ACRONYMS

1D	One dimensional
2D	Two dimensional
AES	Advanced Encryption Standard
ANN	Artificial neural networks
bpp	Bits per pixel
CALIC	Context Adaptive Lossless Image Compression
C-E	Compression-encryption
dB	Decibels
DCT	Discrete Cosine Transform
DES	Data Encryption Standard
DFrCT	Discrete Fractional Cosine Transform
DFrFT	Discrete Fractional Fourier Transform
DFrHT	Discrete Fractional Hartley Transform
DFrST	Discrete Fractional Sine Transform
DFT	Discrete Fourier Transform
DHT	Discrete Hartley Transform
DPCM	Differential Pulse Code Modulation
DWT	Discrete Wavelet Transform
E-C	Encryption-compression
FFD	Fractional Fourier Domain
FFT	Fast Fourier Transform
FrFT	Fractional Fourier Transform
FrHT	Fractional Hartley transforms
FT	Fourier Transform
HDMI	High Definition Multimedia Interface
ICA	Independent Component Analysis
JPEG	Joint Photographic Expert Group
LOCO-I	Low Complexity Lossless Compression for Images
MEM	Maximum Entropy method
MOS	Mean Opinion Score

MPEG	Motion Picture Expert Group
MSE	Mean Square Error
PAD	Pixel absolute difference
PSNR	Peak Signal to Noise Ratio
RE	Relative error
RPM	Random phase mask
RSA	Rivest-Shamir-Adelman
SD	Scrambling degree
SECMPEG	Secure MPEG
SNR	Signal to Noise Ratio
SPIHT	Set partitioning in hierarchical tree
VEA	Video Encryption Algorithm

GLOSSARY OF SYMBOLS

a	Fractional order parameter
$K_\delta(t,u)$	FrFT kernel, a function of (t,u)
$H_n(t)$	nth order normalized Hermite-Gaussian function
$h_n(t)$	nth order Hermite polynomial
$\gamma(t,u)$	Impulse function
π	Pi
α	Fractional order parameter in terms of angle $\delta \cong a\sigma/2$
I	Identity operator
F	Fourier operator
$K(p,q,m,n)$	Two dimensional discrete fractional Fourier transform kernel
v_k	is the DCT-I eigenvector
$C_{N,\delta}$	N-point DFrCT kernel
$K_H^\delta(t,u)$	FrHT kernel, a function of (t,u)
$f(.)$	Function of independent variables
ζ	Angular frequency
$f(m_0,n_0)$	Real valued two dimensional data for image
$p(x,y)$	Random phase masks
$\iota(x,y)$	Randomly generated homogenously distributed functions
$r(i,j)$	Function of two variables i and j used for digital image
$o(i,j)$	Function of two variables i and j used for digital image

CHAPTER 1 | *INTRODUCTION*

The introduction defines important terms, arouse interest and prepare the mind for research in the new area of science and technology. The introduction of Fractional transforms and their applications in image and video processing are given in this chapter.

1.1 PREAMBLE

Fourier Transform (FT) is one of the most powerful, valuable, and frequently used tool in signal processing and analysis. It is a linear transform used to solve linear system problems. The capability of this transform has placed it amongst the three important mathematical advances in the last quarter of the 19^{th} century. But, the FT is unable to solve certain class of ordinary and partial differential equations which arises in the areas of optics, signal processing and quantum mechanics [1], [2].

Given the widespread use of ordinary Fourier Transform in science and engineering, it is important to recognize this integral transform as the fractional power of FT. The Fractional Fourier Transform (FrFT) is a generalization form of FT, which is richer in theory, flexible in application and its implementation cost is at par with FT [3], [4]. FT of a function can be considered as a linear differential operator acting on that function. The FrFT generalizes this differential operator by letting it depends on a continuous parameter 'a' ($\alpha = a\sigma/2$). The parameter 'a' can be interpreted as a rotation by an angle δ in the time–frequency plane. The order parameter or fraction 'a' of FrFT can be freely manipulated as it denotes the a^{th} power of FT operator F [5], [6].

The Fractional Fourier Domain (FFD) decomposition for continuous signals as well as discrete signals and systems has come into existence in evaluation of FT for real time processing [7]. But, when FrFT is analysed in discrete domain there are many definitions of Discrete Fractional Fourier Transform (DFrFT). These definitions are broadly classified according to the methodology of computation adopted. It is also obvious that none of these definitions satisfy all the properties of continuous FrFT and the non-availability of perfect and proper DFrFT expression still persists [2], [3], [8], [9]. The researchers have started the use of available DFrFT's for signal processing,

convolution, filtering and multiplexing, digital watermarking, tomography, image restoration etc. in the FFD [5], [10]-[13]. The Discrete Fractional Cosine Transform (DFrCT) and Discrete Fractional Hartley Transform (DFrHT) have been also used in various signal processing applications [2], [14].

An image is a visual representation of an object or a group of objects. A digital image is composed of a finite number of elements each of which has a particular location and value. A set of moving images is called video. Video is a stream of images composed of frames, containing both audio and pictures. The bottleneck of image and video applications is saving of bandwidth and security [15], [16]. It is concerned with reduction of the number of bits required to store or transmit images or frames without any appreciable loss of information. Reducing the bandwidth will have an effect on significant cost reductions and it results in more affordable image communication. The key issues during compression are efficiency, image quality, computational complexity and security.

The need of security has grown the effective and standardized encryption techniques. Encryption is the process of transforming information to make it readable to authenticate users. Encryption has been used for a long time to facilitate secret communication. Cryptographic algorithms that are used for image encryption are of two types. Sometimes the two end points use same algorithm and most of the time same key is used for encrypting and decrypting the image information, and in other encryption techniques, they must use different but related key for encrypting and decrypting purpose. The idea of compression and encryption of images can further be utilized for joint effort. The joint algorithm reduces the redundant information by which compressed data is encrypted. The advantages of this method are security and less memory storage.

The image compression and encryption techniques are extended to video processing. Video encryption is an extremely useful method for the stopping of unwanted interception and viewing of any transmitted video or other information [17]. It is essential to develop compression methodologies which can produce high compression and preserve good reconstructed videos [18].

The major contribution of research is improvement in the Peak Signal to Noise Ratio (PSNR) and Mean Square Error (MSE) of image and video applications using

discrete Fractional transforms. The images and videos [19] are compressed and encrypted using fractional transforms.

1.2 FRACTIONAL TRANSFORMS

Fractional transforms are linear transforms used to solve linear system problems. The concept of fractional calculus has evolved in pure mathematics. Furthermore, it is only in the last three decades that the application of fractional calculus have emerged in engineering field which lead to a significant impact in several areas and attracted the scientific and technical community to the fractional objects.

With the advent of computers and enhanced computational capabilities the Discrete Fourier Transform (DFT) came into existence in evaluation of FT for real time processing. Further these capabilities are enhanced by the introduction of Digital Signal Processors (DSP) processors and Fast Fourier Transform (FFT) algorithms. On similar lines, so there arises a need for discretization of FrFT. Furthermore, DFT is having only one basic definition and nearly 200 algorithms are available for fast computation of DFT. The FrFT in discrete domain provide definition of Discrete Fractional Fourier Transform (DFrFT). This technique gives an advantage of additional keys i.e. order parameter of the transform. The number of these additional keys can be further enhanced by using repetition the discrete fractional transform with various orders. As discussed earlier, the FrFT has found a lot of applications in signal processing, so the DFrFT for image, video processing for compression and encryption are discussed. The discrete fractional transforms provide much better results than existing methods. The method for computing the DFrFT has been given [20], [21]. DFrFT can be derived as linear combination of identity operation, discrete Fourier transforms (DFT), time inverse operation and inverse DFT.

The generalization of DCT is DFrCT which has an additional free parameter, fractional order. The DCT of a sequence is defined in [22]-[26]. The transform kernel of the widely popular DCT originates from the trigonometric representation of the Chebyshev polynomials of the first kind. The DFrCT uses the eigendecomposition of the DCT kernel. The exclusive eigenvectors are obtained from the even Hermite-Gauss eigenvectors of the Fourier matrix in the cosine case. The kernel matrix of N point DFrCT is defined in [27], [28]. The image encryption has utilized the property

of reality as a real signal has the real DFrCT [29]. The decryption process does not increase the storage or transmission load as in the case of non reality preserving transform.

In 1998, Pei *et al.* defined the definitions of the Discrete Fractional Hartley Transform (DFrHT) and the DFrFT [14]. The matrices of eigenvalues and eigenvectors for the Discrete Fourier and Hartley transform are investigated. Then, DFrHT and DFrFT are defined with the results of the eigendecompositions of the transform matrices. The relationship between DFrHT and DFrFT is also described [30].

The discrete fractional Fourier transforms has fast algorithm and it can work in a similar way of continuous FrFT. It can be used with lesser errors in many signal processing applications [28]. The enthusiasm for image and video processing using fractional transforms is based on the extra degree of freedom provided by fractional orders.

1.3 IMAGE PROCESSING TECHNIQUES

Image processing is an important area of research due to the large amount of data required to represent visual information. With the vast amount of digital images, which is still increasing rapidly, the transmission bandwidth, storage spaces and privacy become the important issues. Compression and encryption is the natural solution for this problem. Researchers are always looking for solutions that can achieve relatively higher compression percentage than existing methods while maintaining a comparable visual quality. Encryption is one of the ways to ensure security. An encryption method transforms the image data such that the transformed data can be read and understood only by someone who has the encryption key. The main purpose of encryption is to provide confidentiality, to keep data secret.

1.3.1 Image Compression

Image compression is reducing the possible number of bits without any loss of information [30], [31]. Effective image compression techniques used as very active research areas in the past two decades. It is essential to develop compression methodologies which can both produce high compression percentages and preserve good reconstructed quality. An image is represented as an array of numbers in a

computer where integers are to be more specific. The image array is usually two dimensional (2D) if it is black and white. Each number in the array represents an intensity value at a particular location in the image and is called a picture element or pixel, for short. The pixel values are usually positive integers and can range between 0 and 255. This means that each pixel of a black and white image occupies 1 byte in a computer memory. In other words, the image has a grayscale resolution of 8 bits per pixel (bpp). The captured images are rectangular in shape. The ratio of width to height of an image is called the aspect ratio. In standard-definition television the aspect ratio is 4:3, while it is 16:9 in a high-definition television. So, if an image has 480 rows, then the number of pixels in each row will be $480 \times 4/3 = 640$ for an aspect ratio of 4:3. For high-definition television, there are 1080 rows and so the number of pixels in each row will be $1080 \times 16/9 = 1920$. The dedicated channels such as high definition multimedia interface (HDMI) capable of transferring uncompressed data at this high rate over a short distance do exist, but it is difficult for long-distance transmission. Therefore, efficient data compression schemes are required to bring down the huge raw image data rates to manageable values so that practical communications channels may be employed to carry the data to the desired destinations in real time.

There are several image compression methods including predictive methods [32]-[35], transform methods [36]-[39], dictionary methods [40], [41] and other miscellaneous methods [42]-[45]. The predictive methods are based on prediction of pixel values using neighboring pixels, context modeling of prediction errors and entropy coding of errors. The current state of the art methods such as Lossless Joint Photographic Expert Group (JPEG), Context Adaptive Lossless Image Compression (CALIC) and Low Complexity Lossless Compression for Images (LOCO-I) are based on predictive techniques. The transform methods are based on decorrelation of pixel values using reversible transforms such as wavelets and entropy coding of transform coefficients. Some popular text compression methods are ZIP, GIF and COMPRESS. Other miscellaneous methods are based on techniques such as segmentation, tree presentation, and vector quantization. But, all these methods need updation of algorithm at transmitting and receiving side, when each time data has been communicated.

1.3.2 Image Encryption

Image encryption is to represent an image in such way that only authorized users can access it. The number of different encryption schemes for visual data types have been proposed in the literature [46], [47]. Various methods are devised to fulfill the security requirements for multimedia applications. Several review papers [48]-[50] have been published on image encryption providing a more or less complete overview of the techniques proposed so far. Security is the important issue in communication and storage.

General purpose image encryption approaches such as Data Encryption Standard (DES) and Advanced Encryption Standard (AES) which are permutation-substitution based iterated product ciphers. These methods encrypt data in small blocks and are not suitable for the encryption of large volume of image or video data. The chaos methods are based on complex dynamics of discretized versions of chaotic maps such as Kolmogorov and Baker maps. These methods have smaller key size. So, there was a need for the large key size encryption algorithm. The present method encrypted the image with three fractional keys. The size of fractional keys can be increased up to any number. The sensitivity of single key has been also discussed that have an effect on the MSE of retrieved image. Security of encrypted image is enhanced with scrambling. Scrambling is shuffling of image pixels to make it unrecognizable and then scrambling degree of encrypted image is measured [51], [52]. However, quality of image is reduced at receiving end with noise attacks. Noise attacks caused unwanted effect on image quality. The affect of salt-pepper and Gaussian noise attacks on the reconstructed image quality is noticed.

1.4 VIDEO PROCESSING TECHNIQUES

Video refers to pictorial (visual) information, including still images and time-varying images. The arrangement of pixels and lines in a contiguous manner of the memory is called a bitmap. There are five parameters of a bitmap: the starting address in memory, the number of pixels per line, the pitch value, the number of lines and the number of bpp. The display devices are driven by analog inputs, so digital to analog converters are used to generate component analog signals from the bitmap for display purposes. The main problem preventing the widespread use of digital video today has

been the huge storage, transmission bandwidth requirement and privacy. Therefore, the compression and encryption of video is essential [53], [54].

1.4.1 Video Encryption

Video encryption is the technique used to provide confidentiality and authenticity to multimedia contents for security. Video encryption based on discrete Cosine transform (DCT) methods is focused on standardized formats like Motion Picture Expert Group (MPEG-1,2,4) or H.26X, therefore all these techniques try to take advantage of the corresponding data formats and bit streams. The straightforward approach for this is to encrypt the entire compressed video stream with a conventional cryptographic method, such as AES [55]. This approach is called the naive algorithm approach [47]. Since the mid-1990s many research efforts have been devoted to the development of specific video encryption algorithms. A large number of algorithms have been proposed to ensure the confidentiality of video data. Early overview of these algorithms is given in [50], [56]. These methods qualify them as selective encryption algorithms. This viewpoint is not completely correct. It grasps an essential characteristic of most video encryption algorithms that only selected parts of the video stream are encrypted to reduce the encryption burden. But not all video encryption algorithms follow this idea. Some of them use their own specific mechanisms rather than applying the selective encryption principle. The video encryption algorithm using fractional transforms is presented in this work. The results are compared with existing methods and improved results are obtained.

1.4.2 Video Compression-Encryption

The video information is compressed to save memory space and then encrypted to get security is known as video compression-encryption technique. For high-definition television, there are 1080 rows and so the number of pixels in each row will be $1080 \times 16/9 = 1920$, where 16/9 is aspect ratio. A video source may produce 30 or more frames per second, in which case the raw data rate will be 221,184,000 bits per second for standard-definition television and 1,492,992,000 bits per second for high-definition television. If this raw data has to be transmitted in real time through an ideal communications channel, which will require 1 Hz of bandwidth for every 2 bits of data, then large bandwidth will be required. There are no such

practical channels in existence that will allow for such a huge transmission bandwidth, so video compression is required [57], [58].

Along with compression, the security of video is also an issue. So, video can be compressed-encrypted collectively also. There are various algorithms available in literature which performs compression followed by encryption or encryption followed by compression. The algorithm is devised for compression-encryption using discrete fractional transforms. This algorithm provides better results than video compression-encryption using SCAN patterns. The quality of frames is measured with PSNR and MSE.

There is little or very little change in the spatial arrangement of objects between two or more consecutive frames in a video [31]. Therefore, it is advantageous to send or store the differences between consecutive frames rather than sending or storing each frame. The difference frame is called the residual or differential frame and may contain far less details than the actual frame itself. So this difference frame is compressed and encrypted in algorithm.

1.5 CONTRIBUTIONS

The major contributions of this thesis can be summarized as follows:

(i) An image compression algorithm using fractional transforms is proposed which is having better PSNR than JPEG method. The block size is determined based on maximum value of PSNR at various compression percentages. The effect of blocking artifacts have been analysed in these block based transformations. The image compression and scrambling algorithm is also proposed.

(ii) An image encryption algorithm that exploits the strength of fractional transforms i.e. using extra degree of freedom provided by fractional orders has been implemented. The security analysis with variation in number of fractional keys has been analysed which is not available in the literature. The encryption and scrambling algorithm is also devised.

(iii) The outcomes of image compression and encryption algorithms are utilized to propose two joint image encryption-compression and compression-encryption algorithms. The joint compression-encryption algorithm is better

than encryption-compression. A few joint methods have been reported in the literature and research in this area is still open.

(iv) Fractional transforms based video encryption and video compression-encryption algorithms have been performed. The security and performance aspects of both approaches are analysed and compared with existing method.

1.6 THESIS ORGANIZATION

This thesis has seven chapters organized as follows:

To start with, **chapter 1** provides a brief introduction to the field of study and various applications.

The **chapter 2** presents the detailed literature review of Fractional transforms. The development of image and video applications with various techniques has been studied. Based on literature review the gaps have been discussed and the objectives are defined.

In **chapter 3,** the two dimensional applications of DFrFT, DFrCT and DFrHT for image compression have been discussed. The block sizes are optimized for fractional transforms. At optimized fractional orders, the compression percentages are calculated. The reconstructed image quality is measured with metrics MSE and PSNR. It has been analysed that in comparison with literature results, the proposed algorithm improves results. The comparison of three discrete Fractional transforms and blocking artifacts are also quantified in this chapter. To get the security of compressed images, compression and scrambling algorithm is performed.

The **chapter 4** presents an algorithm for image encryption using Fractional transforms. The encryption and decryption keys are fractional orders. Security with variation in number of fractional keys has been analysed. The sensitivity of keys has been also discussed. The effect of Gaussian noise and salt-pepper noise are also discussed. An encryption and scrambling algorithm has been analysed to increase the security.

In **chapter 5** two joint image algorithms using Fractional transforms have been devised and compared. Compression before encryption will remove redundancy of data. Brute force attack depends upon exhaustive keys search and is feasible only

for the cryptosystems with relatively small key space. The value of fractional part for each key can be increased up to any number. The increased key size makes this attack infeasible. It has been found that proposed method is robust, secure and sensitive for encryption keys, which has a good prospect and practicability in information security field.

The **chapter 6** presents the two video algorithms. Video encryption using fractional transforms is performed in which frames are extracted and encrypted. The quality metric parameters PSNR and MSE have been calculated. The histogram analysis and security against Gaussian noise is also discussed. Secondly, the video has been compressed-encrypted using fractional transforms. The redundancy of adjacent frames is exploited in the proposed algorithm. The proposed algorithm gives low MSE for the Claire and Trevor video from the existing method at various compression percentages.

The **last chapter** concludes the proposed work with findings. The future scope is proposed which will increase the performance of current state of art.

CHAPTER 2 LITERATURE REVIEW

It is very imperative to go through and understand the history of research area. In this chapter literature review provides the evolution of fractional transforms and their usage in image and video processing. The study of literature provides motivation for further research work.

2.1 INTRODUCTION

The transformation of signal provides extraction of special features that can be utilized in signal processing. The Fourier transform is one of them which in frequency domain characterize the signal. The concept of fraction in FT is called Fractional Fourier transform. The efficacy of computers and enhanced computational capabilities in the continuous domain, a discrete version of FrFT was required which can be calculated in a proficient and faster way. In contrast, to the case of DFT where it has one basic definition and lots of algorithms are available for fast computation, FrFT has several definitions in discrete domain. The literature on discrete fractional Fourier transform is reviewed in next section. The computational load of DFrFT can be reduced with the help of discrete fractional Cosine and Sine transforms (DFrCT and DFrST) given by Pei *et al.* in [29]. The eigen decompositions of the transform matrices are used to define the DFrCT and DFrST kernel matrices. The definitions of the DFrHT and the DFrFT have been presented in 1998 by Pei *et al.* [30]. The historical perspective of DFrCT and DFrHT has been also discussed in the chapter.

The fast growth of computer networks explored the discrete Fractional transforms in various applications, but reinforce in this research field is still necessitating. The rapid use of digital images and videos has made the storage spaces and privacy the primary concern. So, compression and encryption techniques are required for image and video. The literature has several methods for image and video compression techniques [59]-[61]. The protection of the image and video contents during transmission from unauthorized user is security. A number of methods have been available in the literature for the encryption of two-dimensional image and three-dimensional video information [62]-[65]. The literature has been discussed on image, video compression and encryption using various techniques.

2.2 FRACTIONAL TRANSFORMS

The history of fractional concept has found that in the 17th century Bernoulli [66] formulated a question about the meaning of a non-integer order derivative. This was the beginning of the fractional calculus which is the base of the continuous time fractional systems described by the fractional differential equations. The applicability of FT has introduced the concept of fraction in the year 1929 and lead to the development of FrFT [2]. Since then, the concept of fractional calculus has evolved in pure mathematics and developed by pure mathematicians. Afterward, the work on FrFT was conceded by H. Weyl in 1930 followed by E. U. Condon in 1937, H. Kober in 1939, A. P. Guinand in 1956 [67], A. L. Patterson in 1959, V. Bargmann in 1961, De Brujin in 1973 and R. S. Khare in 1974 [2],[68].

2.2.1 Fractional Fourier Transforms

The generalization of FT called FrFT was first introduced in 1980 by Namias [69] apparently unaware of the previous work done by various researchers which dated back to 1929. McBride *et al*. [70] modified fractional operators and also proved some theorems for the modified operators. An application to an ordinary differential equation was also considered. Almeida [71] gave definition and properties of fractional Fourier transform. The FrFT is a time-frequency transform which can reveal the characteristics of signal gradually changing from time domain to frequency domain with its order increasing from 0 to 1. The method for digital computation of FrFT by Kutay *et al.* [72] does not obey the additivity property.

In the history of discrete fractional Fourier transform development, the linear combination of the four parts has been considered as the DFrFT in many documents [2], [4], [81]. The four parts include the original signal, a circular flipped version of the signal, its DFT and a circular flipped version of its DFT. Unfortunately, this method cannot have analogous outputs as the continuous FrFT. For the mismatches, a thorough discussion is presented in [7]. An algorithm for efficient and accurate computation of the fractional Fourier transform was given in 1996 by Ozaktas [8]. This method of calculating the fractional Fourier transform requires oversampling by only a factor of two, regardless of the order of the transform. No additional cost was required to implement the fractional Fourier transform in comparison of ordinary transform. The discrete fractional Fourier transform was also suggested. However, as

is the case with the ordinary DFT, this requirement does not uniquely determine the definition of the discrete fractional transform. This DFrFT an important method for digital computation of FrFT has been also proposed [74], but the angle addition property cannot be perfectly preserved its rotational property. So signals can only be recovered back from their transforms within some approximation errors.

The definition given by Dickinson [73] and Santhanam *et al*. [74] corresponds to a completely distinct definition of the FrFT. For all classes of signals continuous and discrete, periodic and non-periodic, one and multi-dimensional, the definition of FrFT was also reported by Cariolario [5], [6]. These above listed methods were classified, by Pei *et al.* [28] in the year 2000, according to the methodology of their calculations. Although the fractional Fourier transforms have been the mainstay in transforms, a more recent approach using computers has been evolved in the last decade known as discrete fractional Fourier transforms. It simplifies both the mathematics and physical analysis. The advantages of discrete fractional transforms are that it is an evolutionary computational method, high compression ratio and its fractional part provides extra degree of freedom in computations. In 1996, there was a better improvement in DFrFT [75], [76]. The continuous FrFT can have similar results as of the DFrFT with discrete Hermite eigenvectors found by Pei *et al.* [77]. This DFrFT can have the mixed time and frequency characteristics of signals, which is the same as the continuous FrFT.

Thereafter, within a short span of time many definitions of DFrFT came into existence and these definitions are classified according to the methodologies used for calculations in 2000 by Pei *et al.* [28]. The simplest way to define the Direct form DFrFT is sampling the continuous FrFT and computing it. But this form lost many properties so it was confined. The improved sampling type DFrFT was properly sampled and resultant DFrFT was similar to continuous type. Although the computations of this transform were same as that of continuous but the transform kernel of this transform was not orthogonal and additive. The improved sampling type DFrFT was applicable to only set of signals. The linear combination DFrFT is also called discrete rotational Fourier transform or discrete angular Fourier transform. In this transform the transform matrix is orthogonal and additivity property along with the reversibility property is satisfied. The multiplication of DFT and periodic chirps is defined as group theory DFrFT. This DFrFT satisfies the additivity property and

reversibility property of FrFT and rotational property of Wigner distribution. But this DFrFT can be defined with two ways when number of points N is not prime and fractional order of DFrFT is equal some specified angle. The special case of continuous FrFT in which input function is periodic equally spaced impulse train is called impulse train DFrFT. This DFrFT has many constraints and cannot be defined for all fractions. The eigenvector decomposition type DFrFT was suggested by Pei *et al*. [78]. By searching the eigenvalues and eigenvectors of DFT matrix and fractional power of matrix was computed. It satisfies the orthogonality, additivity and reversibility property of continuous FrFT. The eigenvectors cannot be expressed in closed form and it also lacks the fast algorithms. The performance analysis of all these classes is also done [28], [79]. But, it is also clear that none of these definitions satisfy all the properties of continuous FrFT and the non-availability of a perfect and proper DFrFT expression still persists [4, 77, 80].

It has many applications in solution of differential equations, optical beam propagation and spherical mirror resonators, optical diffraction theory, quantum mechanics, statistical optics, optical system design and optical signal processing, signal detectors, correlation and pattern recognition, space or time-variant filtering, multiplexing [11], signal and image recovery, restoration and study of space or time-frequency distributions etc [81]-[83]. The researchers have also started the use of available DFrFT's for signal processing, convolution, filtering and multiplexing, digital watermarking, tomography, signal restoration etc. in the FFD [5], [84], [85].

2.2.2 Fractional Cosine Transforms

The process of decomposing a set of samples into a scaled set of Cosine basis functions is called the Forward discrete Cosine transform. The process of reconstruction the set of samples from the scaled set of Cosine basis functions is called the Inverse discrete Cosine transform. If the sample sequence is longer than eight samples, it can be divided into eight-sample groups and DCT can be computed independently for each group. Because the Cosine basis functions always have the same set of values at each of the discrete sampling points, only the coefficient values change from the one group of samples to the next. The 2D DCT has been used in 2D applications and used in the proposed work for image applications. The one dimensional (1D) DCT can be extended to apply to 2D applications. The horizontally

oriented set of basis functions represents horizontal frequencies and the other set of basis functions represents vertical frequencies. By convention, the DC term of the horizontal basis functions is to the left, and the DC term for the vertical basis functions is at the top.

The general form of DCT is a discrete fractional Cosine transform which has an additional free parameter and can be used in all applications where DCT is found to be useful. It is known that all the DFT and DCT transform kernels have infinite eigenvectors. In [73], [76], [78] a novel matrix is introduced to compute the real-value and complete set of DFT eigenvectors. This particular set of eigenvectors constitutes the discrete analogs of the continuous Hermite-Gaussian functions and also called the DFT Hermite eigenvectors. Because the DFrFT defined with these DFT Hermite eigenvectors can have similar output as continuous FrFT and will have the properties of unitarity, additivity, and reversibility, the DFT Hermite eigenvectors had been used in developing DFrCT and DFrST as reasonable choices.

The DFrCT and DFrST developed by Pei *et al.* in 2001 are not the same as the conventional DCT and DST with real values in the kernel matrices. By taking the real part and imaginary part of the DFrFT kernel, some types of fractional Cosine and Sine transforms have been derived [29]. Although these fractional transforms are real and can be implemented with incoherent light [86], [87], [88]. However, the angle additivity will not exist for these transforms, and more importantly, they do not have simple inverse transforms.

The properties of the DFrCT and the DFrST are inherited from the DFrFT [29]. The reality property affirms that DFrCT of a signal is real if the signal is real. This property is utilized in image encryption. The decryption process does not increase the storage or transmission load as in the case of non reality preserving transform.

2.2.3 Fractional Hartley Transforms

The procedure to compute the fractional Hartley transform has been summarized by Pei [14]. But this 1D Hartley transform has restricted applications. So, Zhao *et al*. [89] defined 2D Hartley transform and its application in optical image encryption. This 2D Hartley transform was inverse transform and convenient for

various practical applications. Li *et al.* in 2008 [90] discussed the image encryption using simplified fractional Hartley transform, which is a real transform. It was concluded that this method was more convenient and efficient in practical applications. Still, the applications of 2D Hartley transform in signal processing are to be explored.

The Hartley transform is a striking substitute and suitable real replacement for the well known complex Fourier transform [91]. In 2008, Sontakke *et al*. discussed the analyticity theorem and inversion formula for the generalized fractional Hartley transform and using that uniqueness theorem was proved. The fractional Hartley transform of selected functions was also proved and operations transform formulae for this transform was obtained. Hartley transform is getting greater importance in several applications.

The relationship between the DFrHT and DFrFT has been found by reviewing the relationship between the DFT and discrete Hartley transform (DHT) [14]. The eigenvalues and eigenvectors of the Discrete Fourier and Hartley transform matrices are investigated. Then, the results of the eigen decompositions of the transform matrices are used to define the DFrHT and DFrFT.

Researchers have presented FrHT by summing the real and imaginary parts of the kernel of FrFT. Although FrHT defined in that way supplies a real output for a real input, it cannot satisfy the additive property. Moreover, it does not have an inverse transform, and accordingly, so primary function cannot be recovered directly, which restricts its applications. Jimenez *et al*. in 2011 [92] suggested a 2D generalization of the 1D FrHT, which had been redefined recently in another form by Pei *et al.* [28]. A new optical encryption method based on FrHT is presented. It was different from those ever-proposed encryption methods. When the image was encrypted by this means, it takes efforts for FrFT or other transforms that we have known to recover the encryption image but in vain. Like many well-known transforms, it also has a number of potential applications. Compared with the conventional HT, it is superior in optical information processing, for FrHT can strengthen the information security to some extent with its additional fractional orders. Besides, with the new definition, the 2D FrHT is a reversible transform analogous to FrFT. These fractional transforms have been used in image and video

processing applications. A brief history of these applications is given in the next section.

2.3 IMAGE COMPRESSION TECHNIQUES

In 1987, JPEG conducted a selection process based on a blind assessment of subjective picture quality, revealed that the 8 × 8 DCT, had produced the best picture quality. The data compression will give better results if it can be represented in statistical form [93], [94]. So, at the input to the encoder, source image samples are grouped into 8 × 8 blocks, shifted from unsigned integers to signed integers and input to the Forward DCT. At the output from the decoder, the Inverse DCT outputs 8 × 8 sample blocks to form the reconstructed image.

Wallace [93] in 1990 highlighted the definition and the overall structure of JPEG. JPEG is an ISO/CCITT working group in the process of developing an international standard for general-purpose, continuous-tone, still-image compression. The overall algorithm structure consists of (i) the Baseline System, a simple coding method sufficient for many applications, (ii) a set of Extended System capabilities, which extend the Baseline System to satisfy a broader range of applications, and (iii) an Independent Lossless method for applications needing that type of compression only. The Baseline System is the heart of the JPEG standard. The history and requirements to develop JPEG method for continuous-tome image compression are also discussed.

The techniques and details of wavelet coding to better understand the JPEG 2000 standard were discussed in 2001 [95]. The focus was on the fundamental principles of wavelet coding and not the actual standard itself. There are two types of filter choices: orthogonal and biorthogonal. Orthogonal filters have the property that there is energy or norm preserving. Nevertheless, modern wavelet coders use biorthogonal filters which do not preserve energy. Reasons for these specific design choices are explained. The basic properties of the wavelet transform which are pertinent to image compression were also discussed. Subband coding or “early” wavelet coding method was also discussed followed by an explanation of the EZW coding algorithm. Other modern wavelet coders that extend the ideas found in the EZW algorithm were also described.

A novel way of representing images based on FFD filtering configurations, leading to a method for compressing images was discussed by Yetik *et al*. in 2001 [96]. Fractional Fourier-domain filtering consists of (i) taking the fractional Fourier transform of the input signal, (ii) multiplication with a filter function, and (iii) taking the inverse fractional Fourier transform of the result. FFD filtering has been further generalized to multi-stage and multi-channel filtering. There are two categories of unknowns, the fractional Fourier transform orders and the filter coefficient in multi-stage and multi-channel filtering configurations. The problem of finding the optimal filter coefficients, given the transform orders can be solved using a minimum mean square approach. For many systems encountered in various applications, it is possible to approximate the system with a multi-stage or multi-channel configuration with acceptable mean square error, by using a small or moderate number of stages or channels. Since the cost of implementing the fractional Fourier transform (optically or digitally) is similar to the cost of implementing the ordinary Fourier transform, this leads to a fast implementation of the space-variant system [96]. The optimal filtering coefficients minimizing the mean square error between the original matrix and its multi-stage and multi-channel approximation are taken as the compressed version of the image. An order of magnitude compression is possible with moderate errors, large compression ratios are accomplished by larger errors.

The image is compressed with a compression ratio of 8, 21.3 and 32 and it has been observed that the compressed version has a lot of error in case of 32 and 21.3. The paper concludes that the idea does not yield better results than presently available compression algorithms. However, it says that the method proposed is a basic one in its rawest and barest form. It encourages further refinement and development of methods and their combination and joint use of other techniques which may lead to full-fledged compression algorithm with better performance. It also expresses the idea that exploring and exploiting these issues seems potentially rewarding.

A scheme for signal compression based on the combination of DFrFT and set partitioning in hierarchical tree (SPIHT) was presented in 2005 [97]. The application of the scheme to different types of signals demonstrates significant reduction in bits leading to high signal compression ratio. The better results were obtained as compared with DCT. However Cebrail *et al.* [98] in 2008 under the theory of data compression analyzed, main subjects of compression (proportion of compression,

repetitions caused by coding, fidelity criteria) components of image compression systems. Thereafter lossy and lossless image compressions were also analyzed. With the passage of time many techniques were developed for image compression using neural network, wavelet transform, phase information etc. The artificial neural network (ANN) [99] was used for image compression by training the net using the image to be compressed. The back-propagation multilayer perceptrons ANN considers two facts, psycho visual features and information contained in images. The algorithms, on application on the image data preserves most of the characteristics of the data while working in a lossy manner and maximize the compression performance. The results were checked with and without the use of quantization, and without median filtering of the image. The further method of image compression was by using only phase information [100]. It described a new spectral lossy compression method which can reduce required memories; adaptively retrieve original images by using only spectral phase information; increase the peak-to-correlation energy at the output of the correlator; and be easily employed in major encryption techniques. An optimal phase coding based on 'a fading grid' was executed to increase the compression ratio of this method. In fact, a variable number of quantization bits had been used to quantize phase information depending on the importance of the spectral phases. The phase information could be classified according to the concept of 'RMS duration'.

Yue *et al.* in 2012 [101] discussed the research status and progress of wavelet-based image compression then points out the main problems. The wavelet transform has local characteristics on the time and frequency domain, it makes up the deficiency of DCT. Moreover, its multi-resolution characteristics can easily associate with the human visual system. Besides, wavelet-based image compression is prone to combine with new image coding methods. It has become the research hotspots at present. The analysis of existing image compression algorithms was also structured in 2012 [102].

The images are subdivided in $n \partial n$ blocks where, $n \cong$ 4, 8, 16, 32 and so on. to reduce the computational complexity in compression techniques. But, blocking artifacts appeared in block-based transforms for image compression. Several papers are available in literature [103]-[107] for pre-processing and post-processing of images using the filters to remove the blocking artifacts. An algorithm had been developed [108], [109] in 2007 to remove the annoying blocking artifacts from low-

bit-rate JPEG compressed images. The blocking artifacts were modeled as 2D step functions. A frequency domain algorithm extracts all the parameters needed to detect the presence of blocking artifacts by using visual perception along with the block statistics. The boundary regions between blocks were identified as smooth and non-smooth regions. It was found that there was a significant improvement in the perceptual quality of the JPEG compressed images after removal of blocking artifact. The values of block boundary measure index, both for vertical and horizontal boundaries between the blocks, were also seen to move very close to the corresponding values of the original images after application of the new algorithm.

A new method [110] in 2011 for blocking artifact detection and reduction was introduced. The key idea of the detection process was based on the power spectrum estimation of the pixel absolute difference (PAD) in both horizontal and vertical directions. The power spectrum of the PAD is estimated using the Maximum Entropy method (MEM) which is a nonparametric method to perform accurate power spectrum estimation. The motivation for the MEM was that it was an intuitively satisfying approach to power spectrum density estimation in that a minimal amount of assumptions were made concerning the input data in the form of PAD. The blocking artifact reduction process was based on the modified projection operator using the best matching pixels vector in four blocks surrounding the boundary vector. Beside the compression the security of images is the main issue.

2.4 IMAGE ENCRYPTION TECHNIQUES

Image encryption is an essential feature of modern communication and data storage. For future multimedia internet applications image encryption is required. Even in the biometric images of fingerprints and retinal scans replaced the password codes to identify individual users. However, such information will likely be sent over a network. An intruder may copy the information when such images are sent over a network. Security can be achieved by encrypting these images. Additionally, by encrypting non-critical images as well, an intruder is less likely to be able to distinguish between important and non-important information [44], [45]. Image encryption can also be used to protect privacy as in medical imaging applications. Recently, in order to reduce the cost and to improve service, electronic forms of medical records have been sent over networks from laboratories to medical centers.

These medical records, which include many images, should not be disclosed to any unauthorized persons. Medical images, therefore, should be encrypted before they are sent over networks [111]. Unlike the conventional cryptographic algorithms, image encryption using fractional transforms provide good results because of extra keys.

In optical communication FrFT is used [112]-[115] from a last decade. Sinha *et al.* [116] presented digital signature technique to encrypt an image. The digital signature of the original image was added to the encoded version of the original image. In 2003, Hennelly *et al.* [117] suggested an image encryption technique, which was based on phase retrieval method using the FrFT. Hennelly *et al.* [118] bring together a review of the number of methods for the encryption of 2D information using optical systems based on fractional Fourier transform. A measure of the strength/robustness of the level of encryption of the various techniques was proposed and a comparison was carried out between the methods. The mean square error measured the level of encryption between original image and decrypted image.

This method gives the review of the first Fourier based optical encryption scheme [119]. This makes use of the FT, which is FrFT of order one and two phase masks. The resulting wave field had an amplitude distribution equal to the original image. But extra degree of freedom offered by the FrFT was not utilized. So, an optical image encryption scheme [120] was further developed. The two phase masks were used in the form of two statistically independent white sequences uniformly distributed. An additional random phase key and an additional FrFT operation had been added in the encryption and decryption, to further encrypt the data [121].

The other method [122] used some arbitrary number of phase keys and FrFT operations to encrypt data. The encryption scheme based on the convolution operation [123] was developed. The fractional convolution integral fractionally convolved two functions.

A new method was outlined for encryption using the FrFT [124]. A three channel system was simulated and the functions were chosen to be white random intensity functions. By making use of the times four periodicity of the Fourier transform [125] a different fractional Fourier transform was derived. A technique based on a random shifting or Jigsaw algorithm [126] do not use any phase keys in order to decrypt the image and yet encrypted the image in a very similar way.

But, the use of FrFT in encryption is more rewarding than compression because of additional key available with the transform as order parameter. One possible definition of the DFrFT has a correlation property which has been used to derive a recursive algorithm for the phase retrieval of a signal provided the availability of the intensities of two fractional Fourier transforms of the original signal [117].

Zhou *et al.* [127] encrypted the image using multiple orders of FrFT in optical communication. The issue also aroused from the use of double random phase encoding in image encryption. The security of an information hiding method using the double-random phase-encoding technique were analyzed and improved by many researchers later on [128]-[135].

Scrambling is a widely used algorithm for the privacy of images. The pixel positions are interchanged using different scrambling algorithms. Two new methods for scrambling digital images were introduced in 2003 [136]. The first was based on the 3-dimensional Arnold transformation and the second method was based on the generalized Gray code transformation. A new digital image scrambling method based on Fibonacci numbers was also presented [52]. The uniformity and periodicity of the scrambling transformation were discussed. This scrambling transformation has the following advantages: (i) encoding and decoding was very simple and they could be applied in real-time situations (ii) the scrambling effect was very good, the information of the image was re-distributed randomly across the whole image; and (iii) the method was endured against common image attacks, noise and loss of data packet. In 2007 [137], a new image scrambling algorithm based on queue transformation needs one step instead of two steps to complete the scrambling. Meanwhile, reference point can be changed in every stage. If the attackers want to decode the hiding image, the step, the reference point and the direction should all been known. So it was difficult for the attacker to decode. Another scrambling algorithm based on wavelet transformation and queue transformation was developed in 2008 [138], [139] with better scrambling degree. The original image in wavelet domain was queue transformed. It was turned out by experiment that the scrambling effect of this algorithm was better. The scrambling algorithm based on random shuffling strategy, which can scramble non equilateral image and has a low cost to build coordinate shifting path was also given in 2008 [51].

Nowadays many algorithms are approaching to improve the scrambling degree. The main attacks are salt-pepper noise, Gaussian noise and brute force attack [122] in scrambling and encryption. For effective encryption these attacks should be infeasible.

2.5 JOINT IMAGE COMPRESSION-ENCRYPTION TECHNIQUES

The present multimedia communication scenario demands exchange of information with more security and reduction in both the space requirement for data storage and the time for data transmission. This can be accomplished by compression and encryption, such kind of system is called joint compression–encryption system. So far, these two technologies have been studied separately. With the progress in technologies the need is to take advantage by joining these two techniques. The Lv *et al.* [140] proposed a method to increase the security by integration of compression and cryptography. The compression technique used was the k-PCA for image compression. The main advantage of using k-PCA is that this technique is data-dependent. Unlike other existing universal compression techniques, such as DCT, discrete Wavelet transform (DWT) and so on, the non-linear transformation function used in k-PCA was found while the data to be compressed. Thus, even if the encrypted data were deciphered by some malicious persons, it was hard for them to reconstruct the images without the transformation function, which was encrypted and kept by the user as a secret key. To encrypt the compressed data and the transformation function, any existing techniques can be used, as long as they are relatively secure. It was shown that without sacrificing the compression performance or the processing speed, security could be achieved. The joint compression-encryption scheme is also discussed in 2006 [141.]

The integrated algorithms are available not only for single image compression-encryption, but for batch of images also. Pal *et al.* [142] presented a novel image compression-encryption scheme for batch of images. A hybrid approach of vector quantization and principal component analysis is presented for improving the compression ratio. The simple shuffling process was used to make the encryption process faster. In this scheme, improved compression ratio with acceptable image quality was achieved. The scheme was also secured from the noise attacks. A survey of the recent progress in multimedia encryption with increasing emphasis on

integrating encryption and compression for improved system efficiency and operability is proposed by Wu *et al.* in his Academic report on Joint Compression-Encryption of Multimedia Contents [143]. By performing compression and encryption jointly in one unified step, one can greatly simplify the system design, reduce energy consumption, and facilitate advanced multimedia manipulations. In order to assess the security level of this class of methods, the joint compression-encryption methods using Huffman coding, Exp-Golomb coding, arithmetic coding, and Lempel-Ziv-Welch coding were investigated and evaluated [143]. It was shown that most of the existing techniques are vulnerable against various attacks. Therefore, solely relying on the randomness offered by the compressors cannot achieve high level of security. Nevertheless, joint compression-encryption is still attractive for applications that require middle-level security and relatively short-term copyright protection.

The joint approach can be used for video data. A survey on video sequences results that joint compression and encryption algorithms reduced 40% of the memory storage size and they increased execution speed up to 21% [144]. Sharma *et al.* [145] describes that Joint multimedia compression and encryption techniques can significantly reduce the computational requirements of video processing systems. The two algorithms Secure Wavelet transform and Chaotic Arithmetic Coding were proposed to reduce the computational cost of multimedia encryption, while also preserving the properties of compressed video (useful for scalability, transcoding, and retrieval), which endanger loss by naive encryption.

Two approaches are available in literature to communicate large size data in secure manner. Encryption before compression: A good encryption algorithm should be successful in forging output stream without patterns. Compression relies on patterns in order to gain any size reduction. Since encryption destroys such patterns, the compression algorithm would be unable to give much reduction in size if it is applied to encrypted data. So, encryption before compression will have lack of patterns in the output. Secondly, the compressor does not have access to the cryptographic key. So it must be able to compress the encrypted data (also called ciphertext) without any knowledge of the original source. It will give minimal compression gain, since the output of an encryption will look very random. But in literature various algorithms are available with this approach.

Kingston *et al.* [146] proposed a technique which takes advantage of the Mojette transform properties. The basic crypto-compression scheme presented was based on a cascade of Radon projection which enables fast encryption of a large amount of digital data. In their method, standard encryption techniques such as AES, DES, 3DES, or IDEA can be applied to encrypt very small percentages of high resolution images and can transmit uncorrelated data along with the encrypted part. Entropy coding was used for lossless compression. The compression ratios provided by the proposed technique cannot compete with lossless JPG2K but advantage is that the percentage of encrypted data can very strongly be reduced allowing the use of public key encryption algorithms, such as RSA.

The compression of encrypted data is possible by using distributed source coding suggested by Kumar *et al.* [147]. They considered the encryption, followed by lossless compression of gray scale and color images. They also proposed to apply encryption on the prediction errors instead of directly applying on the images and use distributed source coding for compressing the cipher texts. Decompression and decryption are performed in a single phase. They achieved compression ratios varying from 1.5 to 2.5 despite encryption. The compression result as 5.39 bits per pixel was obtained on Lena image.

Li *et al.* [148] used a RC5 stream cipher based scalable encryption scheme for low complexity transparent transcoding. CCSDS compression method is used which consist of two part DWT and Bit plane coding. Advantage is that Encryption is scalable. Liu *et al.* [149] used stream cipher based Slepian-Wolf coding for encryption. It is proposed to compress the image progressively, such that the decoder can observe a low-resolution. Also the compression used is lossless. Theoretical analysis shows that, despite the inefficiency of channel codes, their scheme achieves 70% to 90% rate saving of that of the optimal conventional intra-frame coder. The proposed practical system shows better coding performance than the previous approach, which exploits a 2D Markov model in the SWC.

An image encryption algorithm that consists of two parts: scrambling of plain-image and mixing operation of scrambled image using discrete states variables of chaotic maps was proposed by Radha *et al.* [150]. Discrete Cosine transform is used for compression. The proposed algorithm is strong in providing security and is also

very fast. Since the key space is large therefore the attacker cannot decrypt an encrypted image without the correct key. The existing work on encryption before compression is also reviewed in the past [151]. Compression before encryption: It has long been appreciated that there are advantages to eliminating regularities in the plaintext before encrypting. Compression should be done first before encryption because of many advantages. Compression before encryption also slightly increases practical resistance against differential cryptanalysis (and certain other attacks) if the attacker can only control the uncompressed plaintext, since the resulting output may be difficult to deduce. It makes Brute force attack infeasible, discussed at the end of chapter. Brute force attacks perform by trying various keys and decrypting the data and checking if the output data makes any sense. By compressing it first, an attacker must decrypt the data and then decompress it before seeing if the output data makes any sense. So, it make problem for attacker. In this context several algorithms are presented in the literature.

An image data compression-encryption scheme by using the words (patterns, or orders) produced by an image processing language called SCAN in 2004 by Maniccam *et al.* [152]. The SCAN is a formal language-based two-dimensional spatial-accessing methodology which can efficiently specify and generate a wide range of scanning paths or space filling curves. The proposed methodology can compress and encrypt both binary and grey level images. Compression is based on Genetic Algorithm approach using fractal based language G-SCAN. Encryption is carried out using transposition cipher based on SCAN. It is based on permutation of N×N image i.e. (N×N)!. Compression Ratio of 2.8:1 is achieved.

Ito *et al*. [153] proposed a method combining encryption and compression based on Independent Component Analysis (ICA) and discrete Cosine transform. In their method for encryption, target images are covered with an insignificant image to hide them and their mixtures to be transmitted are obtained. The receiver reconstructs the original images applying some Independent Component Analysis algorithm to the mixed images. For compression process they used DCT and simple low pass filter. Using the proposed method the higher frequency components are cut off, that is, the quality of the original image is reduced.

