

Republic of Iraq
Ministry of Higher Education
And Scientific Research
University of Babylon
College of Engineering
Department of Electrical Engineering



VIDEO COMPRESSION TECHNIQUE BASED ON THREE-DIMENSION TRANSFORMATION ALGORITHMS

A Thesis

*Submitted to the department of Electrical Engineering Faculty of
Engineering University of Babylon in Partial Fulfillment
of the Requirements for the Degree of Master in
Engineering / Electrical Engineering /Electronic*

By

Dhrgham Hani Mohammed Obaid

Supervised by

Prof. Dr. Laith Ali Abdul-Rahaim

2021 A.D

1443A.H

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ
قَالُوا سُبْحَانَكَ لَا عِلْمَ
لَنَا إِلَّا بِمَا عَلَّمْتَنَا إِنَّكَ
أَنْتَ الْعَلِيمُ الْحَكِيمُ

صدق الله العظيم

آية (32) من سورة البقرة

Examining Committee Certificate

We certify that we have read this entitled "**VIDEO COMPRESSION TECHNIQUE BASED ON THREE-DIMENSION TRANSFORMATION ALGORITHMS**" and as an examining committee, examined the student **Dhrgham Hani Mohammed** in its contents and that in our opinion it meets the Electronic

Signature:

Name: **Prof. Dr.**

Osama Qasim Jumah Al-Thahab

(chairman)

Date: / /2021

Signature:

Name: **Asst. prof. Dr.**

Moneer Ali Lilo

(member)

Date: / /2021

Signature:

Name: **Asst. prof.**

Sameer Abdul-Kadhim Al-Rufaiaat

(member)

Date: / /2021

Signature:

Name: **Prof. Dr.**

Laith Ali Abdul-Rahaim

(supervisor)

Date: / /2021

Signature:

Name: **Asst. Prof.**

Dr. Shamam Fadhil Alwash

(Head of Electrical Department)

Date: / /2021

Signature:

Name: **Prof.**

Dr. Hatem Hadi Obeid

(Dean of College of Engineering)

Date: / /2021

Supervisors Certification

*I certify that this thesis titled “**VIDEO COMPRESSION TECHNIQUE
BASED ON THREE-DIMENSION TRANSFORMATION ALGORITHMS**” was
prepared by **Dhrgham Hani Mohammed Obaid** under my supervision at the
Electrical Engineering Department, College of Engineering Babylon University, in
a partial fulfillment of the requirements for the Degree of Master in Engineering /
Electrical Engineering /Electronic*

Supervisors

Signature:

Name: Prof. Dr. Laith Ali Abdul-Rahaim

Date: / /2021

In view of the above recommendation, I am forward this thesis for
discussion by the Examination Committee.

Head of the Electrical Engineering Department

Signature:

Name: Assist. Prof .Dr. Shamam Alwash

Date: / 2021

Dedication

Dedicated to
my family

Acknowledgements

My thanks are wholly devoted to God, who has helped me all the way to complete this work successfully.

I would like to express my heartfelt gratitude to my supervisor, Prof. **Dr. Laith Ali Abdul-Rahaim** for his support, encouragement, and valuable guidance.

I am deeply grateful to my family for their tireless love and support throughout my life, and for providing me the best possible environment to continue; this work would simply be impossible without them. I am indebted to **my father** for his care and love (may God have mercy on him), and thanks to **my mother** for all the moral support and prayers she has given me over the years. I would also like to thank my **brothers and sisters** for their support throughout my academic career.

Finally, I would like to express my heartfelt thanks to all teachers, the staff of the Electrical Engineering Department of the University of Babylon, also thankful to my friends for their constant encouragement.

ABSTRACT

Video compression has become especially important nowadays with the increase of data transmitted over transmission channels, the reducing the size of the videos must be done without affecting the quality of the video. This process is done by cutting the video thread into frames of specific lengths and converting them into a three-dimensional matrix. The proposed compression scheme uses the traditional red-green-blue color space representation and applies a three-dimensional discrete Fourier transform (3D-DFT) or three-dimensional discrete wavelet transform (3D-DWT) to the signal matrix after converted the video stream to three-dimensional matrices. The resulting coefficients from the transformation are encoded using the EZW encoder algorithm. Three main criteria by which the performance of the proposed video compression system will be tested; Compression ratio (CR), peak signal-to-noise ratio (PSNR) and processing time (PT). Experiments showed high compression efficiency for videos using the proposed technique with the required bit rate, the best bit rate for traditional video compression. 3D discrete wavelet conversion has a high frame rate with natural spatial resolution and scalability through visual and spatial resolution as well as in quality with other advantages when compared to current conventional systems in terms of complexity, low power, high throughput, low latency and minimum storage requirements.

By working on the proposed system, results were obtained with Compression Ratio ranging between (95.51 - 87.445).

For SNPR, the results were between (47.78-51.4) dB.

It can be noted that the value of MSE is (0.48-1.08).

It can note that the similarity value range is between (94.1-95.3).

Contents	<i>Page No.</i>
Acknowledgement	I
Abstract	II
Contents	IV
List of Figures	VII
List of Tables	IX
List of Abbreviations	X
Chapter One Introduction To Multimedia Compression	<i>Page No.</i>
1-1 Introduction of Data Compression	1
1.2 Motivation	2
1.3 Literature Survey	3
1.4 Thesis Aim	7
1.5 Thesis Organization	8
Chapter Two Concepts of Multimedia Compression	<i>Page No.</i>
2.1 Introduction	9
2.2 Compression Techniques	9
2.2.1 Lossless Compression	10
2. 2.2 Lossy Compression	12

2.2.3 Embedded Zerotrees of Wavelet transforms	14
2.3 Multimedia Compression	16
2.3.1 Image Compression	18
2.3.2 Video Compression	19
2.3.3 The Problem of Video Compression	19
2.3.4 Still Image Compression	21
2.4 Matching Criteria for Motion Estimation	25
2.5 Compression Performance Evaluation	26
2.6 Wavelet Transform	28
2.6.1 Continuous Wavelet Transform	30
2.6.2 The Discrete Wavelet Transform (DWT)	30
2.6.3 2D-DWT for Image Decomposition and Reconstruction	34
2.6.4 Computation of FDWT for 3-D Signal	35
2.6.5 A General Example for Computing 3-D DWT	37
2.7 Inverse Discrete Wavelet Transform	38
2.8 Wavelet Compression Versus Other Compression Techniques	39
2.9 Video Compression System	40
2.10 Applications of Compressed Video	43
2.11 Video Compression Standards	44

CHAPTER THREE Video Compression Using Discrete Wavelet Transform	<i>Page</i> <i>No.</i>
3.1 Introduction	47
3.2 Three-Dimensional Set Partitioned Embedded Block (3dspeck)	48
3.3 Quantization	52
3.4 The proposed compression method using 3D-DWT	53
3.4.1 The Modified Proposed Compression System	54
3.4.2 The Compression System Procedure	56
CHAPTER FOUR SIMULATION AND RESULTS	<i>Page</i> <i>No.</i>
4.1 Introduction	57
4.2 The Case Results	59
4.2.1 Case (1) Results	59
4.2.2 Case (2) Results	65
4.2.3 Case (3) Results	69
4.3 Discussion	73
CHAPTER FIVE CONCLUSION AND FUTURE WORK	<i>Page</i> <i>No.</i>
5.1 Conclusion	76
5.2 Future Works	78
References	79

LIST OF FIGURES		
2.1	Huffman Coding	11
2.2	Huffman code extract	11
2.3	The Effect of a Discrete Cosine Transform	13
2.4	Aspect ratio	17
2.5	The image transmission diagram	18
2.6	Original cameraman picture	22
2.7	Profile of cameraman image along row number 164 .	13
2.8	A two-level 2D DWT of cameraman image	23
2.9	Cameraman image compressed using one-level 2D DWT	25
2.10	Haar Wavelet	28
2.11	A 3 level filter bank	32
2.12	Frequency domain representation of the DWT	32
2.13	One-level ,2D wavelet decomposition (Analysis)	33
2.14	One level, 2D-DWT Decomposition and Reconstruction	33
2.15	Three level wavelet decomposition of an image	34
2.16	Frequency subbands produced by single level of wavelet decomposition of a 3-D image.	36
2.17	Cross section of the ctbrain.vols data set.	36
2.18	One level , One dimensional DWT Reconstruction	39
2.19	JPEG compression and decompression algorithms	41
2.20	MPEG compression algorithm for predictive frames MPEG adds motion estimation to the JPEG model	42
2.21	Octave-band or wavelet decomposition of a still image into unequal sub bands	42
2.22	XYZ compression algorithm	45

3.1	Structure for 3 DSPECK. The numbers on the front lower left corners for each subband are marked to indicate the sorting order	48
3.2	(a)hyperspectral image\coast";(b) non-hyperspectral image\chess".	49
3.3	Video cube from 8 frames	53
3.4	The proposed video compression and decompression system	54
3.5	The Proposed modified video compression & decompression system	55
4.1	Compress-frames for the Rhinos Video.	59
4.2	The size of original video	60
4.3	The size of compressed video	61
4.4	Size frames for the Rhinos Video a)before b)after	62
4.5	Represents the comparison between the original image and the compressed image	62
4.6	PSNR for 40 compress-frames for the Rhinos Video.	63
4.7	MSE for 40 compress-frames for the Rhinos Video.	64
4.8	SSIM for 40 compress-frames for the Rhinos Video.	64
4.9	Compress-frames for the Nature Video.	65
4.10	Size frames for the Nature Video a)before b)after	67
4.11	PSNR for 40 compress-frames for the Nature Video.	67
4.12	MSE for 40 compress-frames for the Nature Video.	68
4.13	SSIM for 40 compress-frames for the Nature Video	68
4.14	Compress-frames for the Music Video.	69
4.15	Size frames for the Music Video a)before b)after	71
4.16	PSNR for 40 compress-frames for the MUSIC Video.	72
4.17	MSE for 40 compress-frames for the MUSIC Video.	72
4.18	SSIM for 40 compress-frames for the MUSIC Video.	73

LIST OF TABLES

2.1	Overview of video compression algorithms	20
2.2	Levels of Frequency domain representation of DWT	32
2.3	summarizes applications of the compressed video by specifying current standards used in various applications, the required bandwidth and typical frame sizes and frame rates	44
4.1	Compression ratio for videos with different resolutions	75

List of Abbreviations	
AVI	Audio Video Interleaved
AWGN	Additive white Gaussian noise
BER	Bit Error Rate
BPP	Bit Per Pixel
BR	Bit Rate
CF	Compression Factor
CR	Compression Ratio
CS	Color Space
CDMA	Code Division Multiple Access
CPM	Continuous Phase Modulation
CWT	Continuous Wavelet Transform
Db	Unit measurement of SNR(decibels)
DCT	Discrete Cosine Transform
DFT	Discrete Fourier Transform
DMWT	Discrete Multi Wavelet Transformation
DMWTCS	Discrete Multi Wavelet Transform critical Sampled
DWT	Discrete Wavelet Transform
EZW	Embedded Zero tree Wavelet
FFT	Fast Fourier Transform
FPGA	Field Programmable Gate Array
IDCT	Inverse Discrete Cosine Transform
IDWT	Inverse Discrete Wavelet Transform
JPEG	Joint Photographic Experts
LTE	Long Term Evolution
MAE	Mean Absolute Error
MSE	Mean Square Error
MPEG	Moving Picture Expert Group

MRA	Multi Resolution Analysis
OFDM	Orthogonal Frequency Division Multiplexing
PSNR	Peak Signal to Noise Ratio
PT	Processing Time
RGB	Red, Green, and Blue
RLE	Run Length Encoding
SSIM	Structural Similarity Index
SAE	Sum Absolute Error
SPIHT	Set Partitioning In Hierarchical Tree
YCrCb	Luminance /Chrominance/ Chrominance Components
YUV	Luminance/Chrominance components
1D	One Dimension
2D	Two Dimension
3D	Three Dimension

symbol Table	
n_1	The number of information carriers (bits) in the original data
n_2	The number of information carriers (bits) in and compressed data
C_{ij}	Sample of the current block.
R_{ij}	Sample of the reference area
$C_{0,0}$	The top samples in the current and reference areas
$R_{0,0}$	The left samples in the current and reference areas
N	Number of Bits
$I(i, j)$	Consider the decompressed image
$\hat{I}(i, j)$	Consider the compressed image
a	Scale parameters
b	Shifting parameters
$\Psi_{a,b}$	The mother wavelet
C	wavelet coefficients which are function of scale and position
$D_{r,k}$	Wavelet coefficients.
$\Phi(x)$	Scaling function.
h_k	Scaling function coefficients
g_k	Wavelet filter coefficients

CHAPTER ONE

Introduction to Multimedia Compression

CHAPTER ONE

Introduction to Multimedia Compression

1-1 Introduction of Data compression

Data compression is the process of converting data files into files with smaller sizes for efficiency of storage and transmission. As one of the essential technologies of the multimedia revolution, data compression is the key in the rapid progress being made in information technology [1].

The history of image data compression started probably about a half of century ago with the works on predictive coding and variable length codes. In the 1960's, advances in digital processing simulated research on compression techniques. These paid their fruits in the marvelous results of the packages issued in the 1980's [1].

Such a technological breakthrough resulted in efficient compression algorithms that have not been standardized until the early 1990's. Presently, they are in common use in multimedia together with modifications and improvements achieved during the last decade [1].

Image and video data compression refers to a process by which the amount of data used to represent an image or a video is reduced to meet a bit rate requirement (below or at most at the maximum available bit rate), awhile the quality of the reconstructed image or video satisfies a certain criteria for a specific application and the complexity of the computation involved should be affordable for that application[1].

1.2 Motivation

All applications of digital video require digital processing for data compression. Some applications require additional processing for motion analysis, standards conversion, enhancement, and restoration to obtain better-quality images. Digital processing of still images have become vital in military, commercial, and consumer applications. Some examples of digital processing are space missions, surveillance imaging, and night vision. Digital video processing differs from still image processing because each frame is processed independently. The utilization of the temporal redundancy by means of multi frame processing techniques enables us to develop more effective algorithms such motion compensation prediction[2].

Multimedia compression is important for many applications that involve huge data storage, transmission and retrieval such as for compress color image, documents, medical imaging and videoconferencing. Uncompressed multimedia require considerable storage capacity and transmission bandwidth.

The objective of multimedia compression technique is to reduce redundancy of the images data in order to be able to store or transmit data in an efficient form. This results in the reduction of file size and allows more data to be stored in the same amount of disk or memory space [3].

This work suggests a multimedia compression scheme with pruning proposal based on Discrete Wavelet Transformation (DWT). The effectiveness of the algorithm has been justified over some real videos and images, and the performance of the algorithm has been compared with other common compression standards.

The compression that is achieved can be calculated by the compression ratio(CR) according to the following formula:

$$CR=n1/n2$$

1-1

Chapter 1: Introduction to Multimedia Compression

Where n_1 and n_2 refer to the number of information carriers (bits) in the original data and compressed data, respectively. A compression ratio such as 10 (or 10: 1) indicates that the original file contains 10 units that carry information (such as bits) for every 1 unit in the compressed data set[4]

Multimedia compression is used in many applications to reduce the information required for representation a series of images. There is a many variety of applications for image and video compression ranging from high definition TV, it has a compressed data rate of several Mbit/s, to reduce the power consumption wireless video transmission at several tens of Kbit/s. Many multimedia compression algorithms are available they depends on application requirements such as the required amount of compression, the type of image and video source, and the required resiliency to errors. This many group of compression techniques makes programmable implementations particularly attractive[5].

1.3 Literature Survey

In this section, some relevant work developed will be mentioned and discussed: Hosam Khalil, 1999. Proposed three-dimensional (3D) video compression based on wavelets decomposition along the temporal axis dictates that a number of video frames must be buffered to allow for the temporal decomposition[6].

V. Rajaravivarma, 1996, the researcher found a step towards reducing the intensity of computations employing fast transform algorithm. In order to work with audio and video image signals, the Discrete Cosine Transform (DCT) is considered. Reducing the intensity of computations and to have amount of data needed to represent the signal, leads to compressed data that could be processed in real-time[7].

Benoit. Le_ Bonhomme, etal, 2008, They have presented the functionalities of the MMW. com (MyMultimediaWorld.com) 3D compression benchmark. The online nature of the MMW. com allowing to continuously updating the database, as well

Chapter 1: Introduction to Multimedia Compression

as the easiness of including new algorithms based on its (Application Programming Interface)API[8].

Chung-Ping Wu, Member, 2005. Two specific encryption schemes are obtained by applying this approach to the Huffman coder and the QM coder. It is shown that security is achieved without sacrificing the compression performance and the computational speed. That modified entropy coding methodology can be applied to most modern compressed audio and video such as MPEG audio, MPEG video, and JPEG/JPEG2000 images[9].

Hanadi Hakami and Zenon Chaczko, they have algorithm that capable of selecting the final threshold value of every image which provides sufficient input for compression, and makes the algorithm an efficient outcome and provides the best quality of an image[10].

Jaya Krishna Sunkara, etal, 2012, they have used the ridgelet transform of areal world images in order to produce low-levels features. Ridgelet based features seems to capture efficiently man-made elements within the scene. The ridgelet transform is relatively fast compared to the Gabor filters because it does not require a filter response calculation [11].

Thomas J. Naughton, 2004, He is outlined the results of combining lossy and lossless techniques to the compression of encrypted digital holograms of 3-D objects. The optical encryption technique, based on phase-shift digital holography, is suitable for secure 3-D object storage and transmission applications[12].

Samir B. etal, 2009, the authors have designed and implemented various algorithms for image and video compression using Discrete Cosine Transformation (DCT). These compression algorithms can be used to achieve various levels of compressions. Various compression ratios along with different quality of recovered video were worked out by using video compression

Chapter 1: Introduction to Multimedia Compression

algorithms. DCT abased implementation have proved to be energy preserving during the implementation phase[13].

Eman Abdelfattah and Asif Mohiuddin, 2010. Researchers evaluated the performance of Huffman's compression and Run Length Encoding algorithms with multimedia data. They bused different types of multimedia formats such as images and text. Extensive experiments with different file sizes were bused to compare both algorithms to assess compression ratio and compression time [14].

D Pavani, D V Sai Sudheer, 2012, they suggested a new video compression technique which exploits objectively the temporal redundancy. With the apparent gains in compression efficiency they predict that the proposed technique could open new horizons in video compression domain. So, bit strongly exploits temporal redundancy with the minimum of processing complexity which facilitates bits implementation in video embedded systems[15].

M. Mozammel Hoque Chowdhury,2012, he proposed a new image compression scheme based on the discrete wavelength transformation in this paper, which provides high enough compression ratios with no significant deterioration in image quality. The effectiveness and robustness of this approach was justified by using a set of real images[16].

K. Sureshraj, 2014, the author applied the Discrete Wavelet Transform (DWT) method for image band video compression. He also calculated the PSNR band CR values after image reconstruction using the Inverse Discrete Wavelet Transform (IDWT) algorithm. Although the video looks like continuous motion, it is actually a series of still images[17].

Andrew Secker and David Taubman, 2001, the researchers proposes a new framework for levitation-based temporal motion compensated DWT. This new transformation is amenable to any motion model and gives excellent initial results

Chapter 1: Introduction to Multimedia Compression

even with simple block-based animation. It also produces high-quality, low-resolution time frames for time scalability[18].

Xiaoli Tanga, et al, 2015, the authors proposed a segmented and compact 3D array for hyperspectral image compression. The 3D waveform automatically takes advantage of inter and depend. Modified Set Partitioned Embedded bloCK (SPECK) sorting algorithm can efficiently find the most important information, thus providing cryptographic gain against 2D compression algorithm. The proposed Three-Dimensional Set Partitioned Embedded bloCK (3DSPECK) is fully included band can be bused for progressive transmission. These features make the proposed programmer a good candidate for compressing (encoding) hyperspectral images before transferring and decompressing (decoding) them at another end for image storage [19].

Divya R. Jariwala, Heta S. Desai, 2016, have been proposed a resolution enhancement technique based on interpolation of high-frequency sub band images obtained by DWT and input image. Most of the techniques bare useful for altering the gray level values of individual pixels and hence the overall contrast of the entire image. But they usually enhance the whole image in a uniform manner which in many cases produces undesirable results[20].

Sai Virali Tummala, 2017, performed compression with both lossy and lossless techniques and then followed by enhancing them. DCT, DWT, RLE and BTC are used for compression. DWT lossy compression gives better results than DCT when enhanced based on PSNR, MSE and SSIM without losing more information. RLE and BTC are compress well without loosing much data. RLE shows good compression rate than BTC from the analysis[21].