Compression using DWT was performed by Younggap *et al.* [154]. For encryption Standard Encryption algorithm AES is used. Algorithm is suitable for Medical image and video. It is fault tolerant algorithm to alleviate error avalanche effect due to the erroneous bits in the received encrypted image data. Maheswari *et al.* [155] employed lossless compression using a novel layer based compound image compression technique that uses XML compression and JPEG to compress data. The FG layer is compressed using an XML compressor and BG layer is compressed using JPEG 2000. The encryption scheme, called, Shuffle Encryption Algorithm, proposed by Yahya *et al.* [156] is used. The average total time (compression time + decompression time) taken by XMLCC to compress an image was 0.79 seconds for compression and 0.68 seconds for decompression. But these results are still slower than DjVu. However the XMLCC technique is superior to both JPEG and DjVu in terms of PSNR showing an average difference of 4.6 dB and 2.13 dB respectively [151].

The literature on image compression and encryption has been discussed. The introduction of additional time dimension or temporal sequence of frames is known as video. Video engineering is quickly becoming digital discipline [31]. The digital videos are huge in volume. So, videos require compression and security especially when transferred over the network in internet applications. In context of these issues, video encryption and compression-encryption techniques are reviewed in the next section.

2.6 VIDEO ENCRYPTION TECHNIQUES

Digital videos are very important in the field of education, entertainment and multimedia applications. So, the transfer of digital videos on World Wide Web requires privacy. The awareness to security has focused the researchers to develop video encryption techniques. Video encryption is a powerful technique for preventing the unwanted interception and viewing of transmitted video for example from law enforcement video surveillance being relayed back to a central viewing centre [157]. It is not easy to use video encryption techniques directly as video data are often of large volumes and require real time operations. Encryption is acknowledged as one of the main components of any organization. It is no longer limited to secure susceptible military information applications only. It is considered industry standard for

providing information security, trust, controlling access to resources, and electronic financial transactions.

All videos that are needed to be protected from suspicious users require encryption. To solve the problem of security, a wide variety of video encryption techniques have been reported [158]-[161]. In past decade, many encryption algorithms have been discussed to get security. In general, video data takes more time for encryption, because of its large size. There are two strategies for encryption, namely, full encryption algorithms and partial encryption algorithms. The fully encryption algorithms consumes more time and memory. The overall system performance decreases due to the huge computation overhead involved.

Raju [162] presented a computationally efficient and secure video encryption algorithm based on the principle of Secret Sharing. The strength of the DC is distributed among the AC values based on Shamir's Secret Sharing scheme. This makes secure video encryption feasible for real-time applications without any extra dedicated hardware. The computational efficiency is achieved by exploiting the frequently occurring patterns in the DCT coefficients of the video data. Computational complexity of the encryption was made proportional to the influence of the DCT coefficients on the visual content. On an average, the algorithm takes only 8.32 ms of encryption time per frame.

AES or DES is the most clear-cut method to encrypt every byte in the complete Moving Picture Experts Group stream. A brief introduction of AES algorithms [163] is presented in this paper. A novel algorithm classification is discussed with the AES algorithm of video encryption. The AES was extended to support a key stream generator for encryption which can overcome the problem of textured zones existing in other known encryption algorithms. The advantages of this were that it offers high security, and can be realized easily in both hardware and software. The algorithm is flexible in supporting any combination of data and key size of 128, 192, and 256 bits.

The security level is make certain by Naive algorithm [164], [165] to the entire MPEG stream by standard encryption schemes because no effective algorithm to break encryption schemes especially AES nor triple DES so far. But this algorithm is very slow especially when we use triple DES. So it is not pertinent solution for big

video. For real time video encryption applications, the delay increases and overload became unacceptable in this method. So, Naive Algorithm is very slow due to applying DES on whole MPEG stream.

By permutation the video contents are scrambled within a frame of MPEG stream in the design of pure permutation algorithm. It is practical in circumstances where the hardware decodes the video, but decryption must be done in software. Slagell in [166] discussed the vulnerabilities of the permutation video encryption algorithm. This is a very fast encryption algorithm that works on a fully encoded MPEG stream. The key is used to divide the frame into blocks in which bytes are set down in a round-robin fashion. A powerful known-plaintext attack against this algorithm is also discussed. The modifications are also suggested that could make this algorithm a viable solution. It has been concluded that frequent re-keying and an outside method of distributing the new keys provides reasonable security while still taking advantage of the fast encryption/decryption rates, but this comes at the cost of a more complicated key distribution system. Mostly, such a solution limits the amount of damage a frame of leaked plaintext can cause. Video encryption algorithms using MPEG and its applications are also given [167].

Qiao *et al.* in [168] suggested a new video encryption algorithm called video encryption algorithm (VEA) in which video stream is separated into chunks. These chunks are divided into two different lists (odd and even lists). Afterward, applying an encryption algorithm like DES to the even list and the final ciphertext is a concatenation of output of encryption algorithm XOR with the odd list streams.

The idea of pure permutation is simply to apply a permutation technique for the intracoded or I-frames. Mutually the both side users have only the correct permutation to encrypt and decrypt the video streams respectively.

Naive Algorithm and VEA are the most secure algorithms. When comparing the algorithms in terms of size metric, video encryption algorithm, pure permutation algorithm and Naive algorithm do not change their size, which is very much enviable. There are many choices when applying different encryption algorithms to MPEG encoded video and its choice depends on the applications [169].

In order to reduce the amount of processing overhead and to meet the security for real time video applications, selective encryption techniques have been presented. The idea of this scheme is to encrypt different levels of selective parts of MPEG stream by using the feature of MPEG layered structures (e.g. encrypting all headers and I-frames, encrypting all I-frames and all I blocks in P- and B-frames). The basic selective encryption is based on the MPEG I-frame, P-frame, and B-frame structure. It encrypts the I-frame only because; conceptually P- and B-frame are useless without knowing the corresponding I-frame [170].

Chaos based algorithms are most prominent algorithms in the field of neural network to perform encryption and decryption as it is a low cost algorithm and is suitable for large amount of data. The performance comparison of various video encryption techniques using chaos was performed. Several video encryption techniques like encryption of raw video data, video encryption with and without region of interest, encryption of compressed video data etc has been discussed. There are also various other algorithms for video encryption as Chaotic Arithmetic Coding [171] in literature. In Chaotic Arithmetic Coding, a large number of chaotic maps can be used to perform coding and the choice of map is managed by a key. This method performs video encryption without effecting coding efficiency.

The security and performance aspects of the digital video encryption from a cryptographic point of view were analysed in 2012 by Mayank *et al.* [170]. The video-Frame encryption provides a good trade-off between encryption robustness, flexibility, and real-time processing. Work had been extended with digital video steganography where it was possible to disguise a given video with another video.

A Survey on several video encryption algorithms has been presented [172], [173]. Several video encryption algorithms with advantages and applications are discussed. But demand of efficient technique for quality in reconstructed video is still persisting.

2.7 VIDEO COMPRESSION-ENCRYPTION TECHNIQUES

The size of video data is immense in volume, it needs to be compressed and encrypted to avoid delay and security threats. So, in compression-encryption algorithm, both the steps, namely, compression and encryption are integrated together.

In general, any compression-encryption algorithm will provide two levels of security and consumes less time when compared to independent compression and encryption algorithms.

Zeng *et al*. [174] in 2003 presented a scrambling and compression framework in which digital video data were efficiently scrambled in the frequency domain without affecting the compression efficiency. Some features of this framework were, scrambling process was very simple and efficient, provided different levels of security, had very limited adverse impact on the compression efficiency and no adverse impact on the error resiliency, allowed more flexible selective encryption, transcodability/scalability, transparency.

Wu *et al*. [175] designed two approaches for integrating encryption with multimedia compression at a low computational cost. It was concluded that security could be achieved without sacrificing the compression performance or the processing speed. The advantages and limitations of partial encryption were also discussed. Tang [176] has contributed the idea of incorporating cryptographic techniques with digital image processing to achieve compression and encryption in single step. A light weight encryption mechanism was provided. Wang *et al*. [177] designed a lightweight, efficient, scalable, format-compliant video encryption algorithm, which was based on the DCT coefficients scrambling. This encryption algorithm was based on the concept of permutation group. Scrambling DCT coefficients of the permutation groups maintains the statistical property of DCT distribution so that the encryption does not suffer from DCT vulnerability attack. In addition, choosing a subset of permutation groups for encryption makes the algorithm efficient and scalable in terms of video data confidentiality. The weakness of video streaming from piracy and malicious attack was reduced with this algorithm. The security mechanism for multimedia data was also discussed by Meyer *et al*. [178]. This security mechanism was named secure MPEG (SECMPEG) and target was to maintain confidentiality and integrity of video contents. The manipulation of JPEG-pictures was performed and found that it was enough to encrypt the DC's and 3-8 AC's to get the best results. To ensure the integrity of a MPEG-I-Video-stream, at least the following information has to be saved by a hash-sum: all headers and trailers, all master-DC and DC's of all I-frames, all master-DC's of P- and B-frames and all motion-vectors. For confidentiality some code based on the MPEG-Decoder of the Portable Video Research Group was

implemented, that decoded the MPEG-I-video stream to get needed information and then did encryption or decryption. It was found that this was not a efficient method; it needed about 30% of the decoding time of a normal MPEG-I-video stream, the basic factor there was the Huffman-decoding with about 12-17%. But then it was realized that it could find most of the needed information without making a Huffman-decoding. The integrity-mechanism was the Cyclic Redundancy Check using the 16- and the 32-bit-variation. It didn't ensure a 100% certificate, but was well known, easy to implement and had quick 16-bit-variations. Other mechanisms like MD-4 or MD-5 were now prepared to get included in the SECMPEG-stream. The speed of the implementation was the main aim of this project for integrity, therefore the decoding of the MPEG stream to recognize all needed information was not possible, the CRC computation was far quicker then everything else in relation to the coding or decoding of a MPEG-I-stream. The confidentiality and integrity of SECMPEG method was also implemented.

Spanos *et al.* [179] limits the amount of data to be encrypted or decrypted by using video compression to reduce the size of transmitted video images. The performance of Aegis (utilizing MPEG video compression across an ATM network) was explored through an extensive simulation study. Three types of video traffic: CATV, Studio TV, and video conference were considered. The simulation results demonstrate the delay performance, as well as, the queue requirements for the Aegis encryption scheme versus the full encryption and no encryption schemes.

The video encryption algorithm which used a secret key to randomly change the signs of all DCT coefficients in an MPEG stream was presented in 1998 by Shi *et al.* [180]. It worked on a small portion of original video, so speed of algorithm was good. It only selectively encrypted a small number of bits of the MPEG compressed video and selected bit is only XORed one time with the corresponding bit of the secret key. Its efficiency was better than DES algorithm. But, this algorithm was vulnerable for plaintext attack. A new version of VEA with less computational complexity was presented by Shi *et al.* [181]. It encrypts the sign bits of differential values of DC coefficients of I-frames and sign bits of differential values of motion vectors of B- and P-frames. The directions of motion vectors were changed when the sign bits of differential values of motion vectors were changed. Modified VEA encrypt DC coefficients of I-frame, and leave AC coefficients of I-frame unchanged. Thus it

significantly reduces encryption computations. Varying a few sign bits of differential values of DC coefficients will have an effect on many DC coefficients during MPEG decoding as DC coefficients of I-frames are differentially encoded. MPEG's differential code of DC coefficients and motion vectors increase the difficulty to break MVEA encrypted videos. The secret key should be used once to provide security. If not, the secret key can be computed by XORing the DCT sign bits.

A novel technique was developed in 2006 by Bose *et al*. [182] for secure encryption and compression of data, which relies on a standard zeroth-order adaptive arithmetic coder for compression and makes the arithmetic coder's statistical model variable in nature using bit stream generated by the chaotic systems pseudorandom bit generator. Chaos-based arithmetic coder and decoder had been designed and developed. The algorithm was resistant to chosen plaintext attacks because model depends on all text that has been coded and the output from the engine was in the form of variable sized words.

Wong *et al*. [183] developed another algorithm for the simultaneous compression and encryption using chaotic maps for lossless data compression and lossy image compression. The effectiveness of this scheme was confirmed by the satisfactory ciphertext-to-plaintext ratio using standard data and image files.

In 2011, Reaz *et al*. [184] designed a Single Core Hardware Module to implement partial Encryption of compressed images. The lossless quadtree compression and RSA encryption algorithms were chosen for implementation due to their computational simplicity in hardware. But it was found that the compression and encryption process was faster than the decompression and decryption process. Although a number of algorithms have been discussed with pros and cons but compressed and secure video method is still required.

2.8 MOTIVATION

The literature has discussed mainly compression and encryption of images in the optical domain. There was a need for exploiting the advantages of fractional transforms in the area of signal processing and especially in compression and encryption. Fractional Cosine and Hartley are real transforms and are good candidates

for image compression. The extra degree of freedom of FrCT and FrHT can be exploited for image compression.

A lot of optical image encryption algorithms using FrFT are available which use different schemes. It is evident that the additional key given by fractional transforms makes them more suitable for encryption. It was concluded that the mean square error is zero when all the keys are matched. But variation in number of fractional keys and performance analysis for security and complexity has not been yet reported. Not only fractional transforms are used, but to increase the robustness certain others algorithms like scrambling is connected in the proposed image encryption algorithm.

The image compression and encryption techniques using various algorithms has been reported separately, but the joint algorithm for both (compression and then encryption) using fractional transforms has not reported so far.

The consolidated study of history of FrFT presented in this chapter is enough to create the interest. The work for image compression and encryption has been further extended to video processing using FrFT. It has been concluded from literature that privacy and less memory storage are main key issues in today's internet applications. So, several video encryption and compression-encryption techniques are available in literature. But a new and efficient algorithm is yet to be explored in comparison of existing methods.

2.9 OBJECTIVES OF WORK

Based on the literature review and motivations, following objectives have been explored in signal processing applications:

- To apply and analyse fractional transforms in image processing for possible improvisation in the existing system.
- To devise an algorithm for joint image compression and encryption system.
- To develop algorithm using DFrFT for video processing.

CHAPTER 3

IMAGE COMPRESSION

To reduce the storage space and large transmission bandwidth image compression is a necessity. It is concerned with reduction of the number of bits required to store or transmit images without any appreciable loss of information. The underlying basis of image compression using fractional transform is the removal of redundant data and better results are attained from JPEG method.

3.1 INTRODUCTION

Image compression techniques have been developed in the fields of medicine, geology oceanography etc. and become the latest area in digital signal processing. The digital image compression is required due to redundancy and irrelevancy of data. This redundancy is proportional to the amount of correlation among the image data samples. For example, in a natural still image, there is usually a high degree of spatial correlation among neighboring image samples. The irrelevancy is image data which is not noted by human visual system [185]. The purpose of image compression is also to reduce the amount of data required for representing sampled digital images and therefore reduce the cost for storage and transmission [186]-[188]. A digital image attained by sampling and quantizing the continuous tone picture requires an enormous storage. For instance, a 24 bit color image with 512 × 512 pixels will occupy 768 Kbyte on a disk, and a picture twice of this size will not fit in a single floppy disk. To transmit such an image over a 28.8 Kbps modem would take almost 4 minutes. So, the limitation of memory or disk space is a familiar challenge in computing and will continue to be, since disk space and memory cannot be unbounded. Eventually, adding hardware does not answer this growing need, since applications continue to demand increasing amounts of space. There is a roundabout: as more space becomes available, data and media that previously could not be stored use all of the newly available space. To help alleviate space limitations, a large number of compression techniques are available in literature [189]. In lossless compression, the reconstructed

The outcome of this chapter has been published in Research Journal as per following detail: N. Jindal, K.Singh, Image and Video Processing using Discrete Fractional Transforms, Signal Image and Video Processing, Springer 2012. DOI 10.1007/s11760-012-0391-4

image after compression is numerically identical to the original image on pixel by-pixel basis. However, only a modest amount of compression is achievable in this technique. In lossy compression on the other hand, the reconstructed image contains degradation relative to the original, because redundant information is discarded during compression. The second categorization is 'predictive coding' and 'transform coding'. In predictive coding, information already sent or available is used to predict future values, and the difference is coded. Differential Pulse Code Modulation (DPCM) is one particular example of predictive coding. Transform coding, also called block quantization, is an alternative to predictive coding. Transform coding on the other hand, first transforms the image from its spatial domain representation to a different type of representation using some well-known transforms, and then codes the transformed values (coefficients). The primary advantage is that, it provides greater data compression compared to predictive methods, although at the expense of greater computations. Several approaches to image compression exist, with a variety of strengths and computational requirements e.g. Huffman coding [190-191], using sparse PCA coding in curvelet domain [192], JPEG 2000 compression coding [193], using DCT [194], wavelet domain [195], visual sensitivity-based low-bit-rate [196], using neural network [99] etc.

The work in this chapter is involving the use of fractional transforms in image compression. Transform coding is also called block coding because it uses an $N \times N$ block of pixels at a time. The transform coefficients are quantized and dequantized; the mean values of the blocks are not reconstructed to the same original values. This imperfection becomes noticeable at low bit rates and appears as distinct blocks known as blocking artifact. Blocking artifacts are clearly noticeable, especially in flat areas [197]. The blocking artifacts are also noticed in image compression using fractional transforms discussed in the chapter. Along with compression, an algorithm for the security of images using scrambling is devised at the end of chapter.

3.2 IMAGE COMPRESSION

Image compression techniques transformed a 2D pixel array into a statistically uncorrelated data set from a mathematical viewpoint. The transformation is applied prior to storage of image. The compressed image is decompressed after some time to reconstruct the original image or an approximation of it.

Image recovery is most often achieved by constructing an objective function to quantify the quality of an image estimate, then optimizing that function to obtain the desired results. Some important objective functions and solutions from estimation theory are reviewed. The loss of information or distortion measure is usually evaluated with the Peak signal to noise ratio, Mean square error and compression percentages to estimate image compression algorithms. The MSE and PSNR are quality parameters for reconstructed image and defined with equations:

$$MSE \cong \left\{ \frac{1}{MN} \sum_{i=0}^{M-1} \sum_{j=0}^{N-1} [r(i,j) - o(i,j)]^2 \right\} \tag{3.1}$$

$$PSNR \cong 10 \log_{10} \left\{ \frac{M \times N}{MSE^2} \right\} \tag{3.2}$$

Compression percentage = (Compressed image size/ Original image size) ×100

where, M × N is the size of the images, $r(i,j)$ and $o(i,j)$ are the matrix elements of the decompressed and the original images at (i,j) pixel. The minimum MSE and larger PSNR (dB) correspond to good reconstructed image. In the final analysis, the human observer determines the quality of the reconstructed image. At various compression percentages, the images are compressed using fractional transforms in the presented work.

3.2.1 Image Compression using Fractional Transforms

The images are compressed using fractional transforms due to fast speed algorithms and small errors [28]. The brief mathematical discussion is analysed for these transforms.

(a) ***Fractional Fourier transforms***

By consecutive applications of the forward Fourier transform F on the signal $y(t)$ numerous times, we will get [198]

$$F^2[y(t)] \cong y(-t) \tag{3.3}$$

$$F^3[y(t)] \cong Y(-w) \tag{3.4}$$

$$F^4[y(t)] \cong y(t) \tag{3.5}$$

Based upon the above notation, in the time-frequency plane, the Fourier transform of a signal can be taken as a π/2 angle rotation of the signal. The FrFT is developed and treated as a rotation of the signal to arbitrary angles in the time frequency plane, and it satisfies the subsequent rotation properties:

(i) Zero rotation: $R_0 \cong I$

(ii) Additivity of rotation: $R_\delta R_\varepsilon \cong R_{\delta . \varepsilon}$

(iii) Consistency with Fourier transforms: $R_{\sigma/2} \cong F$

(iv) 2π rotation: $R_{2\sigma} \cong I$

the rotation operation in the time-frequency plane is signified by R. F is the traditional Fourier transform operation. The rotation angles α and β are parameters between the transformed signal and the time axis in the time-frequency plane. The FrFT transform kernel is defined as follows [69], [71]:

$$K_\delta(t,u) \cong \begin{cases} \sqrt{\dfrac{1 - j\cot\delta}{2\sigma}}\, e^{j((u^2 + t^2)/2)\cot\delta - jut\csc\delta}, & \text{if } \delta \text{ is not multiple of } \sigma \\ \gamma(t - u), & \text{if } \delta \text{ is multiple of } 2\sigma \\ \gamma(t + u), & \text{if } \delta + \sigma \text{ is multiple of } 2\sigma \end{cases} \tag{3.6}$$

$$K_a(t,u) \cong \sum_{n=0}^{\infty} e^{-jan} H_n(t) H_n(u), \tag{3.7}$$

the rotation angle of the transformed signal for FrFT is α. $H_n(t)$ is the nth order normalized Hermite-Gaussian function with unit variance.

$$H_n(t) \cong \frac{1}{\sqrt{2^n n! \sqrt{\sigma}}} h_n(t) e^{-t^2/2} \tag{3.8}$$

where, $h_n(t)$ is the nth order Hermite polynomial [68]. Because the Hermite-Gaussian function is an eigenfunction of the Fourier transform, Eq. (3.8) is treated as the eigen-decomposition of the FrFT kernel. The eigenvalue of continuous FrFT is $e^{-j\delta n}$. By using the FrFT kernel, the FrFT of the signal $y(t)$ by angle α is computed as

$$Y_\delta(u) \cong \int_{-\infty}^{\infty} y(t) K_\delta(t,u)\, dt \tag{3.9}$$

The signal $y(t)$ can be recovered back by an FrFT operation with backward angle (-α):

$$y(t) \cong \int_{0\clubsuit}^{\clubsuit} Y(u) K_{0\delta}(u,t)\,du \tag{3.10}$$

In [198], the 2D FrFT transform kernel with various orders in two dimensions is defined as follows:

$$K_{\delta,\varepsilon}(s,t,u,v) \cong \frac{1}{2\sigma}\sqrt{1 0\ j\cot\delta}\,\sqrt{1 0\ j\cot\varepsilon}\ \partial\, \mathrm{e}^{[j(s^2 .\ u^2)/2\cot\delta\, 0\, jsu\,\mathrm{cos}\,ec\,\delta]}\,\partial \\ \mathrm{e}^{[j(t^2 .\ v^2)/2\cot\varepsilon\, 0\, jtv\,\mathrm{cos}\,ec\,\varepsilon]} \tag{3.11}$$

the rotation angles of the transformed signal are α and β for 2D FrFT. Using this 2D FrFT kernel, the 2D FrFT of the signal $y(s, t)$ by angle parameter (α, β) is computed as

$$Y_{\delta,\varepsilon}(u,v, \cong \int_{0\clubsuit}^{\clubsuit}\int_{0\clubsuit}^{\clubsuit} K_{\delta,\varepsilon}(s,t;u,v, y(s,t, ds\,dt \tag{3.12}$$

The signal $y(s,t)$ can be recovered by a 2D FrFT operation with backward angles $0a, 0\varepsilon$,

$$y(s,t) \cong \int_{0\clubsuit}^{\clubsuit}\int_{0\clubsuit}^{\clubsuit} Y_{\delta,\varepsilon}(u,v) K_{0\delta,0\varepsilon}(u,v,s,t)\,du\,dv \tag{3.13}$$

Although the FrFT have been the mainstay in transform, a more recent approach from the last decade known as DFrFT simplifies both the mathematics and physical analysis. For the 2D DFrFT the Dickinson *et al.* [73] introduced a commuting matrix **M** to compute the real eigenvectors of the DFT kernel matrix **H**:

$$M \cong \begin{Bmatrix} 2 & 1 & 0 & 0 & \ell & 0 \\ 1 & 2\cos\zeta & 1 & 0 & \ell & 0 \\ 0 & 1 & 2\cos 2\zeta & 1 & \ell & 0 \\ \mp & \mp & \mp & \prec & \ell & \mp \\ 1 & 0 & 0 & 0 & \ell & 2\cos N\, 0\, 1, \zeta \end{Bmatrix} \tag{3.14}$$

where, $\zeta \cong 2\sigma/N$ and N is the size of the DFT kernel matrix. Matrix **M** commutes with the matrix **H** and satisfies commutative property: MH=HM. The eigenvectors of matrix **M** and **H** are the same but their eigenvalues are different. The eigenvectors of

M are orthonormal to each other and eigenvalues are real because **M** is a real symmetric matrix. In [75] Pei *et al.* used the DFT eigenvectors obtained from matrix **M** to construct the DFrFT kernel. The eigenvectors of matrix **M** are treated as discrete Hermite functions [76], [77].

The real and orthogonal eigenvectors were not used for further research and practical applications, even a method for computing the DFT real eigenvectors is proposed in [70]. To construct the DFrFT kernel, Pei *et al.* used the DFT eigenvectors obtained from matrix ***S*** in [76], [77]. The eigenvectors of matrix ***S*** are treated as discrete Hermite functions in [73]. In addition to the DFT Hermite eigenvectors, an eigenvalues assignment rule and the eigenvalues are also required for the DFrFT kernel construction. The eigenvalues assignment rule is developed and it is shown in Table 3.1.

Table 3.1: Eigen values assignment rule of DFrFT kernel matrix.

N	The eigen values
$4m$	$e^{0\,jk\delta}$, $k \cong 0,1,2,\ell$, $(4m\,0\,2),4m$
$4m\,.\,1$	$e^{0\,jk\delta}$, $k \cong 0,1,2,\ell$, $(4m\,0\,1),4m$
$4m\,.\,2$	$e^{0\,jk\delta}$, $k \cong 0,1,2,\ell$, $4m,(4m\,.\,2)$
$4m\,.\,3$	$e^{0\,jk\delta}$, $k \cong 0,1,2,\ell$, $(4m\,.\,1),(4m\,.\,2)$

Such an assignment rule can make the constructed kernel consistent with an identity transform when α=0 and a DFT while α=π/2. After the eigenvalues and eigenvectors of the DFT kernel matrix are determined, the transformation kernel of DFrFT can be easily defined by determining the fractional powers of the eigenvalues. The transform kernel of DFrFT is computed as

$$F^{2\delta/\sigma} \cong VD^{2\delta/\sigma}V^{T} \tag{3.15}$$

The DFrFT of a signal can be computed with a transformation kernel with equation:

$$Y_\delta \cong F^{2\delta/\sigma} y \cong VD^{2\delta/\sigma} V^T y \tag{3.16}$$

To compute inverse DFrFT [67], DFrFT is calculated with order –α.

$$y \cong F^{02\delta/\sigma} Y_\delta \cong VD^{02\delta/\sigma} V^T Y_\delta \tag{3.17}$$

In signal processing, many two dimensional unitary transforms have been used, such as discrete cosine transform [59], discrete Walsh transform [199], and so on. The (*M*, *N*) point 2D unitary discrete transform is computed as [198]

$$Y(m,n) \cong \prod_{p\cong 0}^{M01} \prod_{q\cong 0}^{N01} y(p,q) K(p,q,m,n) \tag{3.18}$$

where $K(p,q,m,n)$ is the 2D transform kernel. If $K \cong K_1 \varnothing K_2$, then the transform kernel $K(p,q,m,n)$ is called separable [198], where $\varnothing$ denotes the tensor product. For a 2D separable kernel, its transform can be implemented by row-column computation:

$$Y(m,n) \cong \prod_{p\cong 0}^{M01} \left\{ \prod_{q\cong 0}^{N01} y(p,q) K_2(q,n) \right\} K_1(p,m) \tag{3.19}$$

In [73], the 2D continuous FrFT transform kernel is separable. So the 2D DFrFT is also defined with a separable form. Thus it is defined as

$$R_{(\delta,\varepsilon)} \cong R_\delta \varnothing R_\varepsilon \tag{3.20}$$

where R_α, R_β are the 1D DFrFT transform kernel proposed in [76,77]. These two parameters in DFrFT, α and β, indicate the individual fractional orders in two dimensions. Then the forward and inverse 2D DFrFT are computed as

$$Y_{+\delta,\varepsilon,}+m,n, \cong \prod_{p\cong 0}^{M01} \prod_{q\cong 0}^{N01} y+p,q, Y_{+\delta,\varepsilon,}+p,q,m,n, \tag{3.21}$$

$$y(p,q) = \sum_{m=0}^{M-1}\sum_{n=0}^{N-1} Y_{(\delta,\varepsilon)}(m,n)\, Y_{(0\delta,0\varepsilon)}(p,q,m,n), \tag{3.22}$$

(b) ***Fractional Cosine Transforms***

The definition of DCT has been well reviewed in [29]. In [29], four types of DCT kernel matrices are presented, and they are given as follows:

$$C^{I}_{N-1} = \sqrt{\frac{2}{N}}\left\{k_m k_n \cos\left[\frac{mn\sigma}{N}\right]\right\} \quad for\ m,n = 0,1,.....N \tag{3.23}$$

$$C^{II}_{N} = \sqrt{\frac{2}{N}}\left\{k_m \cos\left\{\frac{m\left[n-\frac{1}{2}\right]\sigma}{N}\right\}\right\} \quad for\ m,n = 0,1,....N-1 \tag{3.24}$$

$$C^{III}_{N} = \sqrt{\frac{2}{N}}\left\{k_n \cos\left\{\frac{\left[m-\frac{1}{2}\right]n\sigma}{N}\right\}\right\} \quad for\ m,n = 0,1,....N-1 \tag{3.25}$$

$$C^{IV}_{N} = \sqrt{\frac{2}{N}}\left\{\cos\left\{\frac{\left[m-\frac{1}{2}\right]\left[n-\frac{1}{2}\right]\sigma}{N}\right\}\right\} \quad for\ m,n = 0,1,....., N-1 \tag{3.26}$$

k_m and k_n in the above four definitions are defined as

$$k_m = \begin{cases} \frac{1}{\sqrt{2}}, & m = 0\ and\ m = N \\ 1 & others \end{cases} \tag{3.27}$$

The DCT-I kernel is periodic with period two and has symmetric structures. The periodicity means that repeated application of DCT-I would give the original sequence. DCT-IV is the same as DCT-I for symmetry and periodicity, but DCT-II and DCT-III operators are the forward and inverse transform pair of each other and are non-periodic [29].

Similar to the DFrFT, the N-point DFrCT kernel can be defined as [29]

$$C_{N,\delta} \cong V_N D_N^{2\delta/\sigma} V_N^t \tag{3.28}$$

$$C_{N,\delta} \cong V_N \begin{Bmatrix} 1 & & & 0 \\ & e^{02j\delta} & & \\ & & \prec & \\ 0 & & & e^{0j2(N01)\delta} \end{Bmatrix} V_N^T \tag{3.29}$$

where $V_N \cong \lfloor v_0 | v_2 | \ell \ | v_{2N02} \rceil$. v_k is the DCT-I eigenvector obtained from the kth-order DFT Hermite eigenvector by (15). While α=π/2 the DFrCT will become the conventional DCT-I. When α=0, $C_{N,\delta}$ is an identity matrix. The steps for constructing the *N* point DFrCT kernel with angular parameter α are summarized as follow:

Step 1: Compute the M_c point DFT Hermite even eigenvectors.where, M_c=*2(N -1)*

Step 2: Use Step 1 to compute the DCT-I eigenvectors from the DFT Hermite even eigenvectors.

Step 3: Determine the DFrCT transform kernel by the following equation. $C^{\delta} \cong HG.D^{2\delta/\sigma}.HG^T$

(c) ***Fractional Hartley Transforms***

The transform kernel $K_H^{\delta}(t,u)$, of fractional Hartley transform is given by [14]:

$$K_H^{\delta}(t,u) \cong \sum_{n\cong 0}^{\clubsuit} o_n^{\delta} e_n(t) e_n(u) \tag{3.30}$$

$$K_H^{\delta} \cong [E(t,u) . \ O(t,u)] e^{0(t^2 . \ u^2)/2} \tag{3.31}$$

where E(t,u) and O(t,u) is defined as

$$E(t,u) \cong \sum_{n\cong 0}^{\clubsuit} \frac{e^{0jn\delta\sigma}}{2^{2n}(2n . \ 1)\sqrt{\sigma}} H_{2n}(t) H_{2n}(u) \tag{3.32}$$

$$O(t,u) \cong \sum_{n\cong 0}^{\clubsuit} \frac{e^{0jn\delta\sigma}}{2^{2n.1}(2n . \ 1)\sqrt{\sigma}} H_{2n.1}(t) H_{2n.1}(u) \tag{3.33}$$

The two dimensional fractional Hartley transform of a function f(s,t) is given as:

$$g_H^{\delta}(v,u) \cong \int_{0\clubsuit}^{\clubsuit}\int_{0\clubsuit}^{\clubsuit} K_H^{\delta}(s,v,t,u) f(s,t)\, ds\, dt \tag{3.34}$$

$$K_H^{\delta}(s,v,t,u) \cong \sqrt{\frac{10\ j\cot\delta}{2\sigma}} e^{j\frac{s^2 .\ v^2 .\ t^2 .\ u^2}{2}\cot\delta} +\cos(sv\csc\iota\ .\ tu\csc\iota\ ),\ .$$
$$e^{j(\iota\ 0\sigma/2)}(\sin(sv\csc\iota\ .\ tu\csc\iota\ )) \tag{3.35}$$

The 2D FrHT is separable in two dimensions.

The relation between fractional Hartley transforms (FrHT) and fractional Fourier transform (FrFT) is given as [14]:

$$g_H^{\delta} +u, \cong \frac{1.\ e^{\frac{j\delta\sigma}{2}}}{2} g_F^{\delta} +u, .\ \frac{10\, e^{\frac{j\delta\sigma}{2}}}{2} g_F^{\delta} +0u, \tag{3.36}$$

The two dimensional FrHT can be computed by applying one dimensional transform row wise and column wise separately. The relation between DFrFT and DFrHT is given by:

$$g_F^{\delta} +u, \cong \frac{1.\ e^{\frac{j\delta\sigma}{2}}}{2} g_H^{\delta} +u, .\ e^{j+\sigma/2,\delta}\ \frac{10\, e^{\frac{j\delta\sigma}{2}}}{2} g_H^{\delta} +0u, \tag{3.37}$$

Moreover, if the real and imaginary parts of the DFrHT are both even symmetric, then $g_F^{\delta} +u', \cong g_H^{\delta}$, means that DFrFT is equal to DFrHT.

3.2.2 Block Sizes

The spatial block size is chosen to give good compression performance while keeping the number of overhead bits small [189]. The images have been compressed using DFrFT, DFrCT and DFrHT. The image can be compressed for any compression percentage with optimized fractional order with different block sizes. The block size are $n \partial n$ blocks where, $n \cong$ 4, 8, 16, 32 and so on. The simulation results of ten images have been executed. The Pyramid image with all possible block sizes using DFrFT, DFrCT and DFrHT has been shown in Figures 3.1, 3.2 and 3.3. From these figures; it has been observed that DFrFT with 8 × 8 block size provides better results but the DFrCT with 32 × 32 block size performed well. The DFrHT has proven that 4 × 4 block size is better with high PSNR at different compression percentages.

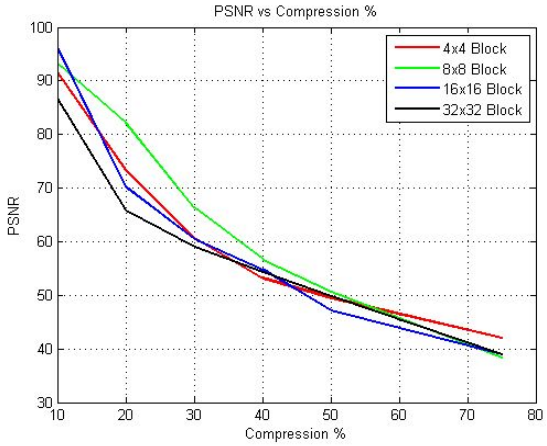


Figure-3.1: Comparison of block sizes using DFrFT for Pyramid image.

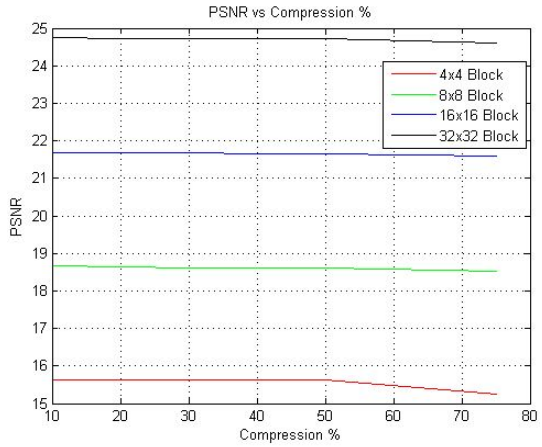


Figure-3.2: Comparison of block sizes using DFrCT for Pyramid image.

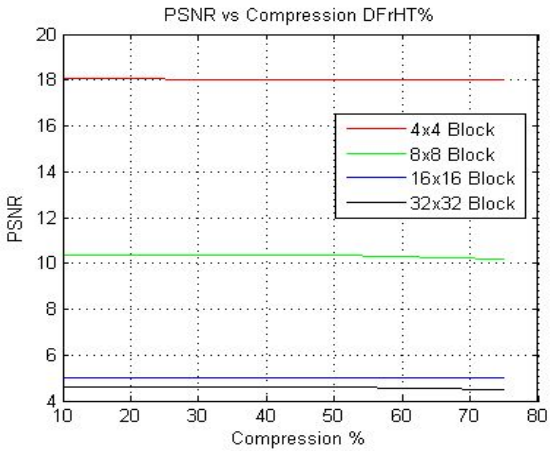


Figure-3.3: Comparison of block sizes using DFrHT for Pyramid image.

3.2.3 Image Compression Using Fractional Fourier Transforms

The algorithm has following steps for image compression with 2D fractional transforms.

Step 1: An image is first separated into non-overlapped sub images. The most popular sub image sizes are 8 × 8 and 16 × 16. For simulation results, implementation scheme sub image size chosen is 8 × 8.

Step 2: A 2D discrete fractional transform (DFrFT, DFrCT and DFrHT) is applied to each block at optimum value of fractional order with selected compression percentage. The degree of data reduction is called compression percentage. For simplicity, the 'α' order along x (rows) and y (columns) directions is taken to be same. This is done to convert the gray-scale levels of pixels in spatial domain into coefficients in transform domain.

Step 3: The quantization of these coefficients is done to selectively eliminate or more coarsely quantize the coefficients that carry the least information. A compromise can be made between image quality and compression percentage by adjusting the coarseness of the quantizer called cutoff value. The optimized quantized coefficients are arranged from lower-frequency to higher-frequency components and further compressed by efficient run-length coding approach.

Step 4: At decoding end, simply contrary process of encoding using inverse 2D discrete fractional transform is performed. Inverse discrete fractional transform is obtained by inverted value of 'α' that was used in forward discrete fractional transform with the same value. The encoder and decoder for image compression are shown as shown in Figure 3.4.

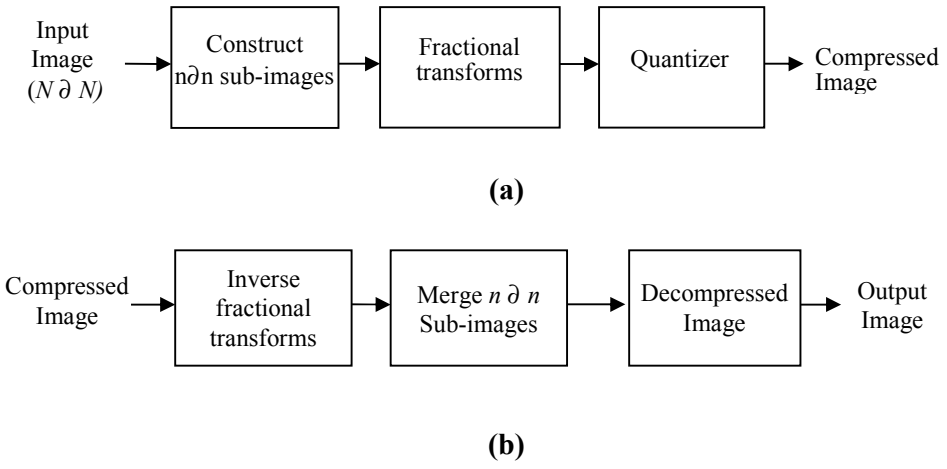


Figure-3.4: Image compression using fractional transforms (a) encoder and (b) decoder.

At the encoder side, an image is first partitioned into non-overlapped 8 ∂ 8 sub-images. Then, a two-dimensional DFrFT is applied to each block to convert the gray levels of pixels in the spatial domain into coefficients in the frequency domain. The Run-length coding is used. The final step in compression process is to quantize the transformed coefficients according to cut off selected and value of *a*. By adjusting the cutoff of the transform coefficients, a compromise can be made between image quality and compression factor. The high compression percentage can be achieved using the DFrFT by varying its free parameter *a*, even for same cut off. Quantizers are designed for optimum value of fractional order at particular compression percentage. At decoder simply inverse process of encoder is performed by using inverse two-dimensional DFrFT.

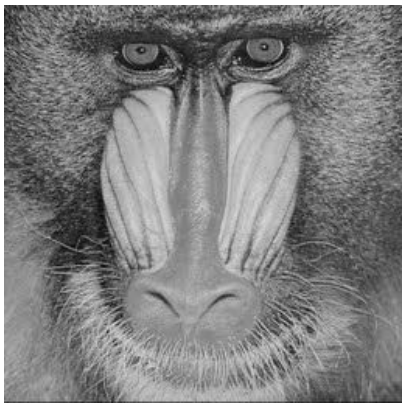
Image compression algorithms were implemented on natural gray scale test images of size 256 ∂ 256 shown in Figure 3.5 (Pyramid, Pentagon, girl, Lena, Baboon, Boat, Flower, House, Barbara and Peppers). These test images are mostly used for research purposes in image compression nowadays.

(a) Pyramid image.

(b) Pentagon image.

(c) Girl image.

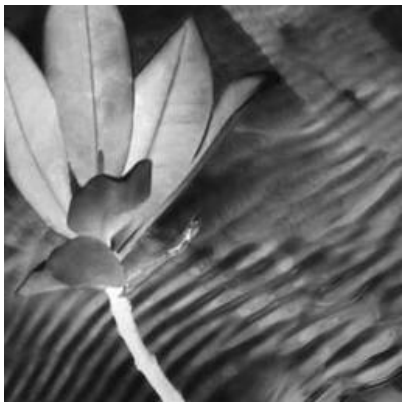
(d) Lena Image.



(e) Baboon image.

(f) Boat image.

Figure-3.5: Different Test images for compression and encryption. (contd.)



(g) Flower image.

(h) House image.

(i) Barbara Image.

(j) Peppers image.

Figure-3.5: Different Test images for compression and encryption.

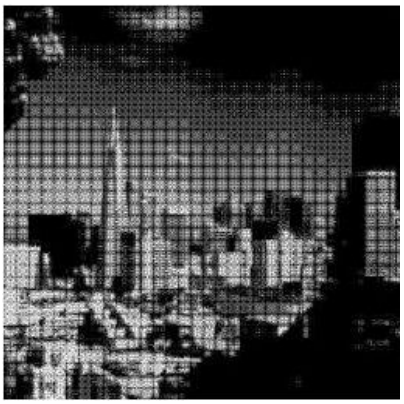
A motivating aspect of using a frequency domain representation of an image is that it is much more efficient to decorrelate an image in the frequency domain than in the spatial domain [197]. The algorithm for image compression using discrete fractional transforms is devised. The test images are compressed at compression percentages of 10%, 20%, 30%, 40%, 50% and 75% for optimized values of fractional order (a). The optimum value of 'a' (a_{opt}) is selected between 0 and 1 for particular compression percentage shown in Figure 3.6. Figure 3.7 shows results of test images. The optimum fractional order is selected with maximum peak signal to noise ratio for all images.

(a) Original Pyramid image.

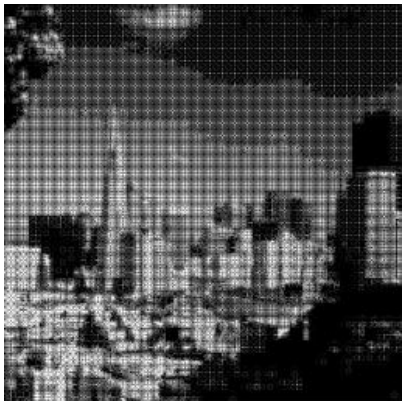
(b) Pyramid image at a=0.1

(c) Pyramid image at a=0 .2

(d) Pyramid image at a=0.3

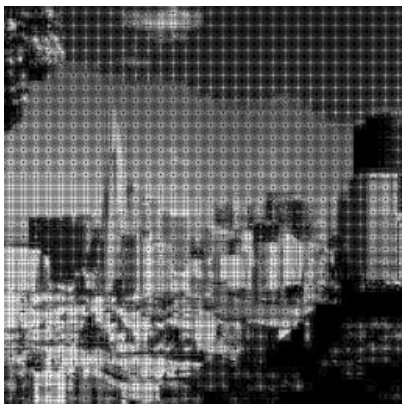


(e) Pyramid image at a=0.4



(f) Pyramid image at a=0.5

Figure-3.6: Simulation results of Pyramid image at different fractional orders with compression percentage 30% using DFrFT. (contd.)



(g) Pyramid image at a=0.6

(h) Pyramid image at a=0.7

(i) Pyramid image at a=0.8

(j) Pyramid image at a=0.9

(k) Pyramid image at a=0.91

(l) Pyramid image at a=0.92

Figure-3.6: Simulation results of Pyramid image at different fractional orders with compression percentage 30% using DFrFT. (contd.)

(m) Pyramid image at a=0.93

(n) Pyramid image at a=0.94

(o) Pyramid image at a=0.95

(p) Pyramid image at a=0.96

(q) Pyramid image at a=0.97

(r) Pyramid image at a=0.98

Figure-3.6: Simulation results of Pyramid image at different fractional orders with compression percentage 30% using DFrFT. (contd.)

(s) Pyramid image at a=0.99

(t) Pyramid image at a=1

Figure-3.6: Simulation results of Pyramid image at different fractional orders with compression percentage 30% using DFrFT.

3.2.3.1 Effect of fractional order

Figure 3.6 shows effect of fractional order at 30% compression percentage. The quality of image is less at a=0.1 and then starts improving as value of fractional order increases. From fractional order 0.9 to 1, quality has again improved and is finest at a=0.99. Figures 3.7(a)-3.7(f), show the Boat, Peppers and Girl images at optimized fractional order 'a' and at a=1. It has been observed from Table 3.2 and Figure 3.7 that PSNR is maximum at optimized fractional order. The MSE is inversely proportional to PSNR i.e as MSE increases for particular value of 'a', PSNR decreases at that value of 'a'.

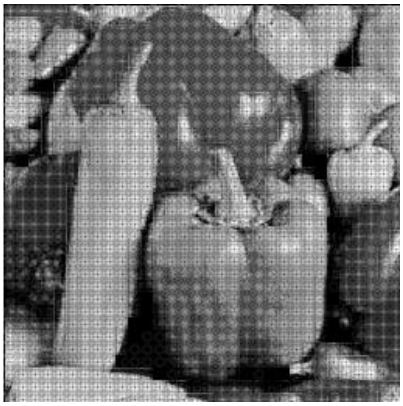
3.2.3.2 Effect of compression percentage

When fractional order is optimized, the image is compressed at various compression percentages. It has been observed that as the amount of compression percentages increases from 10% to 75% the quality of picture degrades. The value of 'a' optimum varies for different compression percentages and for different images as shown in Table 3.2. The Pyramid and Peppers compressed image results are shown in Figures 3.8 and 3.9 respectively.

(a) Boat image with optimized fractional order.

(b) Boat image with fractional order 'a'=1.

(c) Peppers image with optimized fractional order.



(d) Peppers image with fractional order 'a'=1.

(e) Girl image with optimized fractional order.

(f) Girl image with fractional order 'a'=1.

Figure-3.7: Simulation results of test images with optimized fractional order and with a=1.

(a) Compressed Pyramid image at 10%.

(b) Compressed Pyramid image at 20%.

(c) Compressed Pyramid image at 30%.

(d) Compressed Pyramid image at 40%.

(e) Compressed Pyramid image at 50%.

(f) Compressed Pyramid image at 75%.

Figure-3.8: Compressed Pyramid images at optimized fractional orders.

(a) Compressed Peppers image at 10%.

(b) Compressed Peppers image at 20%.

(c) Compressed Peppers image at 30%.

(d) Compressed Peppers image at 40%.

(e) Compressed Peppers image at 50%.

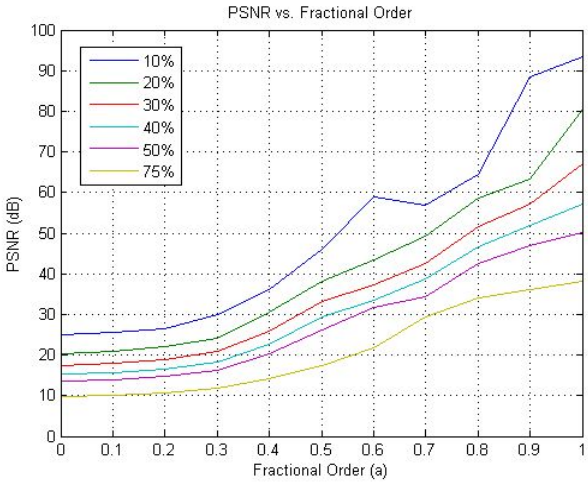
(f) Compressed Peppers image at 75%.

Figure-3.9: Compressed Peppers images at various compression percentages using DFrFT.

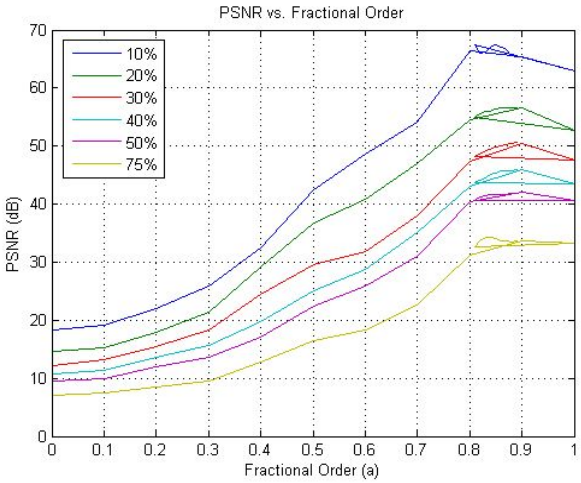
Table-3.2: MSE and PSNR at Optimized fractional orders for Test images using DFrFT.

Compression Percentage	Pyramid			Pentagon			Girl			Lena			Baboon		
	a_{opt}	MSE	PSNR (dB)	a_{opt}	MSE	PSNR (dB)	a_{opt}	MSE	PSNR (dB)	a_{opt}	MSE	PSNR (dB)	a_{opt}	MSE	PSNR (dB)
10	0.93	0.0000152	96.29	0.85	0.0116	67.47	0.85	0.0069	69.73	0.98	0.0036	72.52	0.83	0.0445	61.64
20	0.98	0.0003960	82.14	0.89	0.1397	56.67	0.89	0.0622	59.87	0.99	0.0950	58.35	0.83	0.4091	52.012
30	0.99	0.0151000	66.33	0.87	0.5646	50.61	0.87	0.2784	53.84	0.98	0.2639	53.91	0.86	1.4200	46.59
40	0.99	0.1391000	56.69	0.89	1.6875	45.85	0.89	0.8330	48.92	0.97	0.5402	50.80	0.86	4.100	41.99
50	0.99	0.5663000	50.6	0.89	4.2000	41.88	0.89	2.0800	45.98	0.97	1.2085	44.55	0.89	10.2900	38.02
75	0.99	9.2600000	38.46	0.89	24.3900	34.25	0.89	20.2900	35.93	0.99	9.7100	38.25	0.85	46.0700	31.49

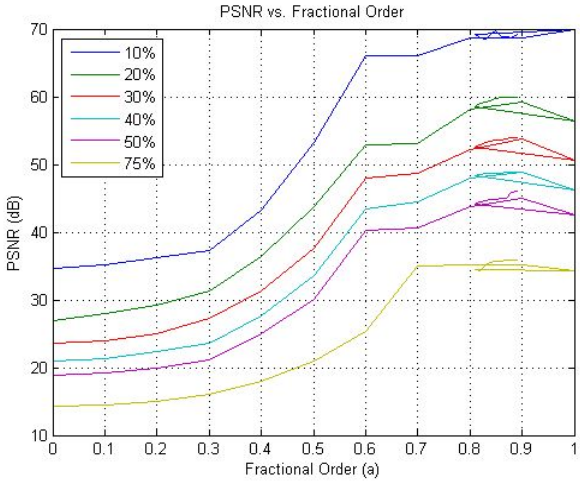
Compression Percentage	Boat			Flower			House			Barbara			Peppers		
	a_{opt}	MSE	PSNR (dB)	a_{opt}	MSE	PSNR (dB)	a_{opt}	MSE	PSNR (dB)	a_{opt}	MSE	PSNR (dB)	a_{opt}	MSE	PSNR (dB)
10	0.81	0.0138	66.71	0.81	0.0052	70.96	0.99	0.0074	69.45	0.99	0.0025	74.20	0.89	0.0256	64.04
20	0.85	0.1185	57.39	0.88	0.0861	58.76	0.99	0.0962	58.29	0.98	0.1001	58.12	0.89	0.1797	55.58
30	0.89	0.4961	51.17	0.89	0.3570	52.60	0.92	0.2422	54.28	0.97	0.3989	52.12	0.89	0.6822	49.79
40	0.88	1.4700	46.45	0.89	0.9681	48.27	0.97	0.5981	50.36	0.96	1.3018	46.98	0.88	1.7700	45.83
50	0.89	3.8600	42.25	0.89	2.4000	44.31	0.99	1.0952	47.73	0.96	3.6400	42.50	0.89	3.5200	42.28
75	0.84	21.4000	34.82	0.89	20.2600	35.06	0.99	6.0800	40.28	0.98	29.970	33.36	0.89	25.530	34.86



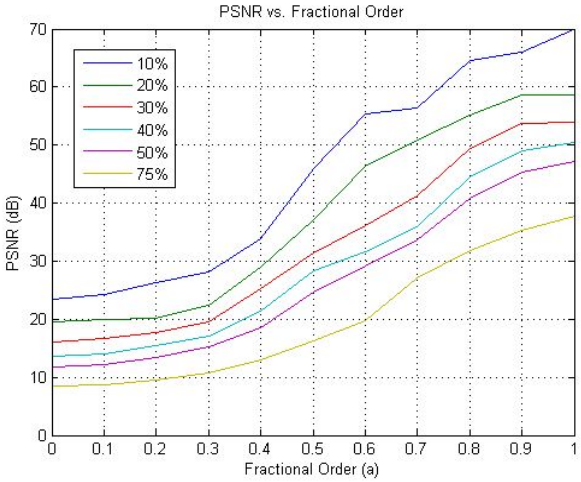
(a) Fractional order vs. PSNR for Pyramid Image.



(b) Fractional order vs. PSNR for Pentagon image.

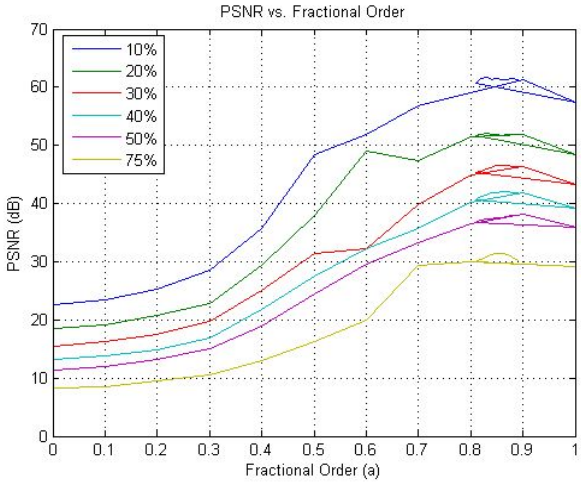


(c) Fractional order vs. PSNR for Girl image.

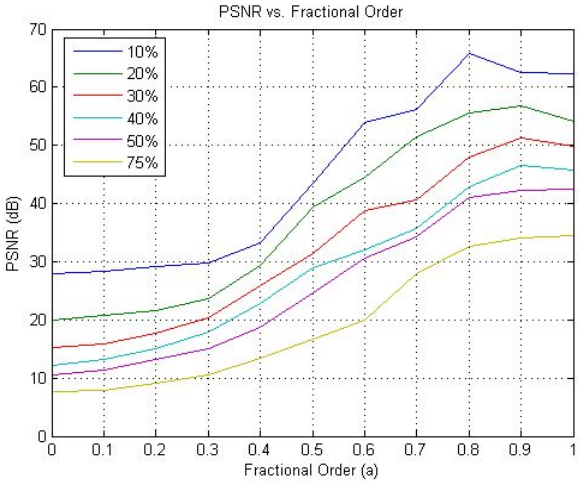


(d) Fractional order vs. PSNR for Lena image.

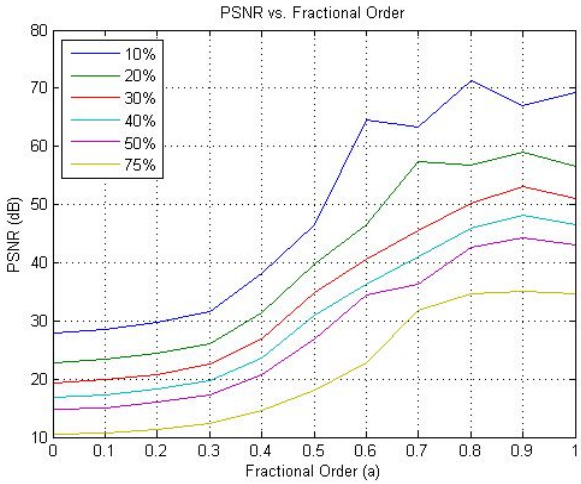
Figure-3.10: Fractional orders vs. PSNR at different compression percentages using DFrFT. (contd.)



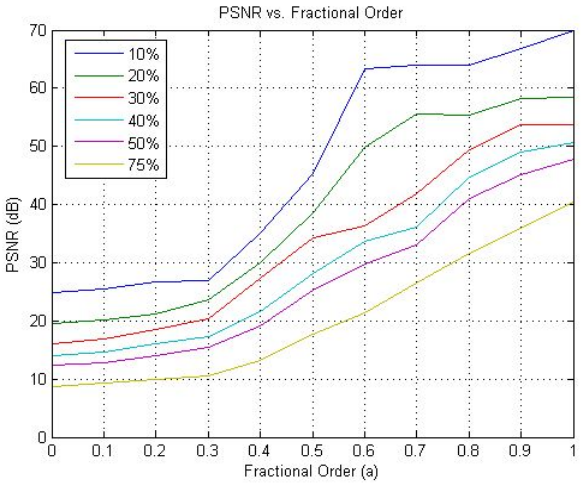
(e) Fractional order vs. PSNR for Baboon image.



(f) Fractional order vs. PSNR for Boat image.

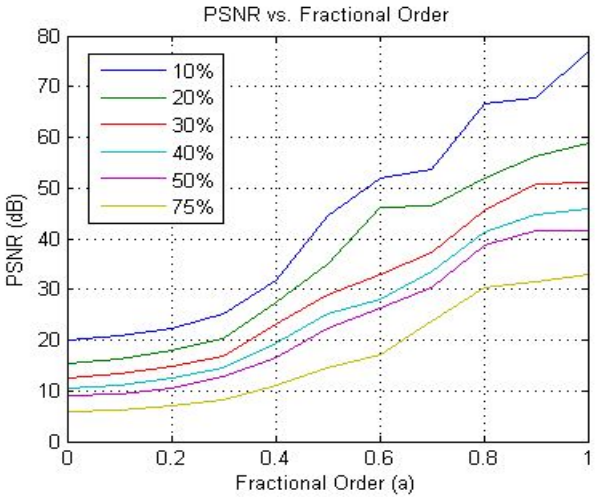


(g) Fractional order vs. PSNR for Flower image.

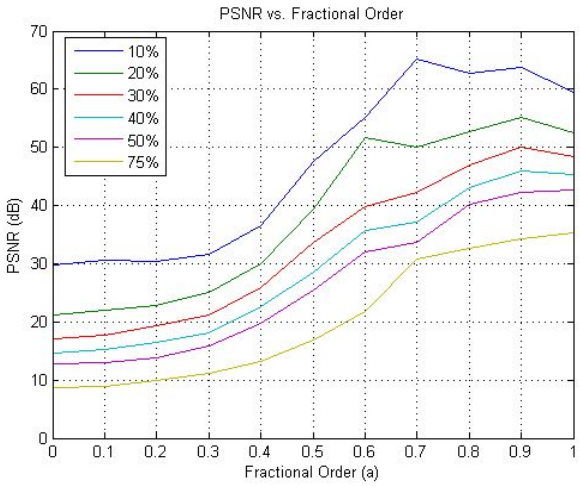


(h) Fractional order vs. PSNR for House image.

Figure-3.10: Fractional orders vs. PSNR at different compression percentages using DFrFT. (contd.)



(i) Fractional order vs. PSNR for Barbara image.



(j) Fractional order vs. PSNR for Peppers image.

Figure-3.10: Fractional orders vs. PSNR at different compression percentages using DFrFT.

3.2.4 Image Compression using Fractional Cosine Transforms

The compressed images using DFrCT are obtained and shown in Figure 3.11. The compressed image quality has decreased, as the compression percentage increases as shown in Figures 3.11(a) to 3.11(f). The PSNR of various images for 8 × 8 and 32 × 32 blocks is mentioned in Table 3.3. It has been observed that the PSNR is better for 32 × 32 block size. The MSE of images increases with the increase in compression percentage from 10% to 75% as mentioned in Table 3.4. The graphical results to optimize the fractional order at different compression percentages are shown in Figure 3.12. It has been observed PSNR is high at optimized fractional orders.

(a) Compressed Girl image at 10%.

(b) Compressed Girl image at 20%.

(c) Compressed Girl image at 30%.

(d) Compressed Girl image at 40%.

(e) Compressed Girl image at 50%.

(f) Compressed Girl image at 75%.

Figure-3.11: Compressed Girl images using DFrCT at various compression percentages using DFrCT.

Table-3.3: PSNR (dB) at Optimized fractional orders using DFrCT.

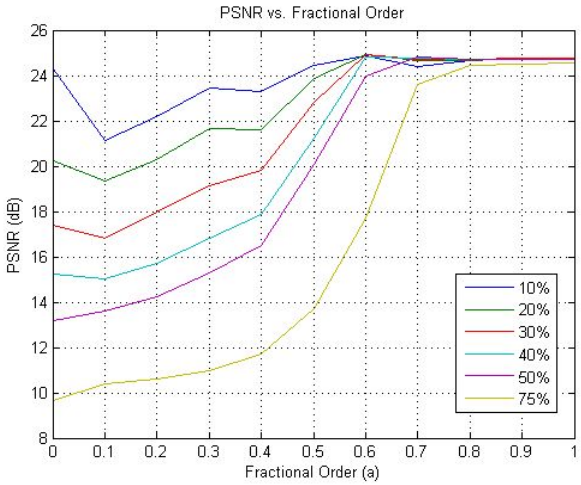
Compression Percentage	Pyramid PSNR (dB)		Pentagon PSNR (dB)		Girl PSNR (dB)		Lena PSNR (dB)		Baboon PSNR (dB)	
	8 × 8	32 × 32	8 × 8	32 × 32	8 × 8	32 × 32	8 × 8	32 × 32	8 × 8	32 × 32
10	18.66	24.74	17.35	23.25	21.41	27.06	17.66	23.69	18.08	24.28
20	18.65	24.74	17.25	23.25	21.40	27.05	17.64	23.69	18.08	24.28
30	18.64	24.73	17.24	23.24	21.44	27.04	17.63	23.68	18.08	24.27
40	18.62	24.73	17.27	23.23	21.42	27.02	17.62	23.68	18.07	24.27
50	18.64	24.72	17.23	23.23	21.43	26.96	17.61	23.67	18.05	24.26
75	18.61	24.56	17.16	22.95	21.28	26.46	17.60	23.66	17.78	23.42

Compression Percentage	Boat PSNR (dB)		Flower PSNR (dB)		House PSNR (dB)		Barbara PSNR (dB)		Peppers PSNR (dB)	
	8 × 8	32 × 32	8 × 8	32 × 32	8 × 8	32 × 32	8 × 8	32 × 32	8 × 8	32 × 32
10	17.40	23.55	18.76	24.89	18.18	24.05	15.47	21.51	17.65	23.68
20	17.46	23.54	18.75	24.88	18.18	24.05	15.45	21.51	17.65	23.68
30	17.44	23.54	18.76	24.88	18.11	24.04	15.44	21.50	17.62	23.67
40	17.44	23.53	18.77	24.87	17.95	24.04	15.43	21.50	17.61	23.67
50	17.44	23.52	18.76	24.87	17.95	24.03	15.42	21.49	17.64	23.66
75	17.36	23.52	18.71	24.85	17.91	23.87	15.40	21.48	17.58	23.65

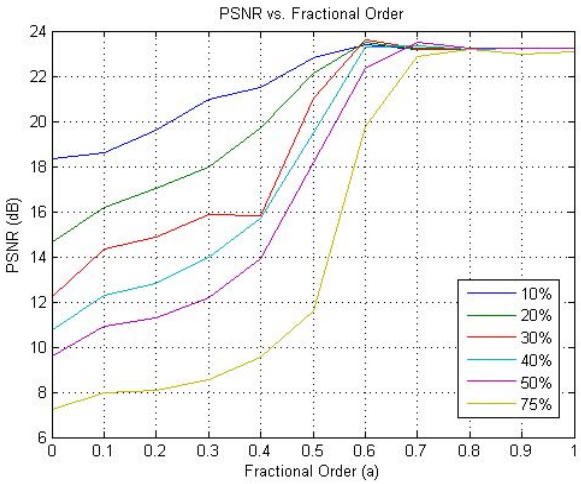
Table-3.4: MSE at Optimized fractional orders using DFrCT.

Compression Percentage	Pyramid MSE		Pentagon MSE		Girl MSE		Lena MSE		Baboon MSE	
	8 × 8	32 × 32	8 × 8	32 × 32	8 × 8	32 × 32	8 × 8	32 × 32	8 × 8	32 × 32
10	883.820	218.05	1130.3	307.15	469.44	127.80	1103.1	278.0	1012.2	242.19
20	883.930	218.04	1130.5	307.17	469.46	128.01	1103.5	280.0	1012.3	242.63
30	883.913	218.09	1130.7	307.10	466.28	128.31	1105.7	283.0	1012.3	243.75
40	884.077	218.10	1130.8	308.10	466.51	129.06	1105.8	278.0	1012.7	245.42
50	884.520	218.20	1130.9	308.50	467.14	130.85	1106.1	278.5	1013.1	251.40
75	895.410	219.10	1131.2	329.50	483.32	179.43	1109.5	283.5	1013.9	295.40

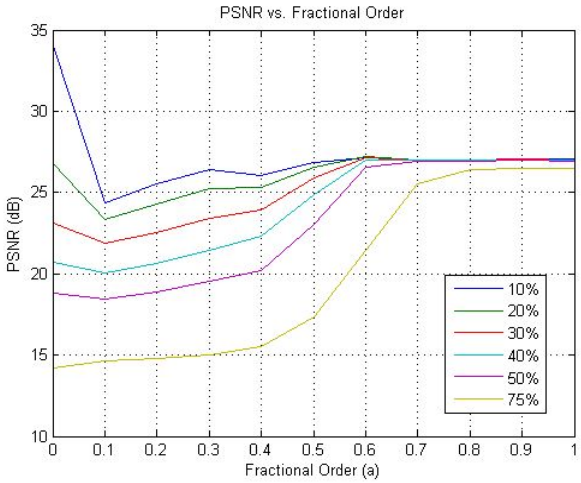
Compression Percentage	Boat MSE		Flower MSE		House MSE		Barbara MSE		Peppers MSE	
	8 × 8	32 × 32	8 × 8	32 × 32	8 × 8	32 × 32	8 × 8	32 × 32	8 × 8	32 × 32
10	1182.3	313.10	864.970	350.2	1051.2	265.40	1832.9	198.2	1121.3	319.4
20	1182.5	313.40	864.990	350.5	1051.4	265.41	1835.7	198.3	1121.5	319.5
30	1182.6	313.50	865.060	350.6	1051.5	265.45	1837.7	198.7	1121.6	319.8
40	1182.8	313.70	865.210	350.8	1051.7	265.46	1837.8	198.8	1121.7	319.9
50	1182.9	313.90	865.639	350.9	1051.8	265.50	1838.2	199.2	1121.9	320.1
75	1183.1	313.94	874.650	351.2	1052.1	267.10	1838.6	198.5	1122.2	320.4



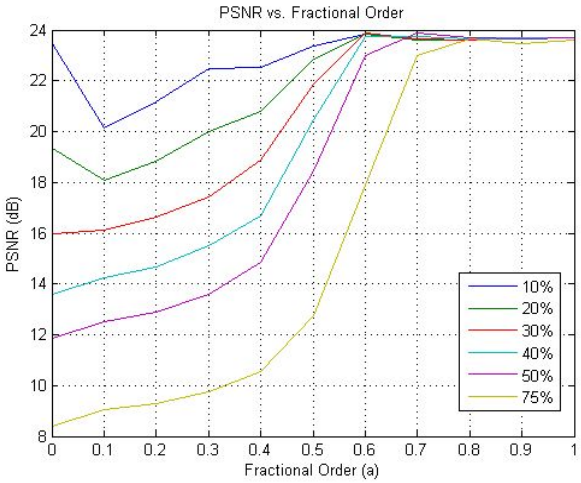
(a) Fractional order vs. PSNR for Pyramid Image.



(b) Fractional order vs. PSNR for Pentagon Image.

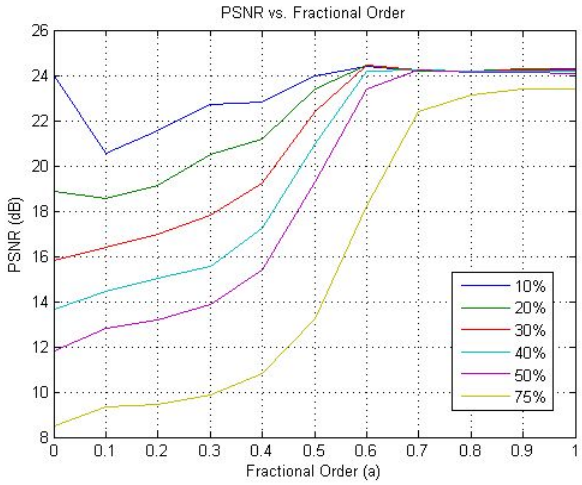


(c) Fractional order vs. PSNR for Girl Image.

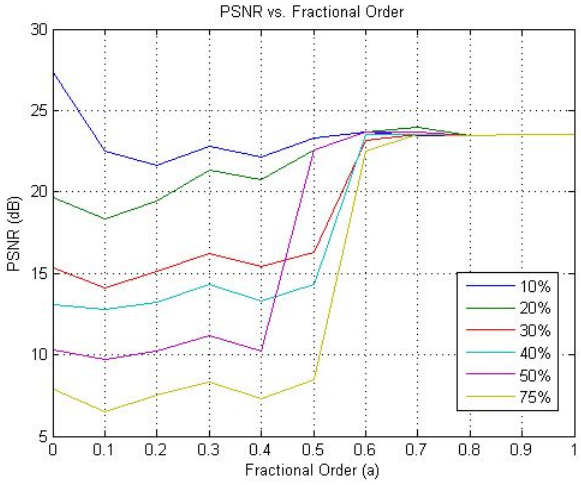


(d) Fractional order vs. PSNR for Lena Image.

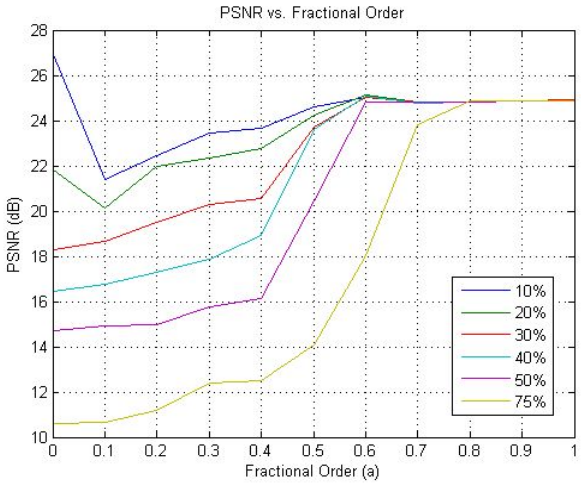
Figure-3.12: Fractional order vs. PSNR at different compression percentages using DFrCT 32 × 32. (contd.)



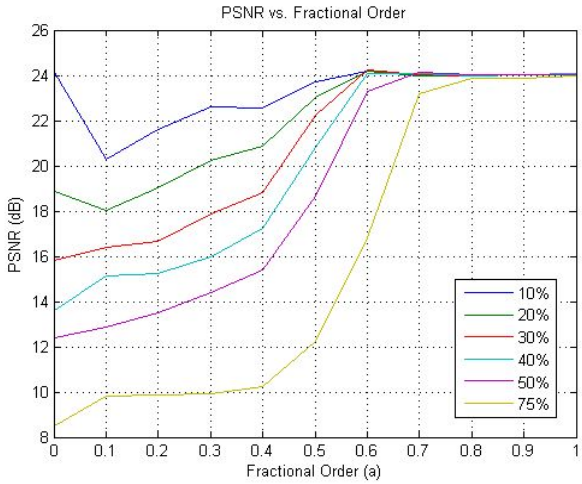
(e) Fractional order vs. PSNR for Baboon image.



(f) Fractional order vs. PSNR for Boat Image.

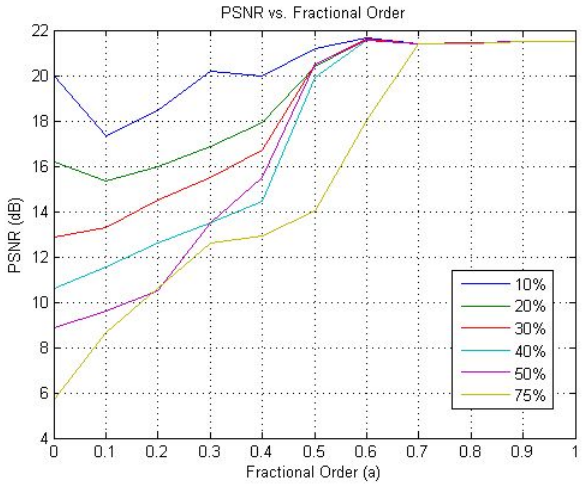


(g) Fractional order vs. PSNR for Flower Image.

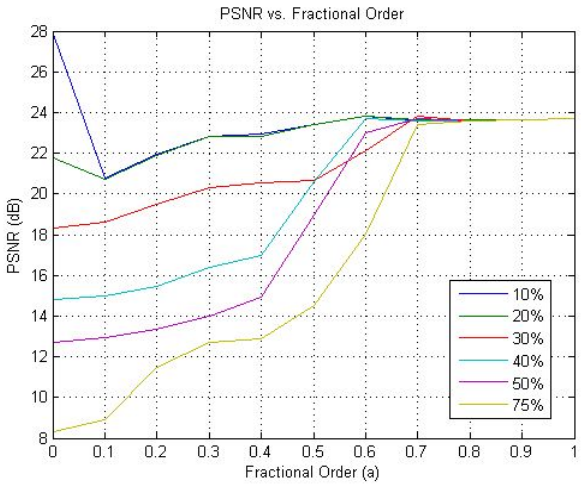


(h) Fractional order vs. PSNR for House Image.

Figure-3.12: Fractional order vs. PSNR at different compression percentages using DFrCT 32 × 32. (contd.)



(i) Fractional order vs. PSNR for Barbara Image.



(j) Fractional order vs. PSNR for Peppers Image.

Figure-3.12: Fractional order vs. PSNR at different compression percentages using DFrCT 32 × 32.

3.2.5 Image Compression using Fractional Hartley Transforms

The DFrHT transform, compress the images for optimized fractional order at required compression percentages as shown in Figure 3.13. The degradation in the compressed image quality from 10% to 75% compression percentage, is shown in Figures 3.13(a) to 3.13(f). The PSNR for 8 × 8 blocks and 4 × 4 blocks is mentioned in Table 3.5 gives. It is observed that PSNR is better for 4 × 4 blocks and decreased with the increase in compression percentage. The graphical results for optimization of fractional orders are shown in Figure 3.14. It has been observed that at optimized fractional orders, the PSNR is maximum.

(a) Compressed Boat image at 10%.

(b) Compressed Boat image at 20%.

(c) Compressed Boat image at 30%.

(d) Compressed Boat image at 40%.

(e) Compressed Boat image at 50%.

(f) Compressed Boat image at 75%.

Figure-3.13: Compressed Boat images using DFrHT at various compression percentages.

Table-3.5: PSNR at Optimized fractional orders for Test images using DFrHT.

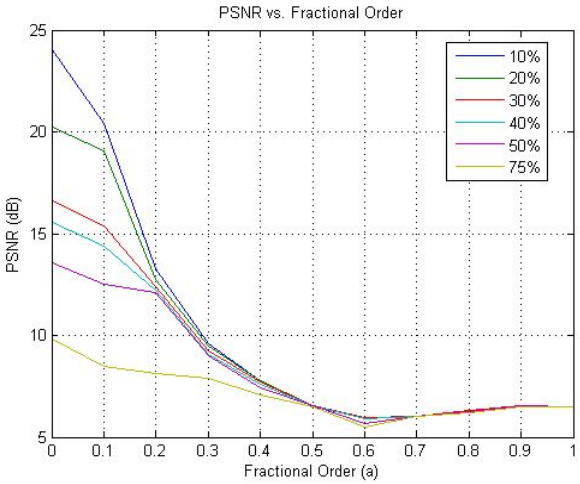
Compression Percentage	Pyramid PSNR (dB)		Pentagon PSNR (dB)		Girl PSNR (dB)		Lena PSNR (dB)		Baboon PSNR (dB)	
	8 × 8	4 × 4	8 × 8	4 × 4	8 × 8	4 × 4	8 × 8	4 × 4	8 × 8	4 × 4
10	4.050	6.532	4.68	5.17	7.05	9.47	3.12	5.62	3.50	5.98
20	4.050	6.531	4.67	5.17	7.04	9.47	3.11	5.61	3.48	5.98
30	4.045	6.530	4.67	5.16	7.04	9.46	3.11	5.61	3.47	5.97
40	4.041	6.530	4.66	5.15	7.03	9.45	3.10	5.60	3.46	5.96
50	4.035	6.529	4.63	5.15	7.02	9.44	3.09	5.59	3.45	5.95
75	4.010	6.520	4.55	5.10	7.01	9.41	3.01	5.55	3.40	5.91

Compression Percentage	Boat PSNR (dB)		Flower PSNR (dB)		House PSNR (dB)		Barbara PSNR (dB)		Peppers PSNR (dB)	
	8 × 8	4 × 4	8 × 8	4 × 4	8 × 8	4 × 4	8 × 8	4 × 4	8 × 8	4 × 4
10	2.87	5.33	4.21	6.67	3.28	5.79	0.945	3.44	3.21	5.67
20	2.87	5.32	4.21	6.66	3.28	5.78	0.941	3.44	3.21	5.67
30	2.86	5.32	4.20	6.66	3.27	5.78	0.930	3.43	3.20	5.66
40	2.85	5.31	4.19	6.65	3.26	5.76	0.930	3.42	3.19	5.65
50	2.83	5.30	4.18	6.64	3.23	5.75	0.920	3.41	3.18	5.64
75	2.80	5.29	4.10	6.60	3.20	5.70	0.900	3.40	3.10	5.62

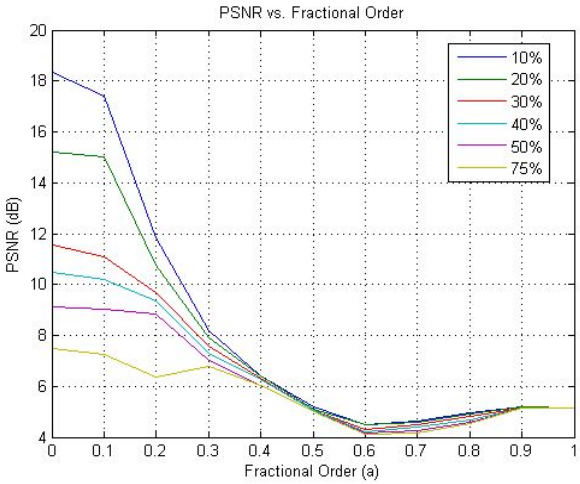
Table-3.6: MSE at Optimized fractional orders for Test images using DFrHT.

Compression Percentage	Pyramid MSE		Pentagon MSE		Girl MSE		Lena MSE		Baboon MSE	
	8 × 8	4 × 4	8 × 8	4 × 4	8 × 8	4 × 4	8 × 8	4 × 4	8 × 8	4 × 4
10	25444.6	16191.1	35111.4	22103.5	1283.4	8269.6	31651.2	19906.6	28999.6	18473.2
20	25445.0	16191.3	35111.7	22103.8	1283.5	8269.8	31653.2	19907.6	28999.7	18473.4
30	25446.0	16192.1	35111.9	22103.9	1283.9	8270.4	31654.2	19907.8	28999.83	18473.5
40	25447.0	16192.6	35112.2	22104.2	1284.3	8270.5	31657.2	19908.2	28999.86	18473.7
50	25447.0	16195.1	35118.4	22103.7	1284.7	8272.6	31657.4	19908.5	29000.5	18473.9
75	25450.0	16195.5	35119.5	22106.1	1286.2	8272.8	31657.7	19909.3	29005.3	18475.1

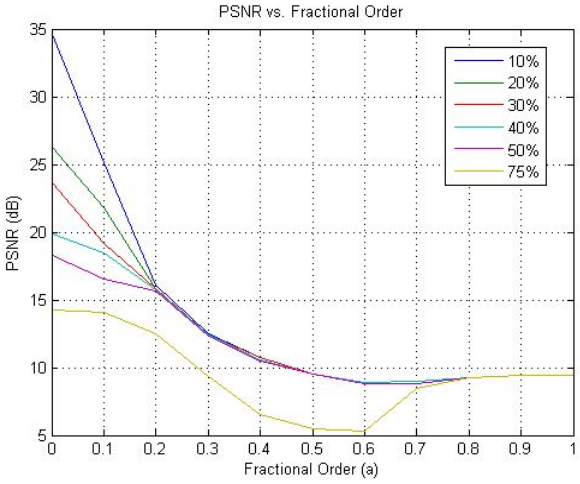
Compression Percentage	Boat MSE		Flower MSE		House MSE		Barbara MSE		Peppers MSE	
	8 × 8	4 × 4	8 × 8	4 × 4	8 × 8	4 × 4	8 × 8	4 × 4	8 × 8	4 × 4
10	33518.1	21324.3	24640.1	15658.3	30501.2	19208.4	52326.6	32887.4	31039.5	19720.6
20	33518.4	21324.5	24640.6	15658.6	30501.6	19208.7	52326.8	32887.4	31039.8	19720.7
30	33518.7	21324.8	24640.8	15658.7	30501.8	19208.9	52326.9	32887.4	31039.9	19720.8
40	33519.3	21324.9	24642.3	15658.9	30501.9	19209.4	52327.2	32887.4	31040.2	19722.3
50	33519.4	21325.3	24642.4	15659.3	30502.2	19209.5	52327.6	32887.4	31040.5	19722.6
75	33519.9	21325.6	24643.3	15659.4	30502.5	19209.7	52328.3	32887.4	31040.6	19723.7



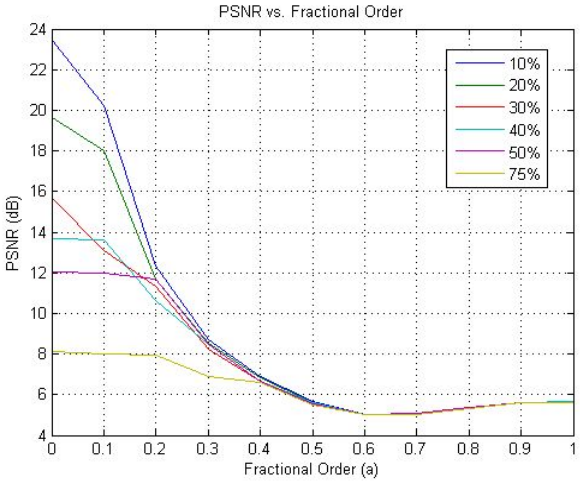
(a) Fractional order vs. PSNR for Pyramid Image.



(b) Fractional order vs. PSNR for pentagon Image.

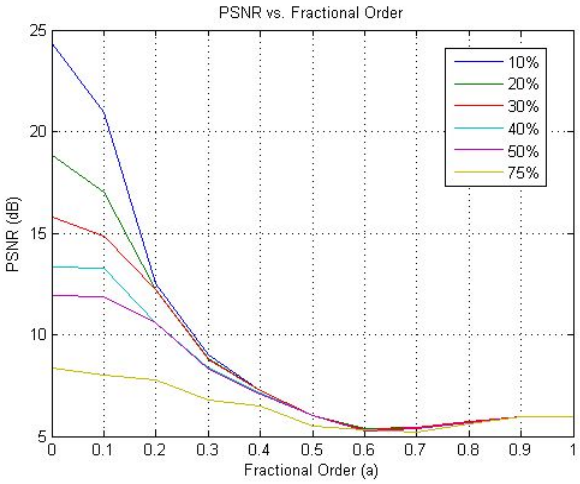


(c) Fractional order vs. PSNR for Girl Image.

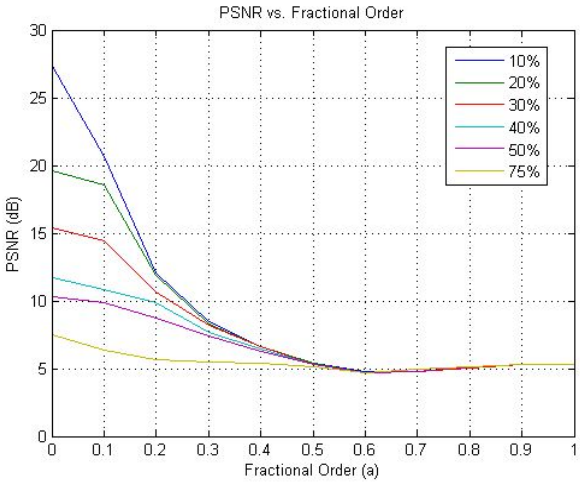


(d) Fractional order vs. PSNR for Lena Image.

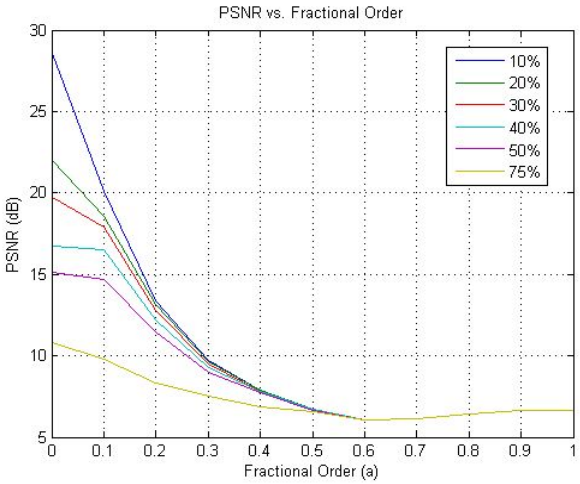
Figure-3.14: Fractional orders vs. PSNR at different compression percentages using DFrHT. (contd.)



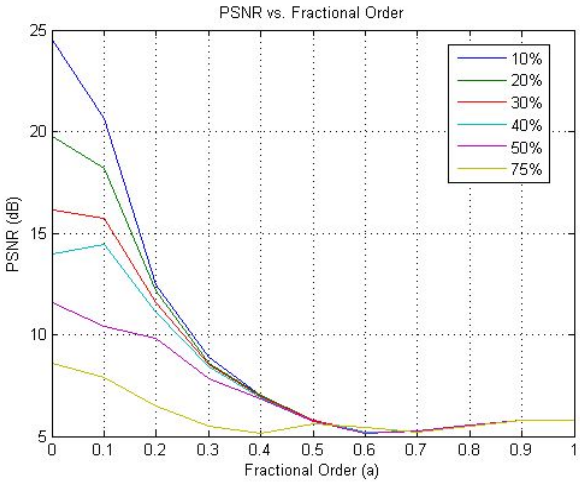
(e) Fractional order vs. PSNR for Baboon Image.



(f) Fractional order vs. PSNR for Boat Image.

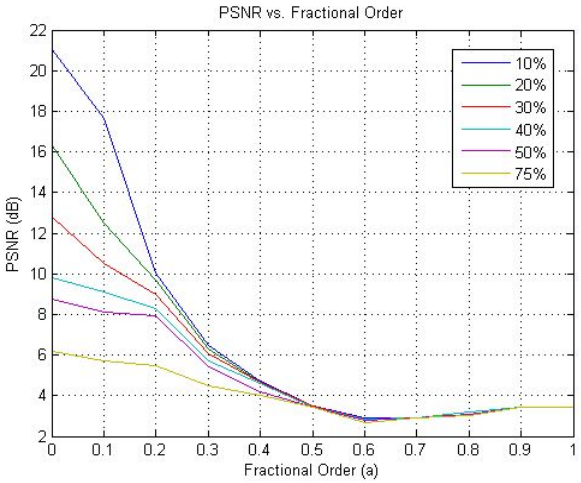


(g) Fractional order vs. PSNR for Flower Image.

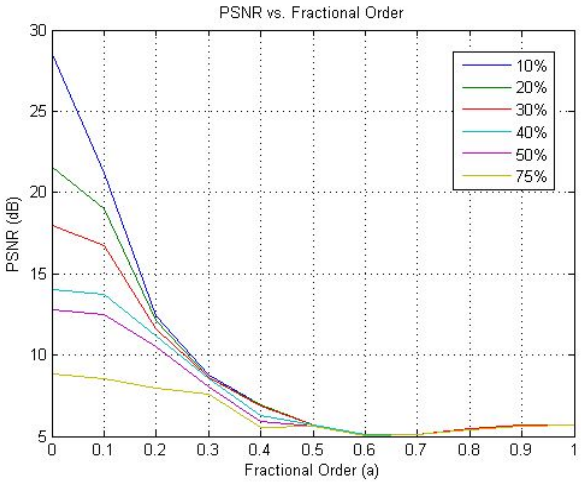


(h) Fractional order vs. PSNR for House Image.

Figure-3.14: Fractional orders vs. PSNR at different compression percentages using DFrHT. (contd.)



(i) Fractional order vs. PSNR for Barbara Image.



(j) Fractional order vs. PSNR for Peppers Image.

Figure-3.14: Fractional orders vs. PSNR at different compression percentages using DFrHT.

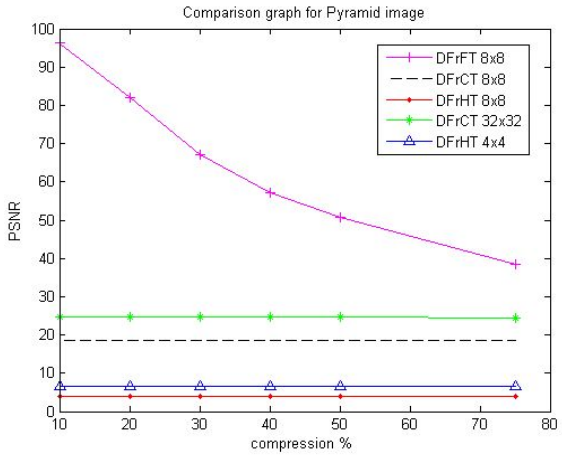
3.2.6 Comparison of Image Compression Algorithms

Image compression algorithms DFrFT, DFrCT and DFrHT are compared for their performance analysis. The PSNR for the test images at various compression percentages is given in Table 3.5 and shown in Figure 3.15. The PSNR for DFrFT is better than DFrCT and DFrHT. The reason is that in time-frequency representations, generally a plane has two orthogonal axes corresponding to time and frequency respectively. If a signal is represented along time axis its Fourier transform is represented along frequency axis. The Fourier transform operator is viewed as a change in the representation of the signal corresponding to a counterclockwise axis rotation of $\pi/2$ rad. In context of DFrFT, for any vector at 45°, it performs same for time and frequency plane equally. So, image compression with DFrFT performs better in frequency domain and tries to save bandwidth by varying fractional order 0 to 1.

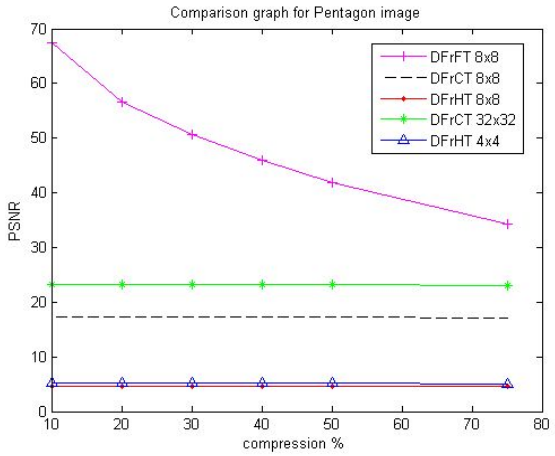
Table-3.7: PSNR at different compression percentages for Test images using DFrFT, DFrCT and DFrHT (8 × 8).

Compression Percentage	Pyramid			Pentagon			Lena			Girl			Baboon		
	PSNR (dB) DFrFT	PSNR (dB) DFrCT	PSNR (dB) DFrHT	PSNR (dB) DFrFT	PSNR (dB) DFrCT	PSNR (dB) DFrHT	PSNR (dB) DFrFT	PSNR (dB) DFrCT	PSNR (dB) DFrHT	PSNR (dB) DFrFT	PSNR (dB) DFrCT	PSNR (dB) DFrHT	PSNR (dB) DFrFT	PSNR (dB) DFrCT	PSNR (dB) DFrHT
10	96.29	18.66	4.05	67.47	17.35	4.68	72.52	17.66	3.12	69.73	21.41	7.05	61.64	18.08	3.50
20	82.14	18.65	4.05	56.67	17.25	4.67	58.35	17.64	3.11	59.87	21.40	7.04	52.012	18.08	3.48
30	66.33	18.64	4.045	50.61	17.24	4.67	53.91	17.63	3.11	53.84	21.44	7.04	46.59	18.08	3.47
40	56.69	18.62	4.041	45.85	17.27	4.66	50.80	17.62	3.10	48.92	21.42	7.03	41.99	18.07	3.46
50	50.6	18.64	4.035	41.88	17.23	4.63	47.30	17.61	3.09	45.98	21.43	7.02	38.02	18.05	3.45
75	38.46	18.61	4.01	34.25	17.16	4.55	38.25	17.60	3.01	35.93	21.28	7.01	31.49	17.78	3.40

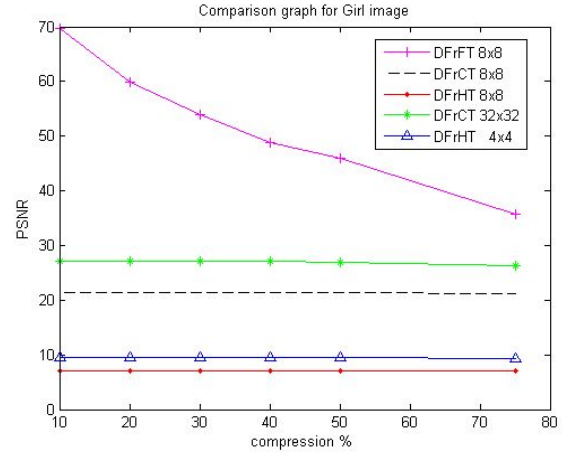
Compression Percentage	Boat			Flower			House			Barbara			Peppers		
	PSNR (dB) DFrFT	PSNR (dB) DFrCT	PSNR (dB) DFrHT	PSNR (dB) DFrFT	PSNR (dB) DFrCT	PSNR (dB) DFrHT	PSNR (dB) DFrFT	PSNR (dB) DFrCT	PSNR (dB) DFrHT	PSNR (dB) DFrFT	PSNR (dB) DFrCT	PSNR (dB) DFrHT	PSNR (dB) DFrFT	PSNR (dB) DFrCT	PSNR (dB) DFrHT
10	66.71	17.40	2.87	70.96	18.76	4.21	69.45	18.18	3.28	74.20	15.47	0.945	64.04	17.65	3.21
20	57.39	17.46	2.87	58.76	18.75	4.21	58.29	18.18	3.28	58.12	15.45	0.941	55.58	17.65	3.21
30	51.17	17.44	2.86	52.6	18.76	4.20	54.28	18.11	3.27	52.12	15.44	0.93	49.79	17.62	3.20
40	46.45	17.44	2.85	48.27	18.77	4.19	50.36	17.95	3.26	46.98	15.43	0.93	45.83	17.61	3.19
50	42.25	17.44	2.83	44.31	18.76	4.18	47.73	17.95	3.23	42.50	15.42	0.92	42.65	17.64	3.18
75	34.82	17.36	2.80	35.06	18.71	4.10	40.28	17.91	3.20	33.36	15.40	0.90	34.86	17.58	3.10



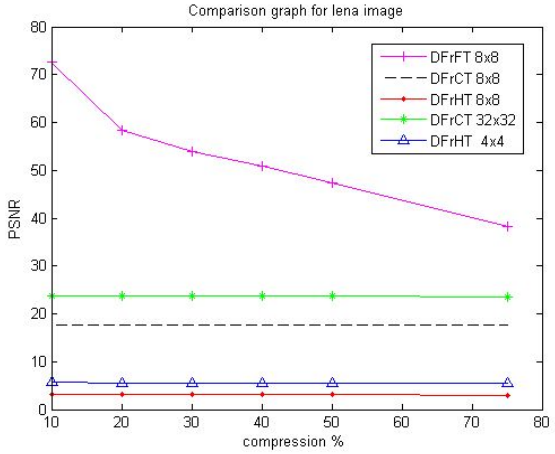
(a) Comparison graph of discrete fractional transforms for Pyramid image.