Thirumoorthi C, Karthikeyan T, Biomedical Research 2017. The authors simulated six methods of compressing images. Those methods are Karhunen-Loeve Transform (KLT), Walsh-Hadamard Transform (WHT), Fast Fourier

Chapter 1: Introduction to Multimedia Compression

Transform (FFT), Sparse Fast Fourier Transform (SFFT), Discrete Cosine Transform (DCT) and Discrete Wave Transform (DWT). The simulation results are presented and the different quality parameters are compared by applying different medical images for a lung cancer computerized tomography (CT) scan. The algorithm of the Discrete Wavelet Transform method (DWT) gave better results such as compression ratio (CR), mean square error (MSE), and peak signal to noise ratio (PSNR) compared to other conversion methods[22].

Siwei M 2019, the authors provided a systematic, comprehensive, and up-to-date review of neural network-based image and video compression technologies. The evolution and development of compression methodologies based on the image and video neural network are presented respectively. More specifically, advanced video coding technologies are presented and discussed by utilizing deep learning and the HEVC framework, which greatly enhance the performance of advanced video coding[23].

1-4 Thesis Aims

The aim of thesis is to design code using Matlab for multimedia compression, it can achieve high compression ratio (CR), acceptable peak signal to noise ratio (PSNR) and favorable processing time.

To achieve these goals, are taken the following steps:

- 1- Design and implementation of video compression code that enables us to achieve high CR and PSNR by using MATLAB (2020a) software.
- 2- One of the main goals of thesis is to provide more space to be used in storing empty information by compressing files and reducing their sizes, which leads to an increase in free space in the disk or any type of storage system.

Chapter 1: Introduction to Multimedia Compression

1-5 Thesis Organization

Thesis is structured as five chapters:

1-Chapter two explains the principles of a video compression, the criteria which is used to evaluate the performance of any compression technique, an introduction about wavelet and multiwavelet transform , then the algorithm and the procedure of multiwavelet critically sampling transformation .

2-Chapter Three is devoted to the proposed video compression system.

3-Chapter four shows the design of the proposed system and the experimental results (tables, figures and reconstructed frames in new files).

4-Chapter five includes the conclusions and some suggestions for future work.

CHAPTER TWO

Concepts of Multimedia Compression

Chapter Two

Concepts of Multimedia Compression

2.1 Introduction

Data compression refers to reduce the amount of data which is used to represent the original data before compression by reducing the size of the original data. The data that has been compressed is eventually transmitted or stored. There are many types of compression, file compression, text compression, audio compression, image compression and video compression. [24].

In order to decompress data, the compressed data is taken and decompressed using the inverse operations conducted in the compressor and extract the original data. Evaluation of compression system depends on the number of times the compressor has compressed the data and the quality of the reconstructed data such as text, quality of audio and resolution of the reconstructed image or video[24].

In this chapter, the general techniques are presented of compression that used in image compression and image decompression technique and the search algorithms used in motion estimation and the parameters for evaluating a compression system.

2.2 Compression Techniques

A Compression algorithm takes an input (Audio, image, video) and generates a representation that hopefully requires fewer bits. There is a reconstruction algorithm that operates on the compressed representation to reconstruct the input.

2.2.1 Lossless Compression

Lossless compression is when the output is identical to the original data, the original data can be recovered exactly from the compressed data. It is used for applications that cannot tolerate any difference between the original and reconstructed data. Examples of lossless methods are Run Length coding, Huffman coding, and Arithmetic coding [25].

A . Run Length Encoding

Run Length Encoding(RLE) is one of the simplest data compression algorithms. It is a data compression for characters that have more than three repetitions.

RLE finds runs of repeated characters in the input stream and replaces them with a three-byte code. The code consists of a flag character, a count byte, and the repeated characters. For instance, the string “AAAAAABBCCDDDD” could be more efficiently represented as “* A6BBCCC * D4”. That saves us eight bytes. It does not make sense to represent runs less than three characters in length with a code[25].

B . Huffman Coding

Huffman coding is a data compression technique which has been used also in image compression. Huffman coding is based on the probabilities of the data occurring in the sequence. Symbols which occur more frequently will need fewer bits than symbols with less frequency. Consider we have a pixel symbol sequence consisting of 6 pixels; the probability of occurrence pixels are shown in Figure(2-1) [26].

Original source		Source reduction			
Symbol	Probability	1	2	3	4
a_2	0.4	0.4	0.4	0.4	0.6
a_6	0.3	0.3	0.3	0.3	
a_1	0.1	0.1	0.2	0.3	0.4
a_4	0.1	0.1			
a_3	0.06	0.1	0.1	0.1	0.1
a_5	0.04				

Figure (2-1) Huffman Coding

First, Huffman code sums together the two lowest probability pixels into a new pixel with a new probability ($0.06 + 0.04 = 0.1$), repeating this until there is only one pixel, and the probability is 1 (where is the compression). The reverse step to code each probability with binary code starts with the smallest source and works back to the original source. Given the binary 0 and 1 to the source on the right, then go backward with the same path, adding to the source 0 and 1. The operation is repeated for each reduced source until the original source is reached. The final code appears on the far left of Figure(2-2) [26].

Original source			Source reduction			
Sym.	Prob.	Code	1	2	3	4
a_2	0.4	1	0.4	1	0.4	1
a_6	0.3	00	0.3	00	0.3	00
a_1	0.1	011	0.1	011	0.2	010
a_4	0.1	0100	0.1	0100	0.1	011
a_3	0.06	01010	0.1	0101	0.1	01
a_5	0.04	01011				

Figure (2-2) Huffman code extract

As can be seen from Figure (2-2), the symbol a_2 gives only the code with only 1 bit, and the a_5 has the code with 5 bits. In other words, the symbols that occur more frequently are coded with fewer bits than those symbols with least frequency.

2.2.2 Lossy Compression

Lossy compression is the technique provides much higher compression than lossless compression but allows the output to be different from input data. Since some applications do not require exact reconstruction, lossy compression techniques involve some loss of some non - significant information, therefore; data cannot be recovered or reconstructed exactly. Higher compression ratios can be possible to obtain with lossless compression[25].

A. Transform Coding

Transform coding is a form of block coding done in the transform domain . The image divided into blocks and the transform is calculated for each block . The simplest type of transform is filtering such as Discrete Cosine Transform(DCT), Discrete Wavelet Transform(DWT). The basic encoding method for transform based on compression works as follows:

- i. Image transform:** Divide the source image into blocks and apply the transformations to the blocks. The transform can be DCT, DWT etc.
- ii. Parameter quantization:** The coefficients generated by the transformation are quantized to reduce the amount of information . This step represents the information within the new domain by reducing the amount of data. This process is usually irreversible.
- iii. Encoding:** Encode the results of the quantization. This last step can be error free by using Run Length encoding or Huffman coding.

Transform coding is one of the most useful applications. Combined with other compression techniques, this technique allows the efficient transmission, storage, and display of images that otherwise would be impractical. DCT based Transform Coding is a widely used transform in image processing , especially for compression. Some applications of two Dimensional DCT involve still image compression and compression of individual video frames, while multi - dimensional is mostly used for compression of video streams [27]. There are

three DCT -based on standards that are widely used and accepted worldwide [28]:

- a- JPEG (Joint Photographic Experts Group).
- b-H.261(Video codec for audiovisual services).
- c-MPEG (Motion Picture Experts Group).

JPEG-DCT is a transform coding method comprising four steps. The source image is first partitioned into sub-blocks of size 8x8 pixels in dimension. Then each block is transformed from spatial domain to frequency domain using a 2D-DCT- basis function. The DCT makes it possible to compress the image by concentrating most of the image information in the lower spatial domain , so that the 64 coefficients produced by that equation is represented as in figure(2-3). The coefficient of the top-left-corner is called DC which represents the average of whole image that represents low frequency, the other 63 coefficients is the high frequency which is called AC[29].

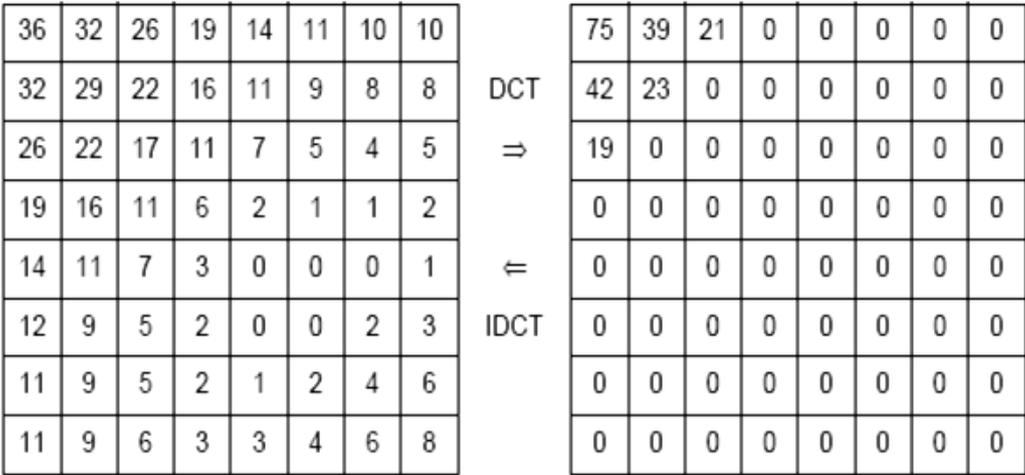


Figure (2-3)The Effect of a Discrete Cosine Transform[30].

In general, these coefficients must be quantified and then compress using one of compression lossless method such as run length coding, Huffman,etc[30].

DCT is efficient in image compression for the following reasons:

1. DCT basis functions are **orthogonal**.
2. DCT has the property of **Separability**.
3. Energy Compaction.

2.2.3 Embedded Zerotrees of Wavelet Transforms

Embedded Zerotrees of Wavelet transforms (EZW) is a lossy image compression algorithm. At low bit rates, i.e. high compression ratios, most of the coefficients produced by a subband transform (such as the wavelet transform) will be zero, or very close to zero. This occurs because "real world" images tend to contain mostly low frequency information (highly correlated). However where high frequency information does occur (such as edges in the image) this is particularly important in terms of human perception of the image quality, and thus must be represented accurately in any high quality coding scheme.

By considering the transformed coefficients as a tree (or trees) with the lowest frequency coefficients at the root node and with the children of each tree node being the spatially related coefficients in the next higher frequency subband, there is a high probability that one or more subtrees will consist entirely of coefficients which are zero or nearly zero, such subtrees are called zerotrees. Due to this, we use the terms node and coefficient interchangeably, and when we refer to the children of a coefficient, we mean the child coefficients of the node in the tree where that coefficient is located. We use children to refer to directly connected nodes lower in the tree and descendants to refer to all nodes which are below a particular node in the tree, even if not directly connected.

In zerotree based image compression scheme such as EZW and SPIHT, the intent is to use the statistical properties of the trees in order to efficiently code the locations of the significant coefficients. Since most of the coefficients will be zero or close to zero, the spatial locations of the significant coefficients make up a large portion of the total size of a typical compressed image. A coefficient

(likewise a tree) is considered significant if its magnitude (or magnitudes of a node and all its descendants in the case of a tree) is above a particular threshold. By starting with a threshold which is close to the maximum coefficient magnitudes and iteratively decreasing the threshold, it is possible to create a compressed representation of an image which progressively adds finer detail. Due to the structure of the trees, it is very likely that if a coefficient in a particular frequency band is insignificant, then all its descendants (the spatially related higher frequency band coefficients) will also be insignificant.

EZW uses four symbols to represent (a) a zerotree root, (b) an isolated zero (a coefficient which is insignificant, but which has significant descendants), (c) a significant positive coefficient and (d) a significant negative coefficient. The symbols may be thus represented by two binary bits. The compression algorithm consists of a number of iterations through a dominant pass and a subordinate pass, the threshold is updated (reduced by a factor of two) after each iteration. The dominant pass encodes the significance of the coefficients which have not yet been found significant in earlier iterations, by scanning the trees and emitting one of the four symbols. The children of a coefficient are only scanned if the coefficient was found to be significant, or if the coefficient was an isolated zero. The subordinate pass emits one bit (the most significant bit of each coefficient not so far emitted) for each coefficient which has been found significant in the previous significance passes. The subordinate pass is therefore similar to bit-plane coding.

There are several important features to note. Firstly, it is possible to stop the compression algorithm at any time and obtain an approximation of the original image, the greater the number of bits received, the better the image. Secondly, due to the way in which the compression algorithm is structured as a series of decisions, the same algorithm can be run at the decoder to reconstruct the coefficients, but with the decisions being taken according to the incoming bit stream. In practical implementations, it would be usual to use an entropy code

such as arithmetic code to further improve the performance of the dominant pass. Bits from the subordinate pass are usually random enough that entropy coding provides no further coding gain.

The coding performance of EZW has since been exceeded by SPIHT and its many derivatives. The EZW algorithm also contains the following features:

- (1) A discrete wavelet transform which can use a compact multiresolution representation in the image.
- (2) Zerotree coding which provides a compact multiresolution representation of significance maps.
- (3) Successive approximation for a compact multiprecision representation of the significant coefficients.
- (4) A prioritization protocol which the importance is determined by the precision, magnitude, scale, and spatial location of the wavelet coefficients in order.
- (5) Adaptive multilevel arithmetic coding which is a fast and efficient method for entropy coding strings of symbols.

2.3 Multimedia Compression

An exact image or still image is represented in a computer as a set of numbers, so that the integers are more specific. An image that stores on a computer is called a digital image. However, The term image is used to give meaning to a digital image. Usually if the image is black and white(BW), it is a two-dimensional(2D) matrix and if it is a color image the matrix is three-dimensional(3D)[31].

Each number in matrix represents the intensity value at a specific location in the image and is called a picture element or a pixel, in other words. Pixel values can be positive integers and can range from 0 to 255. This means that every pixel of a BW image is stored in 1 byte in computer memory. In other words, the image has a gray scale resolution of 8 bits per pixel (bpp). On the other

hand, a color image contains a triangular set of values per pixel: one for each of the primary colors red, green, and blue. Thus, 3 bytes of storage space is needed per pixel.

images captured are rectangular in shape. In standard definition television (SDTV), the aspect ratio is 4:3, while it is 16:9 in High Definition Television (HDTV). The two aspect ratios are illustrated in Fig.2.4, Fig.2.4a corresponds to a 4:3 aspect ratio while Fig.2.4 b corresponds to the same image with an aspect ratio of 16:9. In both images, the height in inches remains the same. [33].

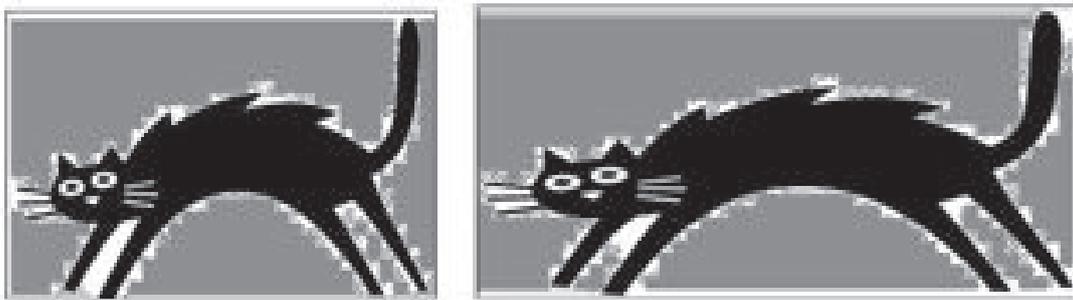


Figure (2-4) Aspect ratio: (a) 4:3 and (b) 16:9. The height is the same in both the pictures.

Which means the number of rows remains constant, So, if the image contains 480 rows, then the number of pixels in each row will be $480 \times \frac{4}{3} = 640$ for a 4:3 aspect ratio for HD TV, there are 1080 rows, so the number of pixels in each row will be $1080 \times \frac{16}{9} = 1920$. Thus, a single SD color image with 24 bpp needs $640 \times 480 \times 3 = 921,600$ bytes of memory, while a high-resolution color image with the same pixel depth needs $1920 \times 1080 \times 3 = 6,220,800$ bytes.

The video source may contain 30 frames per second or more, and in such case the transmitted data rate is 221,184,000 bps for SDTV and 1,492,992,000 bps for HDTV. If this data had to be transmitted in real time through a real communications channel, which would need 1Hz of bandwidth for each 2 bits of raw data, then the required bandwidth would be 110,592,000 Hz for SDTV and 746,496,000 Hz for HDTV. There are no practical channels that allow such a large transmission bandwidth. Note that there are dedicated channels such as HDMI that can transmit uncompressed information at this huge rate over a short

distance that already exist, but we are referring to long distance transmission. It is very evident that effective data compression schemes need to reduce huge video data rates to controllable values, so that practical communication channels can be used to transfer data to desired locations in real time[31].

2.3.1 Image Compression

Image compression is the process of downsizing the file of an image while keeping a good visual quality. The value of the luminance at each pixel in the image is represented to a pre-defined precision(M). Eight bits of precision for luminance is common in imaging applications. The eight-bit per precision is motivated by both the existing computer memory structures(1byte =8 bits) as well as the dynamic range of the human eye[32].

Internet browsing, TV transmission, transmission of remotely sensed images and printing have digital images applications but the vast amount of data required to represent a digital image restricts these applications[33].

Due to the high cost of storage and transmission, digital image application is not practical. The goal of image compression process is to reduce the file size of the image and reduce the bitrate of an image to minimize the communication channel capacity while maintaining the fidelity of the image[30].Figure (2-5) shows the image compression and decompression diagram.

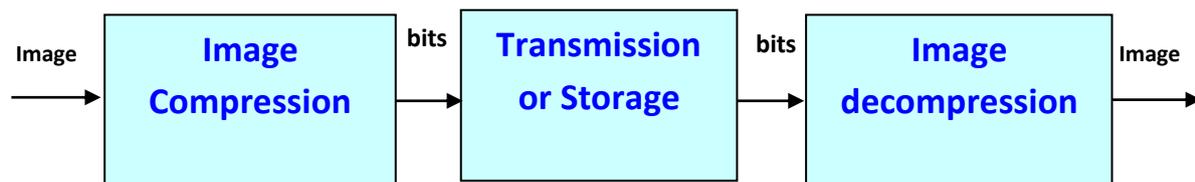


Figure (2-5) The image transmission diagram

2.3.2 Video Compression

Video compression is the process of reducing a video file size and reforming it in to a form that can be stored and played back. It plays a vital role in video processing. Compressed digital video reduces space and bandwidth requirements and therefore is suitable for storage and transmission [36]. The performance of a good video compressor is that the compressed video file is similar to the original video file when played back. A poorly compressed file has a large file size and an unacceptable video quality during playback.

Video Application in telecommunication has been increasingly demandable, but the problem is that image and video data require large data rates typically in the range of 150Mbits/sec[33]. Data rates in these ranges desire large bandwidth during transmission and large file size during storage, therefore; video standards have been developed to eliminate picture redundancy and allow video to be transmitted and stored in an efficient and compact manner. During the '80s and '90s, Discrete Cosine Transform (DCT)- based compression algorithms and international standards were developed to alleviate storage and bandwidth limitations imposed by digital still-image and motion video applications[33].

The basic compression scheme for all video compression standards can be summarized as follows: divide the picture into 8×8 blocks, determine the relevant picture information, discard redundant or insignificant information, and encode relevant picture information with the least number of bits. To express the importance of compression, a 300MHz processor cannot playback a video size of 27MBytes even if the hard disk is not a problem[33].

2.3.3 The Problem of Video Compression

The problem of real-time video compression is a difficult and important one, and has inspired a great deal of research activity. This body of knowledge has been, to a substantial degree, embodied into the MPEG and H. 261/ H263 motion video standards. However, some important questions remain unexplored.

Since about 1989, moving digital video images have been integrated with programs. The difficulty in implementing moving digital video is the tremendous bandwidth required for the encoding of video data. For example, a quarter screen image (320 x 240 pixels) playing on an RGB video screen at full speed of 30 frames per second (fps) requires storage and transmission of 6.9 million bytes per second[34].

This data rate is simply prohibitive, and so means of compressing digital video suitable for real-time playback are a necessary step for the widespread introduction of digital motion video applications .

Many digital video compression algorithms have been developed and implemented. The compression ratios of these algorithms varies according to the subjective acceptable level of error, the definition of the word compression, and who is making the claim. Table 2. 2 summarizes video compression algorithms, their typical compression ratios reported in the literature, and their characteristics [34].

Table 2.1 Overview of video compression algorithms.

Compression Algorithm	Typical Compression Ratio	Characteristics
Intel RTV/Indeo	3:1	A 128X240 data stream is interpolated to 256X240. Color is Subsampled 4:1. A simple 16 bit codebook is used without error correction. Frame differencing is used.
Intel PLV	12:1	A native 256X240 stream is encoded using vector quantization and motion compensation. Compression requires specialized equipment.
IBM Photomotion	3:1	An optimal 8-bit color palette is determined, and run-length encoding and frame differencing are used.
Motion JPEG	10:1	Uses 2-D DCT to encode individual frames. Gives good real-time results with inexpensive but special-purpose equipment. This technique supports random-access since no frame differencing is used.
Fractals	10:1	Fractals compress natural scenes well, but require tremendous computing power.
Wavelets	20:1	2-D and 3-D wavelets have been used in the compression of motion video. Wavelet compression is low enough in complexity to compress entire images, and therefore does not suffer from the boundary artifacts seen in DCT-based techniques.