(b) Comparison graph of discrete fractional transforms for Pentagon image.

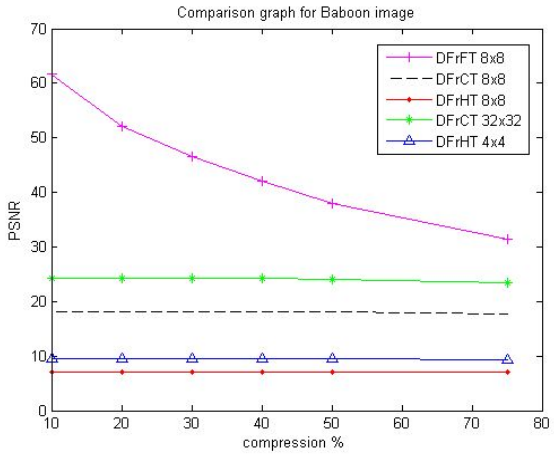


(c) Comparison graph of discrete fractional transforms for Lena image.

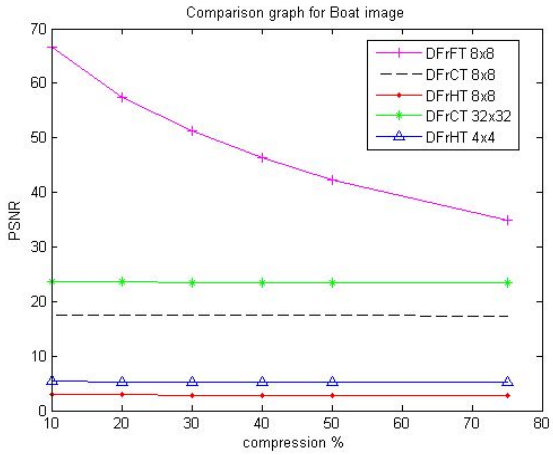


(d) Comparison graph of discrete fractional transforms for Girl image.

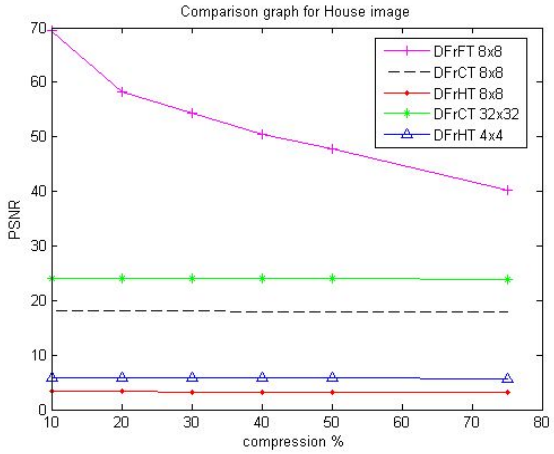
Figure-3.15: Compression percentages vs. PSNR of different images. (contd.)



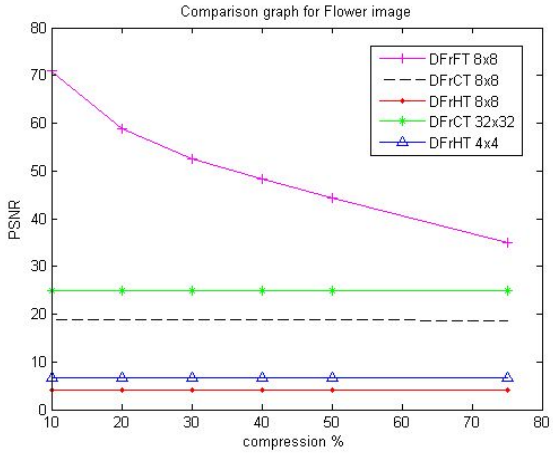
(e) Comparison graph of discrete fractional transforms for Baboon image.



(f) Comparison graph of discrete fractional transforms for Boat image.

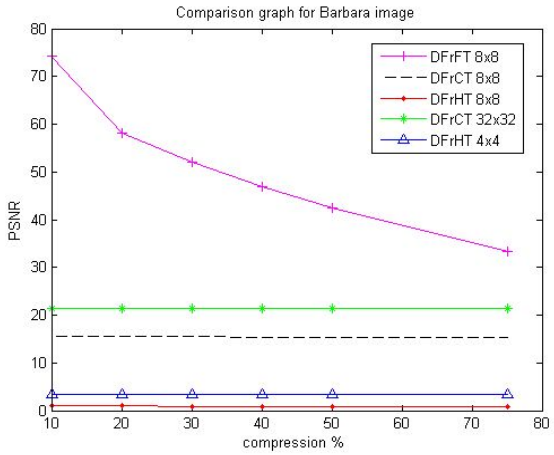


(g) Comparison graph of discrete transforms for Flower image.

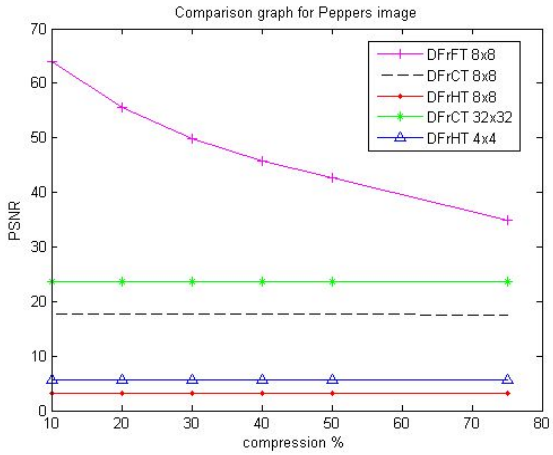


(h) Comparison graph of discrete fractional transforms for House image.

Figure-3.15: Compression percentages vs. PSNR of different images. (contd.)



(i) Comparison graph of discrete fractional transforms for Barbara image.



(j) Comparison graph of discrete fractional transforms for Peppers image.

Figure-3.15: Compression percentages vs. PSNR of different images.

It has been observed from comparison graphs that DFrFT has better performance than DFrCT and DFrHT at different compression percentages. However, in all transforms, as the compression percentage increases, the corresponding PSNR decreases.

The comparison of image compression using discrete fractional transforms and JPEG [200] for Lena and Peppers images is shown in Table 3.8. It was observed that DFrFT shows better performance in image quality at the cost of encoding and decoding time of CPU.

Table-3.8: Comparison of Fractional Transforms and JPEG.

Algorithm	**Image size**		**PSNR (dB)**	**CPU time**	
				Encoding time(s)	**Decoding time (s)**
DFrFT	Lena	(256 × 256)	44.55	5.332	5.330
	Peppers	(256 × 256)	42.28	3.483	3.480
DFrCT	Lena	(256 × 256)	17.66	2.700	2.695
	Peppers	(256 × 256)	17.65	2.652	2.651
JPEG [200]	Lena	(256 × 256)	34.66	0.1200	0.120
	Peppers	(256 × 256)	34.27	0.2000	0.200

3.3 BLOCKING ARTIFACTS

Transform-based data compression is one of the most popular choices in both image and video compression applications. Due to the availability of fast algorithms the block-based transforms are used in most of current image and video compression standards. However, at low bit rates, a major problem associated with the block-based transforms compression is that the decoded images manifest visually objectionable artifacts. One of the well-known artifacts in low-bit-rate transform-coded images is the blocking effect, which is noticeable in the form of undesired visible block boundaries [189]. Block edge artifacts is one of the perceptible types of impairment connected with image compression. Blocking artifacts are visible due to not taking the inter-block correlation into account during encoding. The coarse quantization of transform coefficients at low bit rates and the independent quantization for each block is also the cause for blocking artifact. At higher compression percentages, since very few coefficients are encoded, the blocking artifacts are more visible.

These blocking artifacts cause three types of noise: grid noise in monotone area, staircase noise near the edges and corner outliers at the cross point of block. Many post processing algorithms have been developed for the reduction of blocking artifacts, such as spatial averaging methods, DCT filtering, regularized image reconstruction techniques and wavelet filtering. Most of times, these algorithms are either computationally complex, include multiple iterations, are not adaptive enough to remove different levels of blockiness severity, or result in excessive smoothing of the image textures.

3.3.1 Detection of Blocking Artifacts

The reconstructed images from compression produce noticeable image degradation near the block boundaries, in particular for highly compressed images, because each block is transformed independently. The original image is divided into blocks due to spatially varying statistics within an image and large memory is required. The MSE is observed high at boundary pixels (8, 16), than those of neighboring pixels. So detection of blocking artifacts for complete image is done in horizontal and vertical direction.

Step 1: The MSE of a block (A) for each N^{th} and $N+1^{th}$ column in horizontal direction and each M^{th} and $M+1^{th}$ row in vertical direction is with the equation given by

$$(MSEH + MSEV)_{A1} = \left[\frac{1}{MN}\sum_{i=8}^{9}\sum_{j=8}^{9}[r(i,j) - o(i,j)]^2\right]^{1/2} \quad (3.33)$$

Here N=8 and M=8, i.e. block boundary column and row. The MSE of all blocks of complete image is calculated. The MSE of overlapping pixels available at corners is subtracted and hence blocked MSE is calculated.

Step 2: The MSE of non-blocked pixels is also measured in horizontal and vertical direction and called non-blocked MSE with equation given as

$$(MSEH + MSEV)A2 = \left[\frac{1}{MN}\sum_{i=0}^{7}\sum_{j=0}^{7}[r(i,j) - o(i,j)]^2\right]^{1/2} \quad (3.34)$$

Step 3: The total MSE of an image is sum of blocked and non-blocked MSE. The simulation results are calculated based on the parameters MSE, PSNR and bit rate.

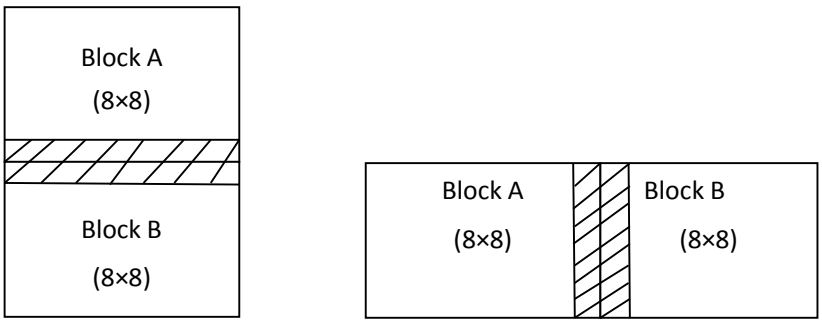


Figure-3.16: Vertical and horizontal block boundaries.

Table-3.9: MSE of blocked and non-blocked per pixels for different Test images.

Title		Blocked MSE per pixel	Non-Blocked MSE per pixel
Pyramid	(256×256)	0.08112	0.0144
Girl	(256×256)	0.041	0.0075
Baboon	(256×256)	0.091	0.017
Peppers	(256×256)	0.097	0.017

The results

in terms of bits per pixel (bpp) are directly proportional to PSNR. If we increase the bpp, then PSNR also increases [104,108].

Table-3.10: PSNR (dB) at various fractional orders for Pyramid image.

bpp	Fractional order (α=0.91)		Fractional order (α=0.95)		Fractional order (α=1)	
	DFrFT	DFrCT	DFrFT	DFrCT	DFrFT	DFrCT
0.1	19.46	13.86	28.33	17.14	21.71	16.03
0.2	20.99	15.94	28.82	17.47	23.96	17.13
0.3	25.41	16.59	30.78	17.64	25.85	17.43
0.4	29.42	17.18	31.80	17.94	27.12	17.78

Blocking artifacts are visually striking and occur because of the loss of high-frequency components that are quantized or eliminated. The blocking artifacts are more visible at lower bpp (higher compression) in the image compression using DFrFT and DFrCT. The comparison of DFrFT and DFrCT results that blocking is more in DFrCT.

3.4 IMAGE COMPRESSION AND SCRAMBLING

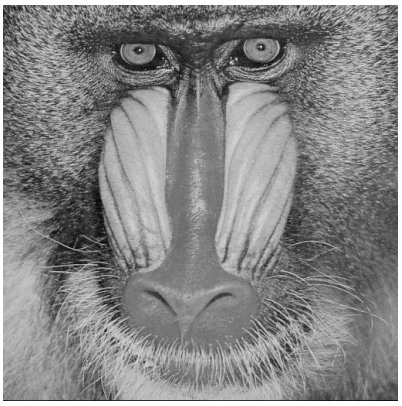
The image scrambling and compression algorithm [201], in which digital image data are efficiently scrambled in the frequency domain without affecting the compression efficiency is presented. The algorithm has the nice feature that the scrambling process is very simple and efficient. It provides security, has very limited adverse impact on the compression efficiency and no adverse impact on the error resiliency. The present method has the advantage that the proposed algorithm can be used in combination with compression, so as to make those schemes more secure with automatic security provided by fractional orders. The other advantage is that the user can be allowed to scramble the image a number of times. This means that a potential attacker will have great difficulties in recovering the original information. This is because even in the case when an attacker knows the algorithm, he/she still will not know the parameters of the algorithms and thus has no way to recover the image.

3.4.1 Scrambling

Digital image scrambling can be used as an effective tool in hiding information in an image, as well as in digital image encryption. Its main purpose is to make a given image scrambled and unrecognizable so that no one is able to find out the true form or meaning of the image or the bidden image. Image scrambling as an encryption technology has become an important method of the digital image transmission and the confidentiality storage. The scrambling used the characteristics that the digital image has the number arrays, to confuse the location of image pixels or color to make it into a medley of images, cannot be identified to achieve the purpose of the original image.

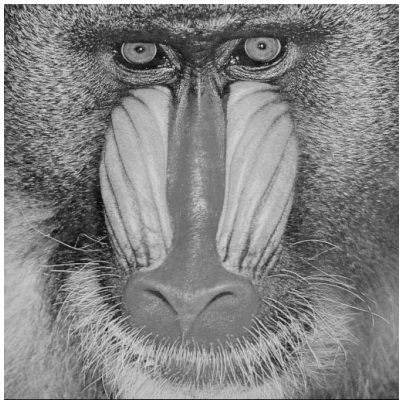
For digital image, scrambling is the interchange of positions of different pixels. One cannot acquire any information about the original image from the scrambled image, but the original image can be retrieved by repositioning the disturbed image according to unique order. The digital image matrix P consists of rows and columns assumed as queues. In simulation, the original image is divided into subsections of $n \times n$ pixels (n = 4, 8, 16…) in column queue, which are repositioned relative to one another and scrambled image is achieved. After one column, element of every column comes from different column in queue scrambling transform [137].

The principal of scrambling is depicted in Figure 3.17 and 3.18. For simplicity the baboon image 256 × 256 is considered as shown in Figure 3.17 (a). The divided 1024 subsections of 8 × 8 pixels are repositioned relative to one another according to reference point (2, 3) and column queue transform [138]. Scrambled image, after random shuffling [121] in pixels is achieved in Figure 3.17(b). Perceptibly, descrambled or original image is the final result which is given in Figure 3.17 (c) by applying all divergent operations. Figure 3.19 shows the scrambled and descrambled Flower image.



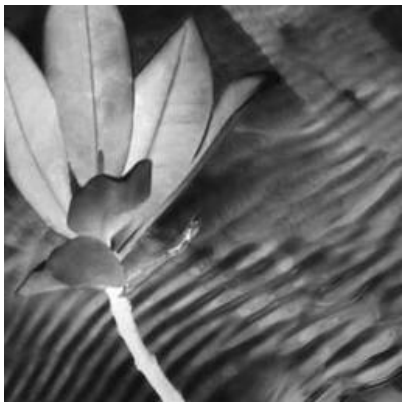
(a) Original Baboon Test Image.

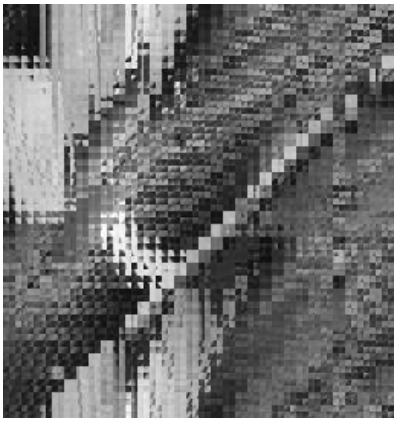
(b) Scrambled Image.



(c) Descrambled Image.

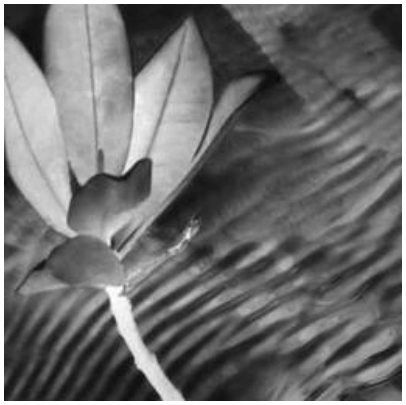
Figure-3.17: Scrambling of Baboon image.





(a) Scrambled Image. **(b) Descrambled Image.**

Figure-3.18: Scrambling of Flower image. (contd.)



(c) Descrambled Image.

Figure-3.18: Scrambling of Flower image.

3.4.2 Algorithm

Image compression algorithm based on DFrFT or DFrCT and scrambling is as follow:

Step 1: The original image is one-dimensional DFrFT or DFrCT transformed, and got compressed image.

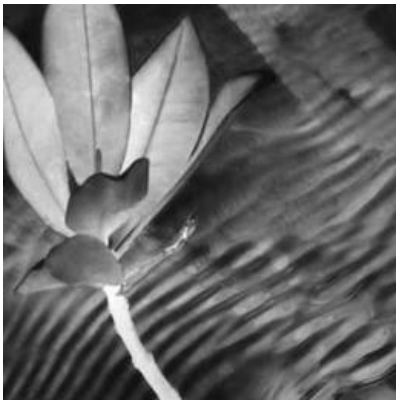
Step 2: Then compressed image has been queue scrambled or column wise scrambled and got the compressed-scrambled image.

Image is also automatically encrypted due to fractional order of discrete fractional Fourier transforms.

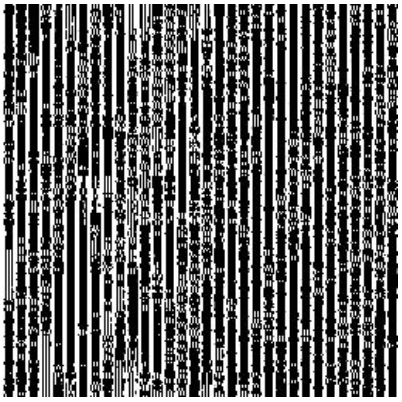
Image extraction algorithm steps are as follows:

Step 1: The compressed-scrambled image is descrambled.

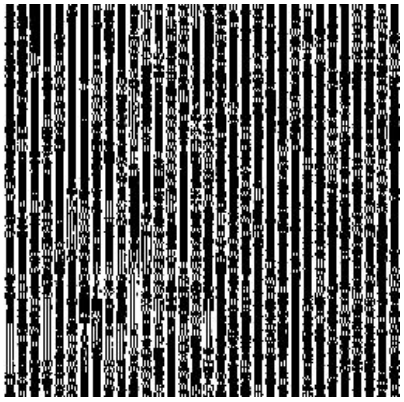
Step 2: Then DFrFT or DFrCT transform with inverse fractional order is devised, and decompressed image is resumed. The results of algorithm are shown in Figure 3.19.



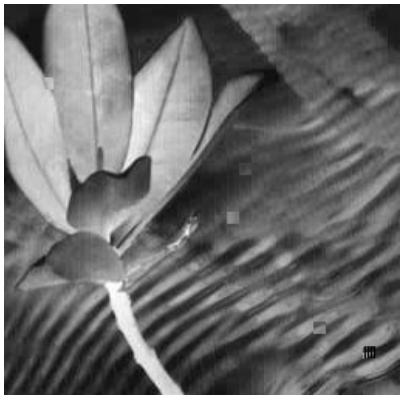
(a) Original Flower test image.



(b) Compressed and scrambled image.



(c) Descrambled image.



(d) Decompressed image.

Figure-3.19: Simulation results of compression-scrambling algorithm.

3.4.3 Scrambling Degree

The evaluation of scrambling degree has theoretical and practical importance in the information hiding field. Scrambling degree (SD) is a standard to measure the effects of scrambling. The definition of the scrambling degree is defined [137,138] with equation

$$SD(M, M') \cong \frac{\sum_{i\cong 1}^{m} \sum_{j\cong 1}^{n} (m_{ij} \; 0 \; m_{ij}')^2}{\sum_{i\cong 1}^{m} \sum_{j\cong 1}^{n} (m_{ij} \; 0 \; e_{ij})^2} \tag{3.35}$$

where, $M \cong (m_{ij})$ is original image, $M' \cong (m_{ij}')$ is scrambled image and $E=(e_{ij})$ is the uniform noise for (i, j) pixel. The comparison of scrambling degree between the proposed algorithm and Zhang method [137] is shown in Table 3.11. The improved value of scrambling degree confirms the performance of presented algorithm. The scrambling degree of some test images is given in Table 3.12.

Table-3.11: Comparison of Scrambling degree (SD).

Image	SD of Proposed algorithm	SD of Zhang algorithm [137]	Improvement
Lena	1.99	1.15	0.8
Cute Baboon	2.004	1.08	0.92

Table 3.12: Scrambling degree (SD) of test images.

Image	SD
Flower	2.003
Pyramid	2.006
Girl	2.007
Boat	2.001
House	2.004

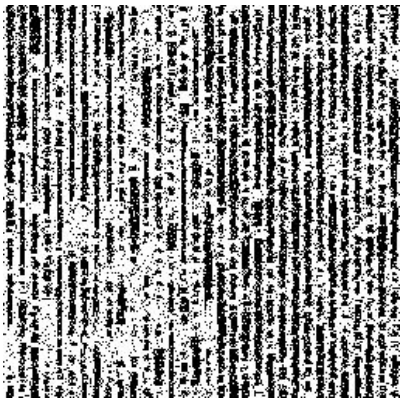
3.4.4 Noise Attacks

There are many aspects of security analysis like assessing a system against common noise attacks; the work is focused on effect of salt and pepper noise and Gaussian noise.

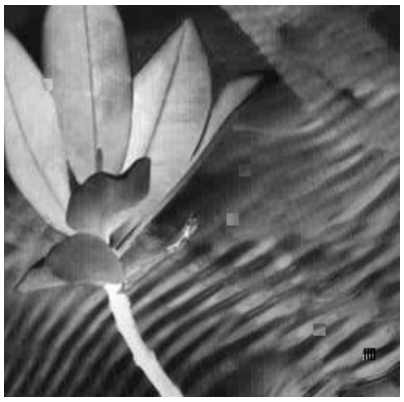
The salt-and-pepper noise [59] also called impulse noise, shot noise or spike noise is caused by malfunctioning pixel elements in the camera sensors, faulty memory locations, or timing errors in the digitization process. In the salt-pepper noise only two values are possible, x and y, the probability of obtaining each of them is less than 0.1 otherwise the noise would vastly dominate the image. The 8bits/pixel image has intensity value close to 255 for salt and nearly 0 for pepper noise.

Gaussian noise [189] is statistical noise that has a probability density function (pdf) of the normal distribution also known as Gaussian distribution. It is a major part of the "read noise" of an image sensor, that is, of the constant noise level in dark areas of the image. The standard model of Gaussian noise is additive, independent at each pixel and independent of the signal intensity, caused primarily by Johnson Nyquist noise (thermal noise), including that which comes from the reset noise of capacitors .

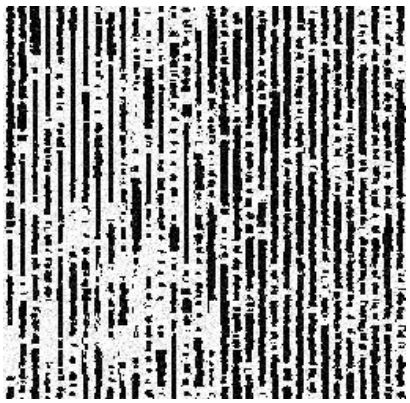
To crack security of the scrambling algorithm, the angles of rotation $\delta = a\sigma/2$ must be identified by attacker, which is not feasible to trace '*a*'. Salt and pepper noise, Gaussian noise in DFrFT transformed and scrambled image, and its resumed image are shown in Figure 3.20.



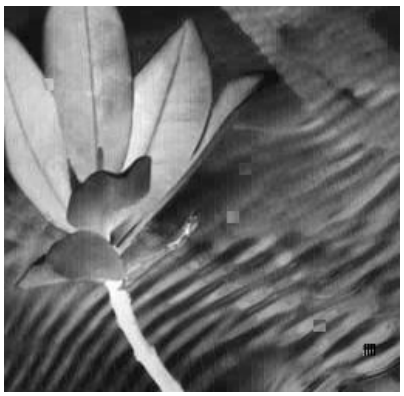
(a) Encrypted-scrambled image with salt-pepper noise.



(b) Retrieved image from salt-pepper noise.



(c) Image with Gaussian noise.



(d) Retrieved image from Gaussian noise.

Figure-3.20: Security check from salt-pepper and Gaussian noise for Flower image.

3.5 SUMMARY

An algorithm for image compression using fractional transforms has been devised. The variation in block size is preferred to give good compression percentage for $n \partial n$ blocks where, $n \cong$ 4, 8, 16, 32 and so on. The choice of block size with fractional Fourier transforms is 8 × 8; with fractional Cosine transforms is 32 × 32 and with fractional Hartley transforms is 4 × 4 because of high PSNR at various compression percentages with these block sizes. The results of DFrFT with are better with high PSNR and low MSE in comparison of DFrCT and DFrHT. It has been observed that as the compression percentage increases, the PSNR decreases. The DFrFT results are also compared with existing JPEG with the improvement in PSNR for Lena and Pepper images are 9.89 dB and 8.21 dB, respectively at the cost of CPU

time. The block transformations have shown the effect of blocking artifacts. These artifacts are analyzed for DFrFT and DFrCT. It has been interpreted that images at high compression percentages using DFrCT shows more blocking effects. The security of compressed images is also a key issue. The image compression and scrambling algorithm gives better scrambling degree as compared to Zhang method for Lena image 0.8 and for Baboon 0.92. The images have been encrypted using fractional transforms in the next chapter.

CHAPTER 4 *IMAGE ENCRYPTION*

Encryption is process of transforming the information to make it non-interpretable to anyone except authenticated users, usually referred to as a key. The larger length of key make algorithm better. Two algorithms have been performed in this chapter for image security i.e. encryption and scrambling.

4.1 INTRODUCTION

The security of image information depends on some very popular cryptographic algorithms. Cryptographic algorithms that are used for image encryption are of two types. Sometimes the two end points use same algorithm and most of the time same key is used for encrypting and decrypting the image information, and in other encryption techniques, they must use different but related key used for encrypting and decrypting purpose. Many methods have been recently proposed in the literature for the encryption of 2D information using digital holography technique [202], scan patterns [152], digital signature [116], and chaos [203]. The advantage of the discrete fractional transforms approach is that high security is provided with the help of fractional orders. The fractional orders are used as fractional keys in the algorithm. The scrambling and encryption algorithm enhanced the security from noise attacks. Scrambling [204] is shuffling of image pixels to make it unrecognizable. The simulation results are also shown that scrambling is also used to encrypt the compressed image. The relative error has been measured and brute force attack is discussed at the last part of chapter.

4.2 IMAGE ENCRYPTION

The goal of image encryption is security and privacy of the image content when it is shared over the internet. The work is presented on image encryption using discrete fractional transforms. By way of increasing the security of the image information, the presented image encryption algorithm use two random phase masks.

The outcome of this chapter has been **accepted** in Research Journal as per following detail: N. Jindal, K.Singh, Image Retrieval Algorithm based on Discrete Fractional Transforms, Journal of Electrical Engineering.

The significant feature of discrete fractional Fourier domain in image encryption benefits from its extra degree of freedom that is provided by its fractional orders. Because this algorithm only uses the fractional order and random phase masks as the secret keys, the complexity of the encryption process is less. In decryption, the quality of image depends on correct fractional order and random phase mask, effect on PSNR with wrong key is also noted.

4.2.1 Image Encryption using Fractional Transforms

The n-stage of discrete fractional transforms can provide n-dimensional extra keys indicated by the fractional orders. Two fractional orders along x-axis and y-axis are used in two-dimensional discrete fractional order transforms. Such a system can have n-1 random phase filters, so that the total encryption keys can be increased to as many as 3n-1. The order 'α' along x and y direction is taken to be same i.e. $\delta_x \cong \delta_y \cong \delta$ and then multiple stages of fractional transforms are cascaded together. In the intermediate planes randomly encoded phase masks are used. Algorithm consists of two parts: encryption to encrypt the image and decryption to retrieve the image. The block diagram of Figure 4.1 has been shown for the encryption using fractional transform.

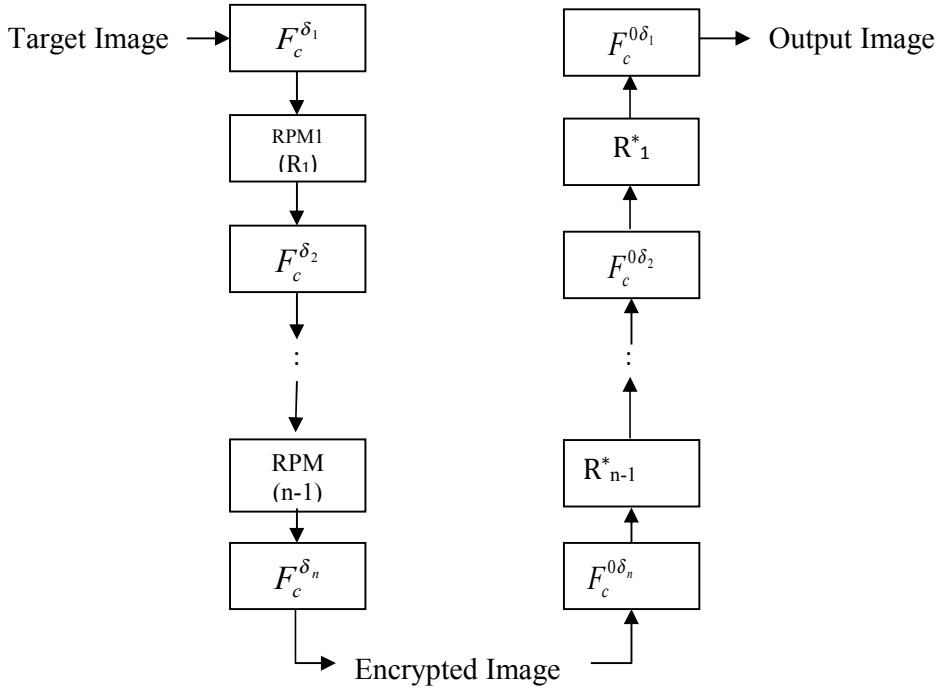


Figure-4.1: Block diagram for image encryption and decryption using fractional transforms.

(a) ***Using Two Fractional Keys***

Let $f(x_0, y_0)$, is a real valued two dimensional image data to be encrypted. The image is fractional cosine transformed two times using fractional orders α_1 and α_2 respectively. In the intermediate stages we put one random phase mask (RPM) serving as phase filters respectively in equation given by [62]:

$$p_1(x_1, y_1) \cong \exp\{j0\lambda 2\sigma\iota_1(x_1, y_1)\} \tag{4.1}$$

The function $\iota_1(x_1, y_1)$, is randomly generated homogeneously distributed function with values (0, 1). Thus the resultant transformed function $Z(x, y)$, can be written as in the following equations:

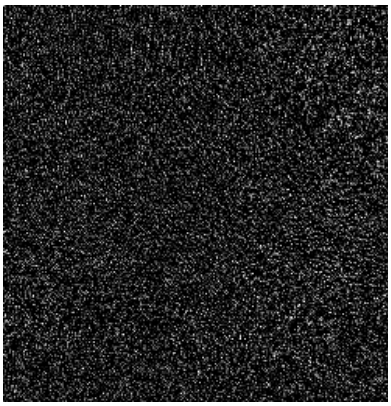
$$Z(x, y) \cong F_c^{\delta 2}\{Z_1(x_1, y_1) p(x_1, y_1)\} \tag{4.2}$$

$$Z_1(x_1, y_1) \cong F_c^{\delta 1}\{f(x_0, y_0)\} \tag{4.3}$$

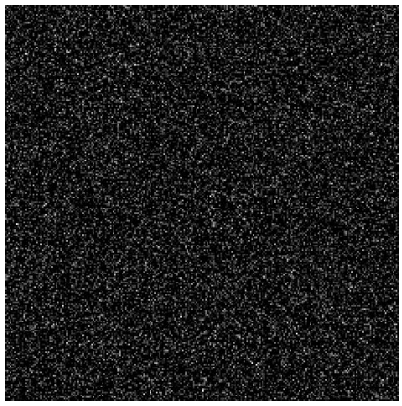
The final resultant function $Z(x, y)$, is the encrypted image. The decryption process is the inverse operation with respect to encryption. First, apply a DFrCT of order $-\alpha_2$ on the encrypted image $Z(x, y)$, and multiply the random phase mask $p_1^*(x_1, y_1)$, , then get the next function $Z_1(x_1, y_1)$, . Finally the original image $f(x_0, y_0)$, has obtained with $-\alpha_1$. Here the random mask $p_1^*(x_1, y_1)$, is the complex conjugate of $p_1(x_1, y_1)$, .

Encryption and decryption results for Pyramid, Pentagon, Girl images of 256 ∂ 256 pixels using discrete fractional transforms (DFrFT, DFrCT) are obtained. Numerical simulations have been performed to examine the performance of discrete fractional transforms (DFrFT, DFrCT). The original images are encrypted with fractional keys {1, 0.95} using DFrCT, shown in Figures 4.2 to 4.4. The simulation results using DFrFT with fractional keys {0.5, 0.5} are shown in Figures 4.5 to 4.7.

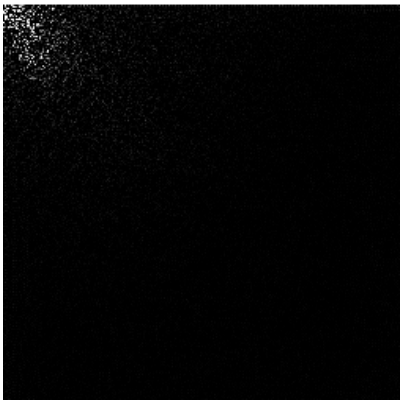
(a) Orig



(b) Encrypted Pyramid image



first wrong key.

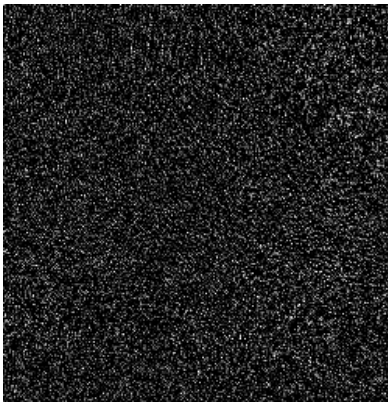


(d) Incorrectly decrypted image with second wrong key.

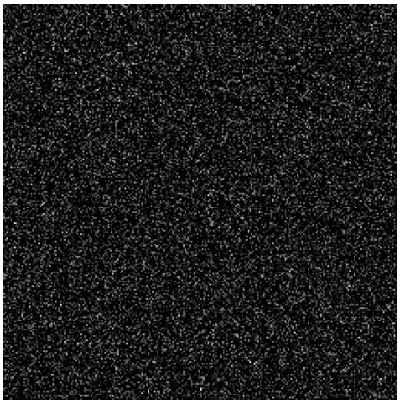
(e) Correctly decrypted image with PSNR=82.72dB.

Figure-4.2: Simulation results of Pyramid image for image encryption using DFrCT.

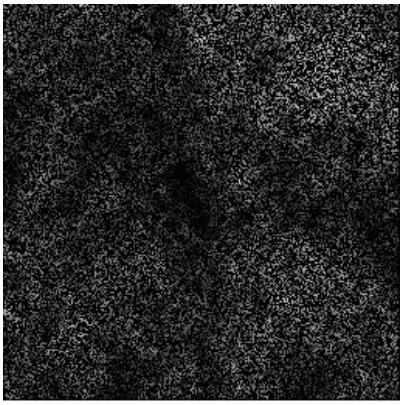
(a) Orig



(b) Encrypted Pentagon image



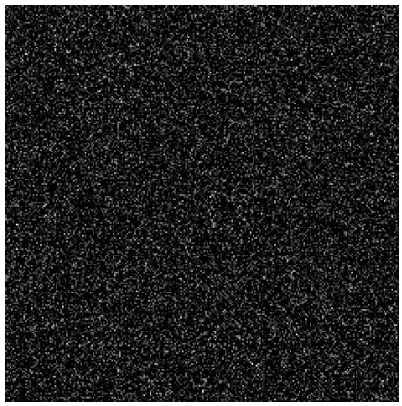
first wrong key.



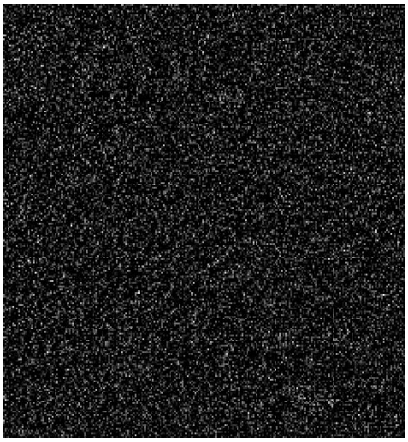
(d) Incorrectly decrypted image with second wrong key.

(e) Correctly decrypted image with PSNR=79.21 dB.

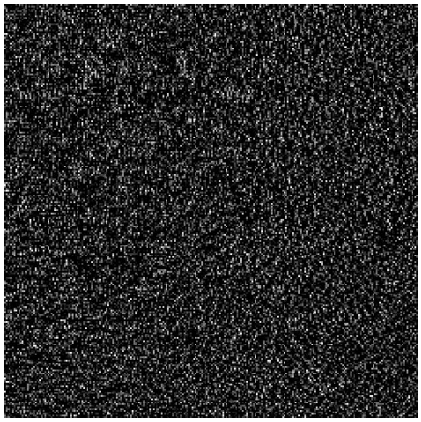
Figure-4.3: Simulation results of Pentagon image for image encryption using DFrCT.



(b) Encrypted Girl image



first wrong key.

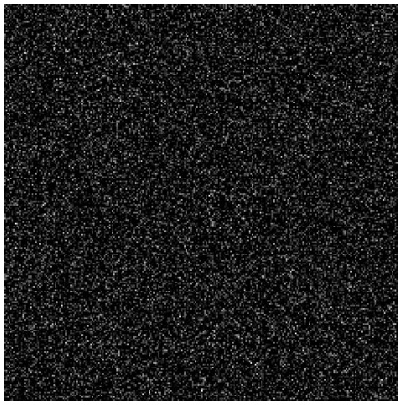


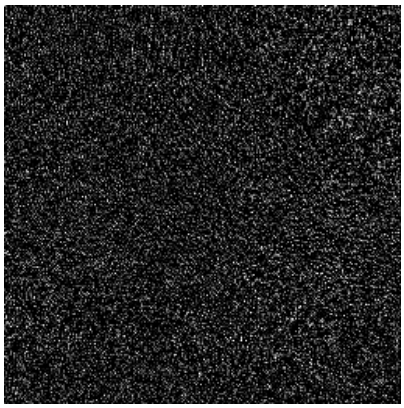
second wrong key.

) Correctly decrypted image with PSNR=81.55 dB.

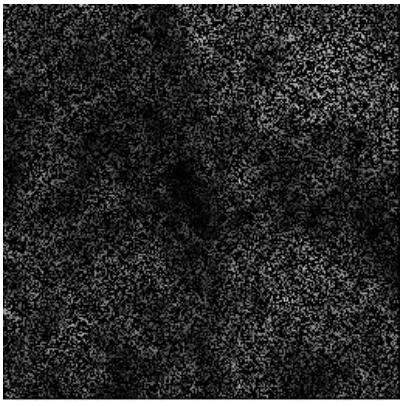
Figure-4.4: Simulation results of Girl image for image encryption using DFrCT.

nid image.





(c) Incorrectly decrypted image with first wrong key.

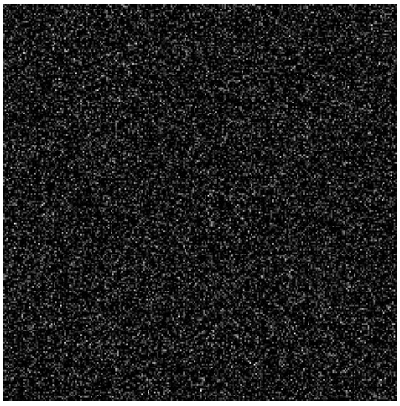


(d) Incorrectly decrypted image with second wrong key.

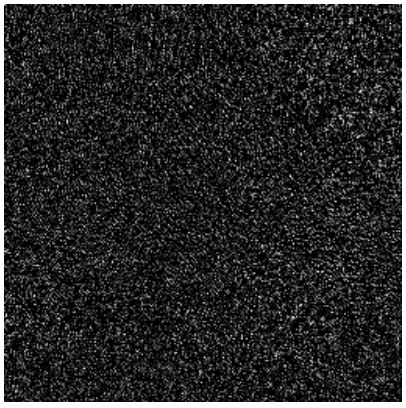
(e) Correctly decrypted image with PSNR=60.04 dB.

Figure-4.5: Simulation results of Pyramid image for image encryption using DFrFT.

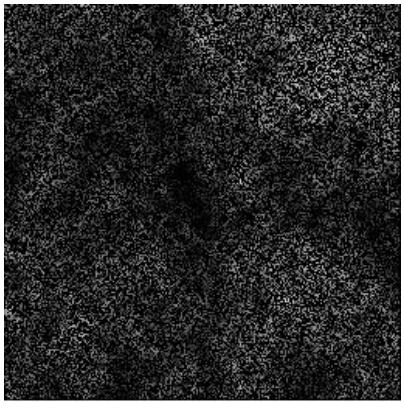
on image.



PSNR=63.38 dB.



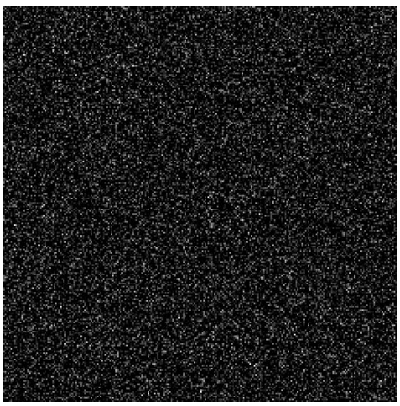
) Incorrectly decrypted image with first wrong key.



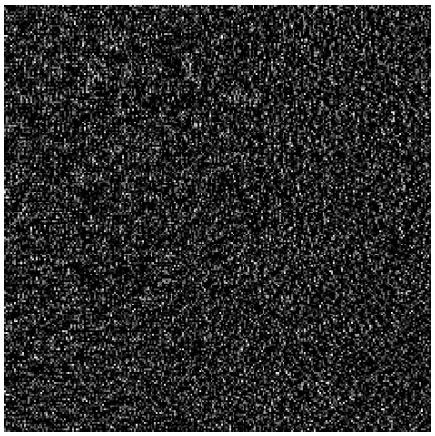
(d) Incorrectly decrypted image with second wrong key.

(e) Correctly decrypted image with PSNR=59.04 dB.

Figure-4.6: Simulation results of Pentagon image for image encryption using DFrFT.



(d) Encrypted Girl image with second wrong key.



(e) Correctly decrypted image with first wrong key.

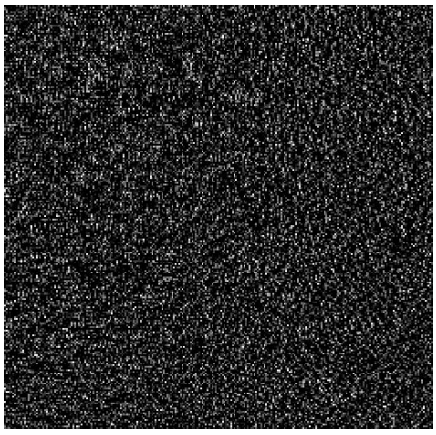


Figure-4.7: Simulation results of Girl image for image encryption using DFrFT.

The simulation results of Pyramid, Pentagon and Girl images shows that encryption with DFrCT performs better than DFrFT. The PSNR of encrypted and decrypted images is high in case of DFrCT. The inverse of two fractional keys and conjugate of random phase mask is used for decryption. First wrong fractional key has decreased the PSNR and original image is not obtained. The second wrong fractional key also did not retrieve the image. Te results of images with two fractional keys are given in Table 4.1.

Table-4.1: PSNR (dB) of images using DFrFT and DFrCT with two fractional keys.

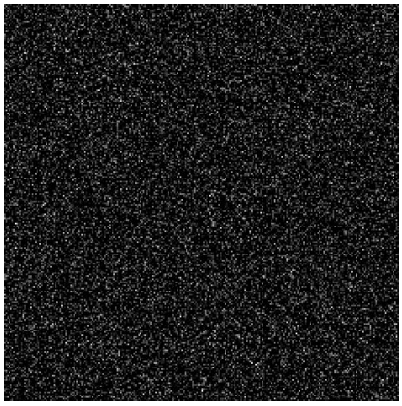
Image	PSNR (dB)	
	DFrFT	**DFrCT**
Pyramid	55.30	82.72
Pentagon	53.74	79.21
Girl	58.44	81.55
Lena	54.31	79.84
Baboon	54.65	83.72
Boat	54.02	80.66
Flower	55.39	83.19
House	54.48	81.40
Barbara	52.11	77.68
Peppers	54.34	78.10

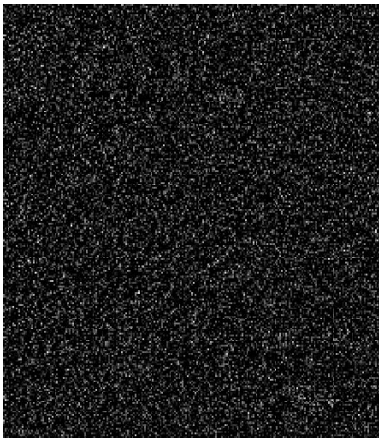
The results have been taken of various images and it has been observed that PSNR is better using DFrCT. The decrypted image using DFrFT has less PSNR, so image quality is not good in comparison of DFrCT.

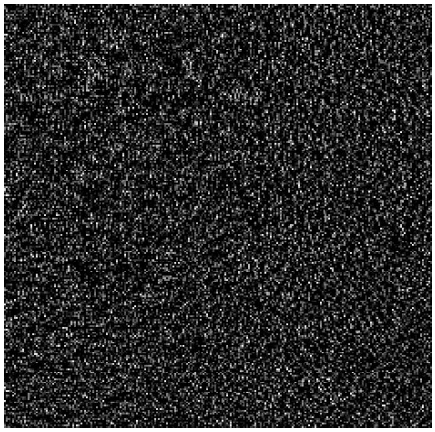
(c) ***Using Three Fractional Keys***

The security can be enhanced by using three fractional keys and two random phase masks. The third fractional key will increase the key space. It will become difficult for the intruders to crack the three fractional keys and two random phase masks. The second random phase mask is given by equation [62]:

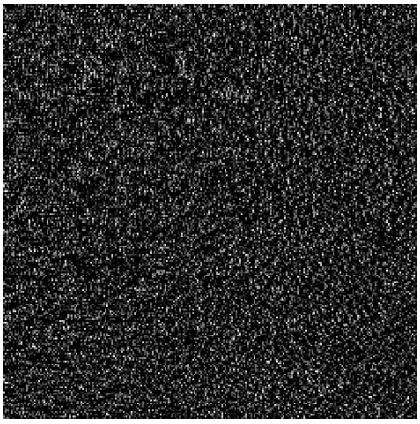
$$p_2(x_2, y_2) \cong \exp[j2\pi r_2(x_2, y_2)] \tag{4.4}$$







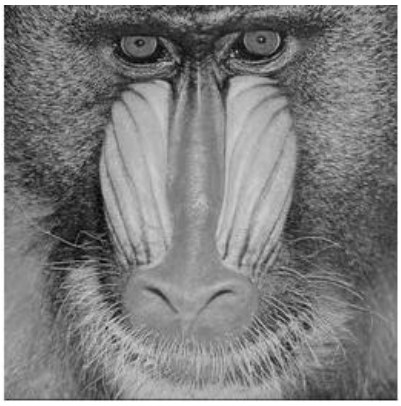
with second wrong key.