H.261/H263	50:1	Real-time compression and decompression algorithm for video telecommunications. It is based on 2-D DCT with simple motion estimation between frames.
MPEG	30: 1	Uses 2-D DCT with motion estimation and interpolation between frames. The MPEG standard is difficult and expensive to compress, but plays back in real-time with inexpensive

An ideal video compression technique should have the following characteristics:

- Will produce levels of compression rivaling MPEG without objectionable artifacts.
- Can be played back in real time with inexpensive hardware support.
- Can degrade easily under network overload or on a slow platform.
- Can be compressed in real time with inexpensive hardware support.

2.3.4 Still Image Compression

In a still image, each pixel in a row may have a value very nearly equal to a neighboring pixel value. As an example, consider the cameraman picture shown in Figure(2-6). Figure(2-7) shows the profile (top figure) and the corresponding correlation (bottom figure) of the cameraman picture along row 164. Generating Figure(2-7) is listed below. Observe that the pixel values are very nearly the same over a large number of neighboring pixels and so is the pixel correlation. In other words, pixels in a row have a high correlation.[31]

Similarly, pixels may also have a high correlation along the columns. Thus, pixel redundancies translate to pixel correlation. The basic principle behind image data compression is to decorrelate the pixels and encode the resulting decorrelated image for transmission or storage. A specific compression scheme will depend on the method by which the pixel correlations are removed.

One of the earliest and basic image compression techniques is known as the differential pulse code modulation (DPCM) . If the pixel correlation along only

one-dimension (row or column) is removed, then the DPCM is called one-dimensional (1D) DPCM or row-by-row DPCM. If the correlations along both dimensions are removed, then the resulting DPCM is known as 2D DPCM. A DPCM removes pixel correlation and requantizes the residual pixel values for storage or transmission. The residual image has a variance much smaller than that of the original image. Further, the residual image has a probability density function, which is a double sided exponential function. These give rise to compression [31].

A second and relatively recent compression method is based on wavelet transform.

As we will see in a later chapter, wavelet transform captures both long-term and short-term changes in an image and offers a highly efficient compression mechanism.



Figure (2-6) Original cameraman picture

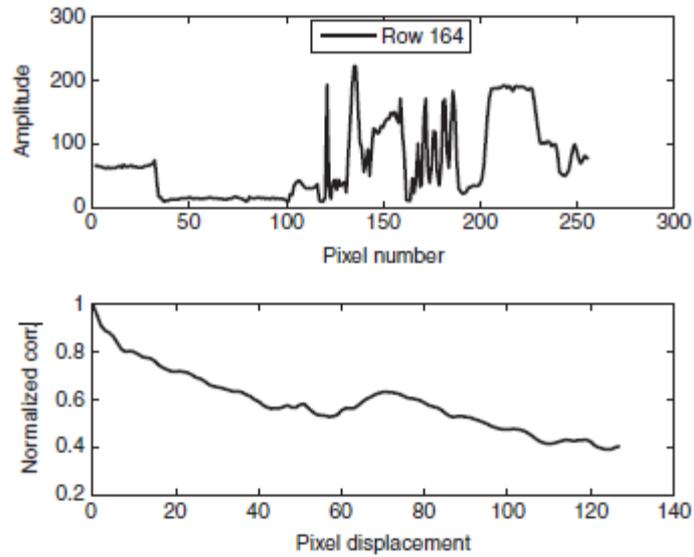


Figure (2-7) Profile of cameraman image along row number 164 . The top graph shows pixel intensity, and the bottom graph shows corresponding normalized correlation over 128 pixel displacements.



Figure (2-8) A two-level 2D DWT of cameraman image.

As a result, it is used in the latest versions of the JPEG standards as a compression tool. It is also adopted by the SMPTE (Society of Motion Pictures and Television Engineers). Even though the wavelet transform may be applied on blocks of an image like the DCT, it is generally applied on the full image and the

various wavelet coefficients are quantized according to their types. A two-level discrete wavelet transform (DWT) of the cameraman image is shown in Figure(2-8) to illustrate how the 2D wavelet transform coefficients look like. Details pertaining to the levels and sub bands of the DWT will be given in a later chapter. Implement multilevel 2D DWT that generates Figure(2-8) is listed below. As we will see in a later chapter, the 2D DWT decomposes an image into one approximation and many detail coefficients.

The number of coefficients sub images corresponding to an L-level 2D-DWT equals $3 \times L + 1$. Therefore, for a two-level 2D-DWT, there are seven coefficient subimages.

In the first level, there are three detail coefficient subimages, each of size $1/4$ the original image. The second level consists of four sets of DWT coefficients one approximation and three details, each $1/16$ the original image. The approximation coefficients are lower spatial resolution approximations to the original image. The detail coefficients capture the discontinuities or edges in the image with orientations in the horizontal, vertical, and diagonal directions. In order to compress, an image using 2D-DWT we have to compute the 2D-DWT of the image up to a given level and then quantize each coefficient sub image. The achievable quality and compression ratio depend on the chosen wavelets and quantization method.

The visual effect of quantization distortion in DWT compression scheme is different from that in DCT-based scheme. (Figure2-9a) is the cameraman image compressed using 2D-DWT. The wavelet used is called Daubechies 2(db2 in MATLAB) and the number of levels used is 1. We note that there are no blocking effects, but there are patches in the flat areas. We also see that the edges are reproduced faithfully as evidenced in the profile (Figure2-9b). It must be pointed out that the amount of quantization applied in (Figure2-9a) is not the same as that used for the DCT example and that the two examples are given only to show the differences in the artifacts introduced by the two schemes[31].

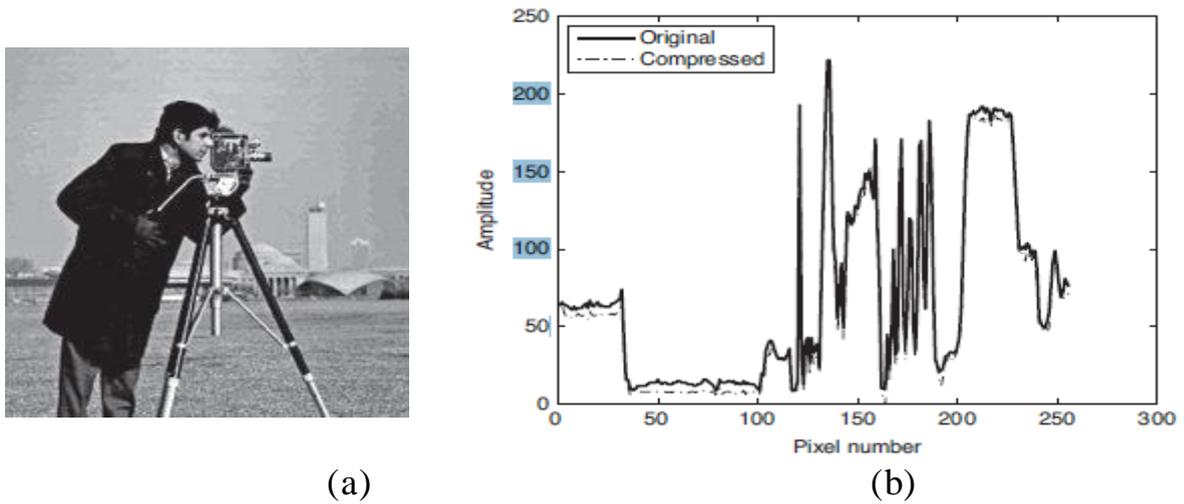


Figure (2-8) (a) Cameraman image compressed using one-level 2D DWT.
 (b) Intensity profile of image along row number 164.

2.4 Matching Criteria for Motion Estimation

Eliminate large amount of temporal and spatial redundancy that exists in video sequences will helps in compressing them. In conventional predictive coding the difference between the current frame and the predicted frame is coded and transmitted. Smaller errors can be achieved by better prediction. Evaluating the goodness of a match is conducted by a number of criterias, block -based motion estimation has three popular criteria:

i. **Mean Squared Error (MSE):**

Mean squared error provides a measure of the energy remaining in the difference block. MSE for a $N \times N$ -sample block can be calculated as follows[38]:

$$MSE = \frac{1}{N^2} \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} (C_{ij} - R_{ij})^2 \quad (2.1)$$

where : C_{ij} , is a sample of the current block.

R_{ij} is a sample of the reference area and $C_{0,0}$ $R_{0,0}$ are the top-left samples in the current and reference areas respectively.

ii. **Mean Absolute Error (MAE):**

Mean absolute error (MAE) provides a reasonably good approximation of residual energy and is easier to calculate than MSE, since it requires a magnitude calculation instead of a square calculation for each pair of samples:

$$MAE = \frac{1}{N^2} \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} |C_{ij} - R_{ij}| \quad (2.2)$$

The comparison may be simplified further by neglecting the term $1/N^2$ and simply calculating the sum of absolute errors (SAE) or sum of absolute differences (SAD):

$$SAE = \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} |C_{ij} - R_{ij}| \quad (2.3)$$

SAE gives a reasonable approximation to matching criterion for block-based motion estimation[35].

iii. **Pel Difference Classification(PDC):**

If the difference is less than some predetermined threshold then there is match. The number of matches is then counted with the help of the Ord (n). If n is true Ord (n) is one and zero if n is false. The smaller the difference is the greater the number of matching pixels, is the better is the match

2.5 Compression Performance Evaluation

The performance of any compression system is measured by three performance parameters:

A. Compression Ratio:

It defined as the ratio of the uncompressed (original) video file size to the compressed file size and the mathematical formula is[36]:

$$\text{Compression Ratio} = \frac{\text{Uncompressed File Size}}{\text{Compressed File Size}} \quad (2.4)$$

Also, it can be written by[39]:

$$\text{Compression Ratio} = (1 - \left(\frac{\text{Compressed_Size}}{\text{Original_Size}} \right)) \times 100 \% \quad (2.5)$$

Another way to state the compression is to use the terminology of bits per pixel. For an NxN image:

$$\text{Bi Per Pixel (BPP)} = \frac{\text{Number of Bits}}{\text{Number of Pixels}} = \frac{\text{Number of bytes} \times 8}{N \times N} \quad (2.6)$$

B. Signal to Noise Ratio:

It is defined as the difference between the original and reconstructed values. Consider the decompressed image $I(i, j)$ to be the “signal” and the error to be “noise”. The signal-to-noise (SNR) metric, the larger number implies a better image. We can define the root-mean-square signal-to-noise ratio as:

$$SNR_{RMS} = \sqrt{\frac{\sum_{i=0}^{N-1} \sum_{j=0}^{N-1} [\hat{I}(i, j)]^2}{\sum_{i=0}^{N-1} \sum_{j=0}^{N-1} [\hat{I}(i, j) - I(i, j)]^2}} \quad (2.7)$$

Another related metric which is called the peak signal to noise ratio, is defined as:

$$PSNR(dB) = 10 \log_{10} \frac{(B-1)^2}{\frac{1}{N^2} \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} [\hat{I}(i, j) - I(i, j)]^2} \quad (2.8)$$

Where B=the number of gray levels (e.g., for 8 bits B=256).

PSNR is the most commonly used value to evaluate the objective image compression quality.

C. Processing Time (PT)

The third parameter to measure the system efficiency is the time required by the system to compress the video file. As long as the time required by the system is less, the system is more efficient and fast.

2.6 Wavelet Transform

Wavelet means small wave. The original term was “Ondelets”, which is French. and the figure (2-10) is one example of Haar wavelet and its mean is zero[37].

A few years later, Ingrid Daubechies constructed families of orthonorma and Stephane Mallet put the wavelet transform in the framework of multi-resolution signal decomposition[38].

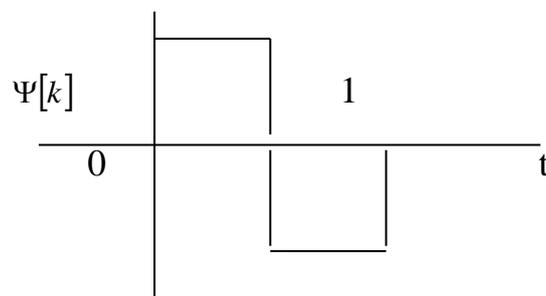


Figure (2-10) Haar Wavelet

The fundamental idea behind wavelets is to analyze the signal at different resolutions, which is called multi-resolution. Wavelets are a class of functions used to localize a given signal in both space and scaling domains. A family of wavelets can be constructed from a mother wavelet. Compared to Windowed Fourier analysis, a mother wavelet is stretched or compressed to change the size of the window. In this way, big wavelets give an approximate image of the signal, while smaller and smaller wavelets zoom in on details. Therefore, wavelets automatically adapt to both the high frequency and the low-frequency components of a signal by different sizes of windows. Any small change in the wavelet representation produces a correspondingly small change in the original signal, which means local mistakes will not influence the entire transform. The

wavelet function may be either continuous or discrete for our purposes, the discrete form is suited for non-stationary signals[39].

The wavelets analyze the input signal in sections by translation of analysis function. The analysis function in wavelet transform is a wavelet function, Ψ . The wavelet function is scaled and translated in time. The Ψ is often called the mother wavelet[36], because it generates a family of wavelets through dilation and translation. A generalized wavelet family, $\Psi_{a,b}$, described in the normalized form is[40]:

$$\Psi_{a,b}(x) = \frac{1}{\sqrt{a}} \Psi\left(\frac{x-b}{a}\right) \quad (2.9)$$

Where a represents the scale and b represents the translation (shifting) parameters, and the constants the scale $(1/\sqrt{a})$ is used for energy normalization throughout different scales.

The scale parameters, a , indicates the level of analysis. Small values of a provide small scale high frequency analysis while large values correspond to large scale low-frequency, analysis. Changing the b parameter moves the time localization center of each wavelet.

Typically, the scale factor between levels increases by two. Thus, scaling is also known as **dilation**. Widely used a and b parameter setting that create an orthonormal basis are $a=2^r$ and $b=2^r$. The wavelet family then becomes[41]:

$$\Psi_{r,k}(x) = 2^{-\frac{r}{2}} \cdot \Psi[2^{-r}x - k] \quad (2.10)$$

Functions of the wavelet transform was put in the framework of multi-resolution analysis the importance of these wavelet bases became obvious [42]

2.6.1 Continuous wavelet Transform

The continuous wavelet transform is defined as the sum over the times of signal multiplied by scaled, shifted versions of the wavelet function.

$$C(\text{scale}, \text{position}) = \int_{-\infty}^{-\infty} f(t)\Psi(\text{scale}, \text{position}, t) \quad (2.11)$$

C: are wavelet coefficients which are function of scale and position[43].

2.6.2 The Discrete Wavelet Transform (DWT)

The wavelet transforms compute the wavelet coefficients by taking the inner product of an input signal $f(x)$ with a function, in this case the wavelet family, $\Psi_{r,k}(x)$. The discrete wavelet transform (DWT) is:

$$D_{r,k} = \langle f, \Psi_{r,k} \rangle = 2^{-\frac{r}{2}} \int_{-\infty}^{\infty} f(x)\Psi(2^{-r}x - k)dx \quad (2.12)$$

where $D_{r,k}$ are the **wavelet coefficients**.

The wavelet coefficients are measures of the goodness to fit between the signal and the wavelet [35]. Wavelet functions are constructed from a father wavelet, or scaling function, Φ .

$$\Phi(x) = \sqrt{2} \sum_{k \in \mathbb{Z}} h_k \Phi(2x - k) \quad (2.13)$$

Where: $\Phi(x)$ is scaling function.

h_k are the scaling function coefficients.

From the scaling function, Φ , it is possible to construct an orthonormal wavelet Ψ such that a signal can be decomposed (analyzed) and reconstructed exactly and efficiently.

$$\Psi(x) = \sqrt{2} \sum_{k \in \mathbb{Z}} g_k \Phi_{j,k}(2x-k) \quad (2.14)$$

where $\Psi(x)$ represents the mother wavelet and g_k represents the wavelet filter coefficients defined by:

$$g_k = (-1)^k h_{1-k} \quad (2.15)$$

Thus, the wavelet function is obtained by convolving the scaling function with the reversed signed form of the scaling filter. The general equation for calculating the wavelet function at any level r is given by:

$$\Psi_{r+1,0}(x) = \sum_{k \in \mathbb{Z}} g_k \Phi_{r,k}(x) \quad (2.16)$$

The wavelet function creates a high-pass wavelet filter (g_k) that provides the **detail coefficients**. The scaling function creates a low-pass wavelet filter (h_k) that provides the **approximation coefficient**.

Convolving the input signal with the wavelet function creates the detail coefficients. The approximations coefficients are calculated in the same way by taking the inner product of the signal, f , and the family of dilated, j , and translated, k , scaling functions:

$$A_{r,k} = \langle f, \Phi_{r,k} \rangle = 2^{-\frac{r}{2}} \int_{-\infty}^{\infty} f(x) \Phi(2^{-r}x - k) dx \quad (2.17)$$

Equation (2.12) and (2.17) define the procedure for complete signal decomposition using wavelet. Although these equations can be implemented algorithmically and would provide accurate results, they don't provide efficient signal decomposition [44].

The decomposition is repeated to further increase the frequency resolution and the approximation coefficients decomposed with high and low pass filters and then down-sampled. This is represented as a binary tree with nodes representing

a sub-space with a different time-frequency localization. The tree is known as a filter bank.[31]

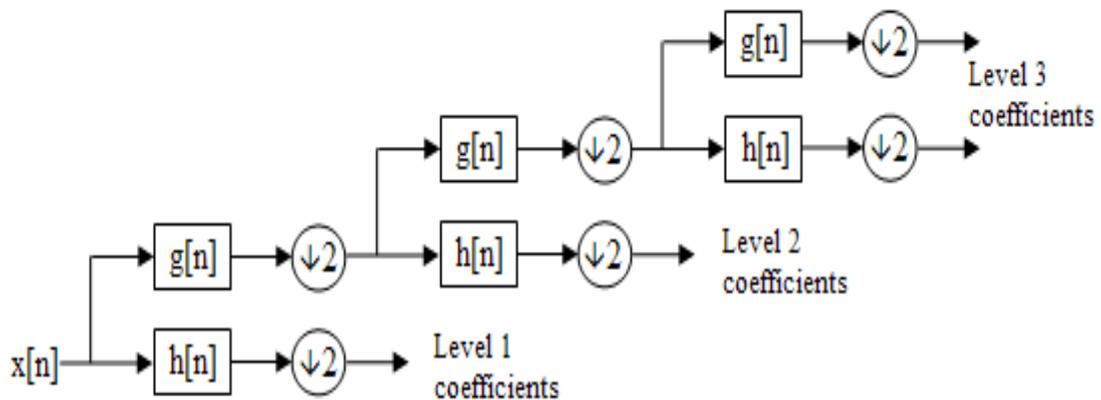


Figure (2-11) A 3 level filter bank

At each level in the above diagram the signal is decomposed into low and high frequencies. Due to the decomposition process the input signal must be a multiple of For example, a signal with 32 samples, frequency range 0 to f_n and 3 levels of decomposition, 4 output scales are produced:

Table 2.2 Levels of Frequency domain representation of the DWT

Level	Frequencies	Samples
3	0 to $f_n/8$	4
3	$f_n/8$ to $f_n/4$	4
2	$f_n/4$ to $f_n/2$	8
1	$f_n/2$ to f_n	16

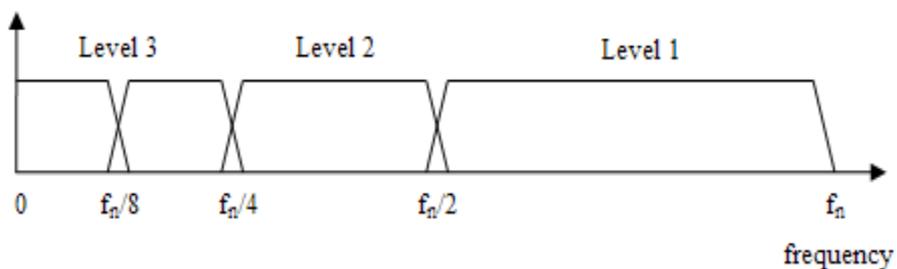


Figure (2-12) Frequency domain representation of the DWT

2.6.3.2D-DWT for Image Decomposition and Reconstruction

A 2D separable discrete wavelet transform is equivalent to two consecutive 1D transforms. For an image, a 2D is implemented as a 1D-row transform followed by a 1D-column transform.

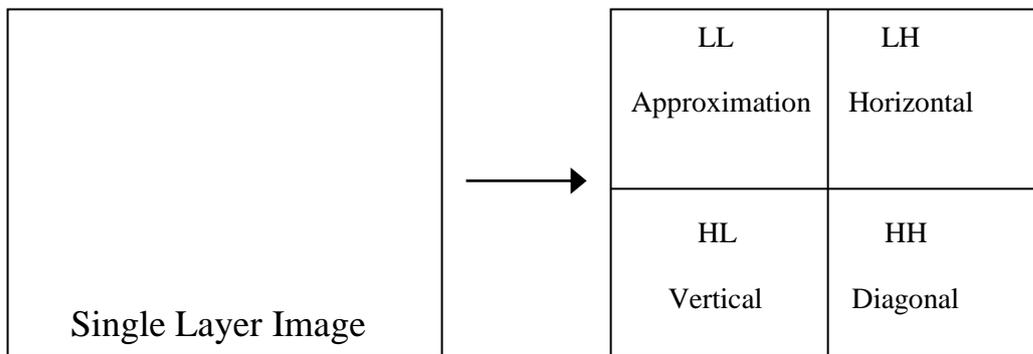


Figure (2-13): One-level 2D wavelet decomposition (Analysis).

The scaling function and wavelet functions corresponding to different sub-bands in the decomposition. A single stage of a 2D-filter bank is shown in figure (2-14):

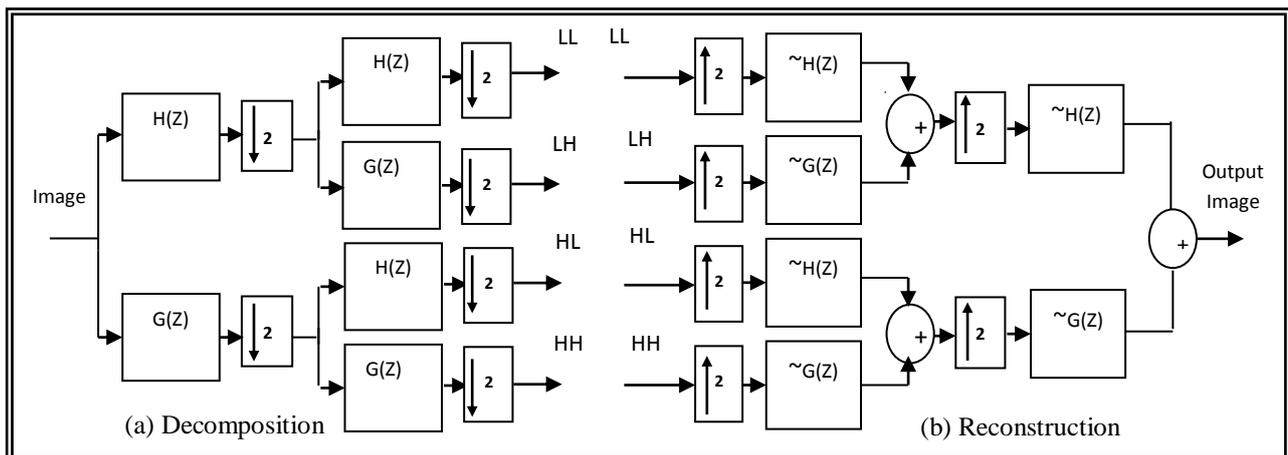


Figure (2-14) One level, 2D-DWT Decomposition and Reconstruction

First, the rows of the input image are filtered by the high pass and low pass filters. The outputs from these filters are down sampled by two, and then the columns of the outputs are filtered and down sampled to decompose the image into four sub-bands. The synthesis stage performs up-sampling and filtering to reconstruct the original image.

Multiple levels of decomposition are achieved by iterating the analysis stage on only the LL band. For (L) levels of decomposition, the image is decomposed into $3L+1$ sub-bands [43].

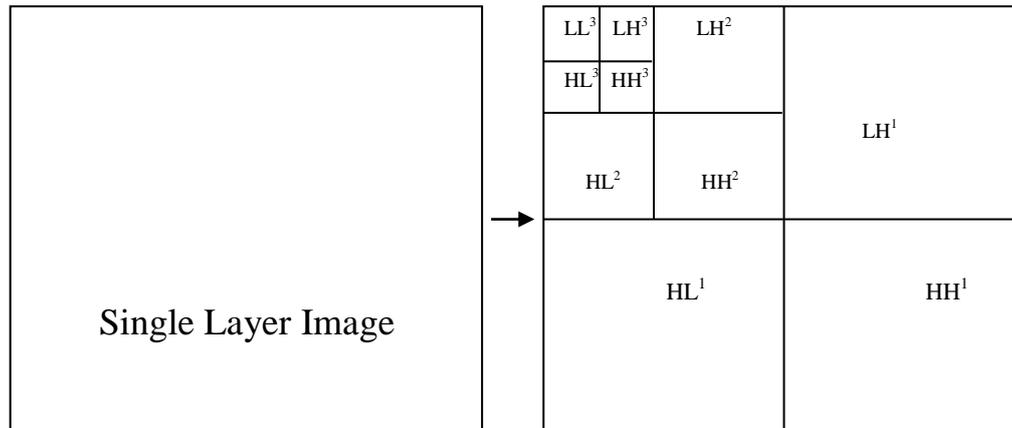


Figure (2-15) Three level wavelet decomposition of an image.

The importance of wavelet transform comes from the principle that wavelet transform coding decompose an image into multi-resolution sub-images. Each sub-image corresponds to a different frequency and energy distribution bands by using multiple scale wavelet bases, which results in flexible time and frequency resolution. These bands are high frequency band with little energy information (corresponding Details bands) and low frequency bands with high-energy information (corresponding Approximation bands).

2.6.4 Computation of FDWT for 3-DSignal

With the advent of the digital age, there has been a rapid increase in the amount of video information available. With the growth in availability of video data, the need to query and access the relevant data becomes critical. The videos have to be stored in a highly compressed form because of the huge storage requirements. Consequently, any efficient video retrieval or matching algorithm has to use the compressed information directly. A classical 3-D Haar wavelet transform (3D-HWT) algorithms are based on a groups of frames (GOFs) for its temporal decomposition. The use of this GOF concept has some disadvantages in terms of efficiency, processing and memory requirement that may limit their practical

implementation. The classical 3D-HWT algorithm works on a group of 4 frames at a time. It starts the temporal decomposition taking the first four frames in the original sequence and it gets two low and two high frequency frames. Then, it uses the resulting two low frequency frames and two new frames from the original sequences to perform the next temporal decomposition. This procedure is repeated until the end of the original sequences[44].

The proposed fast 3-D wavelet transform (DWT) algorithm reduces heavily the processing time for decomposition of video sequences keeping or overcoming the quality of reconstructed sequences. In addition, it cut heavily the memory demands[44].

For a 2-D wavelet transformation, the algorithm is applied in x -direction first, and then in y -direction. Similarly, in 3-D wavelet transformation the structures are defined in 3-D and the transformation algorithm is applied in x -, y - and z -direction successively.

To apply this hierarchical scheme to volume data sets as shown in (Fig2-16a), a 3-D wavelet transformation must be implemented. If this algorithm is applied to data set (*ctbrain. vols*) as shown in Fig(2.17), which consists of 512 x 512 x 231 elements, there is need to scale the size of the array to the closest powers of two, ie., 512 x 512 x 256. The algorithm is initially run in x -direction, row by row for all 231 slices. The algorithm splits the volume into two halves, the left half representing the low-frequency coefficients while the right half represents the detail coefficients, as shown in Fig.(2.16b). In the second stage of the algorithm, the entire volume is then again transformed in y -direction splitting the volume into four quadrants as shown in Fig.(2.16c). For the final run, the volume is transformed in z -direction splitting the volume into eight octants. The upper left front octant contains the low-frequency coefficients that are initially transmitted over the network[45].

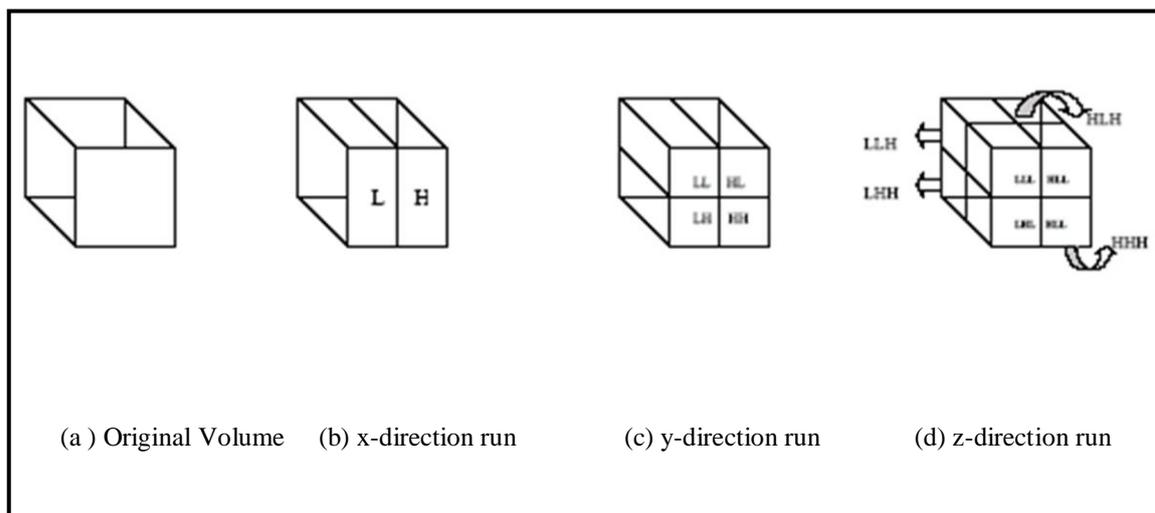


Fig. (2.16) Frequency subbands produced by single level of wavelet decomposition of a 3-D image.

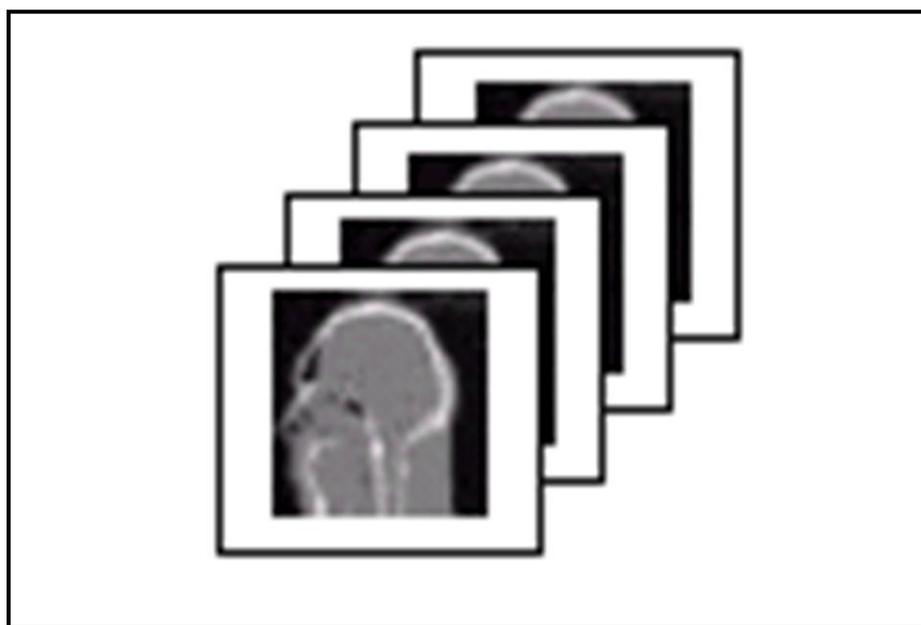
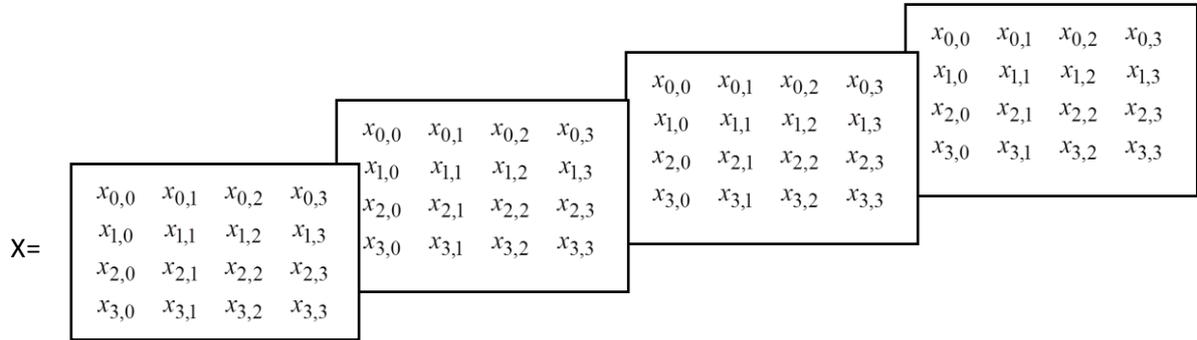


Fig. (2.17): Cross section of the ctbrain.vols data set.

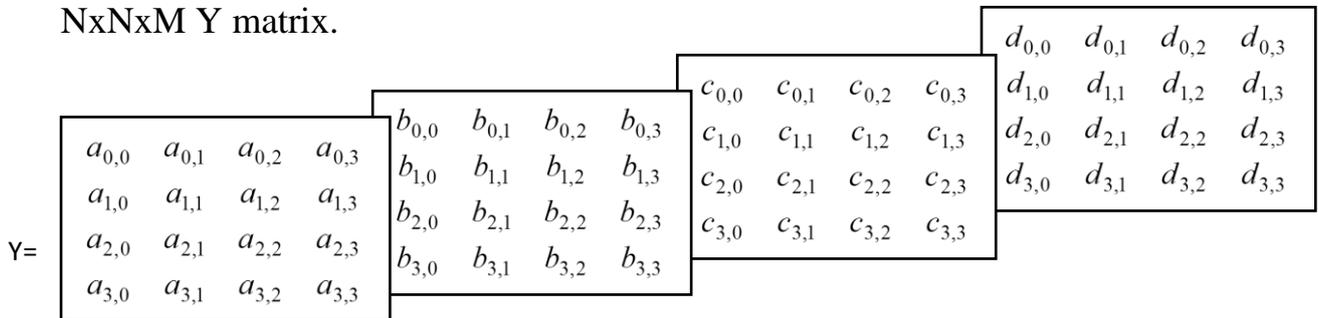
2.6.5 A General Example for Computing 3-D FDWT

Let's take a general 3-D signal, for example any $N \times N \times M$ matrix, and apply the following steps:

1. Let X be the $N \times N \times M$ input 3-D signal,



2. Apply 2-D DWT algorithm to each $N \times N$ input matrix, which result in a $N \times N \times M$ Y matrix.



3. Apply 1-D DWT algorithm to each of the 16 elements in all M matrices in z -direction, which can be done as follows:

- a. For each i, j construct the $M \times 1$ input vector

$$Y(i, j) = [a_{i,j} \quad b_{i,j} \quad c_{i,j} \quad d_{i,j}]_{1 \times M}^T \text{ where } i, j = 0, 1, 2, \dots, N$$

- b. Construct an $M \times M$ transformation matrix; using transformation matrices given in (2.18) and (2.19).
- c. Apply matrix multiplication to the $M \times M$ constructed transformation matrix by the $M \times 1$ input vector.

4. Repeat step 3 for all i, j to get YY matrix ($N \times N \times M$ matrix).

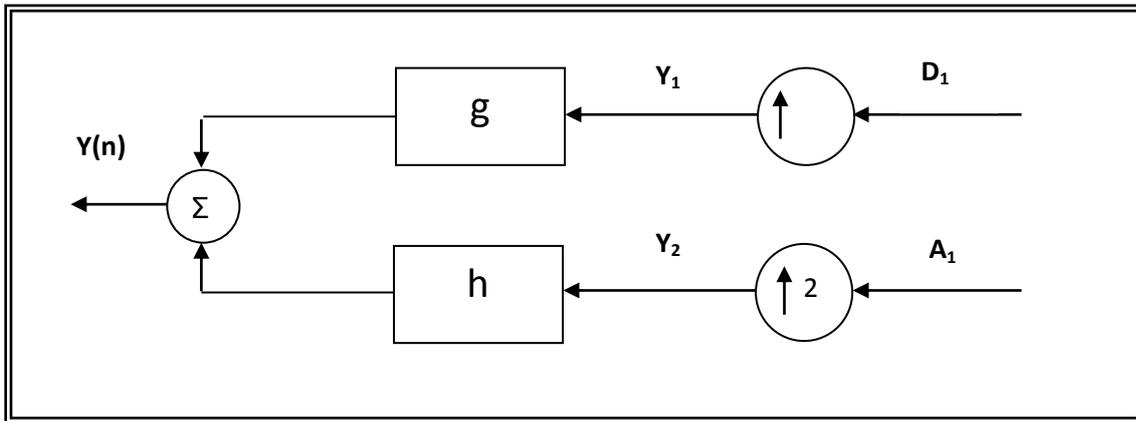


Figure (2-18) One level, One dimensional DWT Reconstruction

Thus, the approximation coefficients ($A_{r,k}$) at any level can be computed from one set of low-level scaling function coefficients ($A_{r+1,m}$) and all the intermediate wavelet coefficients ($D_{r+1,k}$). The coefficient vectors A and D are up-sampled (zeros instead at every other location) prior to convolution with the filters. This is analogous to the down-sampling operation in the decomposition process. The up-sampled and filtered coefficient vectors are then added together to create the next higher level $A_{r,k}$ vector. This process is repeated recursively to recreate the original input signal [41].

2.8. Wavelet Compression versus other Compression Techniques

Considerable effort has been spent in compressing information in order to reduce storage requirements. Techniques used by common compression program such as ZIP, or older compression formats such as JPEG or TIFF, suffer sever disadvantages when trying to compress large digital images. These problems include[47]:

- a. **Limited compression rates:** Because ZIP and related methods are lossless techniques (it is important that a compressed program is exactly decompressed), they don't take advantage of the lossy compression techniques that result in much higher compression rates for digital imagery.

- b. Artifacts and visible errors in the data:** Older techniques compress the image as a series of blocks. This is because older techniques were memory based requiring to limit the size of each block.
- c. Slow speed:** In order to work effectively with large digital images, the user needs to be able to view any subsection of the image, at any zoom factor, with sub-second response times. Older formats do not achieve adequate response rates for several reasons:
 - i-* Many formats require decompressing of the entire image, rather than a selected subset at a select level of detail.
 - ii-* Many formats assume the image will be smaller than computer RAM, making them very slow to access large images.

2.9 Video Compression System

The JPEG still picture compression standard has been extremely successful, having been implemented on virtually all platforms. This standard is fairly simple to implement, is not computationally complex, and gets 10:1 to 15:1 compression ratios without significant visual artifacts. This standard is based upon entropy encoding of quantized coefficients of the discrete cosine transformation of 8x8 blocks of pixel data.

Figure (2.19) shows the block diagram of both the JPEG compression and decompression algorithms. A single frame is subdivided into 8x8 blocks, each of which is independently processed. Each block is transformed into DCT space, resulting in an 8x8 block of DCT coefficients. These coefficients are then quantized by integer division by constants. The quantizing constant for each DCT coefficient is chosen to produce minimal visual artifacts, while maximally reducing the representational entropy of the coefficients. The quantized coefficients are then entropy coded into a compressed data stream[34].

The reduced entropy of the quantized coefficients is reflected in the higher compression ratio of the data.

The Motion JPEG (M-JPEG) uses the JPEG compression for each frame. It provides random access to individual frames, however the compression ratios are too low (same as in JPEG), because the technique does not take advantage of the similarities between adjacent frames.

The MPEG moving compression standard is an attempt to extend DCT-based compression into moving pictures. MPEG will encode frames by estimating the motion difference between the frames, and encoding the differences into roughly JPEG format. Unfortunately, motion estimation is computationally complex, requires specialized equipment to encode, and adds considerable complexity to the algorithm. Figure(2.20) illustrates the MPEG compression algorithm for predictive frames. [34].