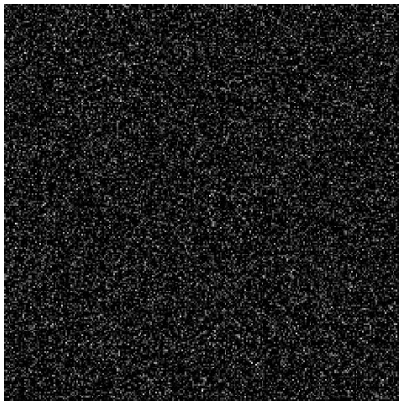


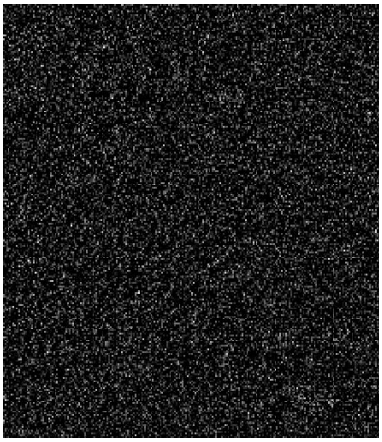
third wrong key.

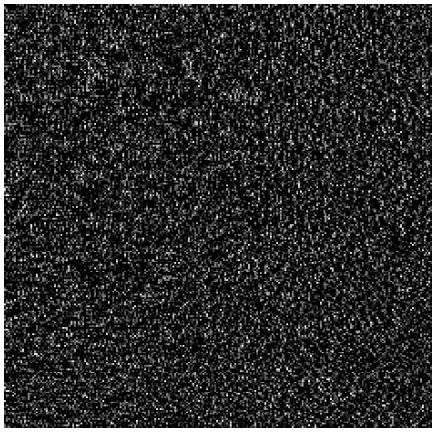
Correctly decrypted image with PSNR=79.84 dB.

Figure-4.8: Simulation results of Lena image for image encryption using DFrCT.

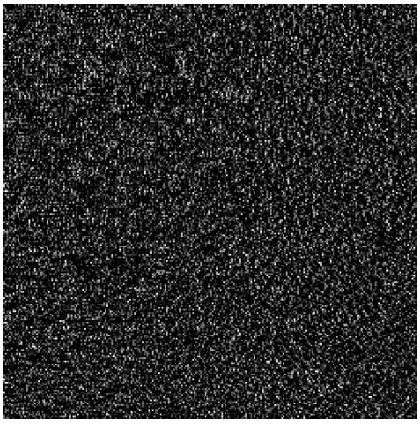




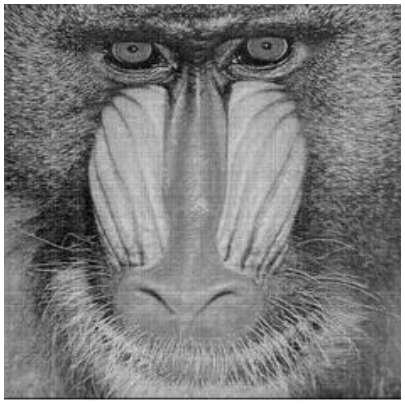




second wrong key.

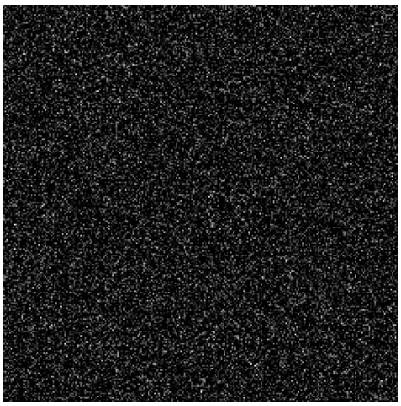


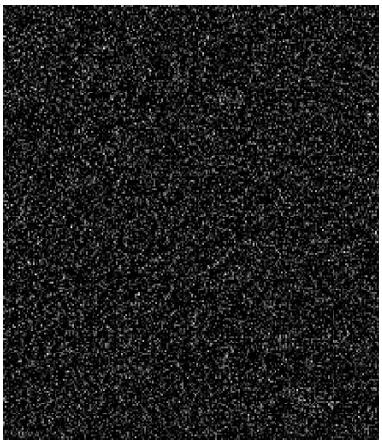
third wrong key.

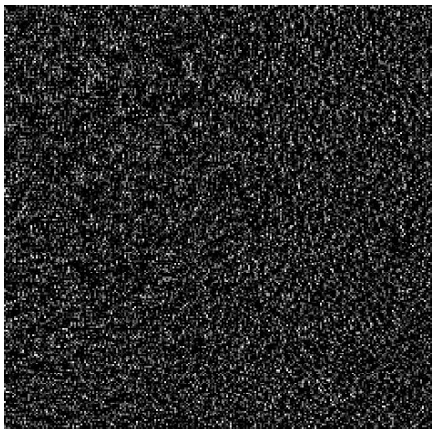


Correctly decrypted image with PSNR=83.72 dB.

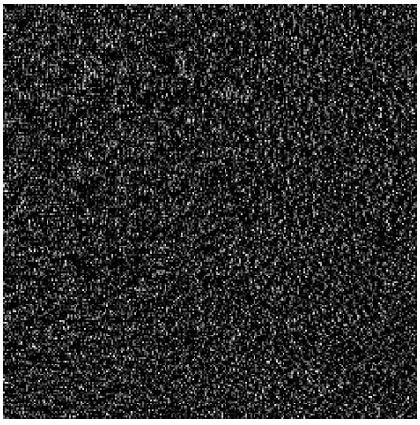
Figure-4.9: Simulation results of Baboon image for image encryption using DFrCT.







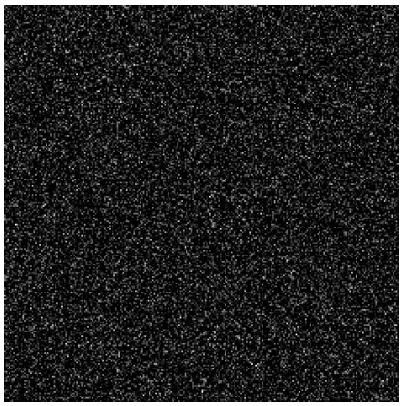
second wrong key.

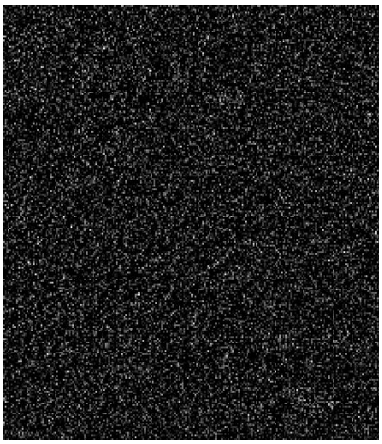


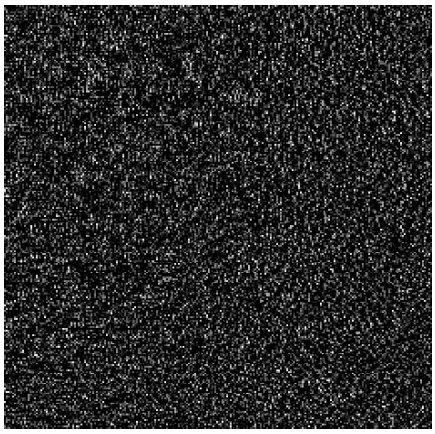
third wrong key.

Correctly decrypted image with PSNR=80.66 dB.

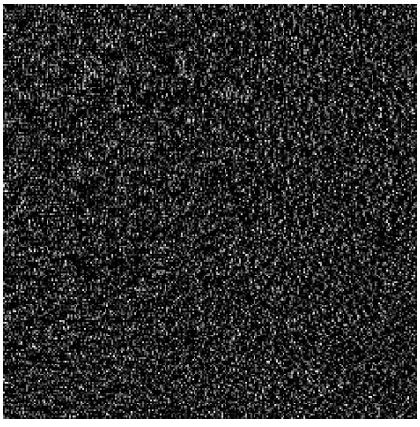
Figure-4.10: Simulation results of Boat image for image encryption using DFrCT.







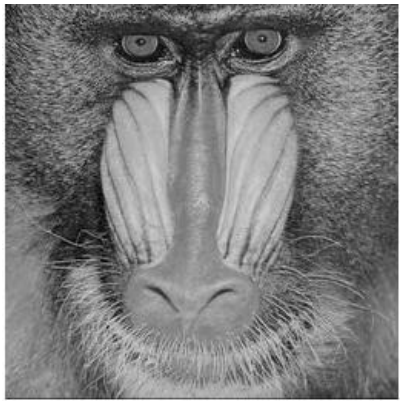
second wrong key.

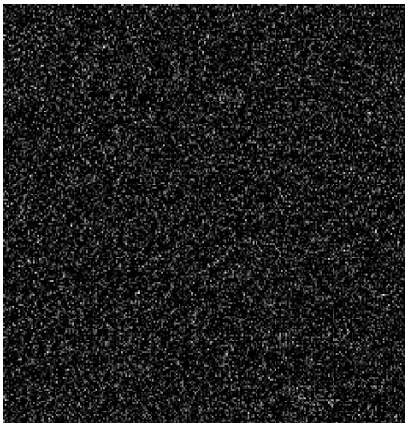


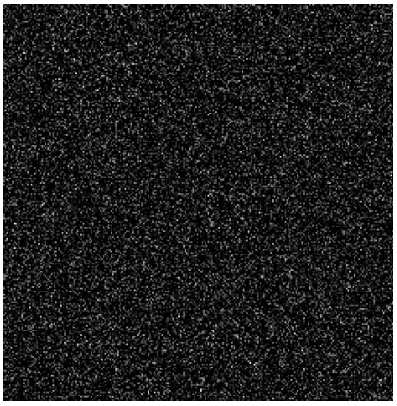
third wrong key.

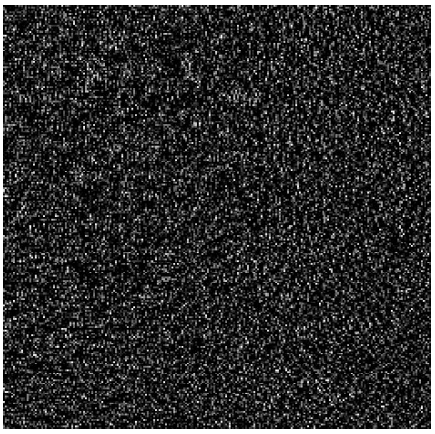
Correctly decrypted image with PSNR=80.04 dB.

Figure-4.11: Simulation results of Lena image for image encryption using DFrFT.

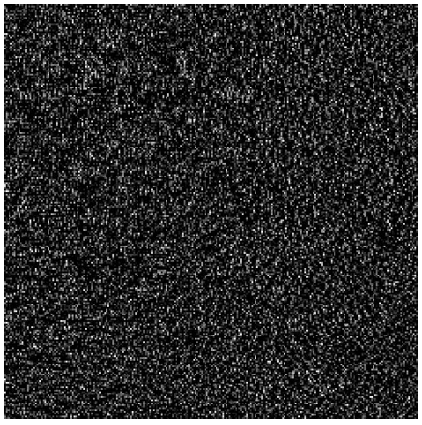




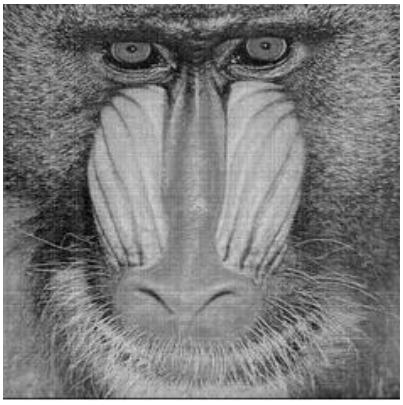




second wrong key.

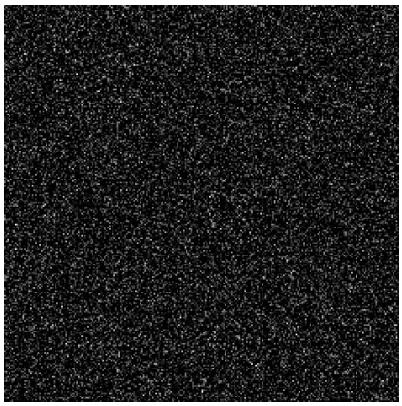


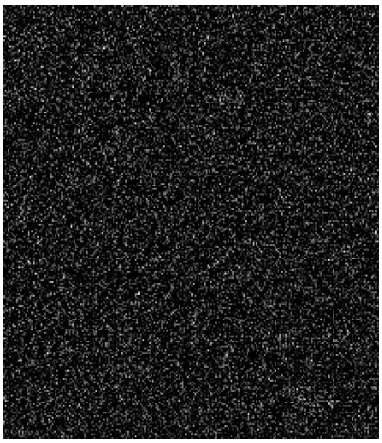
with third wrong key.

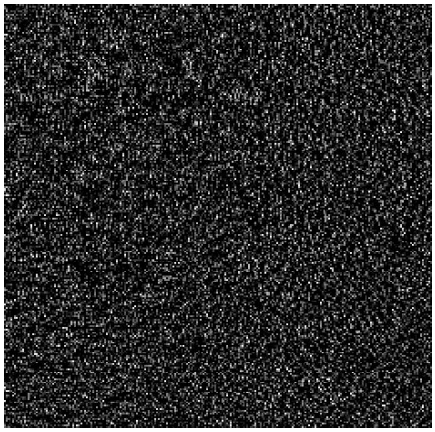


f) Correctly decrypted image with PSNR=59.20 dB.

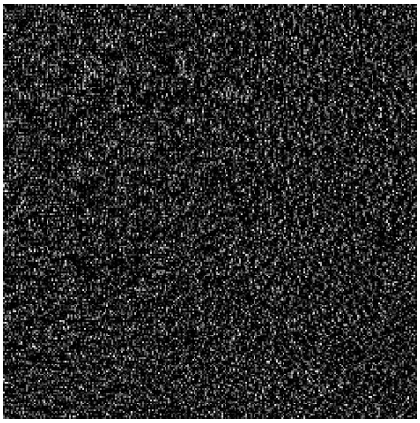
Figure-4.12: Simulation results of Baboon image for image encryption using DFrFT.







second wrong key.



third wrong key.

Correctly decrypted image with PSNR=58.04 dB.

Figure-4.13: Simulation results of Boat image for image encryption using DFrFT.

It has observed that for the decryption all the three keys should be used in inverse order. The effect of single wrong key on PSNR is shown in Figures. The wrong keys will not decrypt the image. The conjugate of random phase masks are used in decryption. It has been observed that the PSNR value of encrypted and decrypted images is less using DFrFT than DFrCT. The PSNR of images is given in Table 4.2.

TABLE-4.2: PSNR (dB) of images using DFrFT and DFrCT with three fractional keys.

Image	PSNR (dB)	
	DFrFT	**DFrCT**
Pyramid	55.30	82.33
Pentagon	53.74	79.16
Girl	58.44	81.29
Lena	54.31	79.48
Baboon	54.65	83.66
Boat	54.02	80.01
Flower	55.39	82.88
House	54.48	81.00
Barbara	52.11	77.64
Peppers	54.34	78.00

(d) ***Using Four Fractional Keys***

The PSNR is better using DFrCT with three fractional keys. So, to enhance the security, DFrCT using four fractional keys and three random phase masks is used to encrypt images. The fractional orders α_1, α_2, α_3 and α_4 used to encrypt the images using DFrFT and DFrCT. The third random phase mask is also used as given in equation:

$$p_3(x_1, y_1) \cong \exp[j0\lambda 2\sigma(t_3(x_3, y_3))] \tag{4.5}$$

Now, resultant transformed function $Z(x, y)$, can be written as in the following equations:

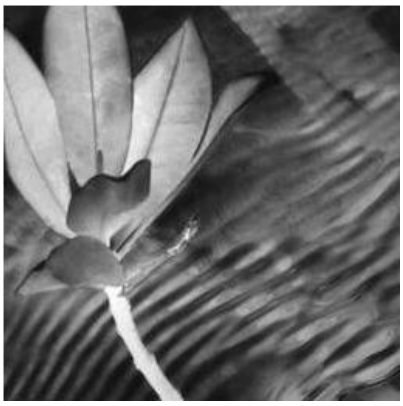
$$Z(x, y) \cong F_c^{\delta 4}\{Z_3(x_3, y_3), p(x_3, y_3)\} \tag{4.6}$$

$$Z_3(x, y) \cong F_c^{\delta 3} \sim Z_2(x_2, y_2, p(x_2, y_2)) \quad (4.7)$$

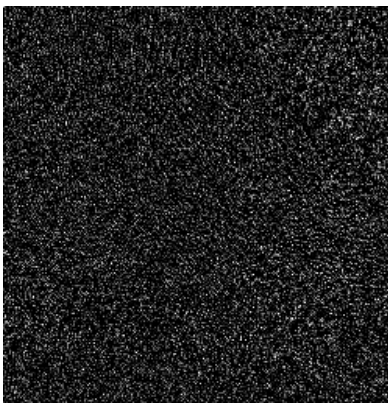
$$Z_2(x, y) \cong F_c^{\delta 2} \sim Z_1(x_1, y_1, p(x_1, y_1)) \quad (4.8)$$

$$Z_1(x_1, y_1) \cong F_c^{\delta 1} \sim f(x_0, y_0) \quad (4.9)$$

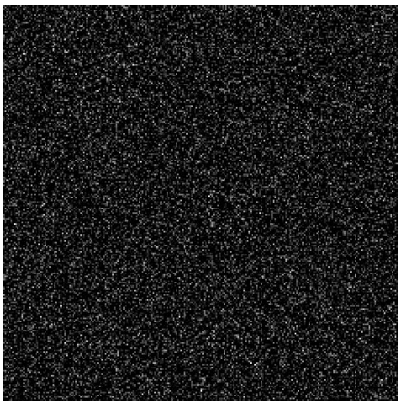
The complex conjugates of random phase masks are used in decryption. It has been resulted that all the four keys should be correct. The simulation results are shown in Figures.



(a) Original Flower image.

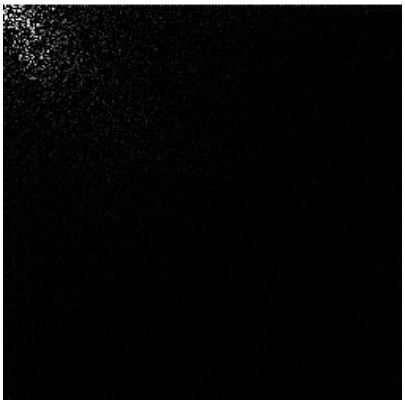


(b) Encrypted Flower image

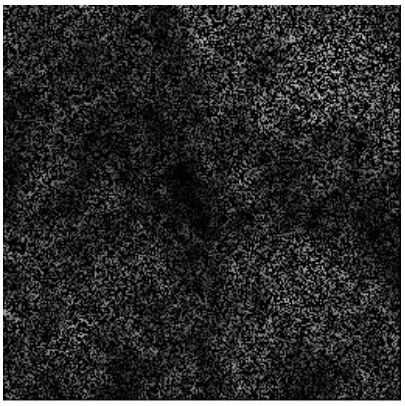


with first wrong key.

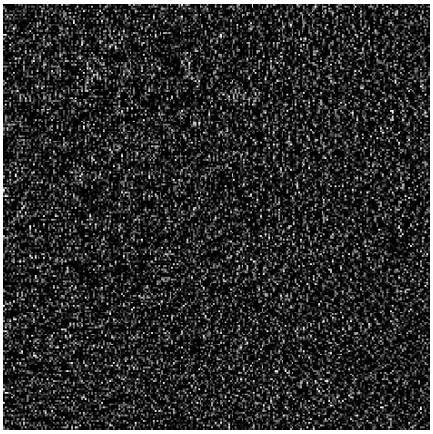
Figure-4.14: Simulation results of Flower image for image encryption using DFrCT. (contd.)



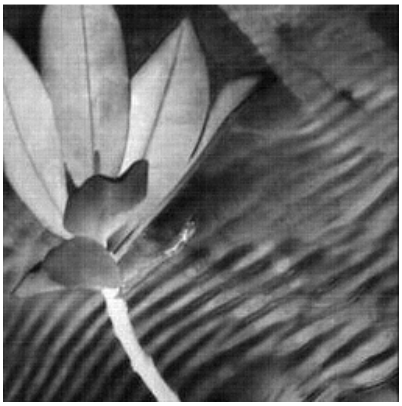
(d) Incorrectly decrypted image



(e) Incorrectly decrypted image with third wrong key.



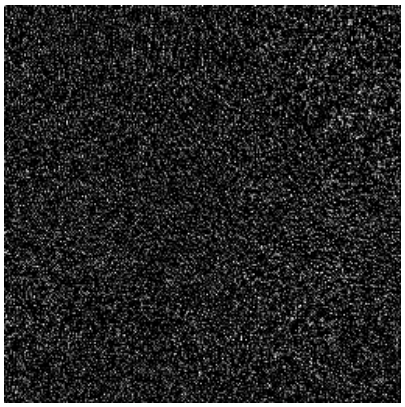
(f) Incorrectly decrypted image with fourth wrong key.



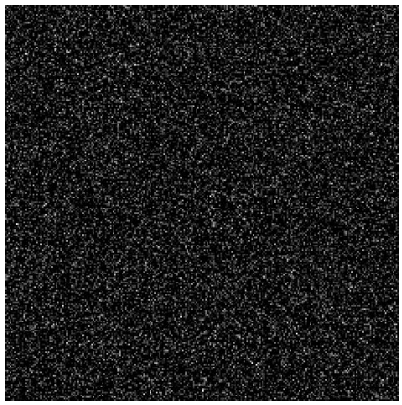
(g) Correctly decrypted image with PSNR=83.19 dB.

Figure-4.14: Simulation results of Flower image for image encryption using DFrCT.

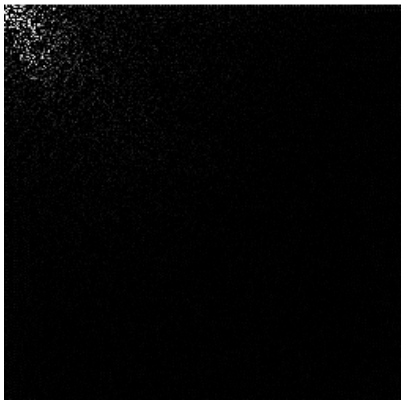
(a) Original House image.



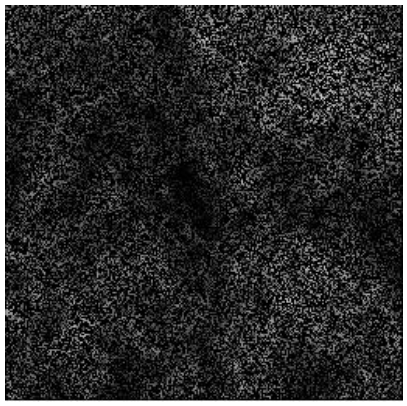
(b) Encrypted House image



first wrong key.

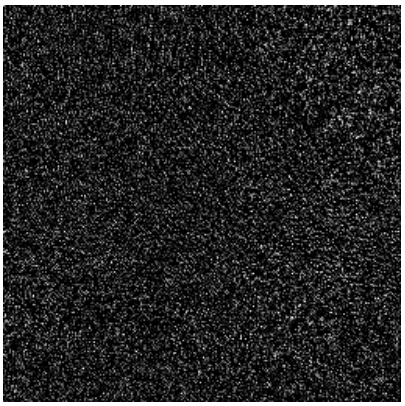


(d) Incorrectly decrypted image with second wrong key.



(e) Incorrectly decrypted image with third wrong key.

Figure-4.15: Simulation results of House image for image encryption using DFrCT. (contd.)

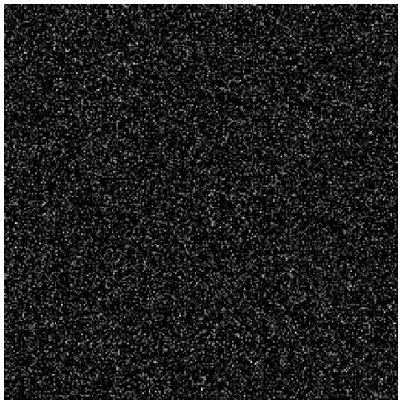


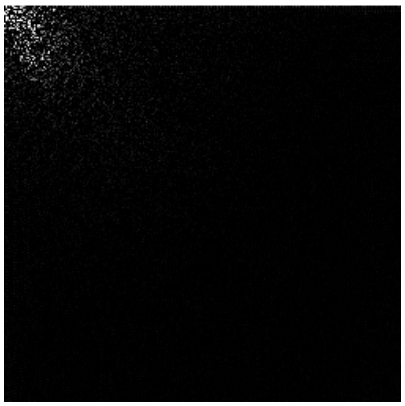
(f) Incorrectly decrypted image with fourth wrong key.

(g) Correctly decrypted image with PSNR=81.40 dB.

Figure-4.15: Simulation results of House image for image encryption using DFrCT.

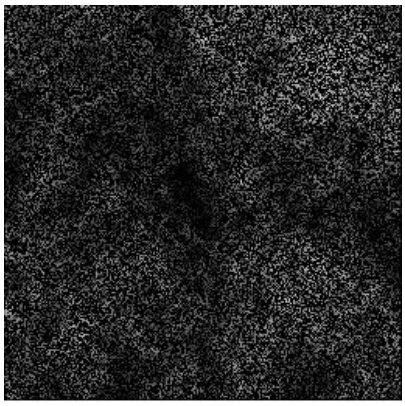
ara image.



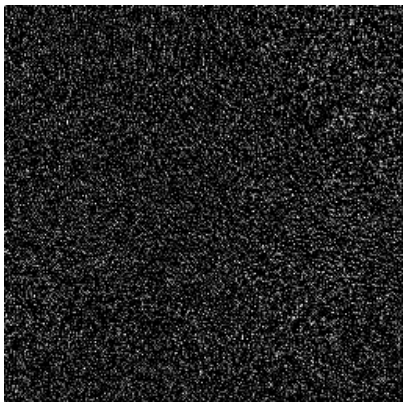


(c) Incorrectly decrypted image with first wrong key.

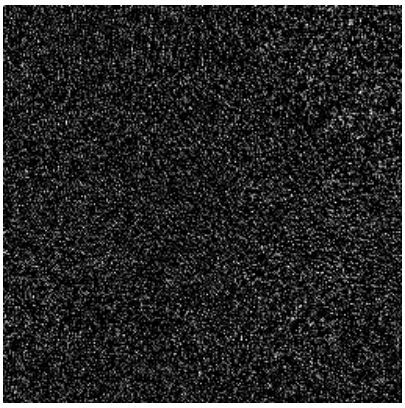
Figure 4.16: Simulation results of Barbara image for image encryption using DFrCT (contd.)



(d) Incorrectly decrypted image with second wrong key.



(e) Incorrectly decrypted image with third wrong key.



(e) Incorrectly decrypted image with fourth wrong key.

(f) Correctly decrypted image with PSNR=77.68 dB.

Figure-4.16: Simulation results of Barbara image for image encryption using DFrCT.

The variation in fractional keys for image encryption using DFrFT and DFrCT has been also observed. Table 4.1 gives PSNR using two, three and four fractional keys. The PSNR value in the table demonstrates that Pyramid image has 82.72 dB PSNR using two fractional keys and one random phase. But it decease 0.39 dB while using the three fractional keys and two random phase masks. The PSNR value has reduced 0.78 dB with four fractional keys and three random phase masks as compare to three fractional keys. The advantage is that numbers of fractional keys increase the key space size. The increased key size creates the difficulty for invaders and the security is increased. But the computational complexity also increases with four fractional keys. So, there should be balance between security and complexity in the applications.

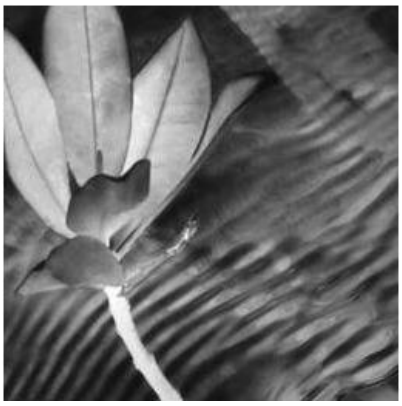
TABLE-4.3: PSNR (dB) of images using DFrCT with Four fractional keys.

Image	PSNR (dB)
	DFrCT
Pyramid	82.72
Pentagon	79.21
Girl	81.55
Lena	79.84
Baboon	83.72
Boat	80.66
Flower	83.19
House	81.40
Barbara	77.68
Peppers	78.10

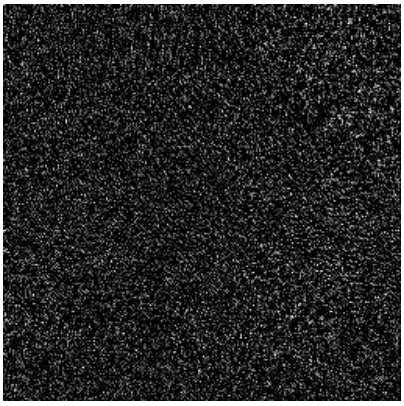
The image encryption using DFrCT is better with improved PSNR, but there are still some noise attacks present. The quality of reconstructed images has been decreased with the noise interference and is discussed in the next section.

4.3 NOISE ATTACKS

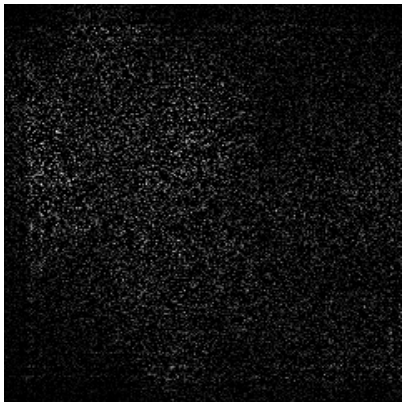
To fulfill the demand of security and privacy, image encryption using fractional transforms is presented [205]. But there are some random degradation called noises present in image encryption like salt-pepper noise, Gaussian noise. It is necessary to discuss these noises so that algorithms can be developed to make the reconstructed images more visual. The results of Flower image has been shown with Salt-pepper noise in Figure 4.18. The reconstructed image quality has been decreased with the interference of noise.



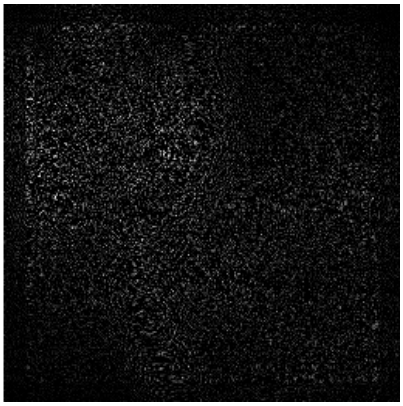
(a) Original Flower test image.



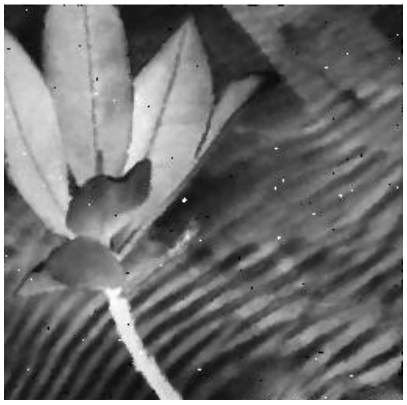
(b) Encrypted Flower image



(c) Encrypted and noisy (Salt-pepper) image.

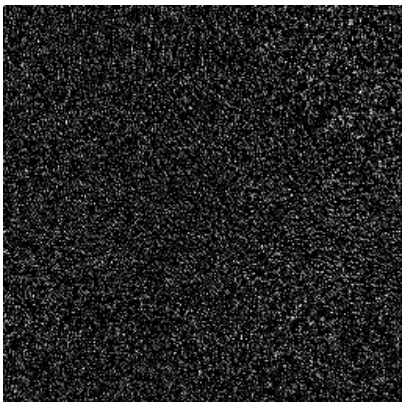


(d) Incorrectly decrypted image.

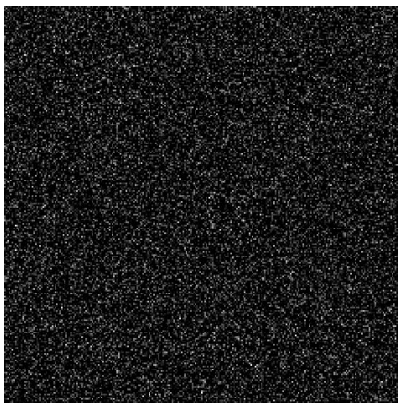


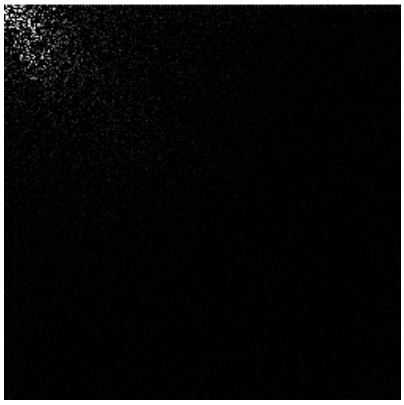
(e) Correctly Decrypted Image.

Figure-4.17: Simulation results with salt-pepper noise.



(b) Encrypted House image





(Gaussian) Image.

) Incorrectly decrypted image.

(e) Correctly decrypted image with PSNR=61.60 dB.

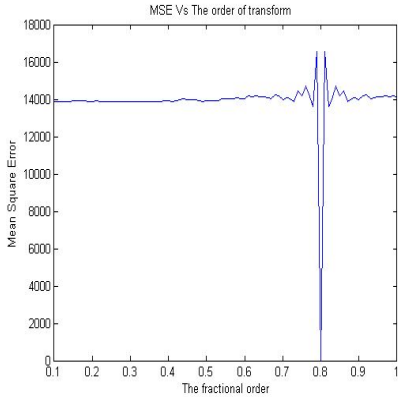
Figure-4.18: Simulation results with Gaussian noise.

The intervention of Gaussian noise in the image encryption is shown in the Figure 4.19. The original House image is encrypted using fractional transforms. The involvement of Gaussian noise in the encrypted is shown in Figure 4.19(c). Figure 4.19(f) is the correctly decrypted image with the right fractional keys and median

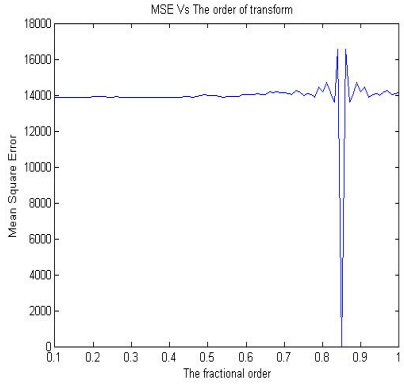
filter. It can be clearly visualized that the effect of Gaussian noise has reduced the quality of image.

4.4 SENSITIVITY OF ENCRYPTION KEYS

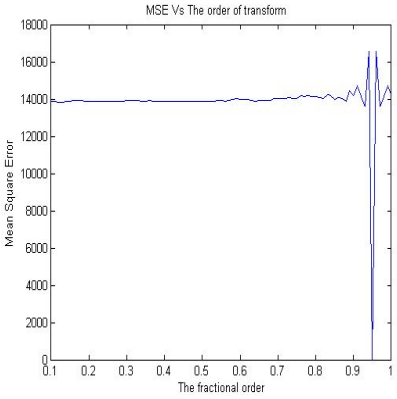
The important feature of encryption algorithm is the ability to specify a 'key' of some kind, and have the encryption method alter itself such that each 'key' produces a different encrypted output, which requires a unique 'key' to decrypt. This can either be a symmetrical key (both encrypt and decrypt use the same key) or asymmetrical key (encrypt and decrypt keys are different). Sensitivity of decryption keys is measured with the effect on mean square error [206]. It has been observed from Figure 4.19 that MSE is minimum at correct fractional order and maximum at other orders for all three right keys.



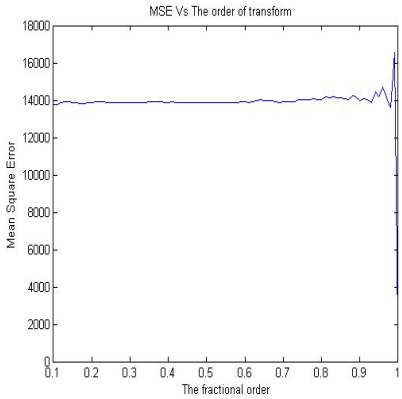
(a) Change in MSE vs. first Fractional order.



(b) Change in MSE vs. second Fractional order.



(c) Change in MSE vs. third Fractional order.



(d) Change in MSE vs. fourth Fractional order.

Figure 4.19: Change in MSE vs. deviation in Fractional orders using DFrCT.

4.5 IMAGE ENCRYPTION AND SCRAMBLING

The security of images can be enhanced images with encryption and scrambling [207]. An algorithm has been presented for encryption based on fractional keys and scrambling.

4.5.1 Algorithm

To retrieve the image correctly, the particular information of fractional keys, the random phase masks and the algorithm of scrambling is required. The encryption keys are used between 0 and 1. The fractional part of encryption keys can be increased upto any number. Hence, it becomes almost impossible for intruder to crack the algorithm with extensive number of combinations. The scrambling enhances the security level. The flow chart of algorithm is given in Figure 4.20.

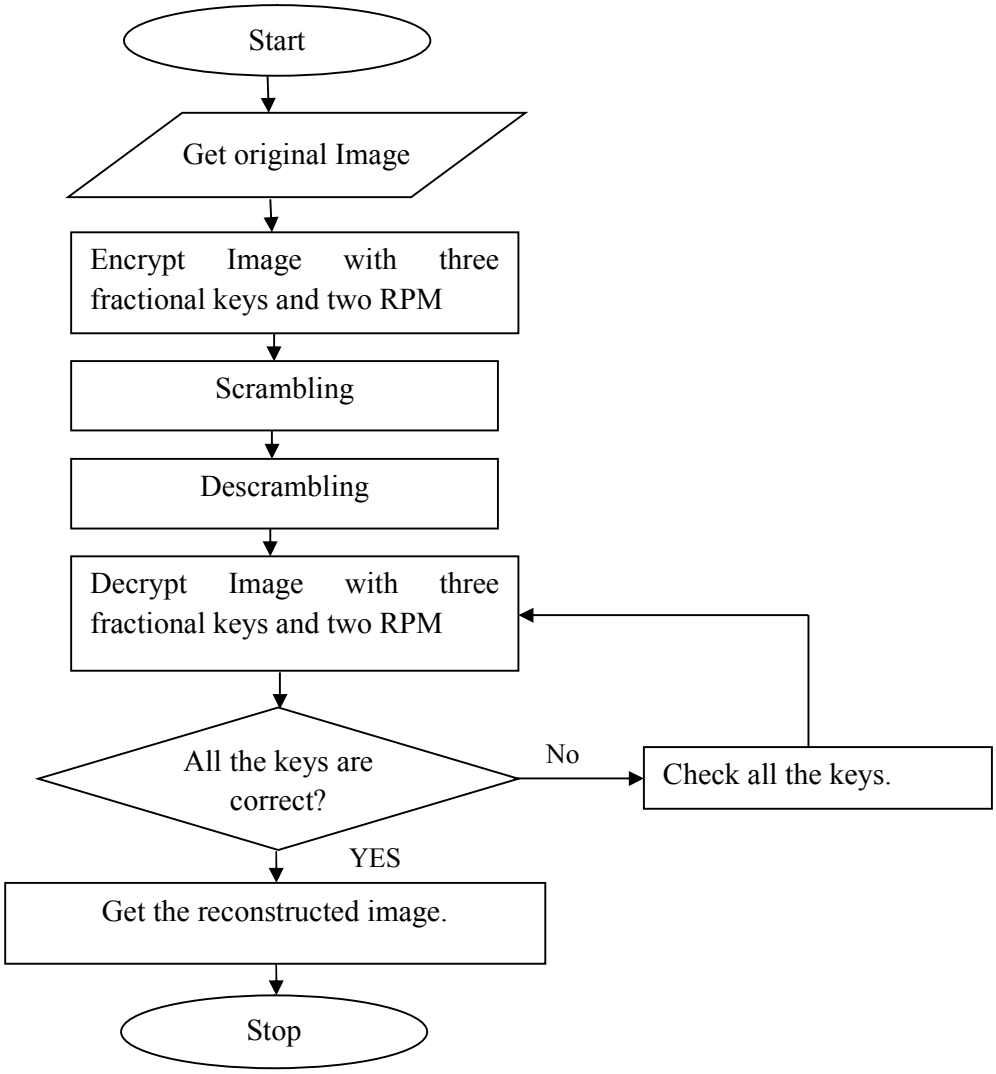
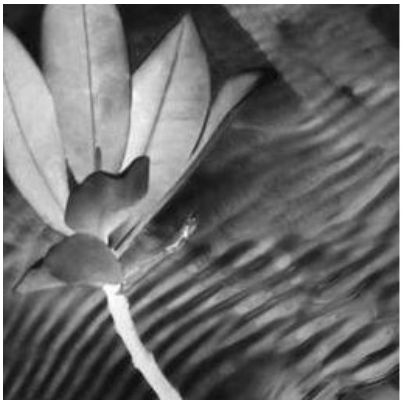
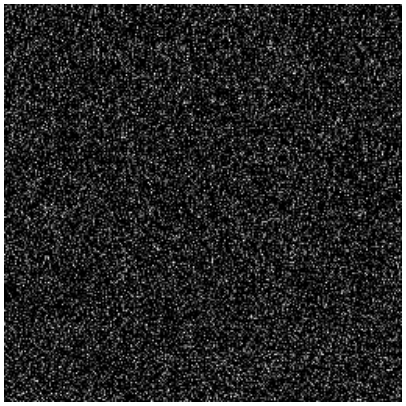


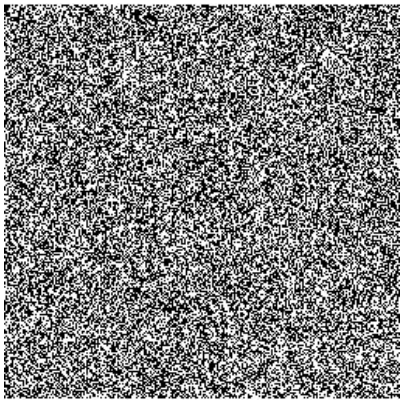
Figure-4.20: Flow chart for the Encryption and Scrambling.



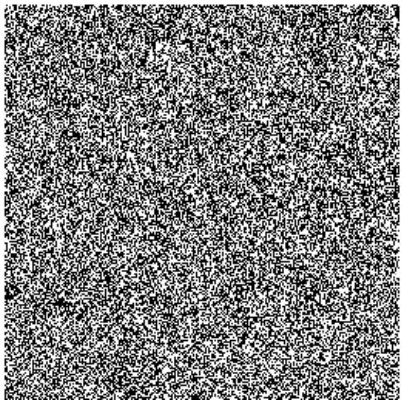
(a) Input Flower image.



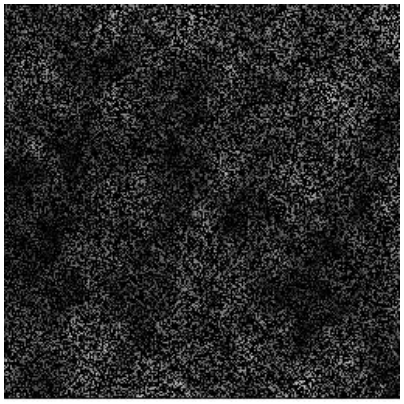
(b) Encrypted image.



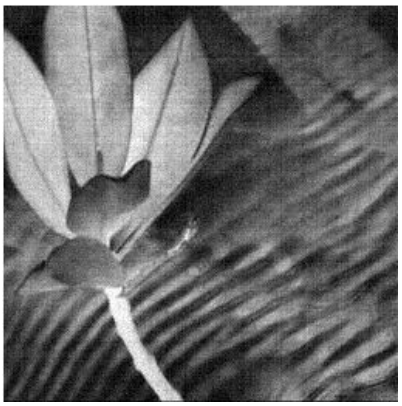
(c) Encrypted-scrambled image.



(d) Encrypted-descrambled image.



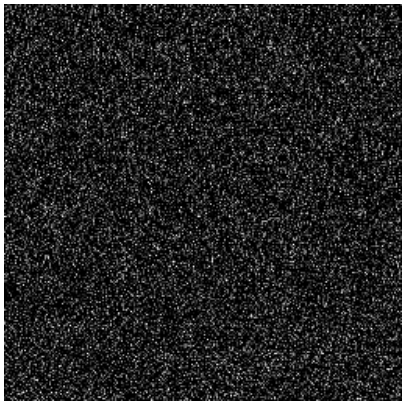
(e) Decrypted image with wrong fractional keys.



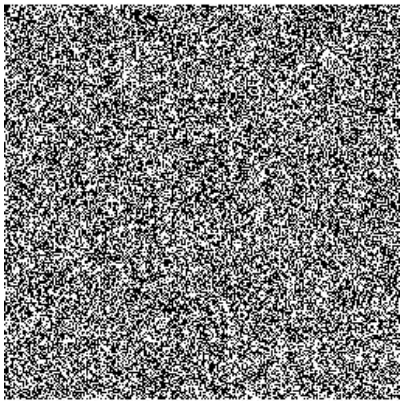
(f) Decrypted image with right fractional keys.

Figure-4.21: Simulation results of encryption based on DFrFT and scrambling.

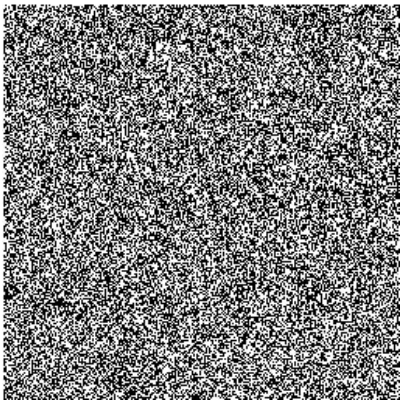
(a) Input Boat image.



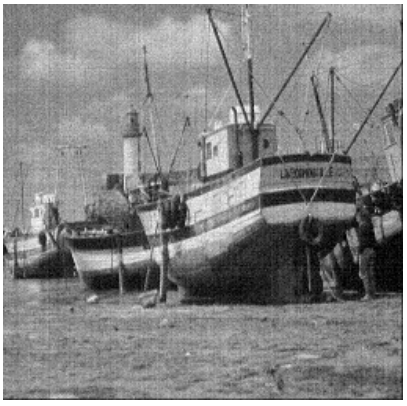
(b) Encrypted image.



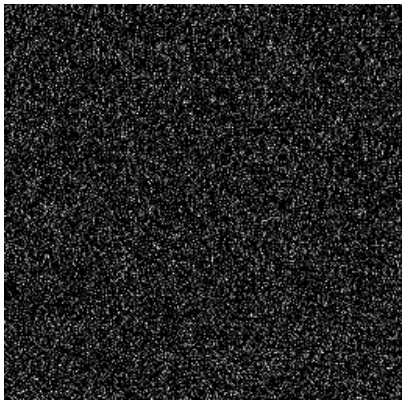
(c) Encrypted-scrambled image.



(d) Encrypted-descrambled image.



(e) Decrypted image with wrong fractional keys.



(f) Decrypted image with right fractional keys.

Figure-4.22: Simulation results of encryption based on DFrCT and scrambling.

4.5.2 Characteristic Measures of Algorithm

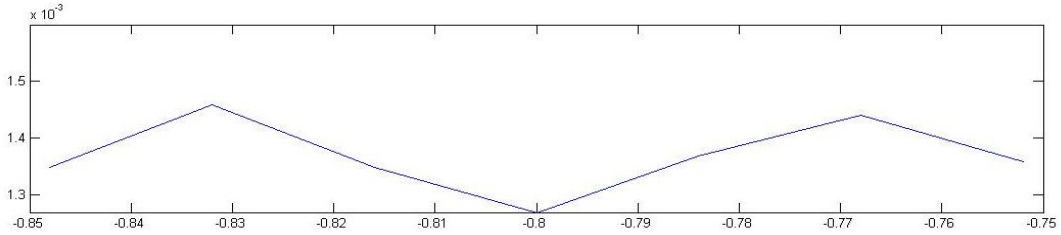
The quality of reconstructed image can be checked by relative error. Various noise interferences are also discussed in this section.

4.5.2.1 Relative error

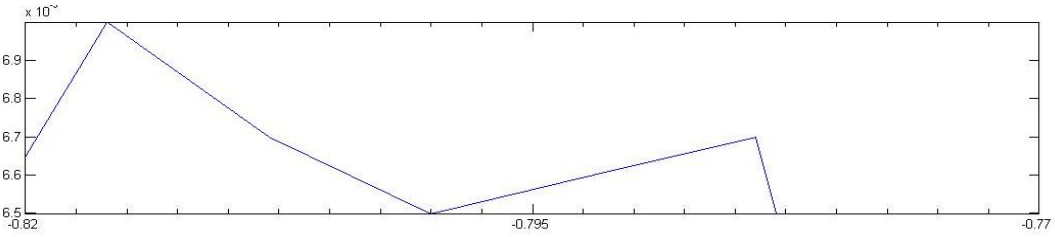
The normalized mean square error called as relative error (RE) between the output image and input image is often used to verify the quality of the reconstructed image [189] defined with the equation:

$$RE \cong \frac{\sum_{i}^{M} \sum_{j}^{N} \left\| r(i,j,) \right| 0 \left| o(i,j,) \right\|^2}{\sum_{i}^{M} \sum_{j}^{N} \left| o(i,j,) \right|^2} \tag{4.6}$$

The dependence of RE on the change of fractional order - δ_3 is shown with T=1, 2, 4,8,16, respectively in Figures 4.23(a)-4.23(e), where T is number of iterations. In the decryption procedure, image can be retrieved with correct fractional orders keys and iterations only.

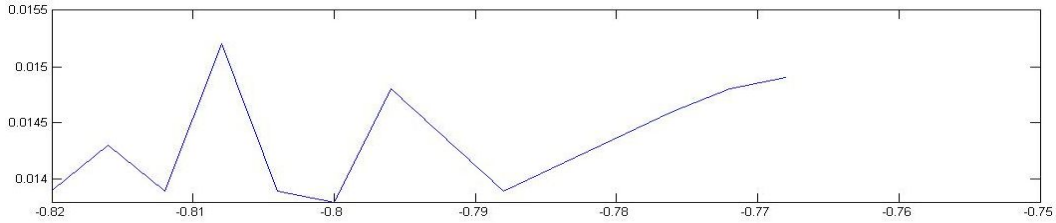


(a) RE (along y-axis) vs. change in fractional order (along x-axis) with T=1.

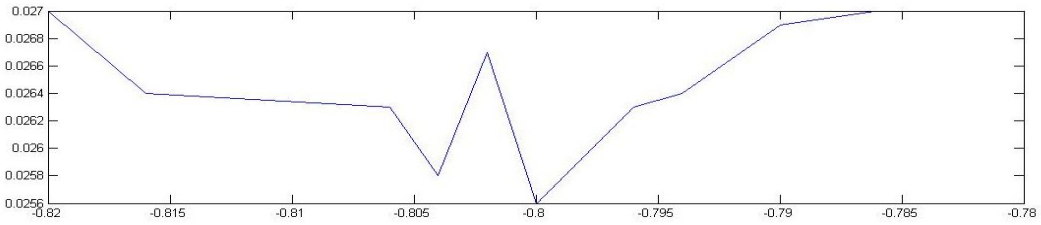


(b) RE (along y-axis) vs. change in fractional order (along x-axis) with T=2.

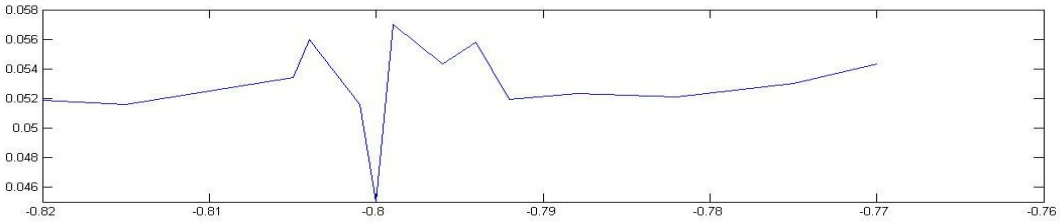
Figure-4.23: RE as a function of change in fractional order. (contd.)



(c) RE (along y-axis) vs. change in fractional order (along x-axis) with T=4.



(d) RE (along y-axis) vs. change in fractional order (along x-axis) with T=8.



(e) RE (along y-axis) vs. change in fractional order (along x-axis) with T=16.

Figure-4.23: RE as a function of change in fractional order.

It has been observed from Figure 4.23; at correct fractional order with value 0.8 the relative error is minimum for all values of T. The corresponding deviations $\Delta\alpha$ in fractional order are 0.016, 0.008, 0.004, 0.002 and 0.001 for T= 1, 2, 4, 8, and 16, respectively. With T increases, the decryption procedure becomes sensitive to change of fractional orders and relative error also increases rapidly. When RE > 0.2, the user failed to distinguish the decrypted image with the naked eye [122]. The sensitivity to fractional orders can be accustomed by alteration in number of iterations. Thus one can adjust the accuracy of right resumption with slight difference in fractional order and the difficulty of brute force breaking attempts.

In case of T = 4, and deviation in fractional order 0.004, the total possible number of searches will be around 3.9×10^{21} [122], this is an extremely large number

for an unauthorized person who tries to access the encrypted image. In our simulations, the original image is divided into 1024 subsections. The security strength of iterative discrete fractional transforms encryption without scrambling will be $15.6 \times 10^{21} \times 2^{256\times256\times4}$ [122]. After introducing scrambling operation, the security strength of proposed method will be $15.6 \times 10^{21}\times2^{256\times256\times4} \times (1024!)^4$, enlarged $(1024!)^4$ times [207]. This is an extremely large number for an eavesdropper to search correct fractional order keys by scanning through all possible combinations. Dividing the image into more number of subsections will further enhance the security strength.

4.5.2.2 Interference of noise

Security to algorithm is also granted by multiple encryption keys in algorithm. In cryptography, the plaintext is the original information to be encrypted. The ciphertext is the encrypted plaintext. The ciphertext-only attack is an attack model in which an attacker tries to deduce the security keys by only studying the ciphertext [64]. This attack can be used to recover the original image data by studying the encrypted images. If fewer portions of the images are encrypted, more portions of the original images can be recovered by an attacker without knowing the encryption algorithm and its security keys. An encryption scheme has an extremely low security level if it cannot withstand this attack. The experiments results that the encrypted images are totally unrecognizable and different from original image.

4.6 SUMMARY

In this chapter, the algorithm for image encryption using fractional transforms has been executed. The change in PSNR with variation in number of keys has been observed. It has been concluded that the fractional Cosine transforms are better. The sensitivity of keys has proven that at correct fractional order MSE is low. An algorithm to enhance the security of images with encryption and scrambling is also devised. The image encryption with scrambling has shown less relative error with various iterations. The presented algorithm is also secured from noise attacks.

The resulting better DFrFT for compression and DFrCT for encryption are utilized to develop joint image encryption-compression and compression-encryption algorithms in the next chapter.

CHAPTER 5 JOINT IMAGE COMPRESSION-ENCRYPTION

The joint algorithms have been analysed to reduce memory space and to provide privacy. The joint image encryption-compression and compression-encryption algorithm have been performed using fractional transforms. To evaluate the performance of algorithms quality metrics PSNR and MSE have been calculated.

5.1 INTRODUCTION

The demand of joint image algorithms is increasing due to growing usage of multimedia data in internet applications. The issues related to the multimedia data transmission have been studied for a long time, but still there are several open issues like limited channel bandwidth, demand for secure access of multimedia based applications. Compression of image data in the past aims at minimizing the number of bits is required to represent. Compression of images for efficient use of storage space and transmission bit rate has become a necessity. Image compression using DCT [105], SPIHT [97] etc. provides low bit rate representation by preserving a high visual quality of decompressed image. Along with compression, the privacy of transmitted image data is also of utmost concern. Several techniques are developed in the last ten years for encryption using fractional Mellin transform [113] based on phase retrieval algorithm [117] and various other methods. Several transforms are implemented in the past decade in optical processing [123-125]. Because of the importance of transforms, its implementation in signal processing in joint form has become an important research area. In 2012, Pande *et al.* [208] proposed a method for securing multimedia contents using joint compression and encryption. This approach lessen the need of additional hardware for encryption in resource constrained scenario, and can be otherwise used to augment existing encryption methods used for content delivery in Internet or other applications. Sharma *et al.* [145] have proposed need and advantages of integrated approach for compression and encryption. The need of compression,

The outcome of this chapter has been **accepted** in Research Journal as per following detail: N. Jindal, K.Singh, Joint Image Compression-encryption using Discrete Fractional Transforms, Imaging Science Journal.

encryption individually is also discussed. The order of joining the two techniques is analysed and it has been observed that compression before encryption is much better.

The motivation of research for joint compression-encryption (C-E) using fractional transforms is from [209] which joined discrete wavelet transform for compression and block cipher data encryption standard for image encryption. It has been proved that joint effort improves transmission rate and security [209].

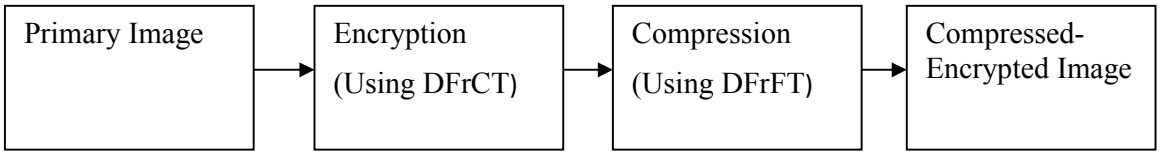
5.2. MOTIVATION FOR JOINT ALGORITHM

The compression and encryption serve two different purposes in computing. The compression algorithm requires a key to be decompressed impressively and hence it gives the impression like a subtle form of encryption. If two processes are different with similar methods then there are still development opportunities to connect these fields.

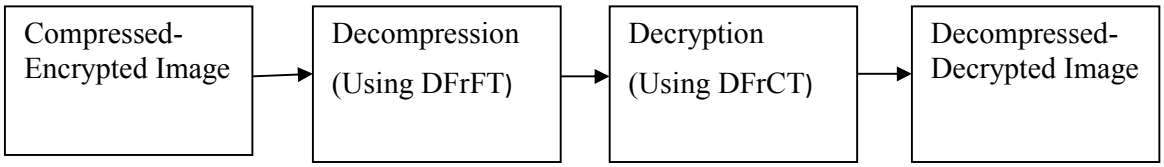
The data can be compressed if necessary, and then encrypted. The problem is that, with the rapid progress in computing technology, the uncompressed encrypted data will not be secured any more after a few years. Once the data are decrypted, all secret will be leaked. It is more difficult for the intruder to know that along with the encryption, the data is compressed or not. The compressed data creates further difficulty for intruder. So, it was concluded that these algorithms should be joined for better results.

5.3 JOINT IMAGE ENCRYPTION-COMPRESSION ALGORITHM

The joint image encryption-compression (E-C) algorithm encrypts the image using DFrCT. The images are encrypted with three fractional keys. Then encrypted images are compressed using DFrFT. Figure 5.1 shows joint image encryption-compression encoder and decoder. The primary image is encrypted with three fractional keys using DFrCT. The encrypted image is then compressed using DFrFT. The encryption is performed using DFrCT and compression using DFrFT as observed from the results of chapter 3. The compressed-encrypted image is given to decoder. The decompression is performed with DFrFT. The decompressed image is decrypted using DFrCT at the output.



(a)



(b)

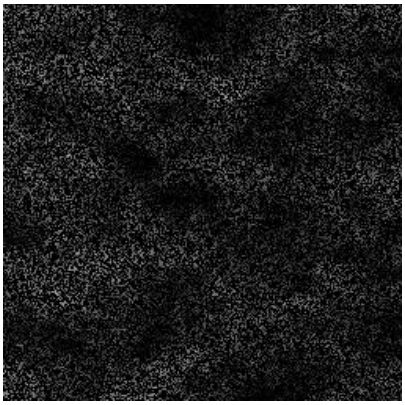
Figure-5.1: Encryption-compression using fractional transforms (a) encoder and (b) decoder.

5.3.1 Simulation Results

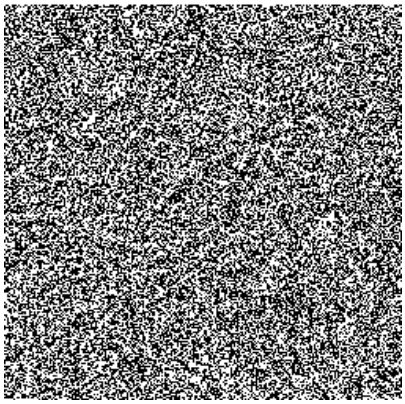
The simulation results are performed on images Pyramid, Pentagon, girl, Baboon, Boat, Flower, House, Peppers, Lena and Barbara images to execute the algorithm. The original images are of size 256 × 256 pixels. The images can be compressed at various compression percentages of 10%, 20%, 30%, 40%, 50% and 75% and then encrypted. Figure 5.2 shows the results of Girl image at 10% compression percentage. The Encrypted image is shown in Figure 5.2 (a). The image is then compressed using DFrFT and shown in Figure 5.2(c). Then the decompressed image is shown in Figure 5.2(d). Finally, the decompressed-decrypted image is shown in Figure 5.2 (e) with PSNR 18.9 dB. Figures 5.3 to 5.7 show the simulation results of Girl images at 20%, 30%, 40%, 50% and 75% respectively.

(a) Original Girl image

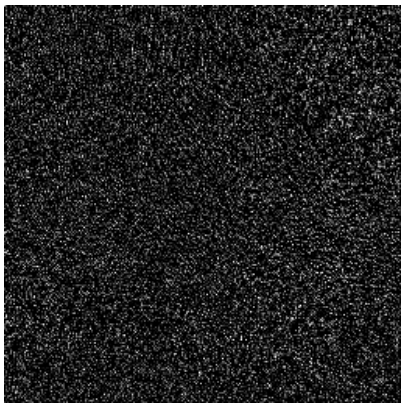
(b) Encrypted image



(c) Encrypted-compressed (at 10%) image



(d) Encrypted-decompressed image



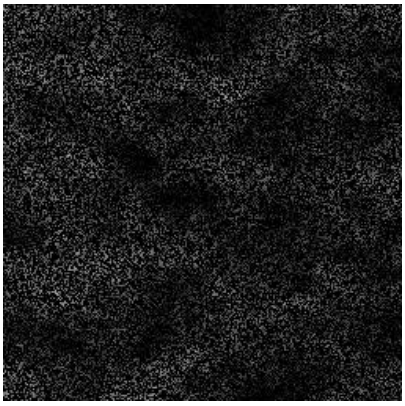
(e) Decrypted-decompressed image with wrong keys

(f) Decrypted-decompressed image with right keys PSNR=18.9 dB

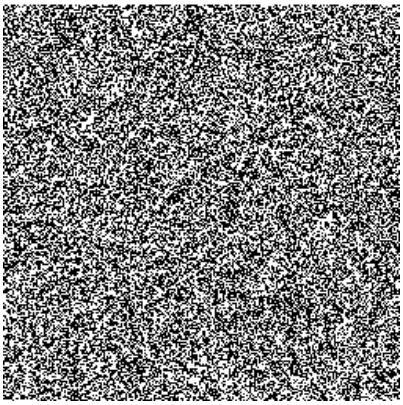
Figure-5.2: Simulation results of Girl image at compression percentage 10%.

(a) Original Girl image

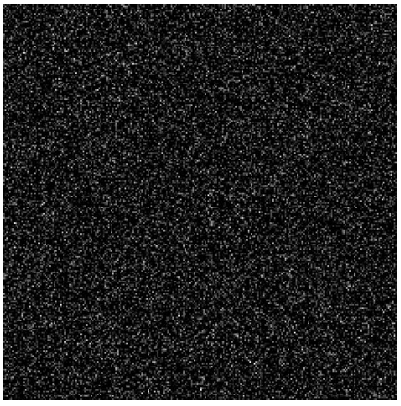
(b) Encrypted image



(c) Encrypted-compressed (at 20%)



(d) Encrypted-decompressed image



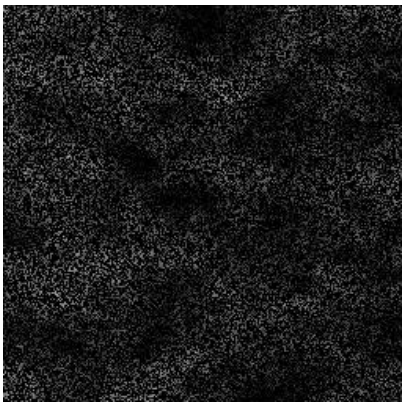
with wrong keys

)ecrypted-decompressed image with right keys PSNR=16.93 dB

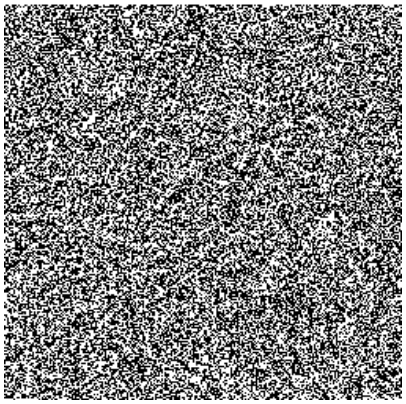
Figure-5.3: Simulation results of Girl image at compression percentage 20%.

(a) Original Girl image

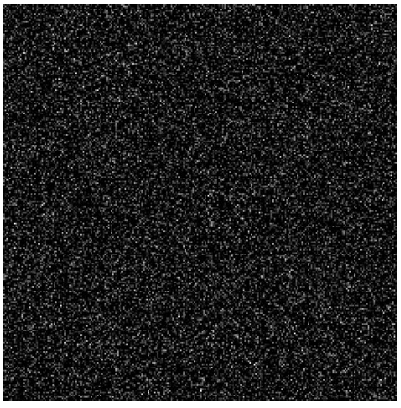
(b) Encrypted image



(c) Encrypted-compressed (at 30%)



(d) Encrypted-decompressed image



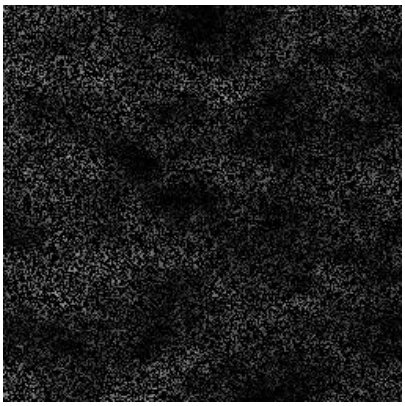
with wrong keys

ecrypted-decompressed image with right keys PSNR=16.8 dB

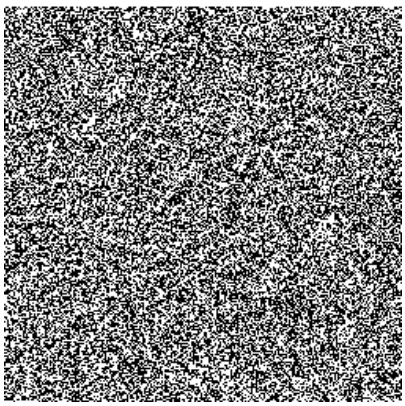
Figure-5.4: Simulation results of Girl image at compression percentage 30%.

(a) Original Girl image

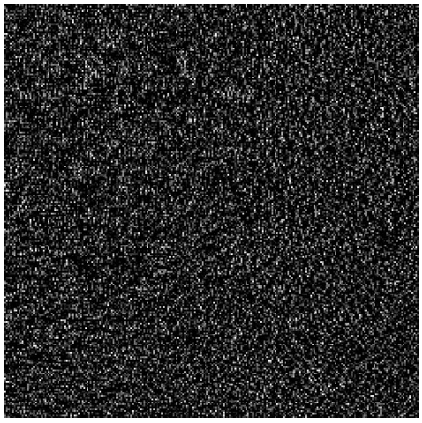
(b) Encrypted image



(c) Encrypted-compressed (at 40%)



(d) Encrypted-decompressed image



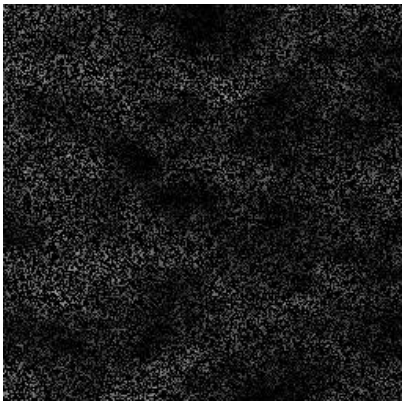
with wrong keys

ecrypted-decompressed image with right keys PSNR=17.09 dB

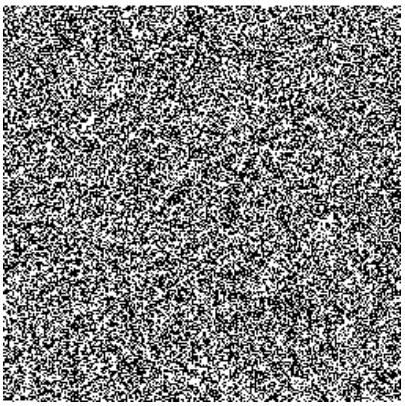
Figure-5.5: Simulation results of Girl image at compression percentage 40%.

(a) Original Girl image

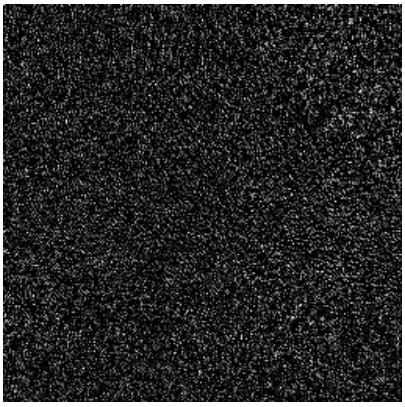
(b) Encrypted image



(c) Encrypted-compressed (at 50%) image



(d) Encrypted-decompressed image



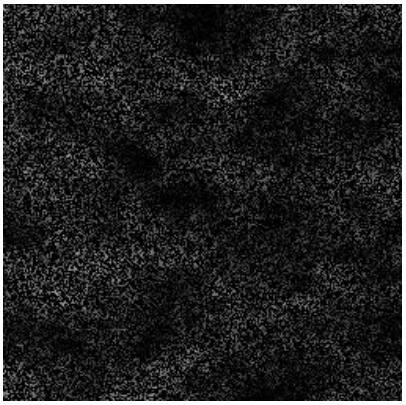
(e) Decrypted-decompressed image with wrong keys

(f) Decrypted-decompressed image with right keys PSNR=16.7 dB

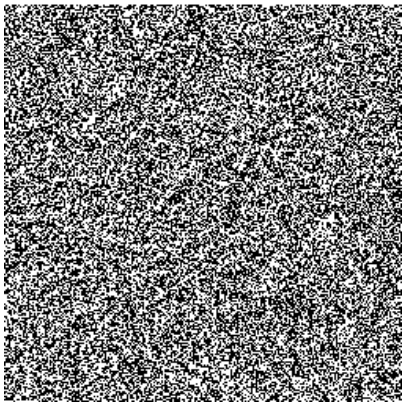
Figure-5.6: Simulation results of Girl image at compression percentage 50%.

(a) Original Girl image

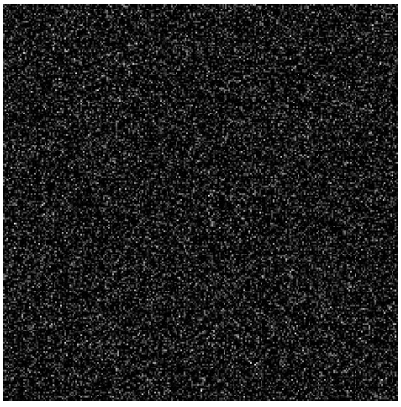
(b) Encrypted image



(c) Encrypted- compressed (at 75%)



(d) Encrypted-decompressed image



wrong keys

Decrypted-decompressed image with right keys PSNR=17.01 dB

Figure-5.7: Simulation results of Girl image at compression percentage 75%.

5.4 JOINT COMPRESSION-ENCRYPTION ALGORITHM

The test images have been compressed using DFrFT and encrypted using DFrCT. To compress the images using DFrFT the fractional orders are optimized at particular compression percentage. Then the compressed image is encrypted using DFrCT. The presented joint approach used for image compression-encryption has several benefits in image communication. It has been concluded that joint compression and encryption algorithms are in general more efficient than compression independent ones [209]. The former add very little computational complexity but provide good results. Figure 5.8 describes the joint model of compression-encryption method using discrete fractional transform.

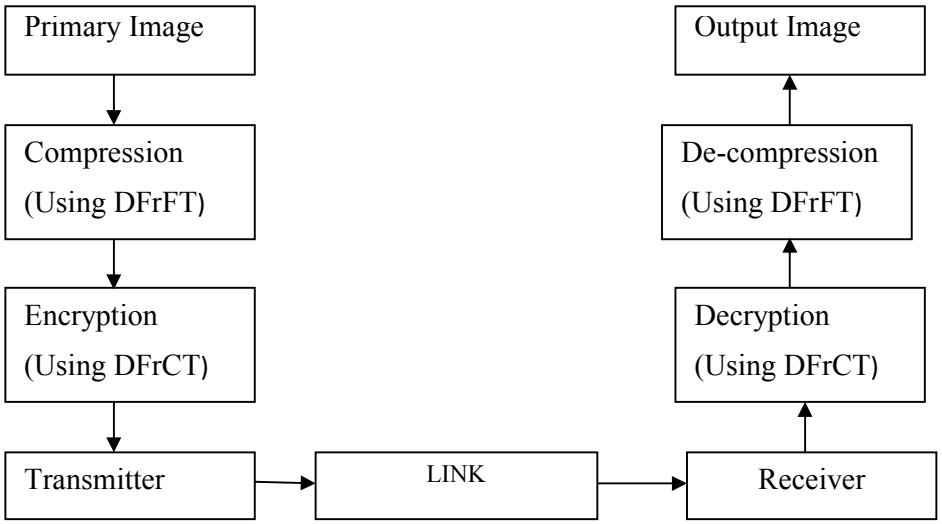


Figure 5.8: Joint model for compression-encryption using Fractional transforms.

The primary image at sending side is compressed using DFrFT at necessitate compression percentage. The fractional order is optimized for required compression percentage. Then compressed image is encrypted using DFrCT. At the other end, the image is decrypted - decompressed using same discrete fractional transform. However, now inverse fractional orders are used. The compression and encryption

keys both are sensitive in algorithm. The change or wrong key will produce incorrect results and effect the mean square error also.

5.4.1 Simulation Results

The simulation results for Girl image are shown in Figures 5.9. Figure 5.9(a) is the original Girl image of size 256 × 256. The Figure 5.9(b) is compressed image at 10% with the DFrFT.

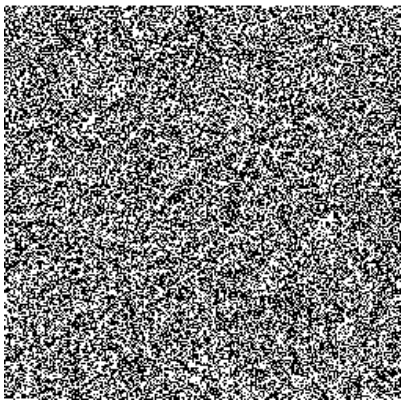
Then Figure 5.9(c) is the compressed-encrypted image. This compressed image is encrypted with DFrCT to provide security. Three fractional orders are used as fractional keys to encrypt the image. The length of fractional keys can be enhanced as the requirement of application. The present algorithm has used three keys each of 32 bit. After that, at the receiving end, with inverse fractional keys, image is processed to get original image. Firstly, the image is decrypted and then decompressed.

The images with wrong fractional keys are shown in Figure 5.9(d). Among of the three keys, each key should be correct. Even with single wrong key, it is not possible to decrypt the image. The PSNR is also changed with the change in single key. In figure 5.9 (e), the original image is obtained.

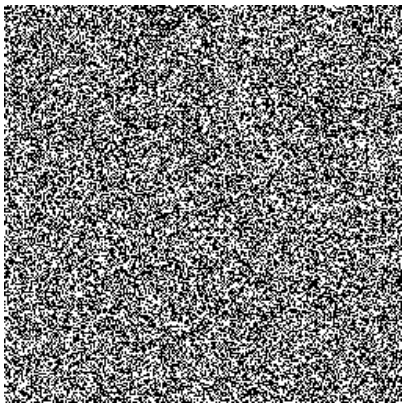
The simulation results at various compression percentages are also shown in Figures 5.9 to 5.14.

(a) Original Girl image

(b) Compressed image (at 10%)



(c) Compressed-encrypted image



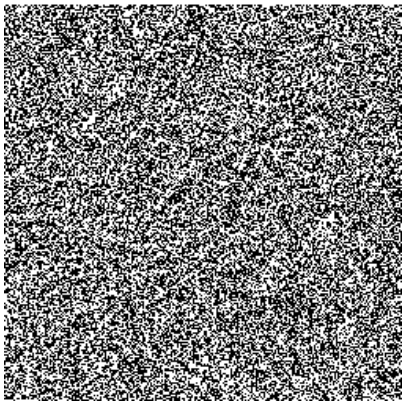
(d) Decrypted-decompressed image (with wrong key)

(e) Decrypted-decompressed image (with right keys) with PSNR=33.1 dB

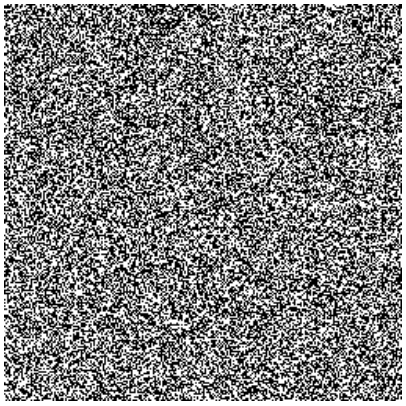
Figure-5.9: Simulation results of Girl image at compression percentage 10%.

(a) Original Girl image

(b) Compressed image (at 20%)



(c) Compressed-encrypted image



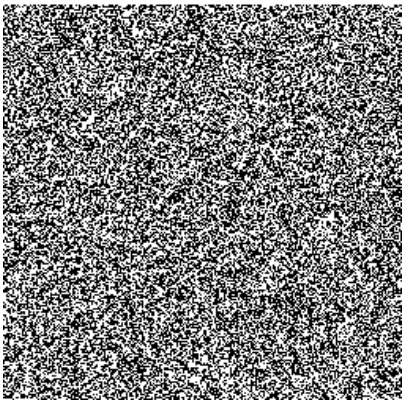
(d) Decrypted-decompressed image (with wrong key)

(e) Decrypted-decompressed image (with right keys) with PSNR=33.0 dB

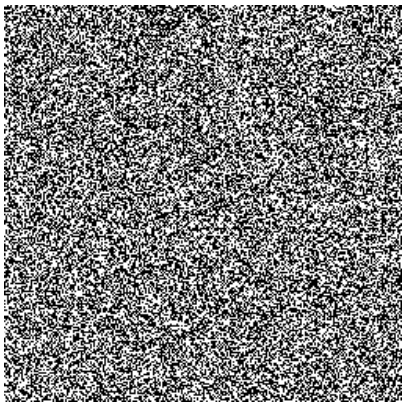
Figure-5.10: Simulation results of Girl image at compression percentage 20%.

(a) Original Girl image

(b) Compressed image (at 30%)



(c) Compressed-encrypted image



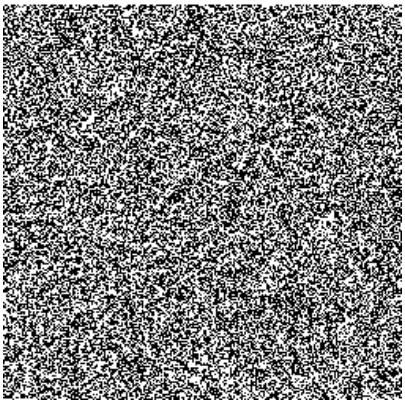
(d) Decrypted-decompressed image (with wrong key)

(e) Decrypted-decompressed image (with right keys) with PSNR=32.79 dB

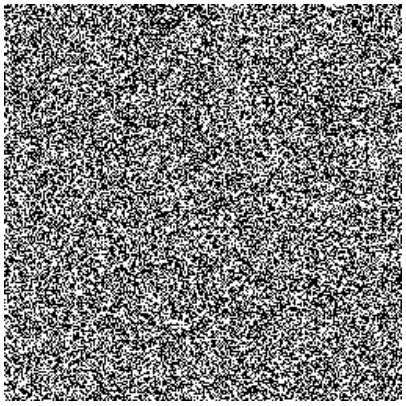
Figure-5.11: Simulation results of Girl image at compression percentage 30%.

(a) Original Girl image

(b) Compressed image (at 40%)



(c) Compressed-encrypted image



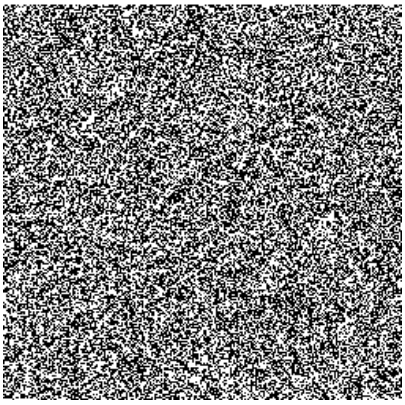
(d) Decrypted-decompressed image (with wrong key)

(e) Decrypted-decompressed image (with right keys) with PSNR=32.50 dB

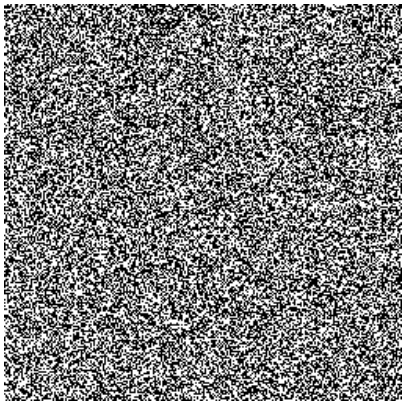
Figure-5.12: Simulation results of Girl image at compression percentage 40%.

(a) Original Girl image

(b) Compressed image (at 50%)



(c) Compressed-encrypted image



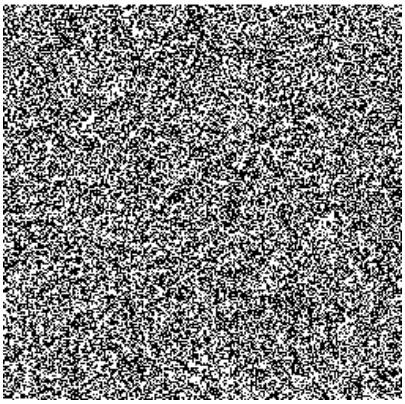
(d) Decrypted-decompressed image (with wrong key)

(e) Decrypted-decompressed image (with right keys) with PSNR=32.34 dB

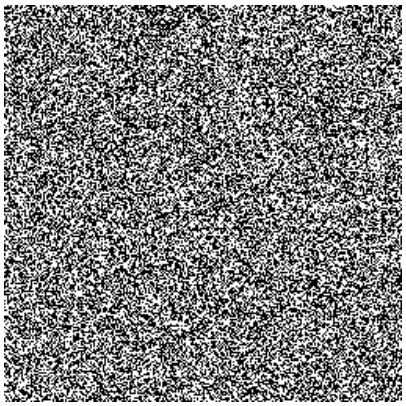
Figure-5.13: Simulation results of Girl image at compression percentage 50%.

(a) Original Girl image

(b) Compressed image (at 75%)



(c) Compressed-encrypted image



(d) Decrypted-decompressed image (with wrong key)

(e) Decrypted-decompressed image (with right keys) with PSNR=30.07 dB

Figure-5.14: Simulation results of Girl image at compression percentage 75%.

5.4.2 Comparison of Joint Algorithms

Two joint algorithms compression-encryption and encryption-compression have been compared. The results are calculated for ten images at various compression percentages. Table 5.1 gives the comparison of two algorithms with the PSNR.

The PSNR using encryption-compression algorithm is between 21 dB to 27 dB. It has been observed from the results that PSNR has been changed for various compression percentages. The PSNR decreases as compression percentage has increased. The important interpretation is that PSNR is minimum at 75% compression percentage.

The compression-encryption algorithm has the PSNR between 27 dB to 33 dB. It has been observed that the change in PSNR from 10% to 50% compression percentage is very less. The value of PSNR is reduced as the image compression percentage is increased.

The comparison of two algorithms interpreted that PSNR values are better in compression-encryption at all compression percentages.

The proposed compression-encryption algorithm at compression percentages of 2.5%, 16% and 28% has mean square errors 0.0000, 0.02 and 0.063 respectively. Alfalou *et.al* [100] calculated MSE 0.0289, 0.08 and 0.122 respectively. So, an improvement in MSE 0.0289, 0.078 and 0.059 respectively was obtained.

The main intention of two joint algorithms is to provide security and saving of memory storage. So, the two algorithms can be used according to the requirement of applications. The security aspects of algorithms have been discussed in the next section.

Table 5.1: Comparison of PSNR (dB) for joint compression-encryption and encryption-compression algorithms.

Compression Percentage	Pyramid		Pentagon		Girl		Lena		Baboon	
	(C-E) PSNR (dB)	(E-C) PSNR (dB)	(C-E) PSNR (dB)	(E-C) PSNR (dB)	(C-E) PSNR (dB)	(E-C) PSNR (dB)	(C-E) PSNR (dB)	(E-C) PSNR (dB)	(C-E) PSNR (dB)	(E-C) PSNR (dB)
10%	33.73	22.50	31.17	24.16	33.10	18.90	31.29	24.20	35.06	23.43
20%	33.40	22.40	30.44	24.00	33.00	16.93	31.45	23.20	34.77	23.08
30%	33.90	22.30	30.30	23.90	32.79	16.80	32.08	23.10	33.79	23.04
40%	33.10	22.08	30.15	23.79	32.50	17.09	31.61	23.57	33.42	23.02
50%	33.20	22.02	30.11	23.74	32.34	16.70	31.23	24.40	32.30	22.94
75%	32.14	21.12	28.38	23.10	30.07	17.01	30.06	23.20	27.06	21.60

Compression Percentage	Boat		Flower		House		Barbara		Peppers	
	(C-E) PSNR (dB)	(E-C) PSNR (dB)	(C-E) PSNR (dB)	(E-C) PSNR (dB)	(C-E) PSNR (dB)	(E-C) PSNR (dB)	(C-E) PSNR (dB)	(E-C) PSNR (dB)	(C-E) PSNR (dB)	(E-C) PSNR (dB)
10%	32.37	23.79	35.01	21.40	33.27	23.13	29.64	25.70	29.89	22.93
20%	31.83	23.55	34.36	21.32	32.57	22.71	29.91	25.87	29.70	22.85
30%	31.61	23.45	34.12	21.28	32.40	22.57	29.94	25.50	29.66	22.80
40%	31.59	23.34	34.01	21.23	32.10	22.50	29.53	26.04	29.61	22.75
50%	31.34	23.12	33.90	21.19	32.00	22.31	29.46	25.40	29.31	22.68
75%	29.18	22.97	31.17	20.70	31.47	21.10	27.37	25.02	29.35	22.10

5.5 SUMMARY

The joint image encryption-compression and compression-encryption algorithms using fractional transforms for several images have been analysed. It has been found from the comparison of two algorithms, that the compression-encryption algorithm gives better PSNR. The advantage of the compression-encryption algorithm is that compression reduced the size of data before encryption. The security of algorithm has been discussed using brute force attack. The brute force attack becomes infeasible due to number of keys provided by fractional orders. In the next chapter, the image processing work is extended to the video processing using fractional transforms.

CHAPTER 6 *VIDEO PROCESSING*

The rapid growth of video processing technologies has given the idea of video encryption techniques. The video compression-encryption algorithm has been developed in this chapter and better results are obtained than SCAN method. The video encryption algorithm using fractional transforms also provide better PSNR than existing method.

6.1 INTRODUCTION

In recent years there has been an incredible improvement and emergence of technologies for communications, coding and retrieval of digital images and videos. This environment has allowed for the realization of many interesting multimedia applications related to nearly all aspects of life. Almost instantaneous delivery of entertainment videos, pictures and music is available to everyone who is connected to a multimedia distribution system. Businesses and other organizations are now able to perform real time video conferencing even over a non-dedicated channel. These all applications necessitate less bandwidth and security as main issues [210]. So, compression and encryption are two important issues in video applications. The goal of data compression is to find a representation of the source data that is easier to store and that uses less bandwidth than the original representation, but can be reconstructed back to the original source data, exactly lossless compression or approximately lossy compression. The decompression techniques will restore the original videos when needed. The compression depends upon information contents, the compression technique used and the desired reconstruction quality. Compression is obtained by applying certain compression transformations to the original data [210]. The second issue is security that addresses the following objectives: confidentiality, authentication, data integrity and non-repudiation. In the past technique, researchers exemplify various techniques for security [50].

The outcome of this chapter has been published in Research Journal as per following detail: N. Jindal, K.Singh Image and Video Processing using Discrete Fractional Transforms, Signal Image and Video Processing, Springer 2012, DOI 10.1007/s11760-012-0391-4

The video encryption algorithm provides better results than the available method in literature. It provides improvement in PSNR 0.81dB for the Foreman video using DFrFT when compared with method [211] based on alternating transforms. The present chapter has also performed the video compression-encryption algorithm using fractional transforms. This algorithm provides better mean square error at various compression percentages than existing methods for the Claire and Trevor videos. The security performance aspect is also measured from noise attacks at the last of chapter.

6.2 VIDEO ENCRYPTION TECHNIQUES

Not surprisingly, the earliest video encryption techniques were developed for analog video signals. While the research in the domain of analog video encryption still continues, the attention switched to little over a decade ago to the emerging area of digital video encryption [50]. Compared with the text communication, video communication is characterized by a number of peculiarities, such as large data size, real-time requirements, and the use of standardized video codecs, standardized data compression formats, and application-specific security requirements. These peculiarities raise a couple of specific requirements to video encryption algorithms [157], [160]. The demand for image and video encryption has triggered the development of several encryption standards.

6.2.1 Types of Video Encryption

In general, there are two basic research methodologies regarding the encryption of digital videos: selective encryption approaches and full encryption approaches. The full encryption approaches are designed to encrypt the entire video bit stream (raw or compressed), selective encryption approaches perform encryption only on certain and carefully selected parts of the video bit stream.

The idea of selective video encryption was introduced independently. This method was used mostly in that days due to its feature of simplicity, fast etc. But with the passage of time some common problems with selective encryption approach were the lack of standard methods for proper security evaluation. Many researchers found ways to exploit the information from the remaining unencrypted bits. In the partial encryption, if the entropy coding method is secretly modified , the method is insecure against several types of attacks since the size of Huffman tables are too small. These

methos may be targeted for speed purpose only. However the full encryption approach encrypt the entire video frame. This approach may be considered before compression or after compression .

6.2.2 Video Encryption using Fractional transforms

An encryption algorithm using Fractional transforms has been proposed. Figure 6.1 gives the block diagram of video encryption. The frames are extracted from the video signal. The frames are encrypted using three fractional keys. The three fractional keys are optimized between the fractional orders 0 to 1. The optimized fractional order is defined where the frame has maximum perceptual quality i.e. maximum PSNR and minimum MSE. The decrypted video can be obtained with right inverse fractional keys only. The decrypted frames are combined and video has been acquired. Several test videos of different frame size are considered for evaluation [19]. Figure 6.1 shows the block diagram and Figure 6.2 shows the test videos. Figures 6.3 to 6.11 are simulation results of Foreman, person and Claire video frames.

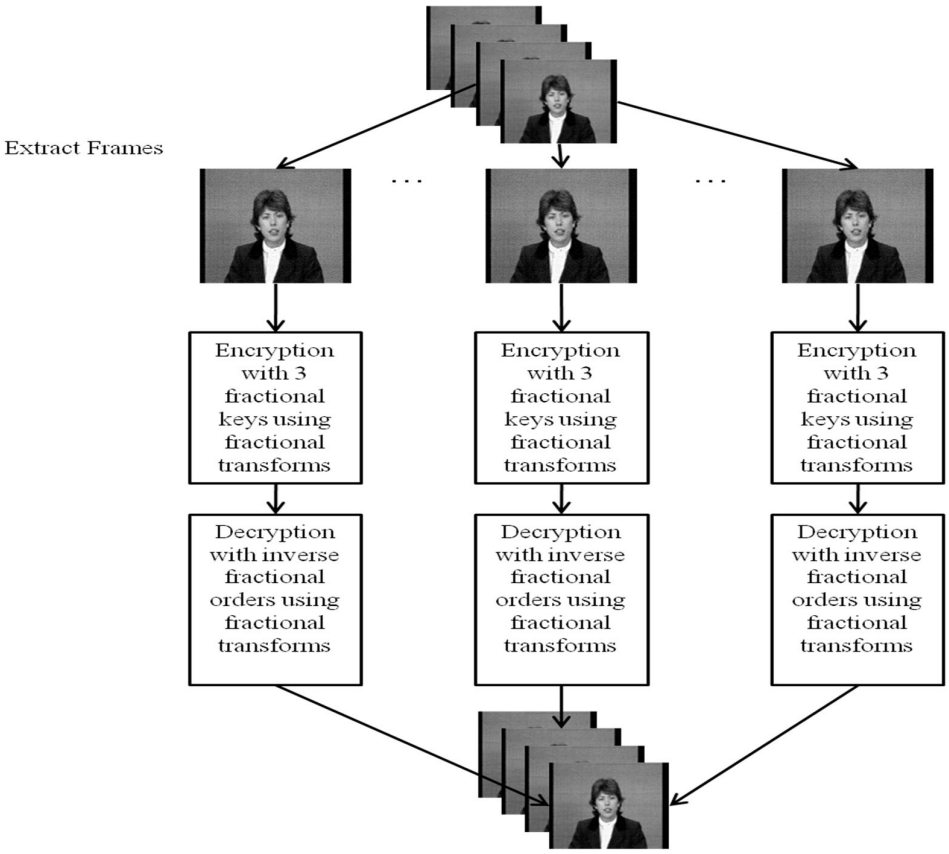
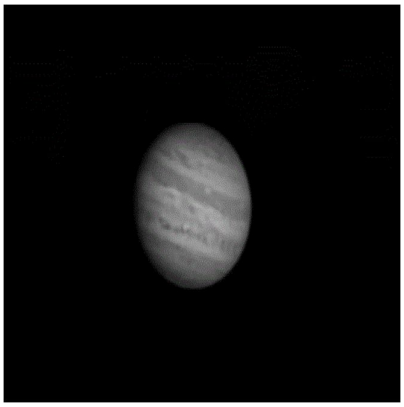


Figure-6.1: Block diagram of video encryption.



(a) Mars

(b) Person

(c) Claire.

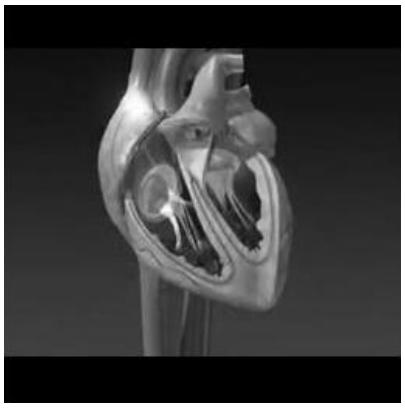
(d) Building.

(e) Foreman.

(f) Tennis.

Figure-6.2: Test videos. (contd.)

(g) Trevor.



(h) Heart.

Figure-6.2: Test Videos.

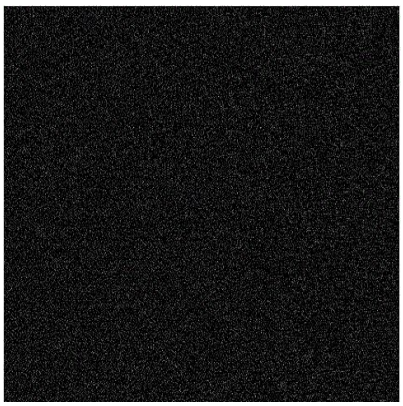
(a) Frame number 3.