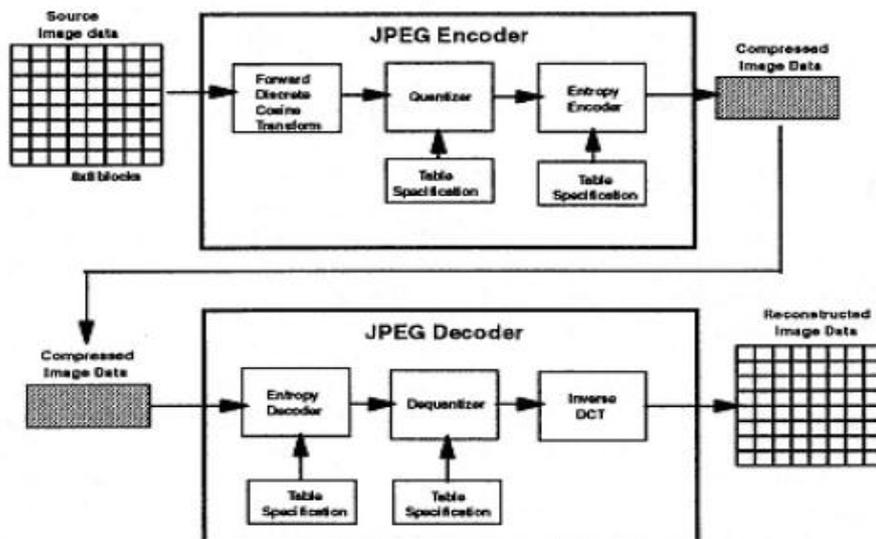


Figure (2-19) JPEG compression and decompression algorithms

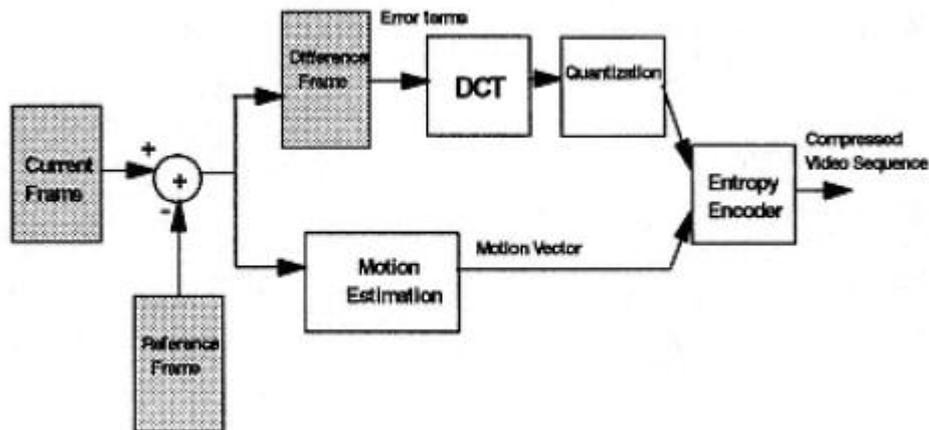


Figure (2-20) MPEG compression algorithm for predictive frames. MPEG adds motion estimation to the JPEG model.

XYZ compression is an alternative extension of DCT encoding to moving pictures. Sequences of eight frames are collected into a three-dimensional block to which a three-dimensional DCT will be applied. The transformed data is then quantized.

These quantizing constants are demonstrated to cause artifacts which are minimally visible. The resulting data stream is then entropy coded. This process strongly resembles the JPEG encoding process, as illustrated in Figure (2.21).

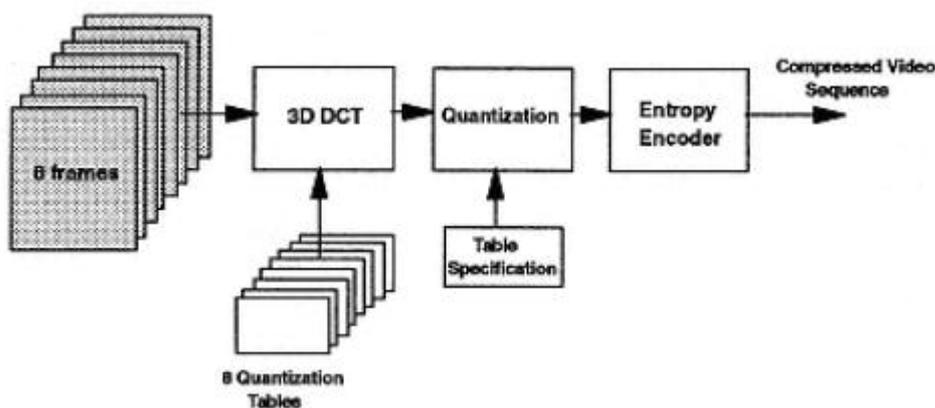


Figure (2-21) XYZ compression algorithm

Here, the three-dimensional DCT has been used to encode errors after motion estimation has been performed, and true three-dimensional DCT-based compression algorithms have been developed where the quantizes were based upon minimization of introduced mean square error[34].

2.10 Applications of Compressed Video

Video compression techniques made feasible a number of applications. Four distinct applications of the compressed video can be summarized as: [34]

(a) Consumer broadcast television.

(b) Consumer playback.

(c) Desktop video.

(d) Video conferencing.

(a) Consumer broadcast television, which includes digital video delivery to homes, typically requires a small number of high-quality compressors and a large number of low-cost decompressors. Expected compression ratio is about 50:1.

(b) Consumer playback applications, such as CD-ROM libraries and interactive games, also require a small number of compressors and a large number of low-cost decompressors. The required compression ratio is about 100:1.

(c) Desktop video, which includes systems for authoring and editing video presentations, is a symmetrical application requiring the same number of encoders and decoders. The expected compression ratio is in the range from 5:1 to 50:1.

(d) Videoconferencing applications also require the same number of encoders and decoders, and the expected compression ratio is about 100:1

Also there are application can be shown in table (2-3)

Table (2-3) summarizes applications of the compressed video by specifying current standards used in various applications, the required bandwidth, and typical frame sizes and frame rates

Application	Bandwidth	Standard	Size Frame	Rate [frames/sec]
Analog Videophone	5-10 Kbps	none	170x128	2-5
Low Bitrate Video Conferencing	26-64 Kbps	H.263	128x96 176x144	15-30
Basic Video Telephony	64-128 Kbps	H.261	176x144	10-20
Video Conferencing	>= 384 Kbps	H.261	352x288 15-	30
Interactive Multimedia	1-2 Mbps	MPEG-1	352x240 15-	30
Digital TV - NTSC	3-10 Mbps	MPEG-2	720x480	30-60
High Definition Television	15-80 Mbps	MPEG-2	1200x800	30-60

2.11 VIDEO COMPRESSION STANDARDS

The first successful standard for still image compression known as JPEG was developed jointly by the International Organization for Standardization (ISO) and International Telegraph and Telephone Consultative Committee (CCITT) in a collaborative effort. CCITT is now known as International Telecommunication Union—Telecommunication (ITU-T). JPEG standard uses DCT as the compression tool for grayscale and true color still image compression. In 2000, JPEG adopted 2D-DWT as the compression vehicle[30].

For video coding and distribution, MPEG was developed under the auspicious of ISO and International Electrotechnical Commission(IEC) groups. MPEG denotes a family of standards used to compress audio-visual information. Since its inception MPEG standard has been extended to several versions. MPEG-1 was meant for video compression at about 1.5 Mb/s rate suitable for CD ROM.

MPEG-2 aims for higher data rates of 10 Mb/s or more and is intended for SD and HD TV applications[30].

MPEG-4 is intended for very low data rates of 64 kb/s or less. MPEG-7 is more on standardization of description of multimedia information rather than compression. It is intended for enabling efficient search of multimedia contents and is aptly called multimedia content description interface. MPEG-21 aims at enabling the use of multimedia sources across many different networks and devices used by different communities in a transparent manner. This is to be accomplished by defining the entire multimedia framework as digital items[30].

Precision determines how many levels of intensity can be represented, and is expressed as the number of bits/sample. According to precision, images can be classified into: (a) binary images, represented by 1 bit/sample, (b) computer graphics, represented by 4 bits/sample, (c) grayscale images, represented by 8 bits/sample, and color images, represented with 16, 24 or more bits/sample.

According to the trichromatic theory, the sensation of color is produced by selectively exciting three classes of receptors in the eye. In a RGB color representation system, shown in Figure(2-22), a color is produced by adding three primary colors: red, green, and blue (RGB). The straight line, where $R=G=B$, specifies the gray values ranging from black to white[34].

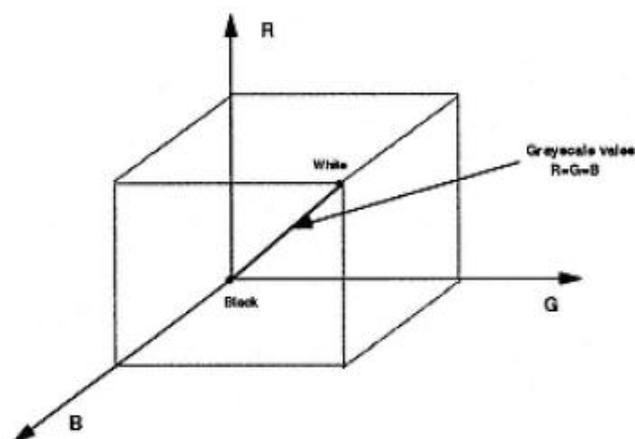


Figure (2-22) The RGB representation of color images

Another representation of color images, YUV representation, describes luminance and chrominance components of an image. The luminance component provides a grayscale version of the image, while two chrominance components give additional information that converts the grayscale image to a color image. The YUV representation is more natural for image and video compression. The exact transformation from RGB to YUV representation, specified by the CCIR 601 standard, is given by the following equations:

$$Y=0.299R+0.587G+0.114B \quad (2.21)$$

$$U=0.564(B-Y) \quad (2.22)$$

$$V=0.713 (R-Y) \quad (2.23)$$

where Y is the luminance component, and U and V are two chrominance components.

This approximation also gives a simplified YUV to RGB transformation, expressed by:

$$R = Y+ 2V \quad (2.24)$$

$$G = Y- (U+V) \quad (2.25)$$

$$B = Y+2U \quad (2.26)$$

Another color format, referred to as YCbCr format, is intensively used for image compression. In YCbCr format, Y is the same as in a YUV system, however U and V components are scaled and zero shifted to produce Cb and Cr, respectively, as follows:

$$Cb = \frac{U}{2} + 0.5 \quad (2.27)$$

$$Cr = \frac{V}{16} + 0.5 \quad (2.28)$$

In this way, chrominance components Cb and Cr are always in the range [0,1].

CHAPTER THREE

Video Compression

Technique Using 3D Discrete

Wavelet Transform

CHAPTER THREE

Video Compression Technique Using 3D Discrete Wavelet Transform

3.1 Introduction

Previous chapters have presented general concepts and techniques of compression in addition to methods of encoding the coefficients of a video. Digital video has a great deal of redundancy, therefore; it can be compressed. A video sequence has two types of redundancy namely, temporal and spatial redundancy. Temporal redundancy stands for the similarity between adjacent frames, while spatial redundancy stands for the similarity between neighboring pixels within the frame which is implemented in the image compression. A video compression system takes advantage of these two kinds of redundancy to achieve the goal of the video compression process.

The Performance of any compression system is evaluated by three essential parameter namely, compression ratio (CR), Peak signal-to-noise ratio (PSNR) and processing time(PT). As the compression ratio increase, the number of bits that represent the video with respect to the original size is less, but this increase will result in a decrease in PSNR. More detail is added to the video by increasing the bits of the video. The time for processing is also increased because the number of coefficients that needs to be processed is also increased. An efficient compression system achieves high compression ratios with considerable picture resolution.

This chapter explains the proposed video compression software in any compression condition. A detailed explanation of the proposed system and its operation which is implemented using discrete wavelet transform and its main drawbacks will be depicted and the systems will be evaluated by the main operating parameters CR, PSNR and PT.

3.2 Three-Dimensional Set Partitioned Embedded Block (3DSPECK)

In order for us to extend 2-D SPIHT or SPIHT technologies to 3D sources, to do this, people usually take a 3-D wavelet transform on the image sequence, and apply the extended sorting and partitioning algorithm to encode the source. As an example, 3-D SPIHT sorts coefficients along the paths of 3-D trees (one pixel corresponds to eight direct descendant pixels) instead of 2-D trees (one pixel to four direct descendants). The inter-band dependence or correlation can be exploited automatically, as was done in the spatial domain. Following the similar idea, we extend and modify 2-D SPECK for three-dimensional hyperspectral image applications. The first step is to decompose images into wavelet coefficients. Discrete wavelet transform is first applied on the spectral dimension, followed by on the horizontal (rows) and vertical (columns) axis. The resulting coefficients have a pyramid structure, and the structure has three levels of decomposition. Figure(3-.2) illustrates this structure.

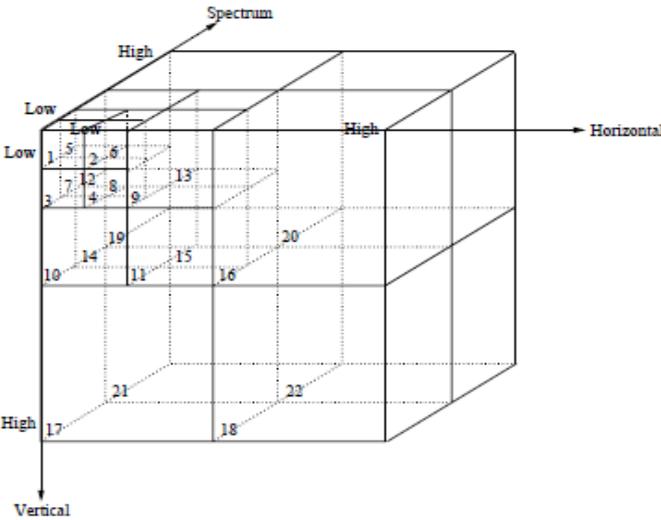


Figure (3-.1) Structure for 3DSPECK The numbers on the front lower left corners for each subband are marked to indicate the sorting order

The next step is to design a sorting algorithm to encode wavelet coefficients based on their magnitudes.

The motivation comes from the underlying physics of hyperspectral images. Most hyperspectral images have more high frequency content than other images in spatial domain. This can be illustrated by comparing the spatial energy distribution of a hyperspectral image "coast" and a non-hyperspectral image "chess" Figure (3-3) for one level of wavelet decomposition.

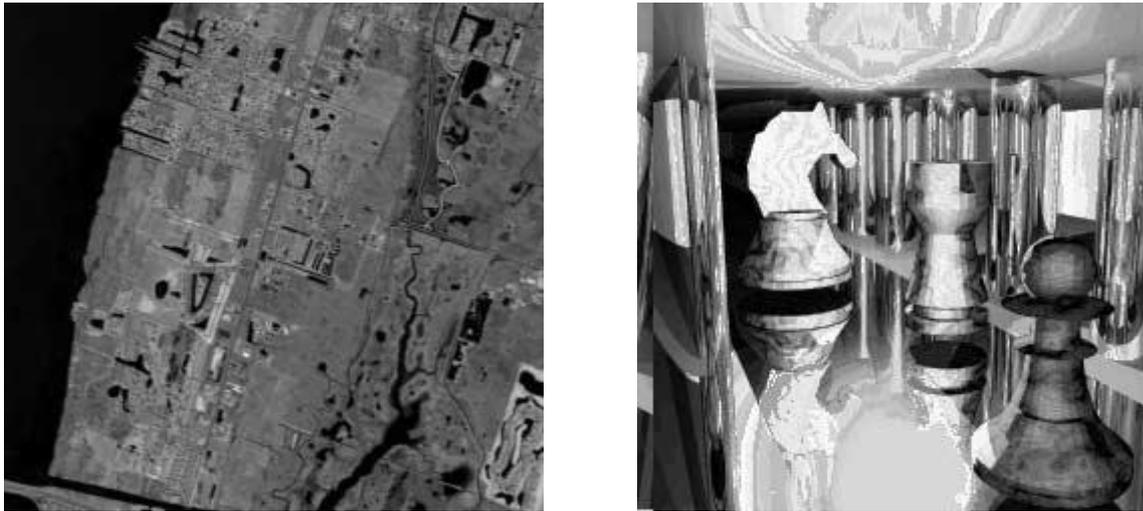


Figure (3-2) (a) hyperspectral image "coast"; (b) non-hyperspectral image "chess".

The energy is defined as the sum of squared value of wavelet coefficients. We calculate the energy proportion for each subband. The energy proportion is calculated by dividing the energy in one sub-band over the total energy. This shows that the "coast" image has more energy distributed over the HH, HL and LH sub-bands than that of "chess" image. In other words, hyperspectral images tend to have more large (significant) coefficients on higher spatial sub-bands by comparing to non-hyperspectral images. On the temporal (wavelength) axis, it is also true that less energy tends to locate on the lowest subband of the hyperspectral image sequences than that of other image sequences.

We want our sorting algorithm to be designed according to these observations. 3DSPIHT searches significant coefficients from the root of the tree structure down to its leaf. One 3D orientation tree includes a pixel on the root and its descendants. If all descendants are found to be insignificant against a certain threshold, 3DSPIHT needs to send only one bit to indicate this information. This kind of tree structure performs excellently if energy concentrates near the root of the tree. Like EZW and SPIHT, they all take advantage of the tree structure to design their algorithms, and they are all proved to be very efficient in general cases. However, in our case of hyperspectral images, energy is not so concentrated near the root of the tree. With higher probabilities than the case of regular image sequences, the descendants of root pixels may be found to be significant during the first several passes. 3DSPIHT will send extra bits to represent the sorting path to locate a significant pixel. In order to locate the large coefficients in the high frequency bands quickly, we extend the quadri-section partitioning algorithm of 2DSPECK to octo-section partitioning in our 3D application. The wavelet coefficient prism is partitioned into code blocks of different sizes. Each subband is treated as a code block. Then, the block splitting into eight is applied on each code block following a certain order to sort significant pixels. This splitting of code blocks allows zooming in quickly to areas of high energy and coding them first. This can be proved by comparing the rate distortion curves of 2DSPECK and 2DSPIHT. It is expected that the 3DSPECK would have the same property. 2DSPECK maintains an array of lists to process code units of varying sizes. This is simplified by maintaining only one list in 3DSPECK. 3DSPECK does not utilize the octave band partitioning in 2DSPECK. Details of the implementation are presented in the following. Later, the experimental results will show the improvements of our algorithm.

For each small (3D) block, we call it type S set. These include 8 sub-bands under the coarsest level low-low-low (LLL) sub-band and the 7 sub-bands at the next finer level and the remaining 7 sub-bands at the next level (e.g. HLL, LHH). All together, there are 22 type S sets.

3DSPECK maintains two linked lists:

LIS { List of Insignificant Sets. This list contains sets of type S of varying sizes.

LSP { List of Significant Pixels. This list contains pixels that have been found significant against threshold n .

- Quantization Step

Decrement n by 1 and go to step 2.

After initialization, there are 22 sets in the LIS. We treat each sub-band on the pyramid as an initial set. All sets are put on the list in the order of extended Z-scan. An example of the extended Z-scan is illustrated in Figure 1, and the order of the sets is marked by numbers. We can see from Figure that the list is a tree-like list. The underlying idea is based on the fact that though energy of hyperspectral images does not tend to concentrate in the LLL subband as much as in the case of non-hyperspectral images, a large portion of the energy still resides in the coarsest level subbands. In other words, the coarsest level subbands always convey the most important information of the original image sequence. If these bands are treated as individual sets, we know with extremely high probability that the algorithm will output when testing the significance of these sets at the first iteration. Since the sizes of the coarsest level subbands are small, the 3 DSPECK algorithm zooms into the significant coefficients quickly. On the other hand, the probability of finding significant coefficients in the sets during the early iterations decreases when going down the subbands of the pyramid. Our experiments show that sometimes no significant coefficient can be found against the first one or two bit planes on the finest subbands. Therefore,

leaving those larger size subbands as individual sets can avoid using extra bit budget to present non-significance. Hence, when we test the significance of sets, we scan sets by following the path from the top of the pyramid down to its bottom. In this way, we can find significant coefficients quickly in the karea of high energy and send the most important information first. This satisfies the requirement of embedded coding and progressive transmission.

3.3Quantization

The loss of data is done at this stage. The quantization stage is a limitation stage where each coefficient is limited to a particular level. The essential role that quantization plays is to reduce the number of bits that are allocated to each frame of the video. The proposed quantization limits the coefficients of all the subbands without severely effecting the picture quality of each frame and the video as a whole. After L-level 3D-DWT decomposition and thresholding process, the quantization stage computes the mean of each 2D-subgroup of the wavelet bands and product the result by a factor such as (0.3, 0.2 1,3,5, ...etc) to limit the step-size with respect to mean value of sub-group for the levels of quantization. The step size is given by equation (3-1):

$$\text{step}=(\text{mean}(s))/SS \quad (3-1)$$

Where: (s) is subband coefficients.

(SS) is a numerical factor.

Each coefficient of the wavelet subband is multiplied by the step and then rounded up to the nearest integer, the result is divided by the step size to obtain the output of the quantization stage as shown in equation(3-2).

$$S_b(w,u)=\text{int}(s(w,u)/\text{step})*\text{step} \quad (3-2)$$

where: (sb) is the quantized coefficients.

(step) is the step size.

(s) is the subband coefficients.

(w,u)is the index of the wavelet subband.

The output of this stage will give large similarity in sub group coefficients, this will reduce the number of bits which are allocated to each coefficient by the entropy encoder.

3.4 The proposed compression method using 3D-DWT

This section defines a proposed compression technique using three- dimensions transformation of Discrete Wavelet Transformation (DWT) The proposed method at the beginning of the work divides the video into parts called tires, and after that a group of frames (8 frames) is combined to form a three-dimensional cube that is the basic idea of compression.

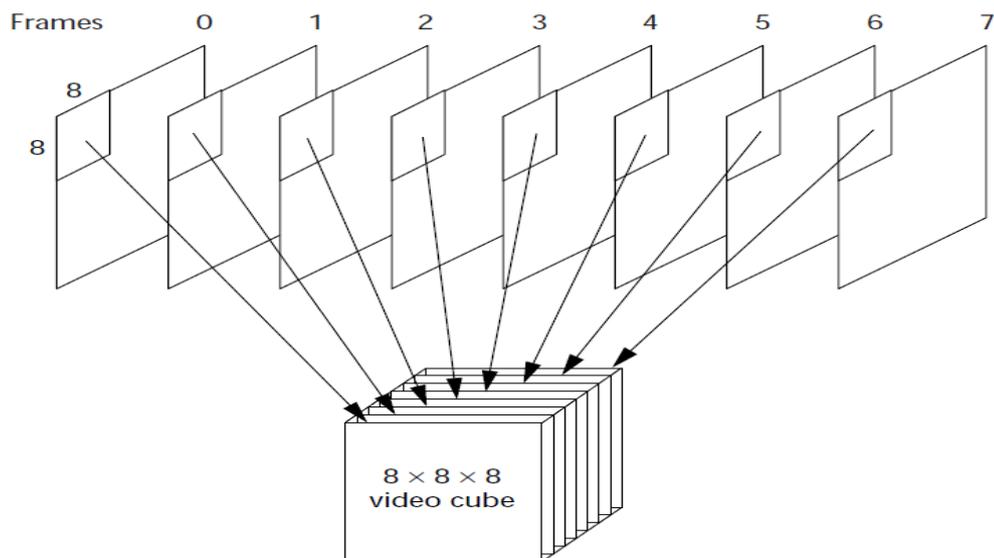


Figure (3-3) video cube from 8 frames

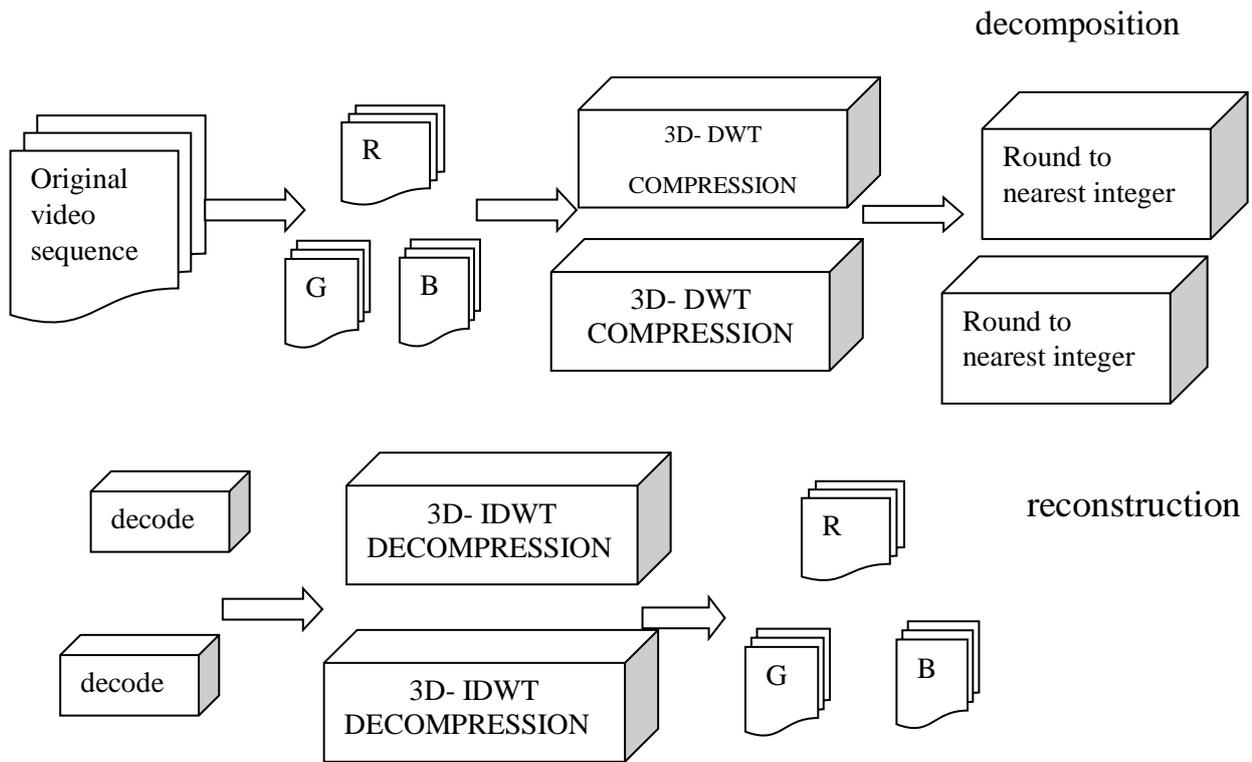


Figure (3-4) The proposed video compression and decompression system

3.4.1 The Modified Proposed Compression System

The proposed quantization and threshold process are used in the same manner as in the previous system. Because of the distortion that results from the previous system, a suggestion has been made to decrease this distortion. The suggestion implies using motion estimation and compensation in the wavelet domain. Motion estimation is an expensive computation operation this means that more time is needed to execute the operation but, motion estimation in the wavelet domain assists in making the algorithm faster (less processing time) than estimating the motion between consecutive spatial frames; Since hard threshold has been used, the number of coefficients that is needed to represent the frame is reduced, therefore; the time required for the motion estimation operation is reduced, eventually, the compression process is also reduced. The frame that is stored in the memory does not suffer severe round up error as a result of the

quantizer, it is obvious the proposed system is more efficient than the conventional system. As for the temporal compression, the modified compression system uses the same block matching algorithm and the type of search as explained in the previous system. The proposed modified video compression and decompression system is shown in figure (3-3).

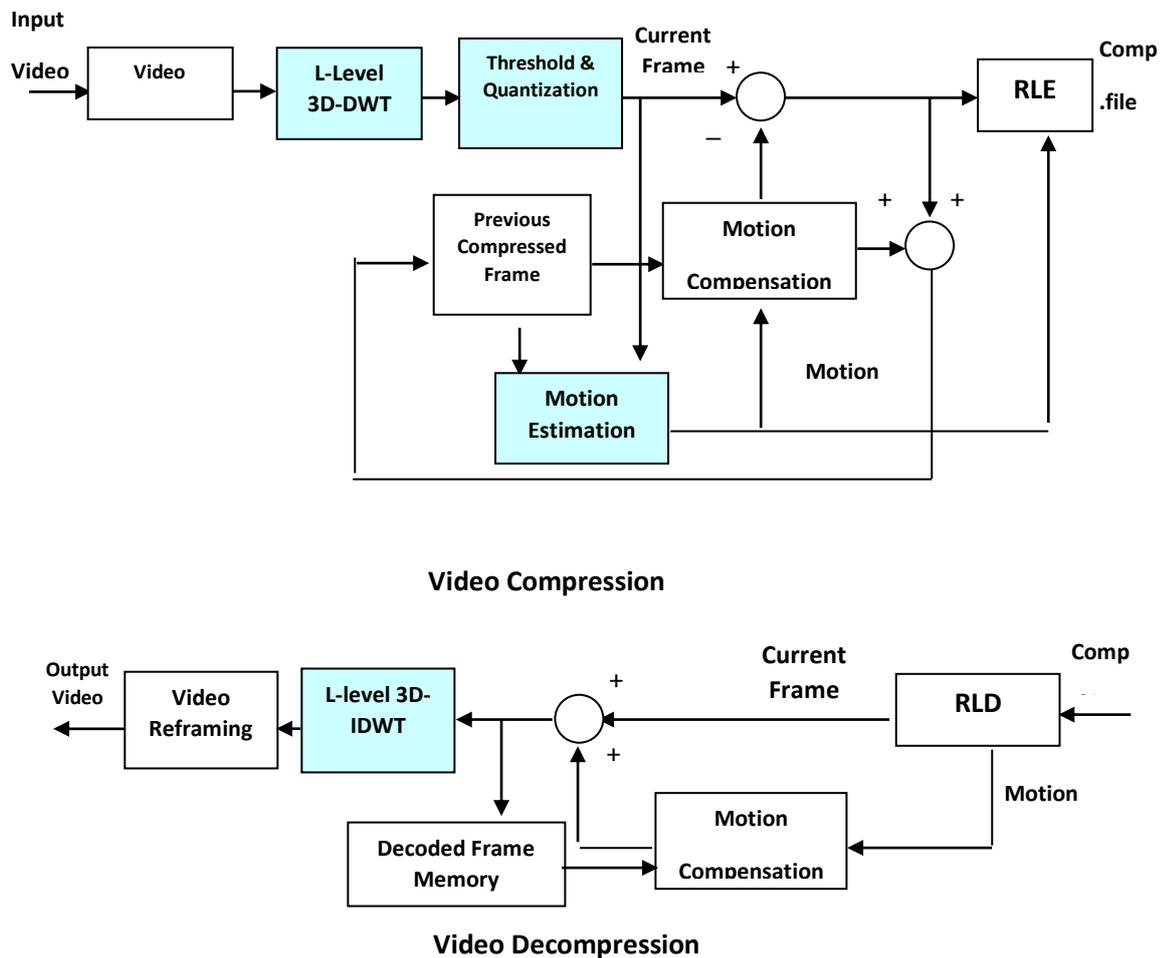
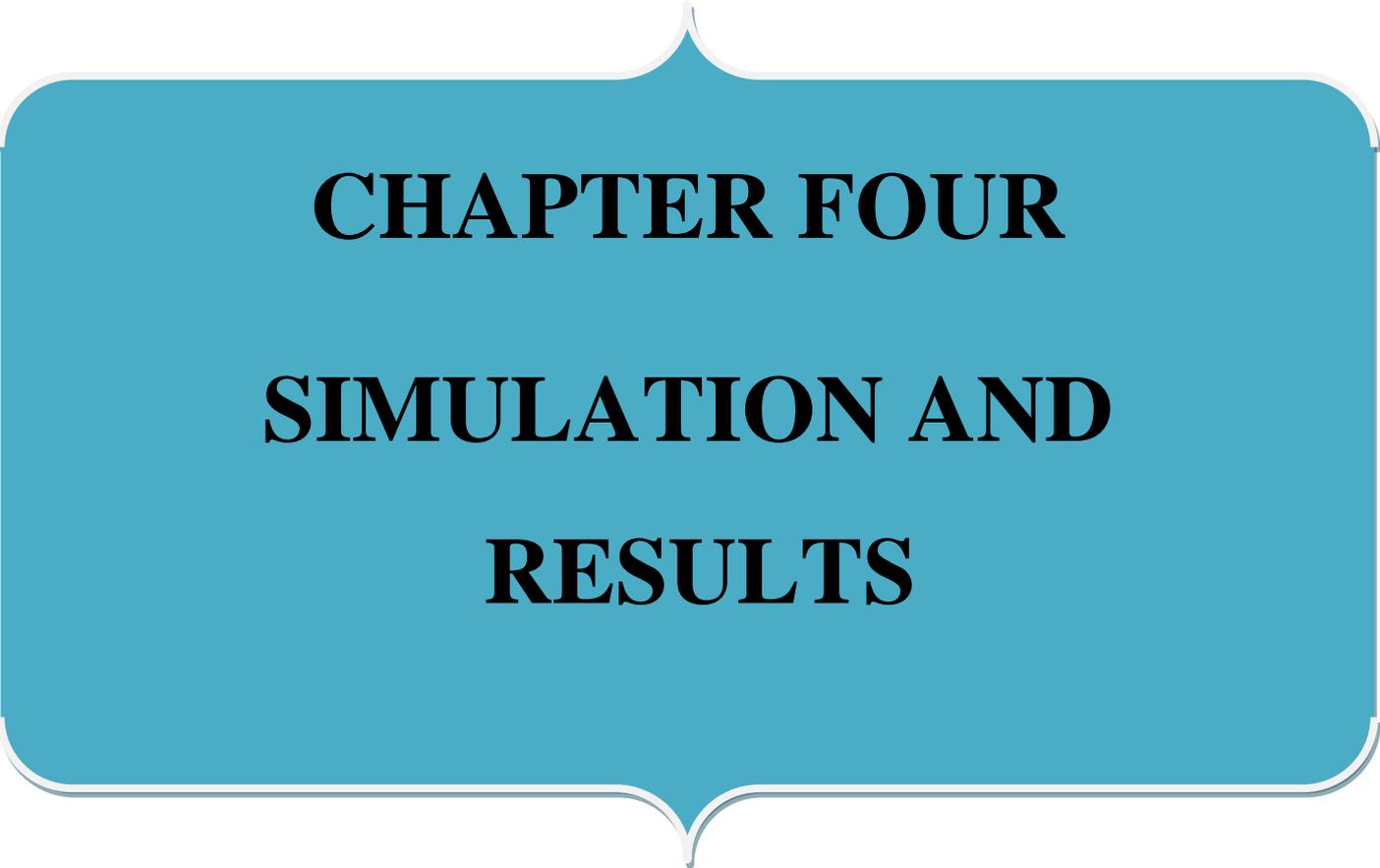


Figure (3-5) The Proposed modified video compression & decompression system

3.4.2 The Compression System Procedure

The compression steps are summarized as follows:

1. Reading the uncompressed input file (AVI file format) and converts the video input to 2D images, number of frames and the video size is also computed.
2. The system will extract the three-color components of the 2D images, The system will be tested using RGB color space, therefore; color space conversion will be applied in the framing process function.
3. The first frame of the input video is spatially compressed. Spatial compression involves executing L-level 3D-DWT on each frame, the resulting detail bands are thresholded. The threshold of the first level ϵ_{L-2} is higher than the threshold of the second level (ϵ_{L-1}) until ϵ_L because second level detail bands has more information than the detail bands in the first level.
4. The spatially compressed video will be stored in the memory of the compression system. The output of the memory is sent to the motion compensation to generate a motion compensated frame. The encoder has a matrix compensating the error prediction.
5. The motion compensated-compressed frame is subtracted from the original compressed frame, residual (error) is encoded by using RLE or using Embedded zero tree wavelet encoding.
6. Motion estimation is applied between consecutive frames and motion vectors are applied to the encoder (DWT).
7. The output data is transferred to the decompression side.



CHAPTER FOUR

SIMULATION AND

RESULTS

CHAPTER FOUR

SIMULATION AND RESULTS

4.1 Introduction

In this chapter, the results of the conventional and proposed system using two kinds of encoding and color spaces is illustrated. The illustration involves figures, tables and frames from the reconstructed frames. three sample videos were fed to the system, the first sample video has a simple motion between frames while the second sample has a complex motion that involves movement of objects, camera movement and zooming while the third video is in high definition.

Images and videos compression aims to search for a more compact representation of visual signals while maintaining high quality, and it becomes more and more important in the era of big visual data. From the surveys presented earlier in this post, it is clear that DWT-based end-to-end image compressions is at an advanced stage that outperforms JPEG2000 and HEVC.

Based on the review, we believe that the advantages of DWT in image and video compression are three times as much. First, the excellent content adaptability of the DWT outperforms the model based on signal processing because network parameters are derived based on a lot of process data while models in the latest encoding standards are handcrafted based on predefined image and video. Second, the larger receptive field is widely used in neural network models that not only use adjacent information, but can also improve compression efficiency by taking advantage of samples from a distance, but traditional encoders only use adjacent samples and are difficult to use far samples. Third, the DWT can well represent both texture and feature, making the combined

pressure optimal for both human vision and machine vision analysis. However, current coding standards only seek to perform high pressure toward the task of human vision.

We envision that deep learning-based image/video compression will play more important roles in representing and rendering images and videos with better quality and lower bit rates.

Generally, video signal has high temporal redundancies between a number of frames. In this code, we suggest a new video compression technique which exploits objectively the temporal redundancy. With the apparent gains in compression efficiency , So, it strongly exploits temporal redundancy with the minimum of processing complexity which facilitates its implementation in video embedded systems.

Experimental results show that the proposed technique Availability sufficient high compression ratios compared to other compression techniques.

4.2 The Case Results

4.2.1 Case(1) Results:

When selecting a video sample and processing it by the proposed system, some header information will be shown, Then, the resulting processed video will appear in its new size and pre-processing frames in the following window:

As a first step, the video frames are saved in a special file Figure (4-1). As we can see in Figure (4-2), the size of the original video is 25 MB, and it contains 114 frames. Whereas, the video resulting from the compression now has a size of 1.16 MB, as shown in Figure (4-3).

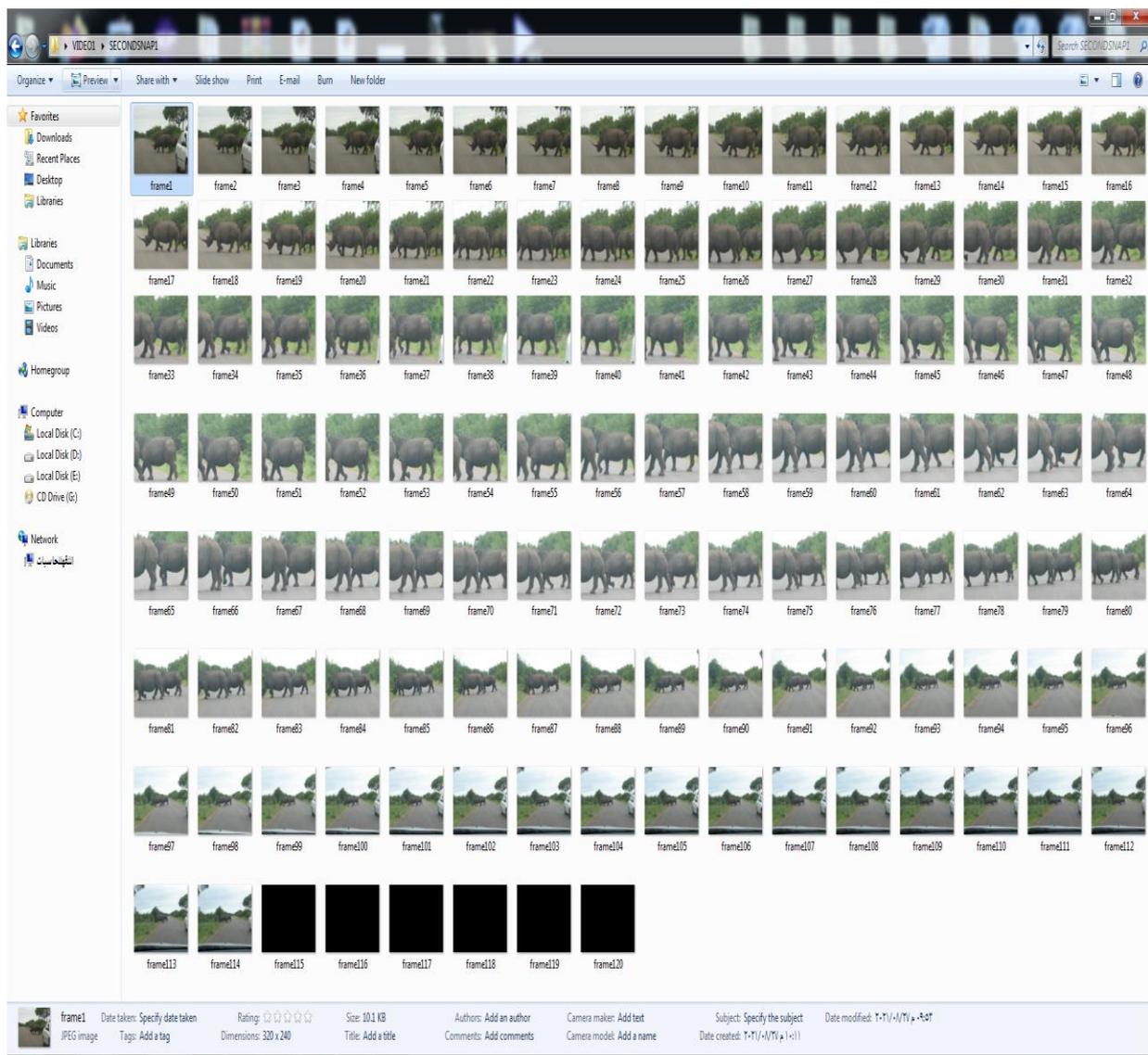


Figure (4-1) compress-frames for the Rhinos Video.