(b) Frame number 25.

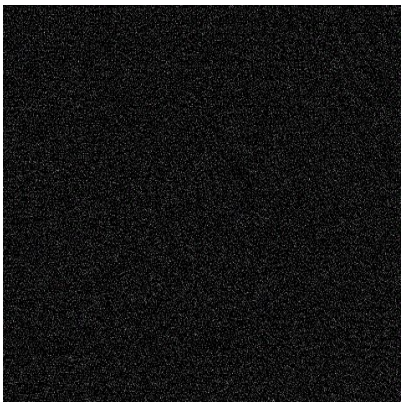
(c) Frame number 60.

(d) Frame number 91.

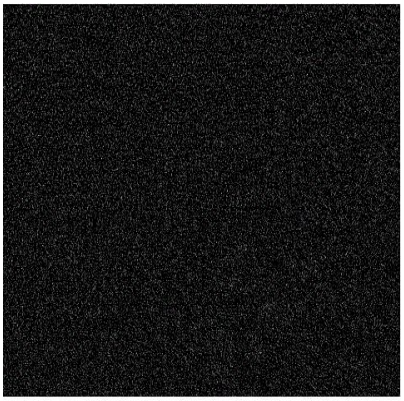
Figure-6.3: Frame number 3, 25,60 and 91 of Foreman.



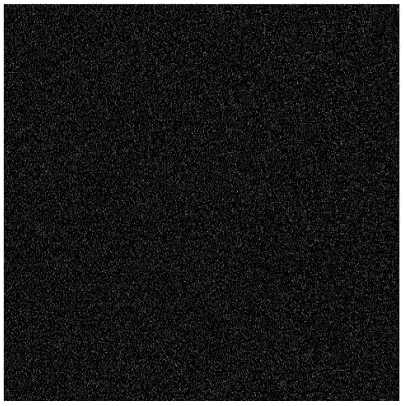
(a) Encrypted Frame number 3.



(b) Encrypted Frame number 25.

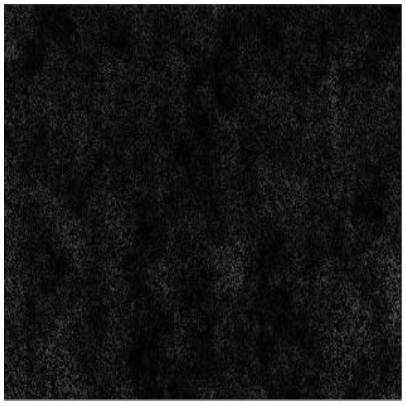


(c) Encrypted Frame number 60.



(d) Encrypted Frame number 91.

Figure-6.4: Encrypted video frames of their respective frames.

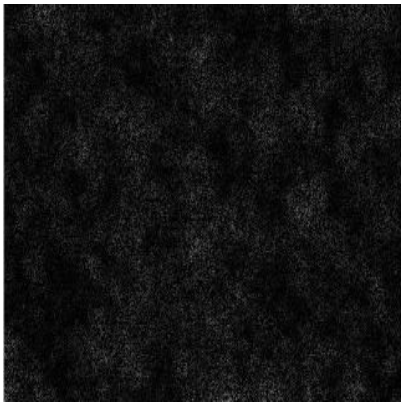


(a) Decrypted Frame number 3.

(with wrong keys).

(b) Decrypted Frame number 3.

(with right keys)

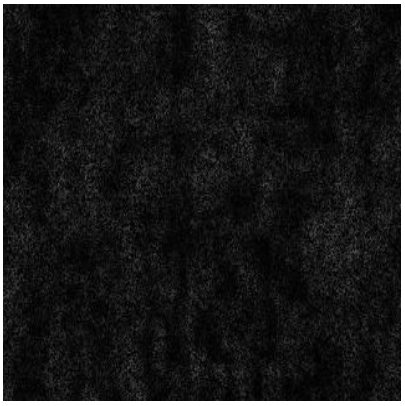


(c) Decrypted Frame number 25.

(with wrong keys).

(d) Decrypted Frame number 25.

(with right keys).



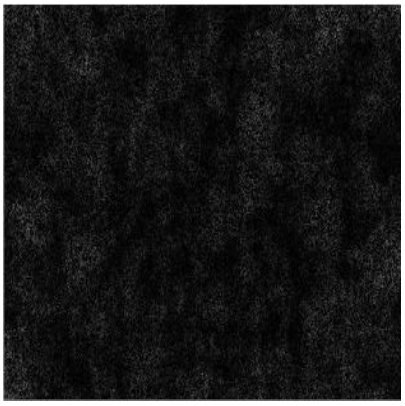
(e) Decrypted Frame number 60.

(with wrong keys).

(f) Decrypted Frame number 60.

(with right keys).

Figure-6.5: Decrypted video frames of their respective frames. (contd.)



(g) Decrypted Frame number 91. (with wrong keys).

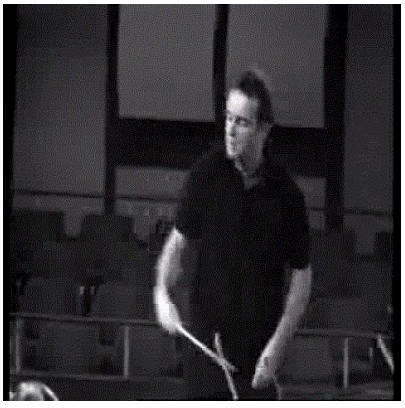
(h) Decrypted Frame number 60. (withwrong keys).

Figure-6.5: Decrypted video frames of their respective frames.

(a) Frame number 5.

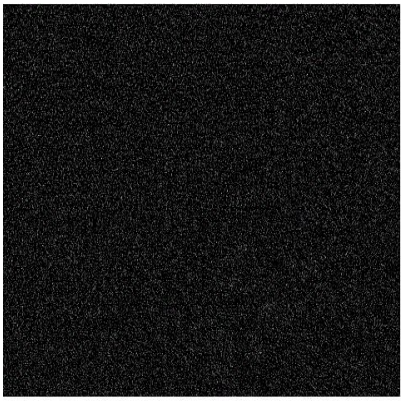
(b) Frame number 12.

(c) Encrypted Frame number 23.

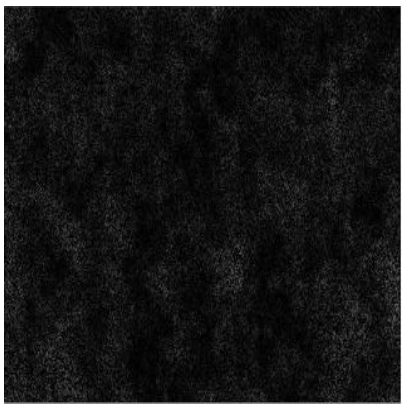


(d) Encrypted Frame number 27.

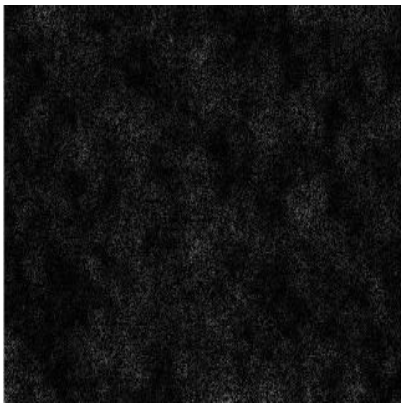
Figure-6.6: A sequence of four frames: note the motion of arms.



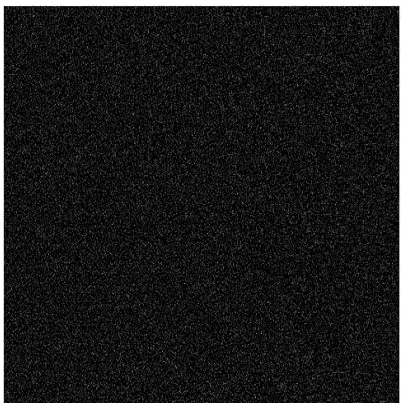
(a) Encrypted Frame number 5.



(b) Encrypted Frame number 12.

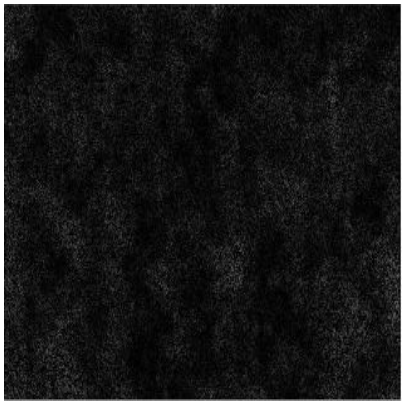


(c) Encrypted Frame number 23.



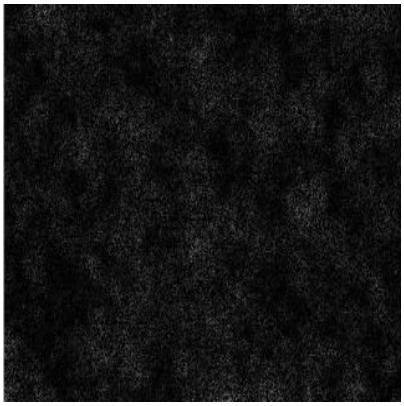
(d) Encrypted Frame number 27.

Figure-6.7: Encrypted Video Frames of their respective Frames.



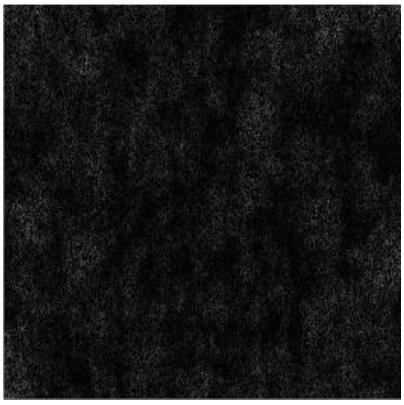
(a)Decrypted Frame number 5.
(with wrong keys).

(b)Decrypted Frame number 5.
(with right keys).



(c) Decrypted Frame number 12.
(with wrong keys).

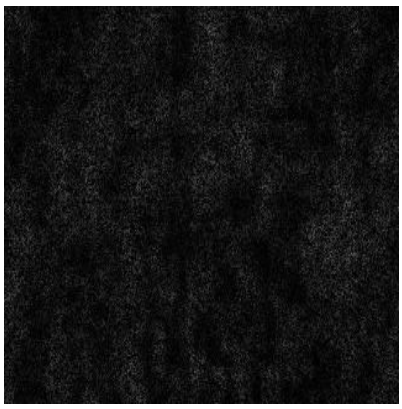
(d) Decrypted Frame number 12.
(with right keys).



(e) Decrypted Frame number 23.
(with wrong keys).

(f) Decrypted Frame number 23.
(with right keys).

Figure-6.8: Decrypted video frames of their respective frames. (contd.)



(g) Decrypted Frame number 27. (with wrong keys).

(h) Decrypted Frame number 27. (with right keys).

Figure-6.8: Decrypted video frames of their respective frames.

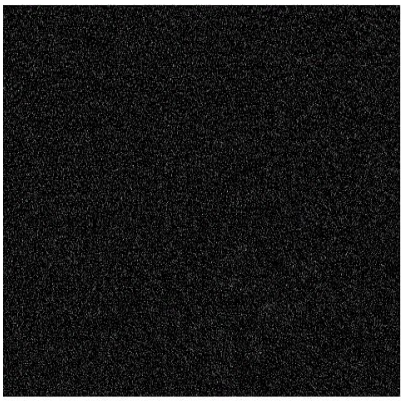
(a) Frame number 4.

(b) Frame number 22.

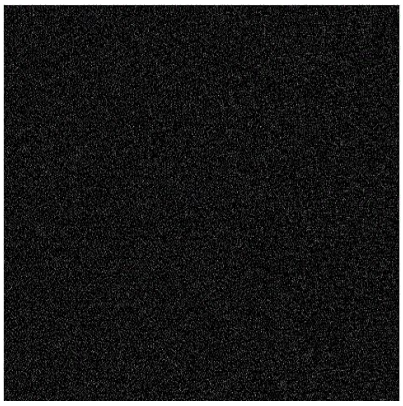
(c) Frame number 70.

(d) Frame number 96.

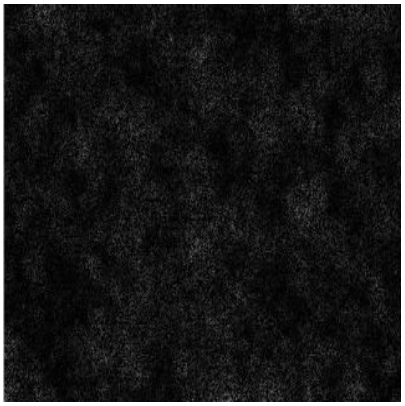
Figure-6.9: Frame number 4, 22,70 and 96 of Claire.



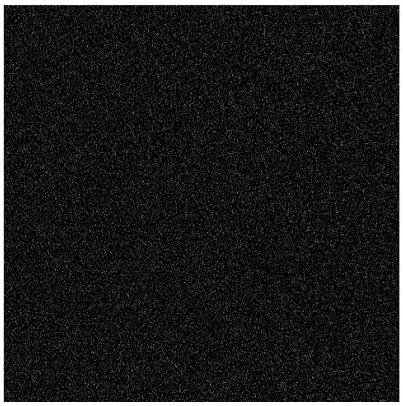
(a) Encrypted Frame number 4.



(b) Encrypted Frame number 22.

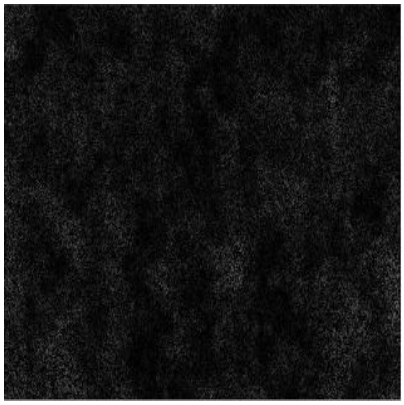


(c) Encrypted Frame number 70.



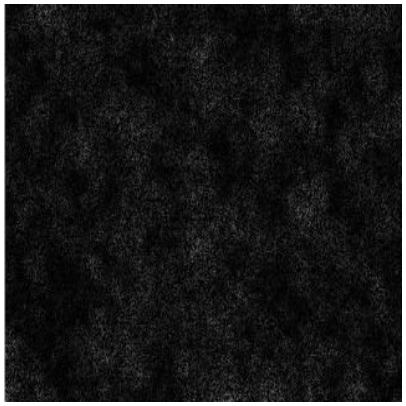
(d) Encrypted Frame number 96.

Figure-6.10: Encrypted video frames of their respective frames.



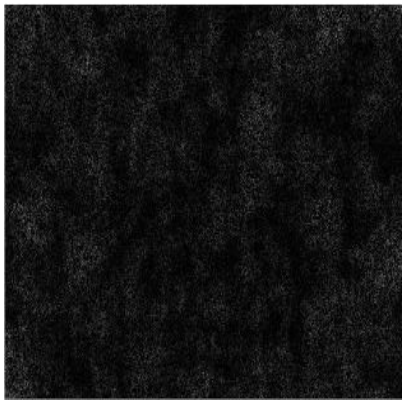
(a) Decrypted Frame number 4. (with wrong keys).

(b) Decrypted Frame number 4. (with right keys).



(c) Decrypted Frame number 22. (with wrong keys).

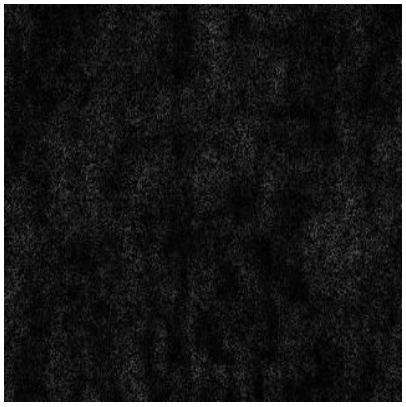
(d) Decrypted Frame number 22. (with right keys).



(e) Decrypted Frame number 70. (with wrong keys).

(f) Decrypted Frame number 70. (with right keys).

Figure-6.11: Decrypted video frames of respective frames. (contd.)



(g) Decrypted Frame number 96. (with wrong keys).

(h) Decrypted Frame number 96. (with right keys).

Figure-6.11: Decrypted video frames of respective frames.

Four Successive frames from a video clip are extracted and their encryption based on DFrCT or DFrFT is performed. At receving end, the authorised user will use decryption keys and get original frames. The PSNR value in decibels was used to measure the difference between decoded frame 'r' and original frame 'o' as shown in Table 6.1. The PSNR is better for the Foreman and Mars video using DFrFT as compared to DFrCT. The Foreman video PSNR using DFrFT is 0.81dB better than Yueng [211] algorithm based on alternating transforms. The Claire, Person and Building videos give better PSNR using DFrCT.

Table-6.1: MSE and PSNR (dB) for videos with DFrFT and DFrCT.

Test Video	Frame size (M × N)	With DFrFT		With DFrCT	
		MSE	PSNR (dB)	MSE	PSNR (dB)
Foremen	290 × 354	0.3399	52.81	0.4828	51.29
Claire	352 × 288	0.4114	51.90	0.3389	52.83
Mars	600 × 480	0.0135	66.80	0.0216	64.78
Person	480 × 360	0.0295	63.42	0.0005	90.63
Building	480 × 640	0.3228	53.04	0.0003	82.39

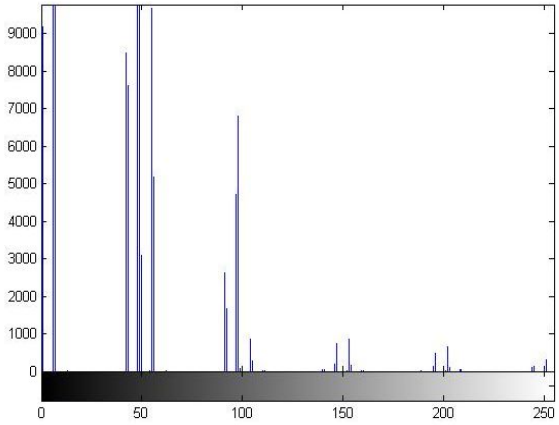
6.2.3 Performance Analysis of Video encryption techniques

This section analyzes the performance aspects of present algorithm.

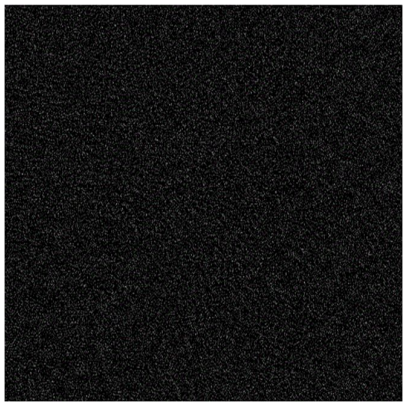
(a) ***Histogram Analysis***

There should be minimum statistical similarities between original frames and encrypted frames to avoid the leakage of information to attackers. The histogram analysis clarifies that, how the pixel values of a frame are distributed. Test video frames are encrypted with proposed algorithm using DFrCT and DFrFT and their histogram is analyzed as shown in Figure 6.12 and 6.13. It has been observed from Figure 6.12(b) and 6.13(b) that histogram components of dark images are concentrated on low side of gray scale and cover a broad range of gray scale. However, from Figure 6.12(d) and 6.13(d) it has been analysed that in high contrast images, the distribution of pixels is not too far from uniform, with very few vertical lines being much higher than others [189].

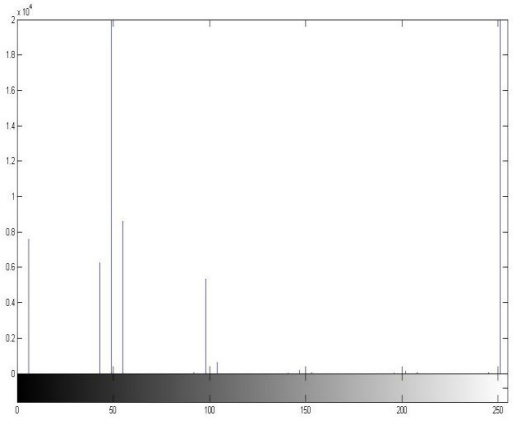
(a) Original frame of Person.



(b) Histogram.

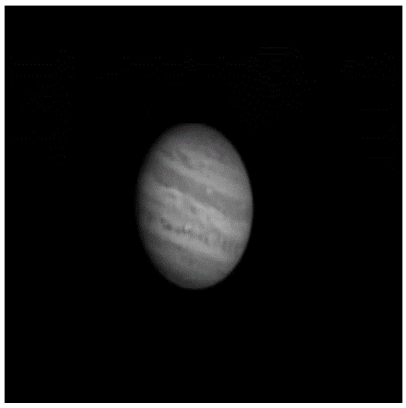


(c) Encrypted Person frame with DFrFT.

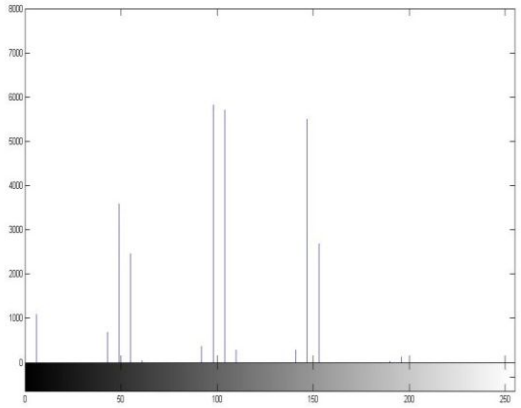


(d) Histogram of Encrypted frame.

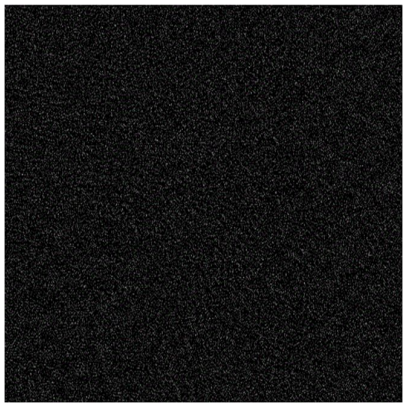
Figure-6.12: Histogram of original Person frame and encrypted frame.



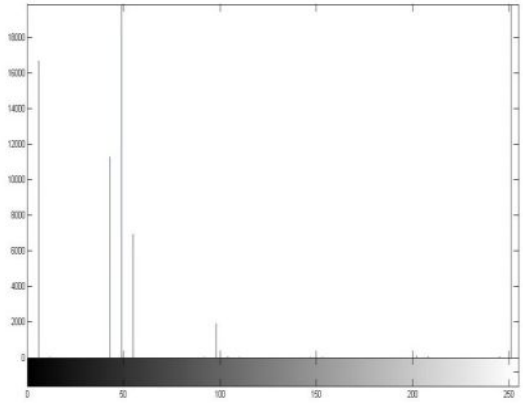
(a) Original frame of Mars.



(b) Histogram.



(c) Encrypted Mars frame with DFrCT .



(d) Histogram of Encrypted frame.

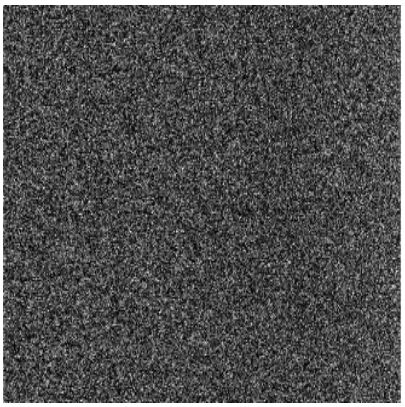
Figure-6.13: Histogram of original Mars frame and encrypted frame.

(b) ***Security***

In the proposed algorithm, the incoming video sequence is divided into frames. Further due to individual frame encryption, if a frame is corrupted or lost during transmission, it does not affect the decryption of other frames. Therefore, time is saved by avoiding iteration transmission of remaining frames due to single frame lost. The strong encryption algorithm must be capable of resisting attack salt-pepper noise in channel. The original Building and Heart video frame and its decrypted frame

at receiver after salt-pepper noise attack are shown in Figures 6.14, 6.15. The PSNR of retrieved frames at receiving end after attack of salt-pepper noise is shown in Table 6.2.

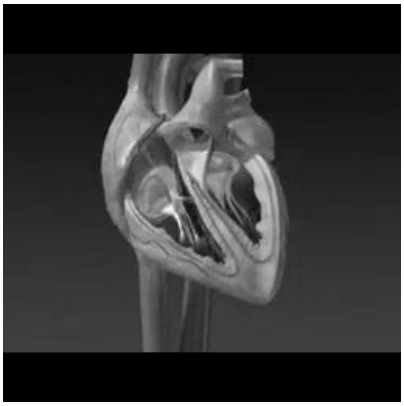
(a) Building Video Frame.



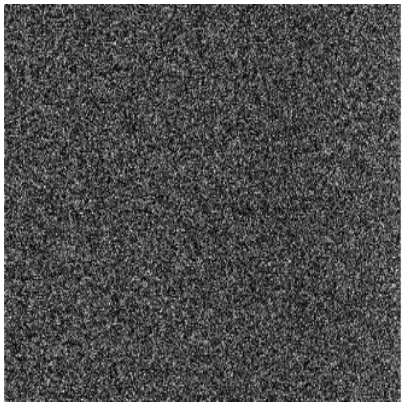
(b) Encrypted & attacked with Salt-Pepper Noise.

(c) Decrypted

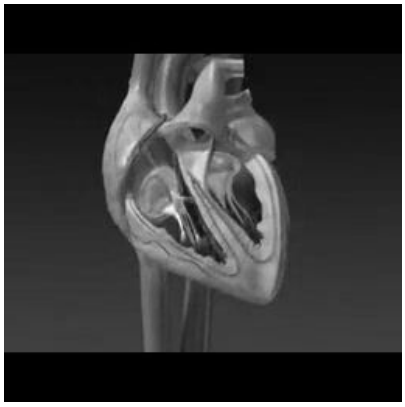
Figure-6.14: Salt-pepper noise attack on building video frames.



(a) Heart Video Frame.



(b) Encrypted & attacked with Salt-Pepper Noise.



(c) Decrypted

Figure-6.15: Salt-pepper noise attack on Heart Video frames.

Table-6.2: PSNR (dB) of frames after Salt-Pepper Noise Attack.

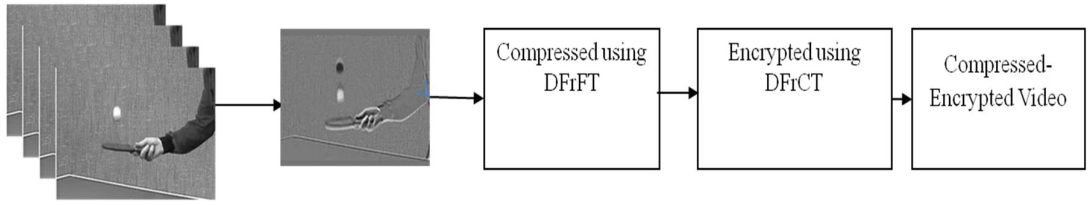
Test video	PSNR (dB) of frames after Salt- Pepper noise attack
Mars	41.06
Person	42.12
Building	31.59

6.3 VIDEO COMPRESSION-ENCRYPTION USING FRACTIONAL TRANSFORMS

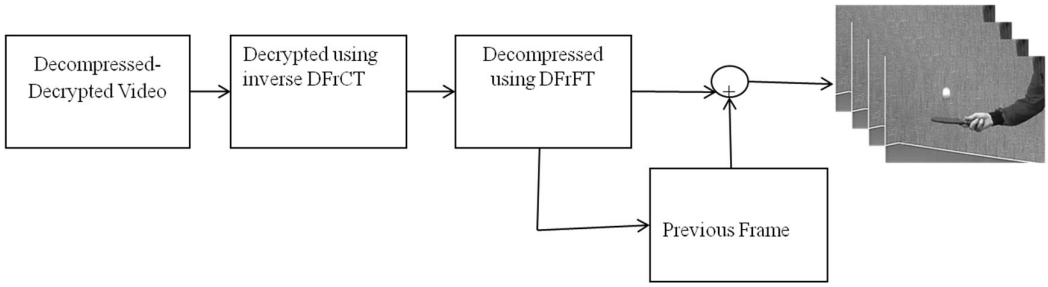
There are three kinds of areas suitable for video compression-encryption in literature. The first one encrypts video stream before the compressed code [212]. A study pointed out that such a method drastically amends the source structure and syntax [213], and the follow-up coding efficiency is exaggerated. The second one encrypts the compressed video stream after being compressed and coded [214]. The early video encryption techniques usually use the security strength of the traditional cryptographic algorithm such as DES, IDEA and RSA algorithm to meet the high security requirements. However, it has controlled some serious disadvantages such as high computation complexity and changed video formats. The third one combines encryption with compression. It partially encrypts video data [152]. In the present algorithm, the video is compressed-encrypted and the temporal difference between adjacent frames is calculated. Comparative mean square error results prove that the present method endows with better video quality at the receiving end.

In general, these algorithms encrypt the key data of video sequence that has great significance for the video reconstruction such as intra-prediction mode, motion vector difference, adjacent frame difference and quantization coefficients, block-matching motion compensation [215] using transforms [211], [216].

In the algorithm, the redundancy of adjacent frames was exploited and difference frame was obtained. Video compression-encryption using fractional transforms is implemented. The encoder and decoder for discrete fractional transforms are shown in Figure 6.16. The video was partitioned into video frames [152].



(a)



(b)

Figure-6.16: Video compression-encryption using Fractional transforms (a) encoder (b) decoder.

The difference frame was compressed using Fractional Fourier transform. The frame could be divided into sub-blocks of sizes $n \partial n$ where n = 8, 16, 32. The smallest sub-block size increases compression and computational complexity. The proposed algorithm had used 8 × 8 sub-block size. The various compression percentages were implemented on algorithm. Then, DFrFT was performed at particular compression percentages. The fractional orders were kept same in the work. Optimization of fractional orders was decided by varying fractional order value between 0 and 1. The value at which PSNR reached maximum was selected as optimum value. The optimum value depends upon the video sequence and the compression percentages. The final step in compression process was to quantize the transformed coefficients and run length coding. The decoding end used divergent processes with same inverse fractional order keys. The block diagram for compression using DFrFT is shown in Figure 6.17.

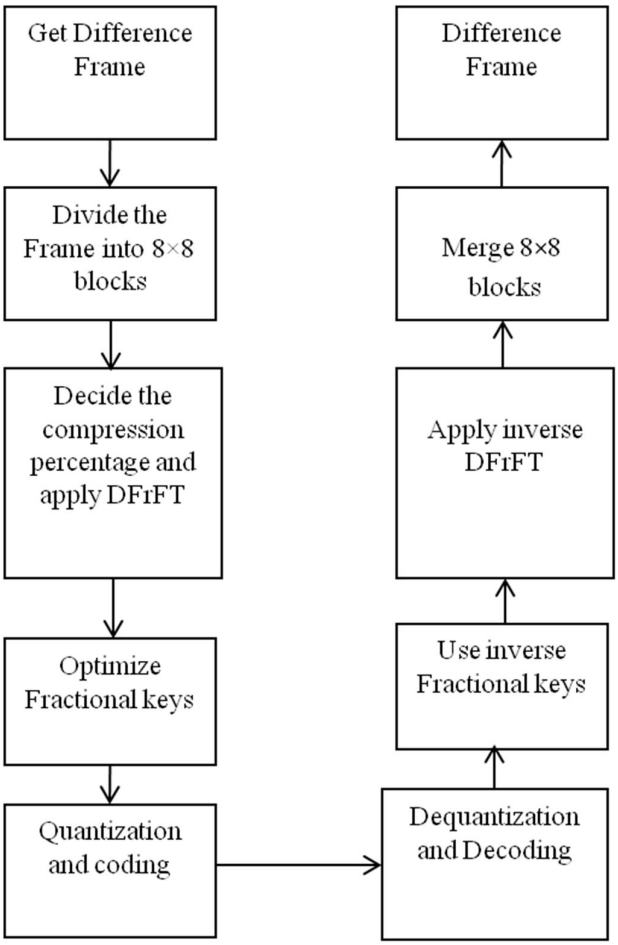


Figure-6.17: Compression using DFrFT.

The compressed frame was encrypted with three fractional keys using DFrCT. The decryption process was the reverse operation with respect to the encryption with inverse fractional keys and complex conjugate of random phase masks.

6.3.1 Simulation Results

The simulation results are evaluated with the video sequences of different frame sizes. The adjacent frames of test video (Tennis) and their difference frames are shown in Figures 6.18 and 6.19. The Figure 6.20 shows compressed and encrypted frames from experiment results of proposed algorithm. Recuperate video frames at the receiving end are shown in Figure 6.21. The simulation results for the Trevor video are shown in Figures 6.22, 6.23, 6.24 and 6.25.

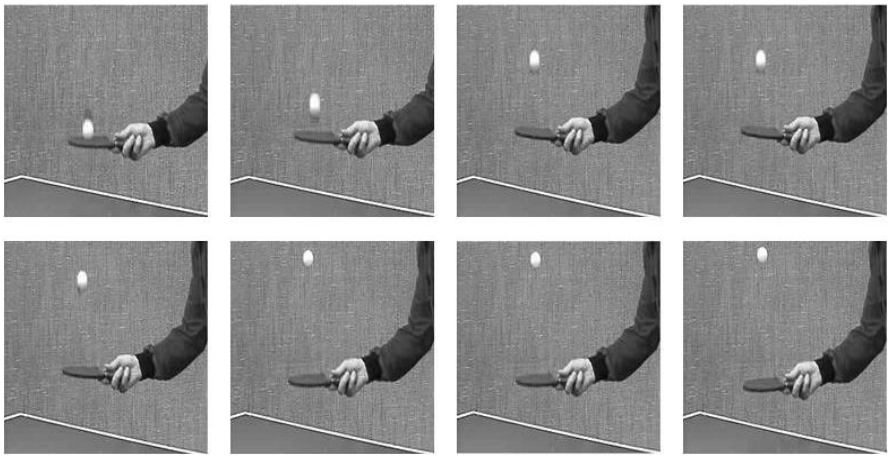


Figure-6.18: Frame numbers 1, 4, 8, 13, 17, 20, 23 and 25: Note the motion of Tennis ball.

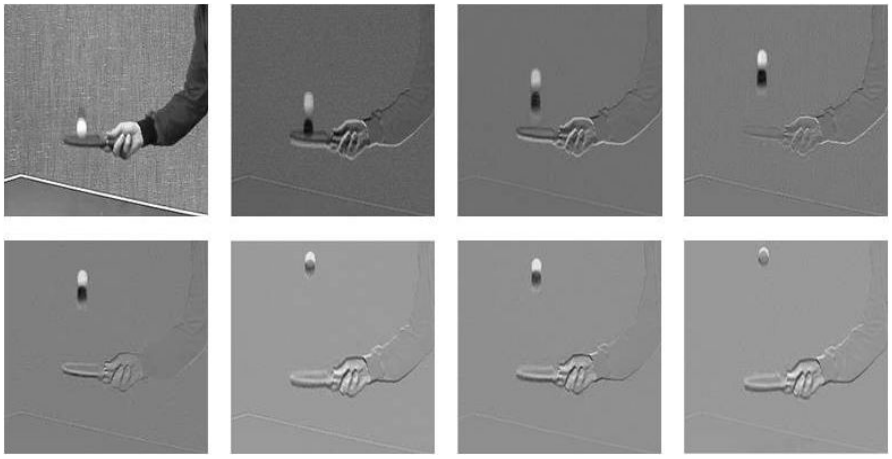


Figure-6.19: Adjacent frame differences.

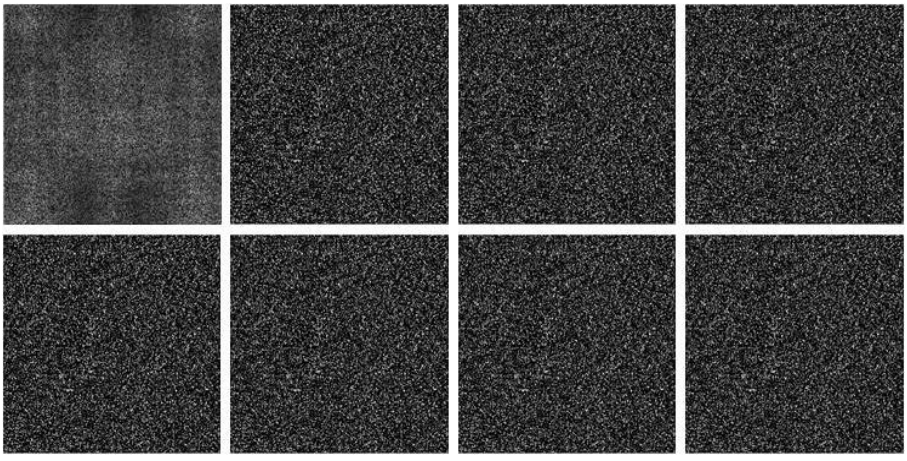


Figure-6.20: Compressed and encrypted frames differences.

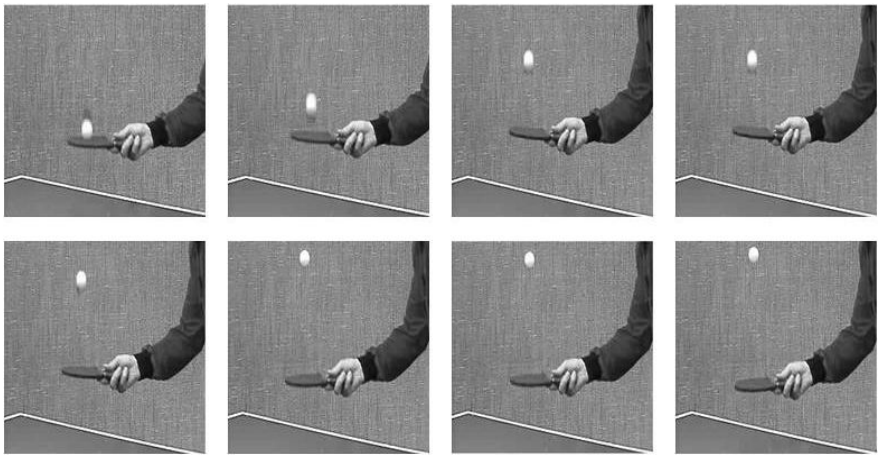


Figure-6.21: Recuperate video frame numbers 1, 4, 8, 13, 17, 20, 23 and 25.

Figure-6.22: Frame numbers 2, 14, 18, 20, 21, 23, 24 and 25: Note the motion of Trevor.

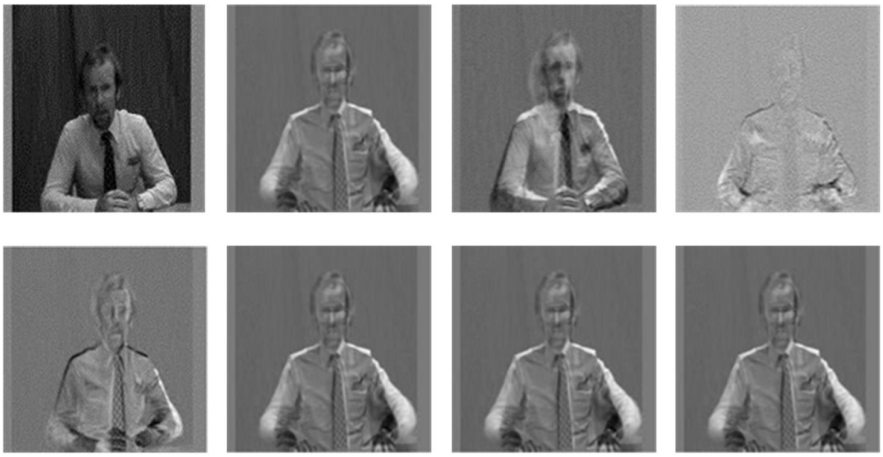


Figure-6.23: Adjacent frame differences.

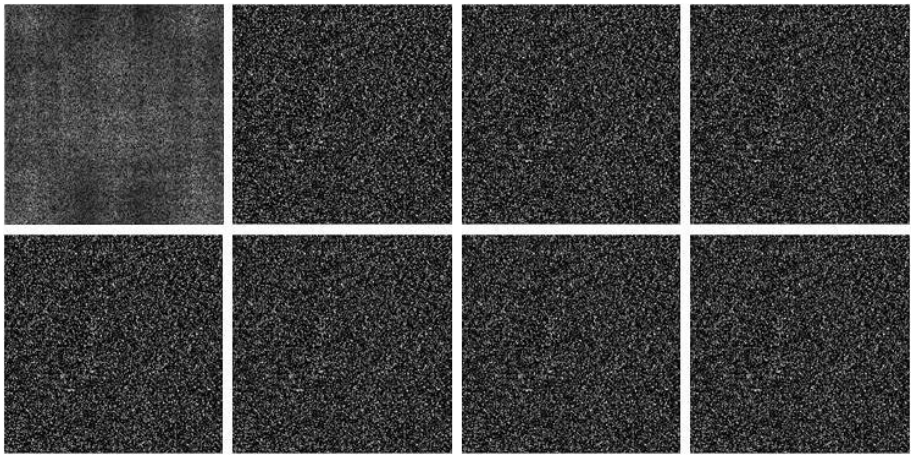


Figure-6.24: Compressed and encrypted frames differences.

Figure-6.25: Recuperate video frame numbers 2, 14, 18, 20, 21, 23, 24 and 25.

The subjective and objective methods are usually adopted to evaluate the performance of video algorithms. The best way to assess the worth of a video is to subjectively evaluate it because in most of the cases, human eyes act decisive receivers. The subjective quality measurement Mean Opinion Score (MOS) has been used for many years [217], [218]. While these tests are the best way to measure "true" perceived quality, they are complex, time-consuming, and expensive. Hence, they are often impractical, when real-time online quality monitoring of several video channels is desired. Looking for faster alternatives, signals could turn to simple error measures such as the MSE or the PSNR. These criteria are considered to be objective due to the fact that they rely on the pixel luma and chroma values of the input and output video frames and do not include any subjective human intervention in the quality assessment process. The MSE is defined with the equation:

$$MSE \cong \frac{1}{TXY} \left| \left| \left| \dashv r \dashv t, x, y, 0\, o \dashv t, x, y, ,, \right.^{2}, \right. \right. \tag{6.3}$$

where x, y, t are height, width and time axis of video frame. The $o(t, x, y)$ is original video frame and $r(t, x, y)$ is the reconstructed frame. The lower value of MSE demonstrates a high quality of video at receiver's end. The worth of the video at receiving end is measured with MSE and PSNR. The proposed algorithm MSE is compared with as shown in Table 6.3 and 6.4[152]. The value of compression percentages 52.26%, 73.09%, 82.65% etc. is considered in Table 6.3 from the Maniccam algorithm [152]. Then proposed algorithm MSE is calculated on these compression percentages for comparison. The Table 6.4 has compared MSE of two algorithms at specific compression percentages taken by Maniccam algorithm.

The PSNR is used as a measure of reconstructed video in proposed algorithm. It is defined with the equation:

$$PSNR \cong 10 \log_{10} \left\{ \frac{\lfloor M \partial N}{MSE^{2}} \right\} \tag{6.4}$$

where M is the maximum value that a pixel can take (e.g. 255 for 8-bit images). The PSNR of videos is calculated at 15% to 90% compression percentage in the proposed algorithm in Table 6.5.

Table-6.3: Comparison of mean square error for Claire video.

Claire			
Compression	**Maniccam *et al.* MSE [152]**	**Proposed algorithm's MSE**	**Improvement in MSE**
52.26%	0.6363	0.000958	0.635342
73.09%	1.0413	0.000951	1.040349
82.65%	1.3416	0.000915	1.340685
87.24%	1.5500	0.000899	1.549101
89.60%	1.7065	0.000883	1.705617
90.96%	1.8374	0.000882	1.836518
91.89%	1.9648	0.000884	1.963916
92.61%	2.0912	0.000880	2.09032
93.24%	2.2191	0.000881	2.218219
93.77%	2.3532	0.000884	2.352316

Table-6.4: Comparison of mean square error for Trevor video.

Trevor			
Compression	**Maniccam *et al.* MSE [152]**	**Proposed algorithm's MSE**	**Improvement in MSE**
31.73%	0.5041	0.000957	0.503143
48.20%	0.9023	0.000927	0.901373
59.99%	1.1752	0.00092	1.17428
65.31%	1.3684	0.000984	1.367416
65.31%	1.3684	0.000984	1.367416
67.79%	1.5469	0.000909	1.545991
69.57%	1.7417	0.000922	1.740778
71.12%	1.9534	0.000945	1.952455
72.52%	2.1785	0.000963	2.177537
73.83%	2.4140	0.000857	2.413143
75.07%	2.6637	0.000935	2.662765

Table-6.5: PSNR of video sequence.

Compression	Tennis PSNR (dB)	Foreman PSNR (dB)	Claire PSNR (dB)	Trevor PSNR (dB)	Heart PSNR (dB)
15%	76.95	76.17	77.33	77.45	76.93
25%	77.30	76.08	77.48	77.30	76.94
35%	77.08	75.80	77.21	77.33	76.78
50%	76.00	75.74	77.31	77.25	77.00
60%	77.25	76.00	77.43	77.49	76.82
70%	76.86	75.60	77.45	77.48	76.89
80%	77.12	76.15	77.50	77.30	77.11
90%	76.83	75.94	77.52	77.53	76.63

The performance of video-compression-encryption algorithm is evaluated with high PSNR and low MSE as compared to existing methods. Yeung *et al.* [219] have also proposed a method for video encryption using multiple 8 × 8 transforms in H.264 and MPEG-4. The PSNR between original frame and the decrypted frame calculated by Yeung *et al.* [219] is 54 dB and 49 dB using H.264 and MPEG-4 respectively. The presented algorithm Table 1 shows that the MSE and PSNR between original image and encrypted image is very small 0.00122 and is very high 76.17 dB respectively at different compression percentages for Foreman video. The increased PSNR 22.17 dB from H.264 and 27.17 dB from MPEG-4 of present algorithm show the successful retrieval of the input frame.

6.4 SUMMARY

Two video encryption algorithms have been developed in this chapter: video encryption and video compression-encryption. Frames are extracted and encrypted using fractional transforms in video encryption. This algorithm gives better PSNR in

comparison of existing method. An algorithm for video compression-encryption algorithm using fractional transforms is also given. The difference frame is compressed and encrypted. This algorithm gives low MSE from the existing SCAN method at various compression percentages. The conclusions and future scope are discussed in the next chapter.

CHAPTER 7

CONCLUSIONS AND FUTURE SCOPE

The contributions given in the previous chapters have been summarized in this chapter. Scope and future prospects of the work is also mentioned.

7.1 CONCLUSIONS

This research work is oriented towards the performance analysis of fractional transforms in image and video processing. Results obtained from research work have been stated in the following section.

In this work, an effort is made to evaluate image compression using DFrFT, DFrCT and DFrHT algorithms. The variation of block sizes at various compression percentages is suggested and it is found that 8 × 8 block size is better for DFrFT, 32 × 32 for DFrCT and 4 × 4 for DFrHT algorithm. The comparison of DFrFT, DFrCT and DFrHT algorithm with one another in image compression makes it clear that DFrFT algorithm provides us with better PSNR. It has been noticed that DFrFT gives improved results for image compression as compared to JPEG standards. This improvement comes at the cost of CPU time. The JPEG method is based on DCT which is discrete fractional Cosine transform when fractional order is one.

The fractional transform algorithms which are based on block-based transforms show visually objectionable artifacts for image compression standards. The blocking effect is noticeable in the form of undesired visible block boundaries. The detection of blocking artifacts at higher compression percentages is noticed more in DFrCT in comparison to DFrFT. Since, very few coefficients are encoded at higher compression percentages, the blocking artifacts become more visible.

A compression and scrambling algorithm is also analyzed for the security of compressed images. The scrambling uses the attributes, i.e. a number of arrays, of digital image to create the confusion. It does so through shuffling in the location of image pixels. The scrambling has very limited influence on the compression efficiency .Scrambling degree (SD) is a standard to measure the outcomes of

scrambling. The improved scrambling degree is obtained in comparison to Zhang's method in 2007 [137] for the Lena and Baboon images respectively.

The image encryption algorithm using fractional transforms with two, three and four fractional keys has been carried out. The variation in the number of encryption keys increases the security. It does so at the cost of increasing complexity however. It has been found that the image encryption using DFrCT provides us with better results than doing it with DFrFT algorithm. The sensitivity of fractional keys is measured for decryption. It is observed that at correct fractional order, the rate of MSE is comparatively lower.

The security of images is augmented with the encryption and scrambling algorithm. The image can properly be retrieved with the correct information of fractional keys, RPMs and algorithm of scrambling. It is also observed that as any change is made in the fractional order, the value of relative error also changes. It happens so for any number of iterations of algorithm. In the presented decryption procedure, image can only be retrieved with the help of correct fractional order keys and correct number of iterations of algorithm.

The transmission of highly informative image contents imposes strict requirements in terms of bandwidth occupancy and security. The joint image encryption-compression and compression-encryption algorithms have been formulated using fractional transforms. The compression-encryption algorithm gives better results than that of encryption-compression in terms of PSNR.

The video has been encrypted using fractional transforms. The video encryption algorithm using DFrFT shows an improvement in PSNR of 0.81dB in comparison with Yueng's method [211]. The proposed algorithm of video compression-encryption gives us lower MSE (better quality) at various compression percentages for Claire and Trevor videos in comparison to existing algorithm [152].

7.2 FUTURE SCOPE

Needless to say, several ideas are left for future research in order to maintain a reasonable scope. Some of the ideas that can be explored in future are listed below:

The presented algorithms of image compression improve the quality at the cost of CPU time. The fractional transforms compress the images and acquire time in seconds. A decrease in the CPU time can be achieved by hardware implementation. So, this parameter can well be considered to bring further improvements in future.

From the more research oriented perspective, the integration and interoperability of different multimedia security techniques (e.g. encryption and robust watermarking or encryption and fragile watermarking) poses a large number of questions. So, efforts in this direction are necessary.

Future directions should include an investigation by extending these applications and thereby achieve more efficient results. The available transforms including fractional Hartley transform can be implemented for image and video applications. The obtained results can be compared with the existing methods.

The purpose of scrambling is to allow only authorized receiver to access the original images and videos. Video scrambling techniques can be implemented using the DFrFT and DFrCT algorithms. The security and other issues can be analyzed with the help of scrambling algorithm.

In future, it will also be in the interest of researchers to develop other fractional transforms such as Haar, Slant and Hadmard transform. In particular, with respect to real-world usage, it will be interesting to see if the entertainment and telecommunication industries will utilize the presented and other new techniques. Therefore, this field will remain vibrant in upcoming time.

List of Publications

Journal Publications

1. Jindal N. and Singh K., (2012) Image and Video Processing using Discrete Fractional Transforms, Signal Image and Video Processing. DOI: 10.1007/s11760-012-0391-4.
2. Jindal N. and Singh K., (2013) Joint Image Compression-Encryption Using Discrete Fractional Transforms, Imaging Science Journal (Maney Publishers) - *Accepted.*
3. Jindal N. and Singh K., (2013) Video Compression-Encryption using Three Dimensional Discrete Fractional Transforms, Research Journal of Applied Sciences, Engineering and Technology, vol. 5, No. 14 pp. 3678-3683.
4. Jindal N. and Singh K., Image Retrieval Algorithm based on Discrete Fractional Transforms, Journal of Electrical Engineering (Versita) - *Accepted.*
5. Jindal N. and Singh K., Secure Image Compression based on Discrete Fractional Transforms and Scrambling –A novel Approach, Defence Science Journal - *Communicated.*
6. Jindal N. and Singh K., Comparison of Image Compression Algorithms based on Discrete Fractional Transforms, International Journal of Electronics (Taylor and Francis) - *Communicated.*
7. Jindal N. and Singh K., Encryption of Video based on Discrete Fractional Transforms, Electronics and Electrical Engineering - *Communicated.*

Conference Publications

1. Jindal N. and Singh K., Image Encryption Using Discrete Fractional Transforms, Proc. IEEE International Conference ARTCOM 2010, Kottyam, pp 165-167. ISBN: 978-0-7695-4201-0.

References

[1]. R. N. Bracewell, *The Fourier Transforms and its Applications*, McGraw-Hill, New York, 1986.

[2]. H. M. Ozaktas, Z. Zalevsky and M. A. Kutay, *The Fractional Fourier Transform with Applications in Optics and Signal Processing*, John Wiley & Sons Ltd., New York, 2000.

[3]. S. Shinde and V. M. Gadre, "An Uncertainty Principle for Real Signals in the Fractional Fourier Transform Domain", *IEEE Trans. Signal Process.*, vol. 49, no. 11, pp. 2545 - 2548, 2001.

[4]. S. G. Samko, A.A. Kilbas and O.I. Marichev, *Fractional Integrals and Derivatives- Theory and Applications*, Yverdon: Gordon and Breach Science Publishers, 1993.

[5]. G. Cariolario, T.Erseghe, P.Kraniauskas and N.laurenti, "A Unified Framework for the Fractional Fourier Transform," *IEEE Trans. Signal Process.*, vol. 46, no. 12, pp. 3206-3212, 1998.

[6]. G. Cariolario, T. Erseghe, P. Kraniauskas and N. Laurenti, "Multliplicity of Fractional Fourier Transforms and Their Relationships," *IEEE Trans. Signal Process.*, vol. 48, no. 1, pp. 247-241, 2000.

[7]. I. S. Yetik, M.A.Kutay, H.Ozaktas, H. M.Ozaktas, "Continuous and discrete fractional Fourier domain decomposition," in Proc. *International Conf. Acoustics, Speech, and Signal Process. 2 (ICASSP 2000)*, 2000, pp. 93-96.

[8]. D. Mendlovic, H. M. Ozaktas and A. W. Lohmann, "Fractional Correlation," *Appl. Opt.,* vol. 34, pp. 303-309, 1995.

[9]. G. S. Agarwal and R. Simon, "A simple Realization of Fractional Fourier Transform and Relation to Harmonic Oscillator Green's Function," *Opt. Commun.,* vol. 110, pp. 23-26, 1994.

[10]. D. H. Bailey and P. N. Swarztrauber, "The Fractional Fourier Transform and Applications," *SIAM Rev.*, vol. 33, no. 3, pp. 389-404, 1991.

[11]. H. M. Ozaktas, B. Barshan, D. Mendlovic and L.Onural, "Convolution, Filtering and Multiplexing in Fractional Domains and their Relation to Chirp and Wavelet Transforms", *J. Opt. Soc. Am. A*, vol. 11, no. 2, pp. 547-559, 1994.

[12]. L. Yu, K.Q. Wang, C.F. Wang and D. Zhang, "Iris Verification based on Fractional Fourier Transform," in Proc. *First International Conf. Machine Learning and Cybernetics,* Beijing, 2002, pp. 1470-1473,.

[13]. S. C. Sharma, B. K. Gupta and M. Lal, "A New Fuzzy Scheme for Multicast Based Shared-Memory ATM Switch," *Int. J. Fuzzy Syst.*, vol. 8, no. 1-3, pp. 125-136, 2002.

[14]. S.C. Pei, C.C. Tseng, M.H. Yeh and J.J. Shyu, "Discrete Fractional Hartley and Fourier Transforms", *IEEE Trans. Circuits Syst.II, Analog Digit. Signal Process*, vol. 45, no. 6, pp. 665-675, 1998.

[15]. W. K. Pratt, *Digital Image Processing*, 2nd edition, John Wiley, New York, 1991.

[16]. V. Bhaskaran and K. Konstantinides, *Image and Video Compression Standards: Algorithms and Architectures,* Kluwer Academic Publishers, Boston, MA, 1995.

[17]. M. Yang, S. Li, and N. Bourbakis, "Data-Image-Video Encryption, " *IEEE Potentials*, vol. 23, no. 3, pp. 28-34, 2004.

[18]. G. Sullivan and T. Wieg, "Video Compression-from Concepts to the H.264/AVC Standard," in Proc. *IEEE*, vol. 93, no. 1, 2005, pp 18-31.

[19]. Available: http://sipi.usc.edu/database/database.php?volume=misc

[20]. H. M. Ozaktas, O. Ankan, M. A. Kutay and G. Bozdaki, "Digital Computation of the Fractional Fourier Transforms," *IEEE Trans. Signal Process.*, vol. 44, no. 9, pp. 2141-2150, 1996.

[21]. C. Candan, M. A. Kutay and H. M. Ozaktas, "The Discrete Fractional Fourier Transform," *IEEE Trans. Signal Process.*, vol. 48, no. 5, pp. 1329-1338, 2000.

[22]. O. N. Gerek, M. F. Erden, "The discrete fractional cosine transform", in Proc. *IEEE Balkan Conf. Signal Processing, Communications, Circuits and Systems,* Istanbul, Turkey, 2000.

[23]. A. W. Lohmann, D.Mendlovic, Z. Zalevsky, R. G. Dorch, "Some important Fractional Transformations for Signal Processing", *Optics Communications,* vol. 125, pp. 18-20, 1996.

[24]. G. Strang, "The Discrete Cosine Transform," *SIAM Rev.*, vol. 41, no. 1, pp. 135-147, 1999.

[25]. A. P. Prudnikov, Y. A. Brychkov and O.I. Marichev, *Integrals and Series VI: Elementary Functions*, Gordon and Breech Science Publishers, New York, 1986.

[26]. N. Ahmed, T. Natarajan and K. R. Rao, "Discrete Cosine Transform", *IEEE Trans. Computers*, vol. C-23, no. 1, pp. 90-93, 1974.

[27]. M. J. Narasimha and A. M. Peterson, "On the Computation of the Discrete Cosine Transform," *IEEE Trans. Commun.(COM)*, vol. 26, no. 6, pp. 934-936, 1978.

[28]. S. C. Pei and J. J. Ding, "Closed-form Discrete Fractional and Affine Fourier Transforms," *IEEE Trans. Signal Process.*, vol. 48, no. 5, pp. 1338-1353, 2000.

[29]. S. C. Pei, M. H. Yeh, "The Discrete Fractional Cosine and Sine Transforms", *IEEE Trans. Signal Process., vol.* 49, pp. 1198-1207, 2001.

[30]. I. Pitas, *Digital Image Processing Algorithms and Applications*, John Wiley & Sons, 2000.

[31]. A. C. Bovik, *Handbook of Image and Video Processing*, Elsevier, Academic Press, Burlington USA, 2005.

[32]. J. C. Russ, *The Image Processing Handbook*, Fifth Edition, CRC Press, Taylor & Francis Group, Sound Parkway, NW 2007.

[33]. M. J. Weinberger, G. Seroussi and G. Sapiro, "The LOCO-I Lossless Image Compression Algorithm: Principles and Standardization into JPEG-LS" *IEEE Trans. Image Process.,* vol. 9, no.8, pp. 1309-1324, 2000

[34]. W. Pennebaker and J. Mitchell, *JPEG: Still Image Data Compression Standard*, Van Nostrand Reinhold, New York, 1993.

[35]. X. Wu and N. Memon, "Context-based, Adaptive, Lossless Image Codec," *IEEE Trans. Commun.*, vol. 45, no. 4, pp. 437-444, 1997.

[36]. N. Memon and X. Wu, "Recent Developments in Context-based Predictive Techniques for Lossless Image Compression", *The Comp. J.*, vol.40, no. 2/3, pp.127-136, 1997.

[37]. A. Said and W. A. Pearlman, "An Image Multiresolution Representation For Lossless And Lossy Compression," *IEEE Trans. Image Process.*, vol. 5, no. 9, pp. 1303-1310, 1996.

[38]. S. Dewitte and J. Cornelis, "Lossless Integer Wavelet Transform," *IEEE Sig. Process. Lett.*, vol. 4, no. 6, pp. 158 -160, 1997.

[39]. A. Zandi , J. D. Allen , E. L. Schwartz and M. Boliek, " CREW: Compression with reversible... Wavelets," in Proc. *IEEE Conference Data Compression* , 1995, pp. 212 -221.

[40]. Y. Wang, "A Set of Transformations for Lossless Image Compression," *IEEE Trans. Image Process.*, vol. 4, no. 5, pp. 677-679, 1995.

[41]. J. Ziv and A. Lempel, "A Universal Algorithm for Sequential Data Compression," *IEEE Trans. Inf. Theory*, vol. 23, no. 3, pp. 337-343, 1977.

[42]. T. A. Welch, "A Technique for High-Performance Data Compression," *IEEE Computer*, vol. 17, no. 6, pp. 8-19, 1984.

[43]. J. A. Robinson, "Efficient General-purpose Image Compression with Binary Tree Predictive Coding," *IEEE Trans. Image Process.,* vol. 6, no. 4, pp. 601-608, 1997.

[44]. B. Schneier, *Applied Cryptography*, John Wiley & Sons, New York, 1994.

[45]. G. Brassard, *Modern Cryptology*, Springer-Verlag, New York, 1988

[46]. G. R. Chen, Y. B. Mao and C. K. Chui, "A Symmetric Image Encryption Scheme based on 3D Chaotic Cat Maps," *Chaos, Soliton. Fract.*, vol. 21, pp. 749-761, 2004.

[47]. B. Furht and D. Kirovski, *Multimedia Security Handbook*, CRC Press, Boca Raton, Florida, 2005.

[48]. Z.H. Guan, F. Huang and W. Guan, "Chaos-based Image Encryption Algorithm," *Phy. Lett. A*, vol. 346, no. 1, pp. 153-157, 2005.

[49]. M. A. Sid-Ahmed, *Image Processing-Theory, Algorithms, and Architectures*, McGraw-Hill, New York, 1995.

[50]. G. Kropatsch, H. Bischof, *Digital Image Analysis: Selected Techniques and Applications*, Springer-Verlag, New Tork, 2001.

[51]. S. Lping, Q. Zheng, L. Bo, Q. Jun and L. Huan, "Image Scrambling Algorithm based on Random Shuffling Strategy," *IEEE Trans. Sig. Process*., pp. 2278-2283, 2008.

[52]. J. Zou, R. K. Ward and D. Qi, "A New Digital Image Scrambling Method based on Fobonacci Numbers," in Proc. *International Symp. on Circuits and Systems* (ISCAS), 2004, pp. III-965-968.

[53]. L. Itti, "Automatic Foveation for Video Compression using a Neurobiological Model of Visual Attention," *IEEE Trans. Image Process*., vol. 13, no. 10, pp. 1304-1318, 2004.

[54]. S. Lian, J. Sun, G. Liu and Z. Wang, "Efficient Video Encryption Scheme based on Advanced Video Coding," *Multimed. Tools Appl.*, vol. 38, no. 1, pp. 75-89, 2008.

[55]. A. K. Lenstra and E. R. Verheul, "*Selecting Cryptographic Key Sizes*," *J. Cryptol.*, vol. 14, no. 4, pp. 255-293, 2001.

[56]. X. Liu, and A.M. Eskicioglu, "Selective Encryption of Multimedia Content in Distribution Networks: Challenges and New Directions," in Proc. *IASTED International Conf. on Communications, Internet and Information Technology* (CIIT 2003), Scottsdale, AZ, 2003, pp. 1-10.

[57]. S. Lian, Z. Liu, Z. Ren and H. Wang, "Commutative Encryption and Watermarking in Video Compression," *IEEE Trans. Circuits and Syst. Video Technol.,* vol. 17, no. 6, pp. 774-778, 2007.

[58]. V. Madisett , *Digital Signal Processing Fundamentals*, vol. 1, CRC Press, 2010

[59]. A. K. Jain, *Fundamentals of Digital Image Processing*, Prentice Hall of India, Englewood Cliffs, NJ , 1989.

[60]. M. Rabbani, P. W. Jones, *Digital* Image Compression *Techniques*, SPIE, Washington, 1991.

[61]. I. E. Richardson, *The H.264 Advanced Video Compression Standard*, John Wiley & Sons. 2011.

[62]. N. K. Nishchal, J. Joseph and K. Singh, "Fully Phase-encrypted Memory using Cascaded Extended Fractional Fourier Transform," *Opt. Laser. Eng.*, vol. 42, pp. 141-151, 2004.

[63]. S. W. Baik, M. S. Jeong, R. Baik, "Aerial Photo Image Retrieval Using Adaptive Image Classification," *Knowledge-Based Intelligent Information and Engineering Systems*. Springer Berlin Heidelberg, vol. 4253, 2006, pp. 284-291.

[64]. A. J. Menezes, C. Paul, V.Oorschot, S. A. Vanstone, *Handbook of Applied Cryptography*, CRC Press, NewYork, 1997.

[65]. A. Uhl, A. Pommer, *Image and Video Encryption: From Digital Rights Management to Secured Personal Communication*, Springer, 2005.

[66]. A. Bultheel and H. Martinez, "A shattered survey of the Fractional Fourier Transform", TW Reports, TW337, Belgium, pp. 1-42, 2002.

[67]. A. P. Guinand, "Matrices Associated with Fractional Hankel and Fourier Transformations," in Proc. Glasgow Mathematical Association, vol. 2, 1956, pp. 185-192.

[68]. G. Sansone, *Orthogonal Functions*, Inderscience Publishers, New York, 1959.

[69]. V. Namias, "The Fractional Order Fourier Transform and its Application in Quantum Mechanics," *IMA J. Appl. Math.*, vol. 25, pp. 241-265, 1980.

[70]. A. C. McBride and F. H. Kerr, "On Namias's Fractional Fourier Transform", *IMA J. Appl. Math.*, vol. 39, pp. 159-175, 1987.

[71]. B. L. Almeida, "The Fractional Fourier Transform and Time-frequency Representations," *IEEE Trans. Signal Process.*, vol. 42, pp. 3084-3091, 1994.

[72]. M. A. Kutay, H. M. Ozaktas, O. Arikan and L. Onural, "Optimal filtering in Fractional Fourier Domain," *IEEE Trans. Signal Process.*, vol. 45, no. 3, pp. 1129-1143, 1997.

[73]. B. W. Dickinson and K. Steiglitz, "Eigenvectors and Functions of the Discrete Fourier Transform," *IEEE Trans. Acoust., Speech, Signal Process.,* pp. 25-31, 1982.

[74]. B. Santhanam and J. H. McClellan, "The Discrete Rotational Fourier Transform," *IEEE Trans. Signal Process.*, vol. 42, no. 4, pp. 994-998, 1996.

[75]. S. C. Pei and M. H. Yeh, "Discrete fractional Fourier transform," in Proc. *IEEE International Symp. on Circuits and Systems*, 1996, pp. 536-539.

[76]. S. C. Pei and M. H. Yeh, "Improved Discrete fractional Fourier Transforms", *Opt. Lett.*, vol. 22, no. 14, pp. 1047-1049, 1997.

[77]. M. H. Yeh, S. C Pei, "A Method for the Discrete Fractional Fourier Transform Computation", *IEEE Trans. Signal Process.*, vol. 51, no. 3, pp. 889-891, 2003.

[78]. S. C. Pei, M. H. Yeh, and C. C. Tseng, "Discrete Fractional Fourier Transform based on Orthogonal Projections," *IEEE Trans. Signal Process.*, vol. 47, no. 5, pp. 1335-1348, 1999.

[79]. K. Singh, "Performance of Discrete Fractional Fourier Transform Classes in Signal Processing Applications," Ph.D Thesis, Dept. of Elect. Comm. Eng., Thapar Univ., Patiala, India, 2005.

[80]. T. Erseghe, P. Kraniauskas and G. Cariolaro, "Unified Fractional Fourier Transform and Sampling Theorem", *IEEE Trans. Signal Process.*, vol. 47, no. 12, pp. 3419-3424, 1999.

[81]. H. M. Ozaktas and M. A. Kutay. David Mendlovic, "Introduction to the Fractional Fourier Transform and its Applications," *Adv. Imag. Electr. Phys.,* vol. 106, pp. 239-291, 1999.

[82]. S.C. Pei, C.C. Tseng, "A New Discrete Fractional Fourier Transform based on Constrained Eigendecomposition of DFT Matrix by Largrang Multiplier Method," *IEEE*, pp. 3965-3968, 1997.

[83]. A. W. Lohmann, D. Mendlovic and Z. Zalevsky, "Fractional Transformation in Optics," *Prog. Optics*, vol. 38(C), pp. 263-342, 1998.

[84]. X.G. Xia, "On Bandlimited Signals with Fractional Fourier Transforms," *IEEE Signal Process. Lett.*, vol.3 , no. 3, pp. 72-74, 1996.

[85]. M. F. Erden, M. A. Kutay and H.M. Ozaktas, "Repeated Filtering in Consecutive Fractional Fourier Domains and its Application to Signal Restoration," *IEEE Trans. Signal Process.*, vol. 47, no. 5, pp. 1458-1462, 1999.

[86]. P. Yip, "Sine and Cosine Transforms," in *The Transforms and Applications Handbook, A. D. Poularikas, ed., Alabama: CRC Press*, 1996, pp. 1-67.

[87]. D. Mendlovic, Z. Zalevsky, N. Konforti, R. G. Dorsch and A. W. Lohmann, "Incoherent Fractional Fourier Transform and its Optical Implementation," *App. Optics*, vol. 34, no. 32, pp. 7615-7620, 1995.

[88]. J. Mielikainen, "A Novel Full-Search Vector Quantization Algorithm Based on The Law of Cosines," *IEEE Signal Process. Lett.,* vol. 9, no. 6, pp. 175-176, 2002.

[89]. D. Zhao, X. Li, L. Chen, "Optical Image Encryption with Redefined Fractional Hartley Transform," *Opt. Commun.*, vol. 281, pp. 5326-5329, 2008.

[90]. X. X. Li, D.M. Zhao, "Optical Image Encryption With Simplified Fractional Hartley Transform," *Chin. Phys. Lett.*, vol. 25, no. 7, pp. 2477-2480, 2008.

[91]. P. K. Sontakke and A. S. Gudadhe, "Analyticity and Operation Transform on Generalized Fractional Hartley Transform," *Int. Journal of Math. Analysis*, vol. 2, no. 20, pp. 977-986, 2008.

[92]. C. Jimenez, C. Torres, L. Mattos, "Fractional Hartley Transform Applied to Optical Image Encryption," *J. Phy. : Conf. Ser.*, vol. 274, no. 1, pp. 1-6, 2011.

[93]. G. K. Wallace, "The JPEG Still Picture Compression Standard," *IEEE Trans. Consum. Elect.*, vol. 38, no. 1, pp. 30-44, 1991.

[94]. G. K. Wallace, "Overview of the JPEG (ISO/CCITT) Still Image Compression Standard," in Proc. *SPIE Image Processing Algorithms and Techniques*, vol. 1244, 1990, pp. 220-233.

[95]. B. Usevitch, "A Tutorial on Modern Lossy Wavelet Image Compression: Foundations of JPEG 2000," *IEEE Signal Process. Magazine*, vol. 18, no. 5, pp. 22-35, 2001.

[96]. I. S. Yetik, M. A. Kutay, H. M. Ozaktas, "Image Representation and Compression with the Fractional Fourier Transform," *Opt. Commun.*, vol. 197, pp. 275-278, 2001.

[97]. C. Vijaya and J. S. Bhat, "Signal Compression using Discrete Fractional Fourier Transform and Set Partitioning in Hierarchical Tree," *Signal Process.*, vol. 86, no. 8, pp. 1976-1983, 2006.

[98]. T. Cebrail, S. Serder, "An Overview of Image Compression Approaches," in Proc. *Third International Conf. on Digital Telecommunication*, 2008, pp. 174-179.

[99]. D. P. Dutta, S. D. Choudhury, Md. A. Hussain and S. Majumder, "Digital Image Compression using Neural Networks," in Proc. *International Conf. on Advances in Computing, Control, & Telecommunication Technologies* (ACT '09), 2009, pp. 116-120.

[100]. A. Alfalou, M. Elbouz, A. Mansour and G. Keryer, "New Spectral Image Compression Method based on an Optimal Phase Coding and the RMS duration Principle", J. Opt., vol. 12, no. 11, pp. 1-12, 2010.

[101]. Y. Li, Z. Zhang and Y. Li, "Recovery of the Optimal Approximation from Samples in Wavelet Subspace," *Digit. Signal Process.*, vol. 22, no. 5, pp. 795-807, 2012.

[102]. A. Kaushik and M. Gupta, "Analysis of Image Compression Algorithms," *Int. J. Eng. Res. App.*, vol. 2, no. 2, pp.773-779, Mar. 2012.

[103]. Y. L. Lee, H.C. Kim and H.W. Park, "Blocking Effect Reduction of JPEG Images by Signal Adaptive Filtering," *IEEE Trans. Image Process.*, vol. 7, no. 2, pp. 229-234, Feb. 1998.

[104]. B. Zeng, "Reduction of Blocking Effect in DCT-coded Images using Zero-masking Techniques," *Sig. Process.*, vol. 79, no. 2, pp. 205-211, 1999.

[105]. Y. Luo and R.K. Ward, "Removing the Blocking Artifacts of Block-based DCT Compressed Images," *IEEE Trans. Image Process.*, vol. 12, no. 7, pp. 838-842, 2003.

[106]. R. Palaparthi and V.K. Srivastava, "A Simple Deblocking Method for Reduction of Blocking Artifacts," in *IEEE Students' Conference on Electrical, Electronics and Computer Science* (SCEECS), 2012, pp. 1-4.

[107]. T. Chen, H.R. Wu and B. Qiu, "Adaptive Postfiltering of Transform Coefficients for the Reduction of Blocking Artifacts," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 11, no. 5, pp. 594-602, 2001.

[108]. S. Singh, V. Kumar and H.K. Verma, "Optimization of Block Size For DCT-Based Medical Image Compression," *J. Med. Eng. Technol.*, vol. 31, no.2, pp. 129-143, 2007.

[109]. S. Singh, V. Kumar and H.K. Verma, "Reduction of Blocking Artifacts in JPEG Compressed Images," *Digit. Signal Process.*, vol. 17, no.1, pp. 225-243, 2007.

[110]. U. S. Mohammed, "A Pixel-domain Post-Processing Technique to Reduce the Blocking Artifacts in Transform-Coded Images," in *IEEE International Symp. on Signal Processing and Information Technology* (ISSPIT), 2010, pp. 266-270.

[111]. W. Puech, "Image Encryption and Compression for Medical Image Security," in *First Workshop on Image Processing Theory*, *Tools and Applications* (IPTA), 2008, pp 1-2.

[112]. G. Cristobal, P. Schelkens, H. Thienpont, *Optical and Digital Image Processing*, Germany, 2011.

[113]. N. Zhou, Y. Wang and L. Gong, "Novel optical image encryption scheme based on fractional Mellin transform," *Opt. Commun.*, vol. 284, no. 13, pp. 3234-3242, 2011.

[114]. N. Singh, A. Sinha, "Optical Image Encryption using Fractional Fourier Transform and Chaos," *Opt. Laser Eng.*, vol. 46, no. 2, pp. 117-123, 2008.

[115]. R. Tao, Y. Xin and Y. Wang, "Double Image Encryption based on Random Phase Encoding in the Fractional Fourier Domain," *Opt. Express*, vol. 15, no. 24, pp. 16067-16079, 2007.

[116]. A. Sinha, K. Singh , "A Technique for Image Encryption using Digital Signature," *Opt. Commun.*, vol. 218, no. 4-6, pp. 229-234, 2003.

[117]. B. Hennelly and J. T. Sheridan, "Fractional Fourier Transform-based Image Encryption: Phase Retrieval Algorithm," *Opt. Commun.*, vol. 226, no. 1-6, pp. 61-80, 2003.

[118]. B. Hennely, J.T. Sheridan, "Image Encryption and Fractional Fourier Transform," Optik, vol. 114, pp. 251-265, 2003.

[119]. P. Refregier and B. Javidi, "Optical Image Encryption based on Input Plane and Fourier Plane Random Encoding," *Opt. Lett.*, vol. 20, pp. 767-769, 1995.

[120]. G. Unnikrishnan, K. Singh, "Double Random Fractional Fourier Domain Encoding for Optical Security", *Opt. Eng.*, vol. 39, pp. 2853-2859, 2000.

[121]. S. Liu, L. Yu and B. Zhu, "Optical Image Encryption by Cascaded Fractional Fourier Transforms with Random Phase Filtering", *Opt. Commun.*, vol. 187, pp. 57-63, 2001.

[122]. Y. Zhang, C. H. Zheng and N. Tanno, "Optical Encryption based on Iterative Fractional Fourier Transform", *Opt. Commun.*, vol. 202, pp. 277-285, 2002.

[123]. B. Zhu and S. Liu, "Optical Image Encryption based on the generalized Fractional Convolution Operation", *Opt. Commun.*, vol. 195, pp. 371-381, 2001.

[124]. B. Zhu and S. Liu, "Optical Image Encryption with Multistage and Multichannel Fractional Fourier-Domain Filtering," *Opt. Lett.*, vol. 26 , pp. 1242-1244, 2001.

[125]. B. Zhu, S. Liu and Q. Ran, "Optical Image Encryption based on Multifractional Fourier Transforms", *Opt. Lett.*, vol. 25, pp. 1159-1161, 2000.

[126]. B. Hennelly and J. T. Sheridan, "Optical Image Encryption by Random Shifting in Fractional Fourier Domains," *Opt. Lett*., vol. 28 , pp. 269-271, 2003.

[127]. N. Zhou, Y. Wang, L. Gong, H. He and J. Wu, "Novel Single-Channel Color Image Encryption Algorithm Based on Chaos and Fractional Fourier Transform," vol. 284, no. 12, pp. 2789-2796, 2011.

[128]. W. Jin and C. Yan, "Optical Image Encryption based on Multichannel Fractional Fourier Transform and Double Random Phase Encoding Technique," *Optik*, vol. 118, pp. 38-41, 2007.

[129]. Y. Sheng, Z. Xin, M. S. Alam, L. Xi and L. Xiao-feng, "Information Hiding based on Double Random-Phase Encoding and Public-Key Cryptography," *Opt. Express*, vol. 17, no. 5, pp. 3270-3284, 2009.

[130]. Y. Frauel, A. Castro, T. J. Naughton and B. Javidi, "Resistance of the Double Random Phase Encryption against various Attacks," *Opt. Express*, vol. 15, no. 6, pp. 10253-10265, 2007.

[131]. N. Saini and A. Sinha, "Key Management of the Double Random-Phase-Encoding Method using Public-Key Encryption," *Opt. Laser Eng.*, vol. 48, pp. 329-334, 2010.

[132]. J. Sang, H. Xiang , L. Fu and N. Sang, "Security Analysis and Improvement on a Double-Random Phase-Encoding Technique based Information Hiding Method," *Opt. Commun.*, vol. 282, pp. 2307-2317, 2009.

[133]. S. Yuan, X. Zhou, D.H. Li and D.F. Zhou, "Simultaneous Transmission for an Encrypted Image and a Double Random-Phase Encryption Key," *Appl. Optics*, vol. 46, pp. 3747-53, 2007.

[134]. X. Zhou, D. Lai, S. Yuan, D.H. Li and J.P. Hu, "A Method for Hiding Information utilizing Double-Random Phase-Encoding Technique," *Opt. Laser Technol.*, vol. 39, no. 7, pp. 1360-1363, 2007.

[135]. Y. Zhang, C. Zheng and N. Tanno, "Optical Encryption based on Iterative Fractional Fourier Transform", *Opt. Commun.*, vol. 202, pp. 277-285, 2002.

[136]. J. Zou and R. K. Ward, "Introducing Two New Image Scrambling Methods," *IEEE Trans. Image Process.,* pp. 708-711, 2003.

[137]. H. Y. Zhang, "A New Image Scrambling Algorithm based on Queue Transformation," in *Proc. IEEE International Conf. on Machine Learning and Cybernetics,* pp. 1526-1530, 2007.

[138]. H. Y. Zhang, "A New Image Scrambling Algorithm", in *Proc. IEEE International Conf. on Machine Learning and Cybernetics,* pp.1088-1092, 2008.

[139]. W. Ji and, C. Yan, "Optical Image Encryption based on Multichannel Fractional Fourier Transform and Double Random Phase Encoding Technique," *Optik,* vol. 118, pp. 38-41, 2007.

[140]. C. Lv and Q. Zhao, "Integration of Data Compression and Cryptography: Another Way to Increase the Information Security," in *21st International Conference on Advanced Information Networking and Applications Workshops* (AINAW '07), vol. 2, 2007, pp. 543-547.

[141]. W. Zeng, H. Yu and C.Y. Lin, *Multimedia Security Technologies for Digital Rights Management*, vol. 18, Academic Press, London, UK, 2006.

[142]. A. K. Pal, G. P. Biswas and S. Mukhopadhyay, "An Efficient Compression-Encryption Scheme for Batch-Image," in Proc. *First International Conf. on Technology Systems and Management Communications in Computer and Information Science* (ICTSM), Springer, vol. 145, 2011, pp. 85-90.

[143]. W. Xiaolin and P.W. Moo, "Joint Image/Video Compression and Encryption via High-Order conditional Entropy Coding of Wavelet Coefficients," *IEEE International Conf. on Multimedia Computing and Systems,* vol. 2, 1999, pp. 908-912.

[144]. K. J. Singh and R. Manimegalai, "A Survey on Joint Compression and Encryption Techniques for Video Data," *J. Comp. Sci.,* vol. 8, no. 5, pp. 731-736, 2012.

[145]. M. Sharma and S. Gandhi, "Compression and Encryption: An Integrated Approach," *Int. J. Eng. Res. Technol.*, vol. 1, no. 5, pp. 1-7, 2012

[146]. A. Kingston, S. Colosimo, P. Campisi and F. Autrusseau, "Lossless Image Compression and Selective Encryption using a Discrete Radon Transform," in Proc. *IEEE International Conference on Image Processing* (ICIP 2007), vol. 4, pp. IV-465-468.

[147]. A. Anil Kumar and A. Makur, "Distributed Source Coding based Encryption and Lossless Compression of Gray Scale and Color Images," in *Proc. IEEE 10th Workshop on Multimedia Signal Processing*, 2008, pp. 760-764.

[148]. L. Mingyu, Y. Xiaowei and M. Hengtai, "A Scalable Encryption Scheme for CCSDS Image Data Compression Standard," in Proc. *IEEE International Conference on Information Theory and Information Security* (ICITIS), 2010, pp. 646-649.

[149]. W. Liu, W. Zeng, D. Lina and Y. Qiuming, "Efficient Compression of Encrypted Grayscale Images," *IEEE Trans. Image Process.*, vol. 19, no.4, pp. 1097-1102, 2010.

[150]. V. Radha, D. Maheswari, "Secured Compound Image Compression Using Encryption Techniques," in Proc. *World Congress on Engineering and Computer Science* (WCECS), San Francisco, USA, 2011, ISBN: 978-988-18210-9-6.

[151]. A. Razzaque and N. V. Thakur, "An Approach to Image Compression with Partial Encryption without sharing the Secret Key," *Int. J. Comp. Sci. Network Security*, vol.12, no.7, pp. 1-6, 2012.

[152]. S. S. Maniccam and N.G. Bourbakis, "Image and Video Encryption using SCAN Patterns," *Pattern Recogn.*, vol. 37, no. 4, pp. 725-737, 2004.

[153]. M. Ito, N. Ohnishi, A. Alfalou and A. Mansour, "New Image Encryption and Compression Method Based on Independent Component Analysis," in Proc. *Third International Conf. on Information and Communication Technologies: From Theory to Applications* (ICTTA), 2008, pp. 1-6.

[154]. Y. You, H. Kim, "Endoscopy Image Compression and Encryption under Fault Tolerant Ubiquitous Environment," in Proc. *IEEE Biomedical Circuits and Systems Conference* (BioCAS), 2009, pp. 165-168.

[155]. D. Maheswari, V. Radha, "Enhanced Hybrid Compound Image Compression Algorithm Combining Block and Layer-based Segmentation," *The Int. J. Multi. App.*, vol.3, no.4, pp.119-131, 2011.

[156]. A. A. Yahya and A. M. Abdalla, "A Shuffle Image-Encryption Algorithm," *J. Comp. Sc.*, vol. 4, no. 12, pp. 999-1002, 2008.

[157]. Y. Negi, "A Survey on Video Encryption Techniques," *Int. J. Emer. Technol. Adv. Eng.*, vol. 3, no. 4, pp. 234-237, 2013.

[158]. B. Bhargava, C. Shi, and S. Wang, "MEPG Video Encryption Algorithms," *Multimed. Tools Appl.,* vol. 24, no. 3, pp. 57-79, 2004.

[159]. H. Kezia and G. F. Sudha, "Encryption of Digital Video Based on Lorenz Chaotic System," in Proc. *International Conference on Advanced Computing and Communication*, 2008, pp. 40-45.

[160]. F. Liu and H. Kornig, "A Survey of Video Encryption Algorithms," *Comput. Secur.*, pp. 3-15, 2010.

[161]. E.T. Lin, A.M. Eskicioglu, R.L. Lagendijk and E.J. Delp, "Advances in Digital Video Content Protection," in Proc. *IEEE,* vol. 93, no. 1, 2005, pp. 171-183.

[162]. C.N. Raju, G. Umadevi, K. Srinathan and C. V. Jawahar, "Fast and Secure Real-Time Video Encryption," in Proc. *Sixth Indian Conference on Computer Vision, Graphics & Image Processing* (ICVGIP '08), 2008, pp. 257- 264.

[163]. M. Zeghid, M. Machhout, L. Khriji, A. Baganne, and R. Tourki, " A Modified AES Based for Image Encryption", *Int. J. Comp. Sc. Eng.*, vol. 1, no.1, pp. 70-75, 2007.

[164]. S. Lian, *Multimedia Content Encryption: Techniques and Applications*, CRC, 2008.

[165]. C.P. Wu, C.C. J. Kuo, "Design of Integrated Multimedia Compression and Encryption Systems," *IEEE Trans. Multimedia*, vol. 7, no. 5, pp. 828-839, 2005.

[166]. A. J. Slaggel. (2004). *Known-Plaintext Attack against a Permutation Based Video Encryption Algorithm*, pp.1-10. Available:http://eprint.iacr.org

[167]. M. C. Angelides and H. Agius, *The Handbook of MPEG Applications: Standards in Practice*, John Wiley & Sons, 2011.

[168]. L. Qiao and K. Nahrstedt, "A New Algorithm for MPEG Video Encryption," in Proc. *First International Conf. on Imaging Science, Systems and Technology*, Las Vegas, Nevada, 1997, pp. 21-29.

[169]. M. Abomhara, O. Zakaria and O. O. Khaifla, "An Overview of Video Encryption Techniques," *Int. J. Comp. Th. Eng.*, vol. 2, no. 1, pp. 103-110, 2010.

[170]. M. A. Chandra, R. Purwar and N. Rajpal , "A Novel Approach of Digital Video Encryption," *Int. J. Comp. App.*, vol. 49, no. 4, pp.38-42, 2012.

[171]. A. Pande, P. Mohapatra and J. Zambreno, "Using Chaotic Maps for Encrypting Image and Video Content," in Proc. *IEEE International Symp. on Multimedia*, Danapoint, California, USA, 2011, pp. 171-178.

[172]. J. Shah and V. Saxena, "Video Encryption: A Survey," *Int. J. Comp. Sc.*, vol. 8, no. 2, pp. 525-534, 2011.

[173]. S. Dhanani, M. E. Parker, *Digital Video Processing for Engineers: A Foundation for Embedded Systems Design*, Elseveir, USA, 2011.

[174]. W. Zeng and S. Lei, "Efficient Frequency Domain Selective Scrambling of Digital Video," *IEEE Trans. Multimedia*, vol. 5, no. 1, pp. 118-129, 2003.

[175]. W. Zeng, H. Yu and C.Y. Lin, *Multimedia Security Technologies for Digital Rights Management*, vol. 18, Academic Press, London, UK, 2006.

[176]. L. Tang, "For Encrypting and Decrypting MPEG Video data Efficiently," in Proc. *The Fourth ACM International Multimedia Conference* (ACM Multimedia'96), Boston, MA, 1996, pp. 219-229.

[177]. H. Wang and C. W. Xu, "A new Lightweight and Scalable Encryption Algorithm for Streaming Video over Wireless Networks," in Proc. *International Conf. on Wireless Networks*, Las Vegas, Nevada, USA, 2007. Available: http://science.kennesaw.edu/~hwang7/icwn07.pdf

[178]. J. Meyer and F. Gadegast. (1995). *Security Mechanisms for Multimedia-Data with the Example MPEG-I- Video*, Project description of SECMPEG, pp.1-10, 1995. Available:http:// gadegast.de/frank/doc/secmeng.pdf

[179]. G.A. Spanos and T. B. Maples, "Performance Study of a Selective Encryption Scheme for the Security of Networked Real Time Video," in Proc. *Fourth International Conf. on Computer Communications and Networks*, 1995, pp. 2-10.

[180]. C. Shi and B. Bhargava, "An Efficient MPEG Video Encryption Algorithm," in Proc. *Seventeenth IEEE Symposium on Reliable Distributed Systems,* 1998, pp. 381-386.

[181]. C. Shi, S. Y.Wang, and B. Bhargava, "MPEG Video Encryption in Real-time using Secret Key Cryptography, " in Proc. *International Conf. on Parallel and Distributed Processing Algorithms and Applications*, 1999, pp. 191–201.

[182]. R. Bose and S. Pathak, "A Novel Compression and Encryption Scheme Using Variable Model Arithmetic Coding and Coupled Chaotic System," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 53, no. 4, 2006, pp. 848-857.

[183]. K. W. Wong and C. H. Yuen. *Performing Compression and Encryption Simultaneously using Chaotic Map*, Available: http://inds08.uni-klu.ac.at/INDS2008/ INDS08_Performing_Compression_ and_Encryption _Simultaneously_using_Chaotic_Map.pdf

[184]. R. Curtmola, J. Garay, S. Kamara and R. Ostrovsky, "Searchable Symmetric Encryption: Improved Definitions and Efficient Constructions," *J. Comp. Secur.*, vol. 19, no. 5, pp. 895-934, 2011.

[185]. S. Jayaraman, S. Esakkirajan and T. Veerakumar, *Digital Image Processing*, Tata McGraw-Hill Education, 2011.

[186]. K. A. McIntyre, "Dynamic Bandwidth Adaptive Image Compression/ Decompression Scheme," U.S. Patent 7024045, 2006.

[187]. T. L. B. Yng, B. G. Lee and H. Yoo, "A Low Complexity and Lossless Frame Memory Compression for Display Devices," *IEEE Trans. Cons. Electron.*, vol. 54, no. 3, pp. 1453-1458, 2008.

[188]. J. Shukla, M. Alwani and A. K. Tiwari, "A Survey on Lossless Image Compression Methods," in Proc. *Second International Conf. on Computer Engineering and Technology* (ICCET), vol. 6, 2010, pp. 136-141.

[189]. R. C. Gonzalez and R. E. Woods, *Digital Image Processing*, 3rd edition, 2008.

[190]. C. Saravanan, R. Ponalagusamy, "Lossless Grey-scale Image Compression using Source Symbols Reduction and Huffman Coding," *Int. J. Image Process.*, vol. 3, no. 5, pp 246-251, 2009.

[191]. M. Sharma, "Compression Using Huffman Coding," *Int. J. Comp. Sc. Net. Secur.*, vol. 10, no. 5, pp. 133-141, 2010.

[192]. A. Majumdar , "Image Compression by Sparse PCA Coding in Curvelet Domain," *Sig. Image Video Process.*, vol. 3, pp. 27-34, 2009.

[193]. A. Skodras, C. Christopoulos and T. Ebrahimi, "The JPEG 2000 still Image Compression Standard," IEEE Signal Processing Magazine, vol. 18, no.5, pp. 36-58, 2001.

[194]. A. B. Watson, "Image Compression using the Discrete Cosine Transform," Math. J., vol. 4, no. 1, pp. 81-88, 1994.

[195]. S. Grgic, M. Grgic, B. Zovko-Cihlar, "Performance Analysis of Image Compression using Wavelets," *IEEE Trans. Ind. Electron.*, vol. 48, no.3, pp. 682-695, 2001

[196]. A. Mayache, T. Eude, H. Cherifi, "A Comparison of Image Quality Models and Metrics based on Human Visual Sensitivity," in Proc. *International Conf. on Image Processing* (ICIP 98), vol. 3, 1998, pp. 409-413.

[197]. K. S. Thyagarajan, *Still Image and Video Compression with MATLAB,* John Wiley & Sons, Inc. Hoboken, New Jersey, 2011.

[198]. S. C. Pei, M. H. Yeh, "Two-dimensional Discrete Fractional Fourier Transform," *Sig. Process*., vol. 67, no. 1, pp. 99-108, 1998.

[199]. Z. Wang , "Fast Algorithms for the Discrete W Transform and for the Discrete Fourier Transform," *IEEE Trans. Acoust., Speech, Signal Process.* (ASSP), vol. 32, 1984, pp. 803-816.

[200]. C. C. Chen, "On the Selection of Image Compression Algorithms," in Proc. *IEEE 14th International Conf. on Pattern Recognition* (ICPR'98), 1998, pp.1-5.

[201]. S. Deng, Z. Pu, D. Zhang, "An Algorithm of Image Processing based on Discrete Wavelet Transform Compression and Chaotic Scrambling," *Journal of Chongqing University,* vol.8, 2008.

[202]. X. Wang, D. Zhao and L. Chen, "Image Encryption based on Extended Fractional Fourier Transform and Digital Holography Technique," *Opt. Commun.,* vol. 260, no. 2, pp. 449-453, 2006.

[203]. Z. H. Guan, F. Huang and W. Guan, "Chaos-based Image Encryption Algorithm," *Phy. Lett. A*, vol. 346, no. 1-3, pp. 153-157, 2005.

[204]. G. Ye, "Image Scrambling Encryption Algorithm of Pixel Bit based on Chaos Map," *Patt. Recogn. Lett.*, vol. 31, no. 5, pp. 347-354, 2010.

[205]. I. Ozturk and I. Sogukpinar, "Analysis and Comparison of Image Encryption Algorithms," *Int. J. Infor. Technol.*, vol. 1, no. 2, pp. 64-67, 2005.

[206]. K. R. Castleman, *Digital Image Processing*, Pearson Education, India, 2007.

[207]. J. Zhao , H. Lu, X. Song, J. Li and Y. Ma, "Optical Image Encryption Based on Multistage Fractional Fourier Transforms and Pixel Scrambling Technique," *Opt. Commun.*, vol. 249, no. 4-6, pp. 493-499, 2005.

[208]. A. Pande and J. Zambreno, "Securing Multimedia Content using Joint Compression and Encryption," *Embedded Multimedia Security Systems*, Springer, pp. 23-30, 2013.

[209]. P. P. Dang and P.M. Chau, "Image Encryption for Secure Internet Multimedia Applications," in Proc. *IEEE trans. Consum. Electron.*, 2000, pp. 395-403.

[210]. O. Marques, *Practical Image and Video Processing Using MATLAB,* John Wiley & Sons, 2011

[211]. S. A. Yeung and S. Zhu, "Partial Video Encryption based on Alternating Transforms", *IEEE Signal Process. Lett.*, vol. 16, no. 6, pp. 893-896, 2009.

[212]. S. Lian, J. Sun and Z. Wang, "Quality Analysis of Several typical MPEG Video Encryption Algorithms," *J. Image and Graphics*, vol. 9, no. 4, pp. 483–490, 2004.

[213]. X. Bao, J. Jiang, W. Yuan and Y. Li, "Study of CABAC-based Digital Video Encryption in the H.264/AVC Standard," *J. Commun.*, vol. 28, no. 6, pp. 24-29, 2007.

[214]. H. Cheng and X.B. Li, "Partial Encryption of Compressed Images and Videos," *IEEE Trans. Signal Process.,* vol. 48, no. 8, pp. 24 -39, 2000.

[215]. S. Lian, Z. Liu, Z. Ren and Z. Wang, "Selective Video Encryption based on Advanced Video Coding," *Advances in Multimedia Information Processing-PCM 2005*. Springer Berlin Heidelberg, 2005, pp. 281-290.

[216]. S. Ishwar, P. K. Meher and M. N. S. Swamy, "Discrete Tchebichef Transform-A Fast 4 × 4 Algorithm and its Application in Image/Video Compression," in Proc. *IEEE International Symp. on Circuits and Systems* (ISCAS 2008), 2008, pp. 260-263.

[217]. "Methodology for the Subjective Assessment of the Quality of Television Pictures," ITU Recommendation BT.500-9, 1998.

[218]. S. Winkler, *Digital Video Quality: Vision Models and Metrics*, Wiley Press, 2005.

[219]. S. K. A. Yeung, S. Zhu and B. Zeng, "Perceptual Video Encryption using multiple 8×8 Transforms in H.264 and MPEG-4," in Proc. *International Conf. on Acoustics, Speech, and Signal Processing 2* (ICASSP 2011), 2011, pp. 2436-2439.