Then ,the total original size of the video file is:

Original size = No. of frames * frame size

=114*224

= 25536 KB

While the original compressed size is (224 KB),and the resulting size after processing is (10 KB),therefor

$$\text{Compression Ratio(CR)} = \frac{\text{the size of compressed image}}{\text{the size of original image}}$$

$$\text{PSNR} = 10 \log_{10} \left| \frac{255}{\sqrt{\text{MSE}}} \right|$$

Where MSE is mean square error

$$\text{MSE} = \frac{1}{M.N} \{ \sum_{m=1}^M \sum_{n=1}^N (f(m,n) - \hat{f}(m,n))^2 \}$$

A. Compression Ratio:

Compression ratio: It is defined as the ratio of the uncompressed or original video file size to the compressed file size. The mathematical formula is:

$$\text{CR} = \frac{25536}{1164} = 22$$

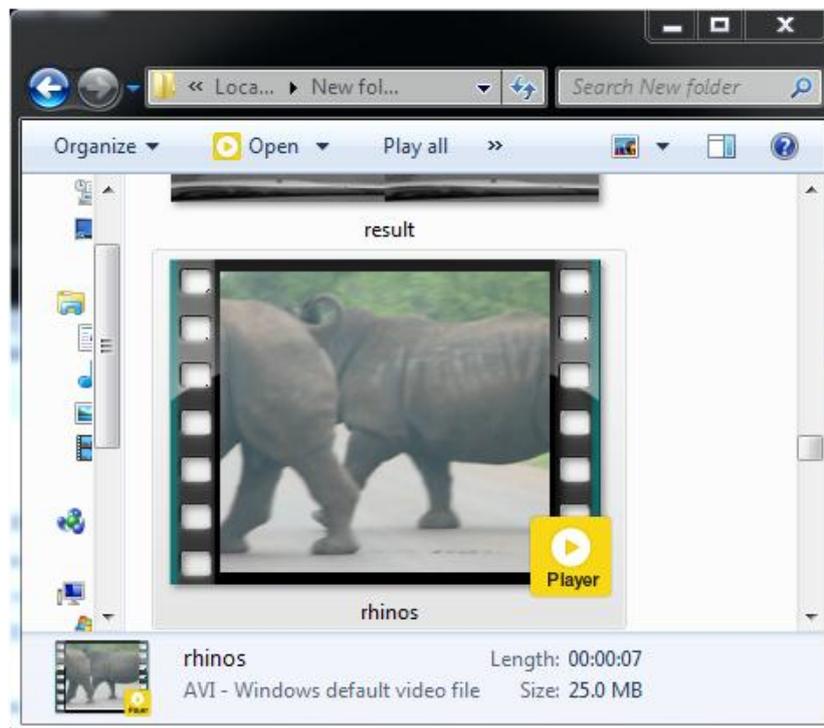


Figure (4-2) The size of original video

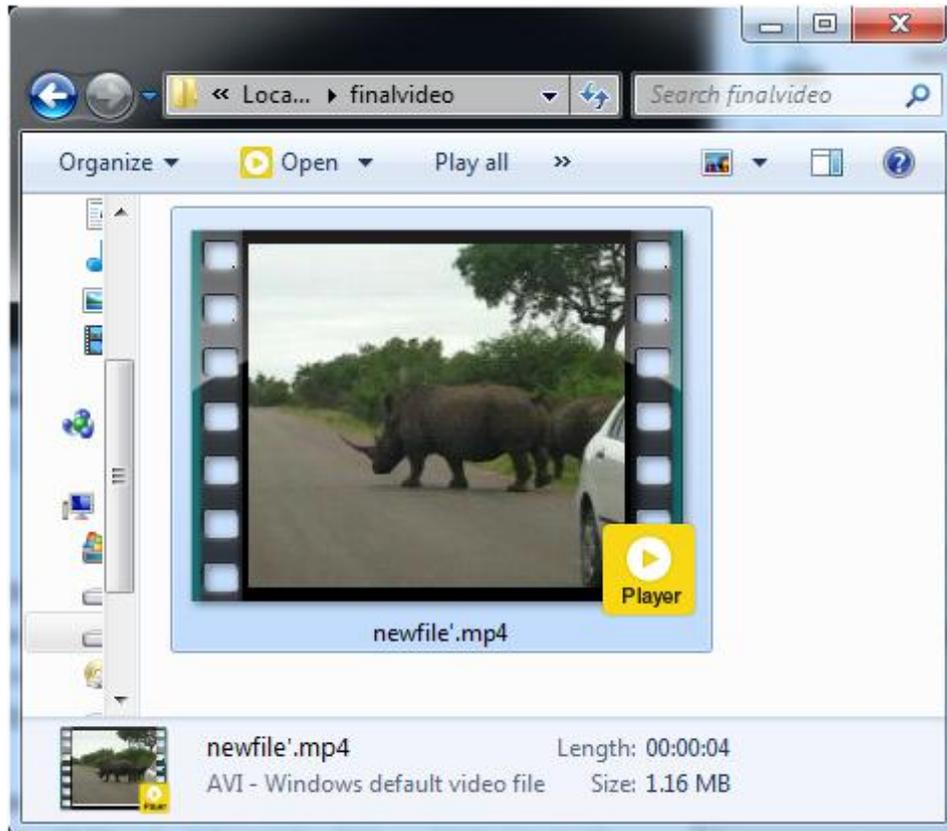


Figure (4-3) The size of compressed video

Also, it can be written by :

$$\text{Compression Ratio} = \left(1 - \left(\frac{\text{Compressed_Size}}{\text{Original_Size}}\right)\right) \times 100$$

$$= \left(1 - \left(\frac{1164}{25536}\right)\right) \times 100\% = 95.44\%$$

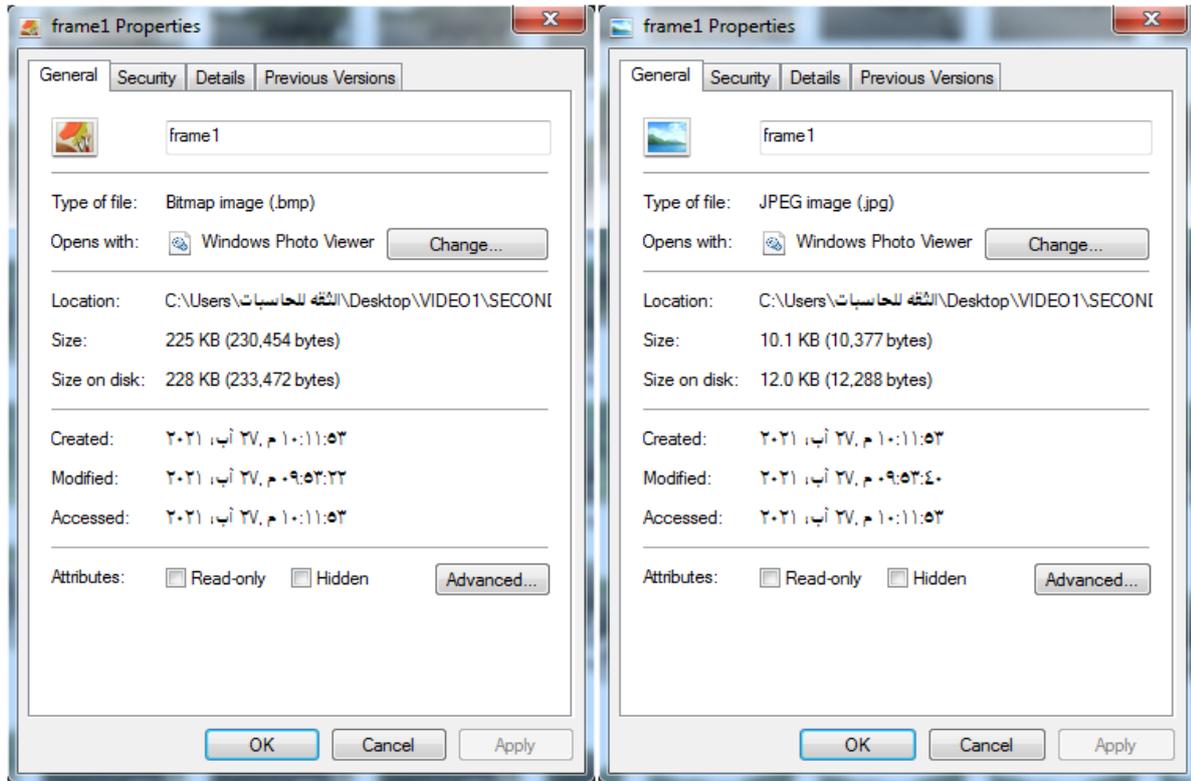
For one frame

$$CR = \frac{225}{10.1} = 22.27$$

$$\text{Compression Ratio} = \left(1 - \left(\frac{\text{Compressed_Size}}{\text{Original_Size}}\right)\right) \times 100$$

$$= \left(1 - \left(\frac{10.1}{225}\right)\right) \times 100\%$$

95.51%



(a)

(b)

Figure (4-4) size frames for the Rhinos Video a)before b)after



Figure (4-5) represents the comparison between the original image and the compressed image

B. Visual Quality

The PSNRs of 40 compress-frames in the Rhinos video are presents on Figure (4-6). The qualities of the outcomes frames that have been resulted from our suggestion system are very similar to the quality of the original digital videos frame before the compress process. In generic, the PSNRs value is between (47.78-51.4) dB.

- ❖ Size of the video that using here is 25 MB.
- ❖ size of result compress video 1.16 MB.

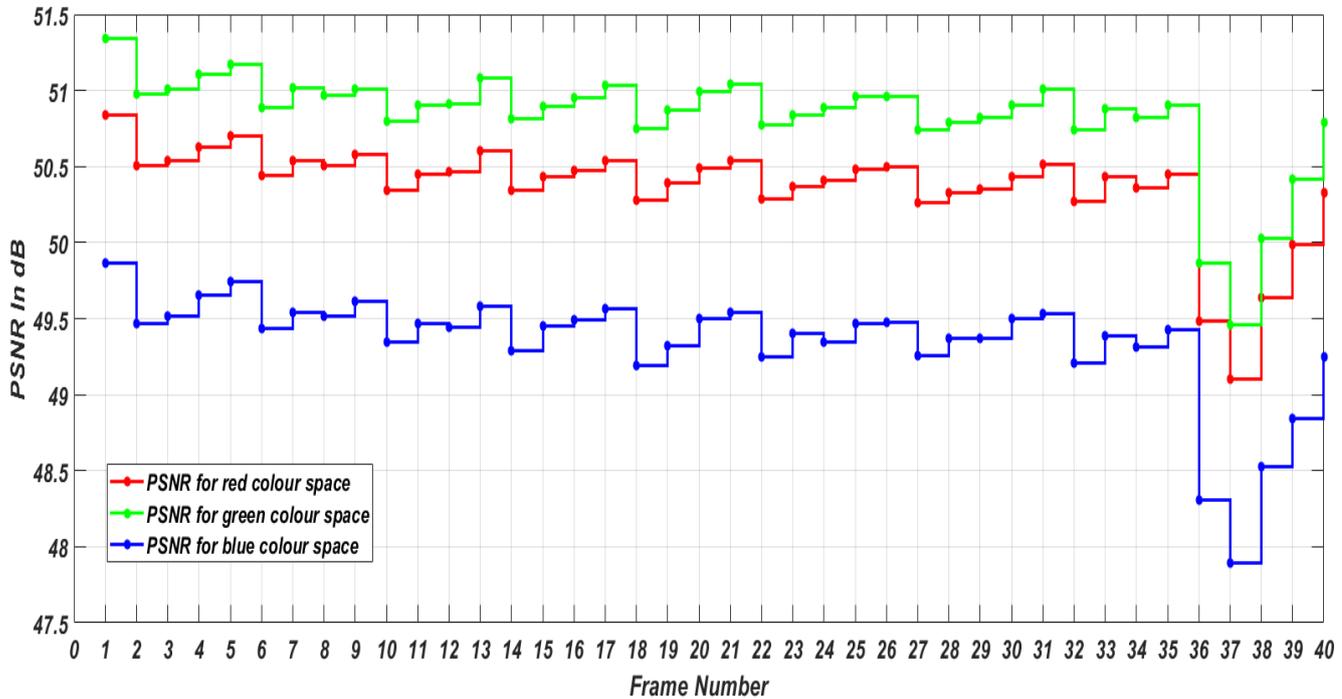


Figure (4-6) PSNR for 40 compress-frames for the Rhinos Video.

C. MSE

The average of the squares of errors (MSE) for 40 compress frame is display on figure 4.7. Which results from the difference among the original and the compress frame. From the figure, it can be noted that the value of MSE is very tiny (0.48-1.08),which is smaller than all the previous compress systems mentioned early. Figure (4-8) displays the average SSIM for the 40 compress frame. It can note that the similarity value range is between (94.1-95.3) % with an amount of differencing is very tiny which cannot be noted by natural human eyes.

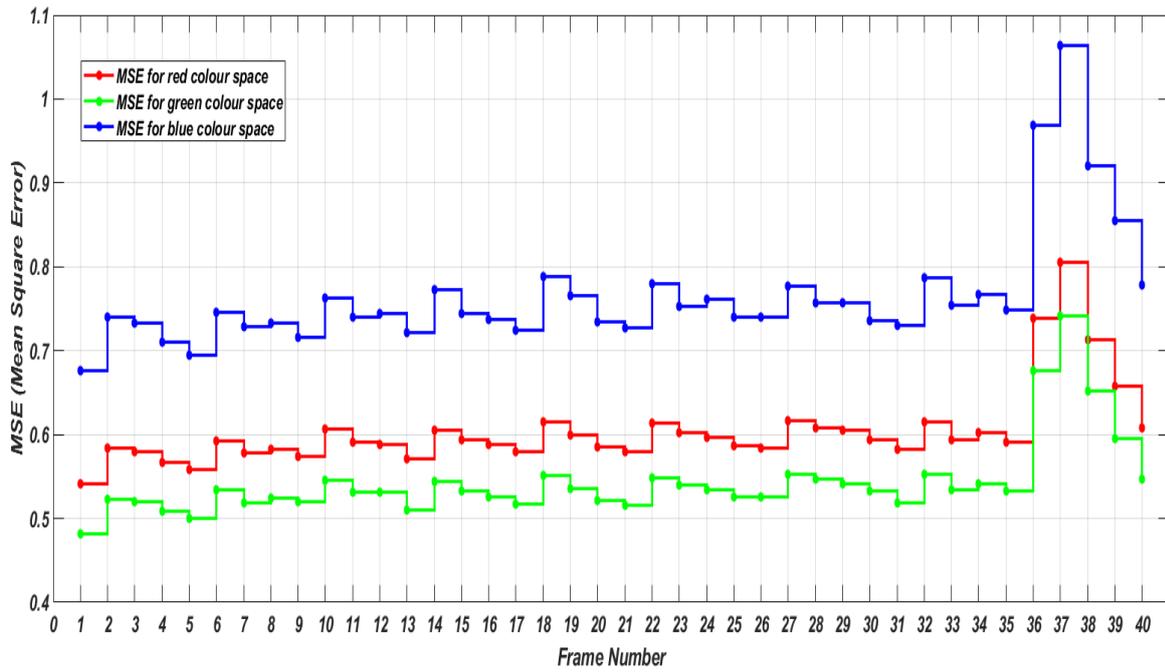


Figure (4-7) MSE for 40 compress-frames for the Rhinos Video.

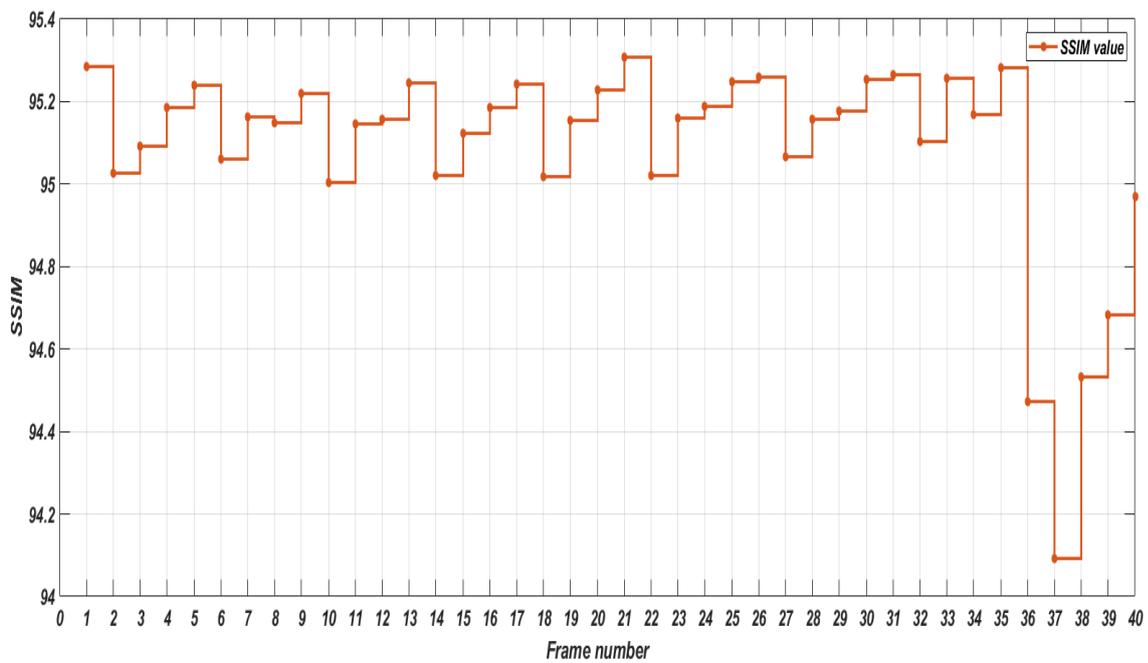


Figure (4-8) SSIM for 40 compress-frames for the Rhinos Video.

4.2.2 Case(2)results:

When a second video sample is selected and processed by the proposed system, the main information will appear, then the resulting processed video will appear with its new size and preprocessing frames in the following window:

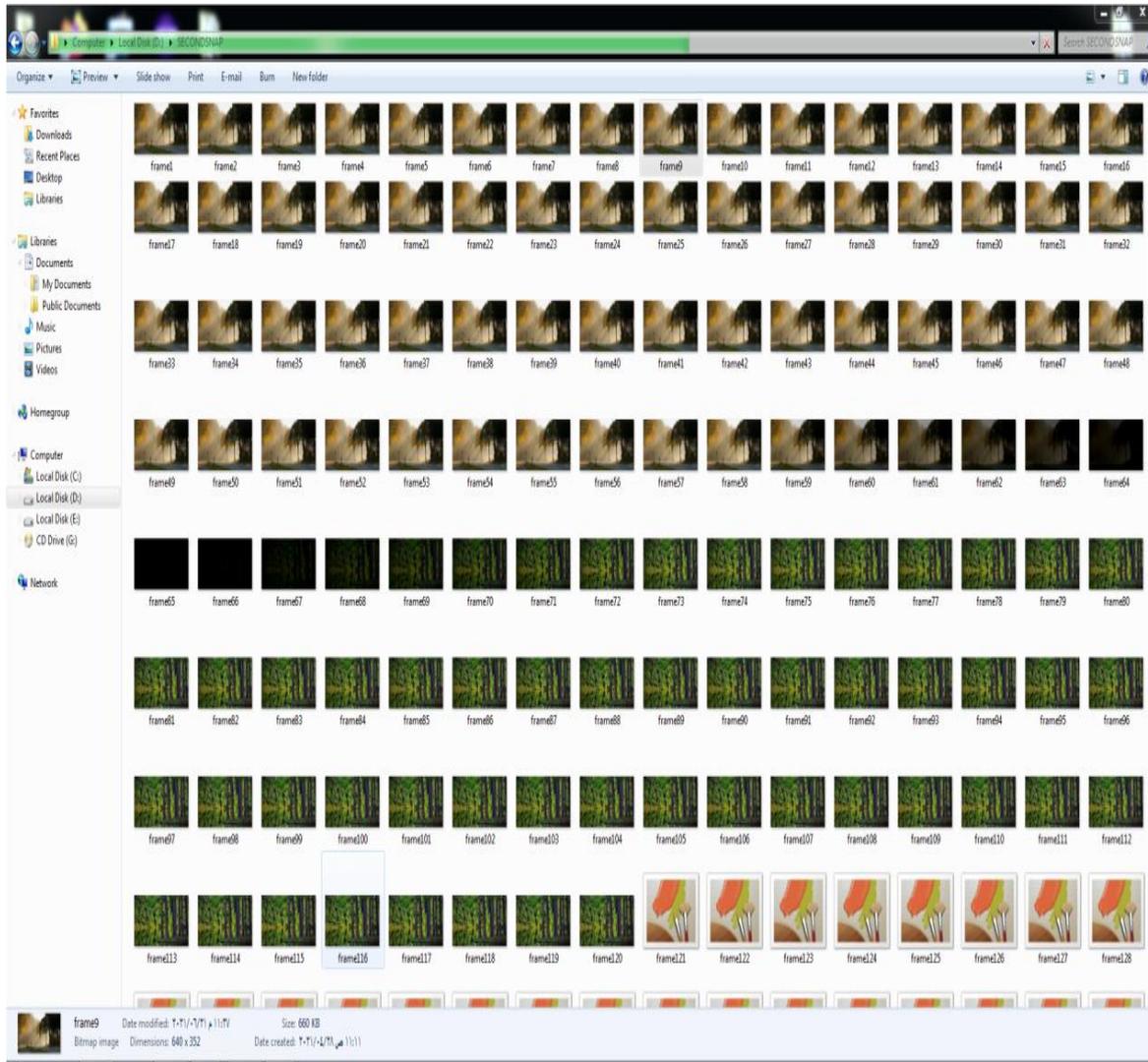


Figure (4-9) compress-frames for the Nature Video.

Then ,the total original size of the video file is:

Original size = No. of frames * frame size

=552*600

= 331,200 KB

While the original compressed size is (660 KB),and the resulting size after processing is (36 KB),therefor

A. Compression Ratio:

It defined as the ratio of the uncompressed (original) video file size to the compressed file size and the mathematical formula is:

$$\text{Compression Ratio(CR)} = \frac{\text{the size of compressed image}}{\text{the size of original image}}$$

$$\text{CR} = \frac{15.900}{1.500} = 10.6$$

$$\text{Compression Ratio} = \left(1 - \left(\frac{\text{Compressed_Size}}{\text{Original_Size}}\right)\right) \times 100$$

$$= \left(1 - \left(\frac{1500}{15900}\right)\right) \times 100\%$$

$$= 90.566\%$$

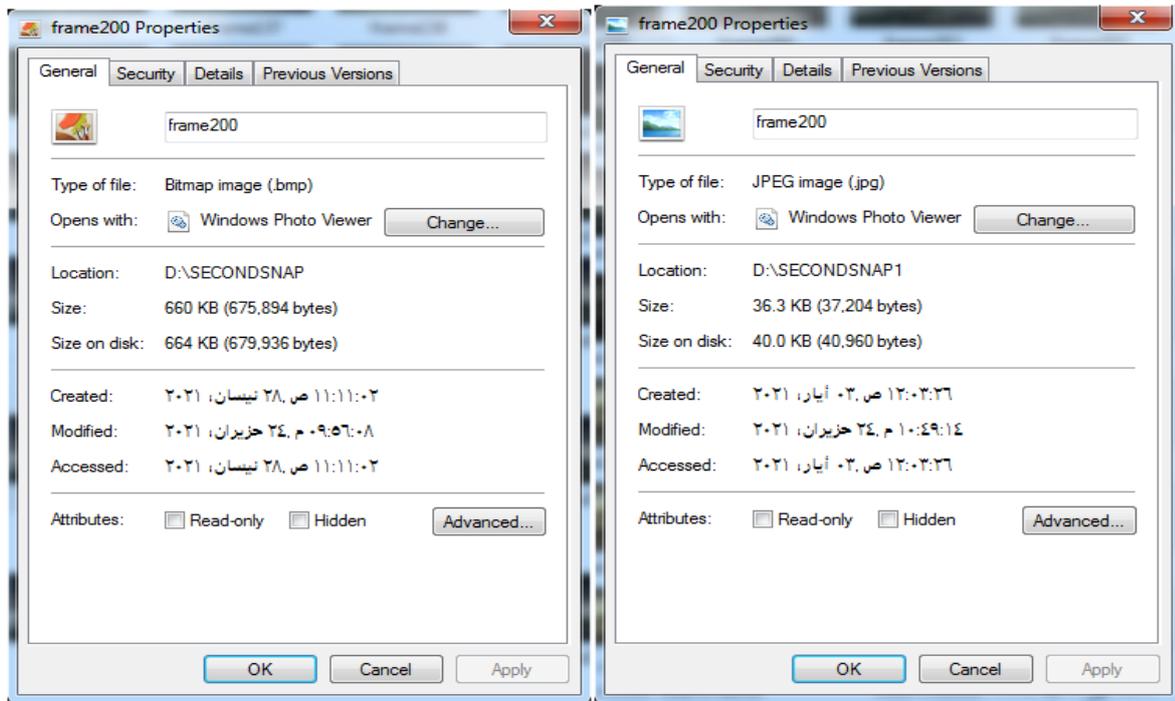
For one frame

$$\text{CR} = \frac{660}{36} = 18.333$$

$$\text{Compression Ratio} = \left(1 - \left(\frac{\text{Compressed_Size}}{\text{Original_Size}}\right)\right) \times 100$$

$$= \left(1 - \left(\frac{36}{660}\right)\right) \times 100\%$$

$$94.54\%$$



(a)

(b)

Figure (4-10) size frames for the Nature Video a)before b)after

The PSNRs , MSE and SSIM of 40 compress-frames in the Nature video are presents on Figures (4-11,4-13).

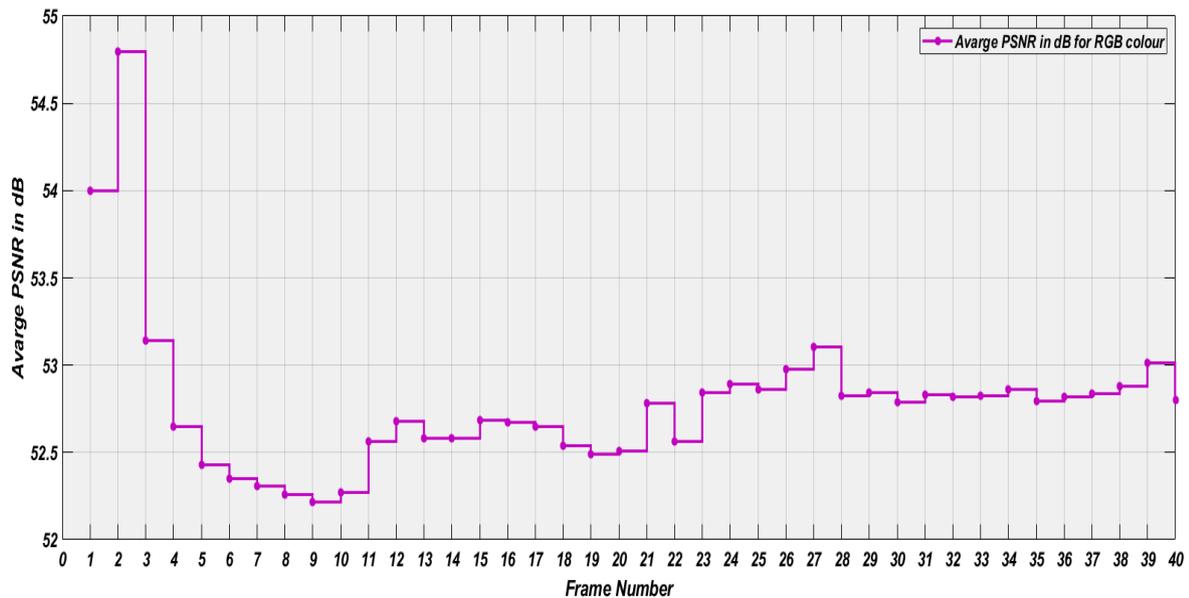


Figure (4-11) PSNR for 40 compress-frames for the Nature Video.

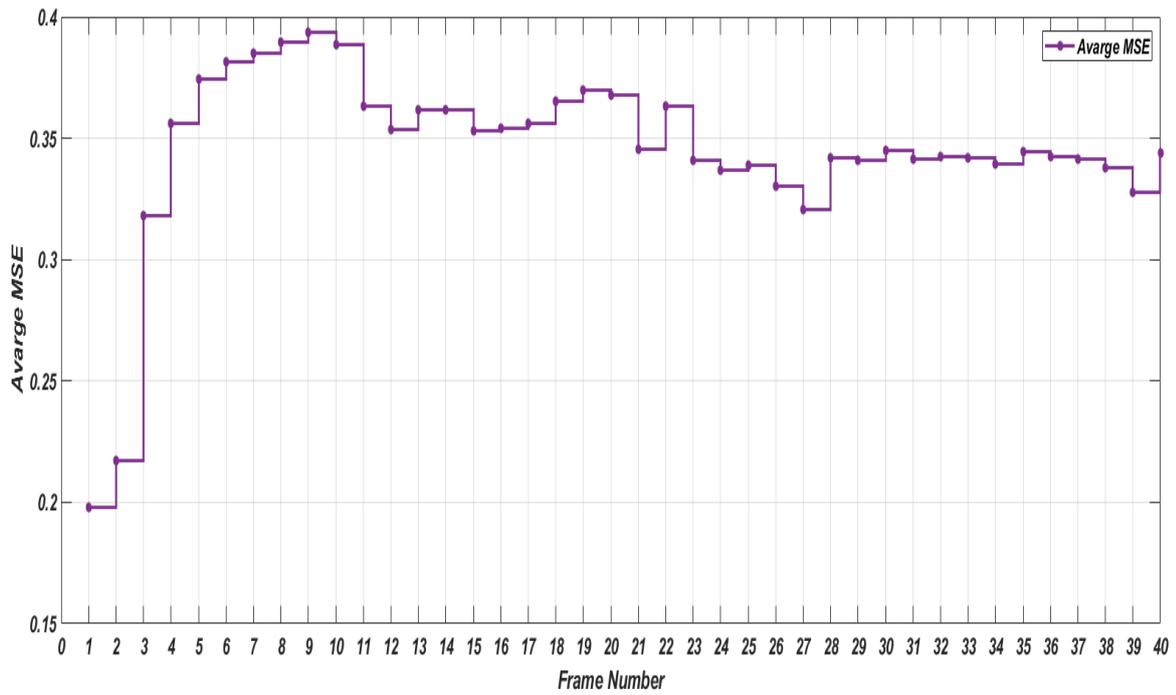


Figure (4-12) MSE for 40 compress-frames for the Nature Video.

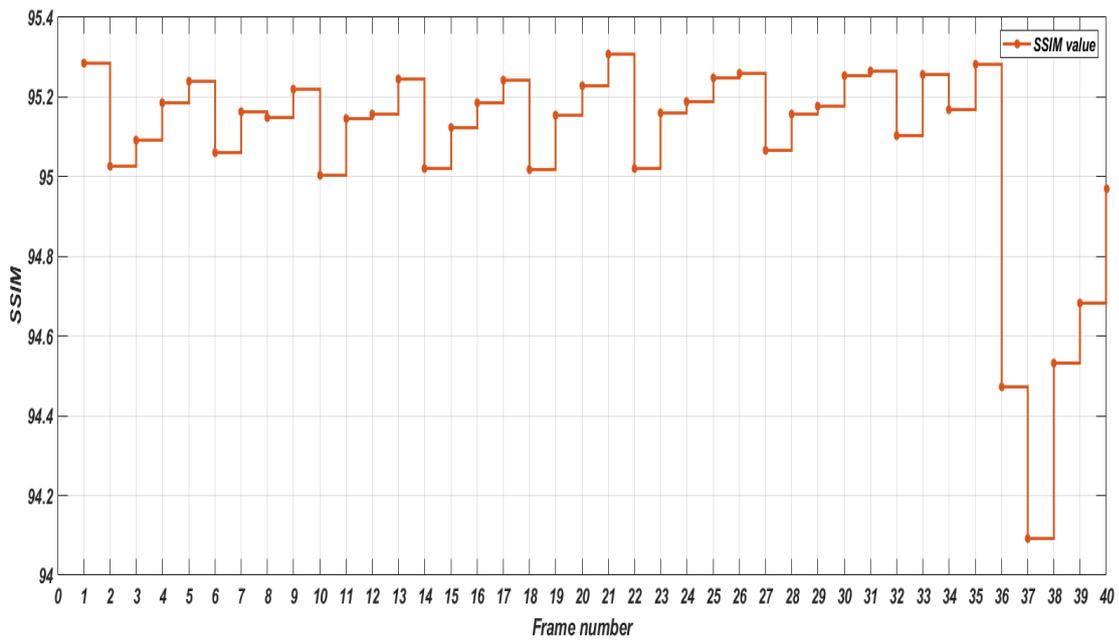


Figure (4-13) SSIM for 40 compress-frames for the Nature Video

4.2.3 Case(3)results:

Then we select a third video sample and process it by the proposed system, the main information will appear, then the resulting processed video will appear with its new size and preprocessing frames in the following window:

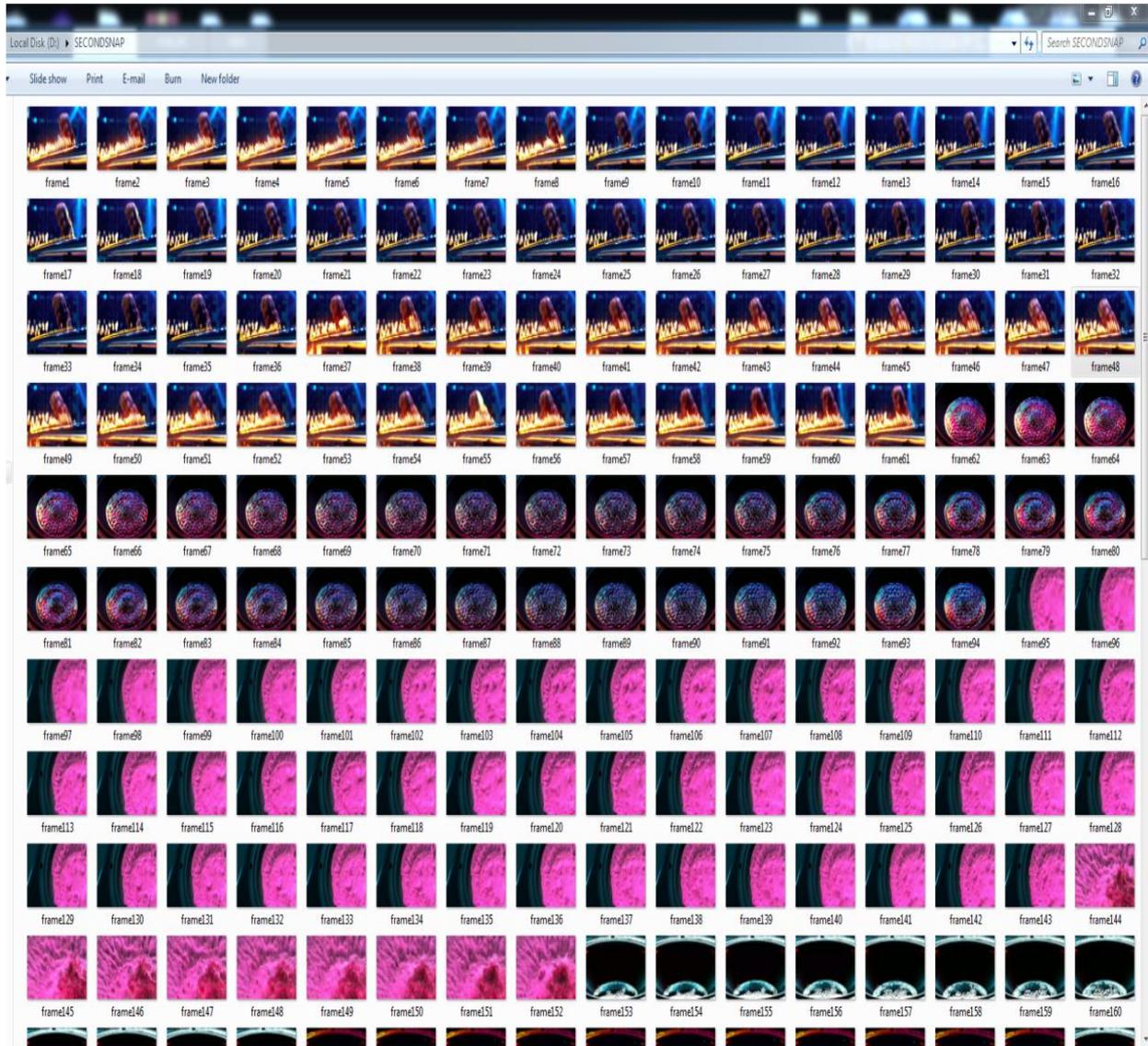


Figure (4-14) compress-frames for the Music Video.

Then ,the total original size of the video file is:

$$\text{Original size} = \text{No. of frames} * \text{frame size}$$

$$= 336 * 0.98 \text{MB}$$

$$= 329.280 \text{ MB}$$

While the original compressed size is (0.98 MB),and the resulting size after processing is (54.9KB),therefor

A. Compression Ratio:

It defined as the ratio of the uncompressed (original) video file size to the compressed file size and the mathematical formula is:

$$\text{Compression Ratio(CR)} = \frac{\text{Uncompressed File Size}}{\text{Compressed File Size}}$$

$$\text{CR} = \frac{13.700}{1.720} = 7.965$$

Also, it can be written by :

$$\text{Compression Ratio} = \left(1 - \left(\frac{\text{Compressed_Size}}{\text{Original_Size}}\right)\right) \times 100$$

$$= \left(1 - \left(\frac{1.720}{13.700}\right)\right) * 100\%$$

$$= 87.445\%$$

For one frame

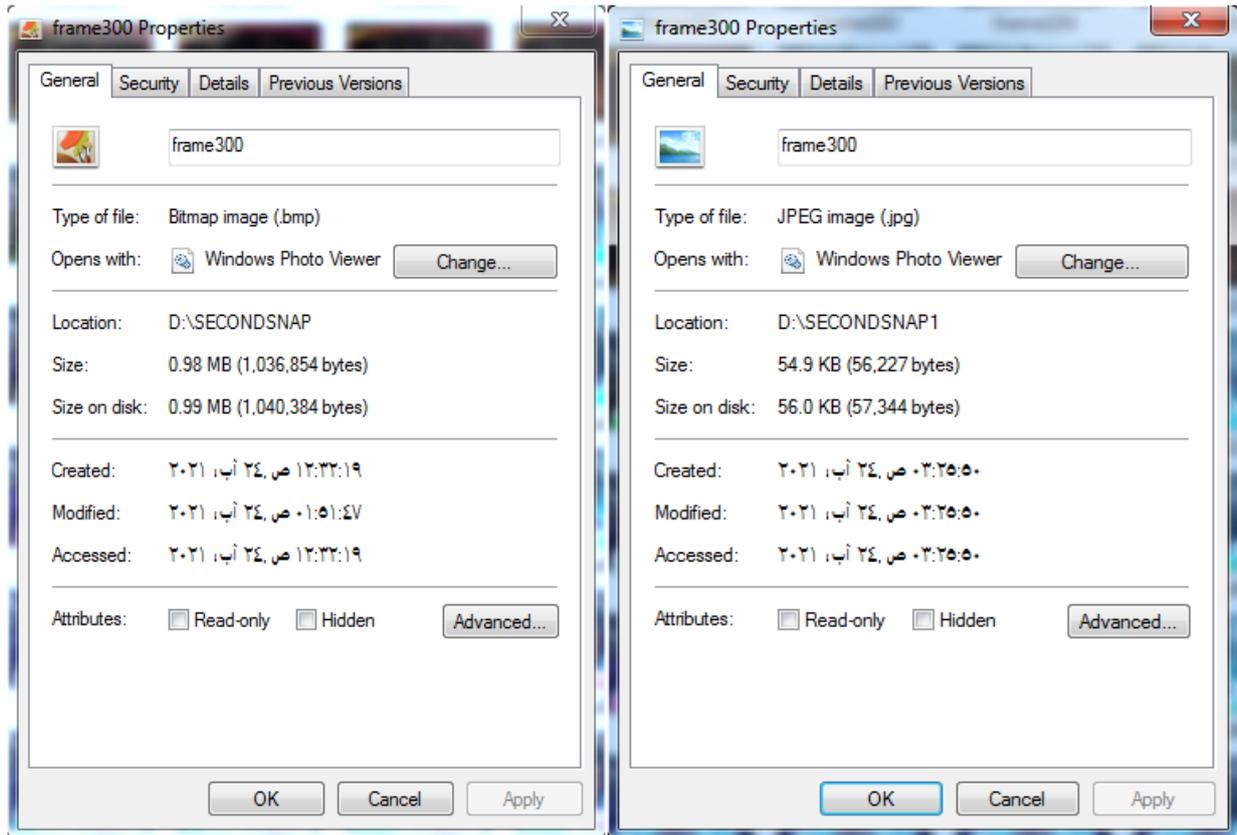
$$\text{Compression Ratio(CR)} = \frac{\text{Uncompressed File Size}}{\text{Compressed File Size}}$$

$$\text{CR} = \frac{980}{54.9} = 17.850$$

$$\text{Compression Ratio} = \left(1 - \left(\frac{\text{Compressed_Size}}{\text{Original_Size}}\right)\right) \times 100$$

$$= \left(1 - \left(\frac{54.9}{980}\right)\right) * 100\%$$

$$94.397\%$$



(a)

(b)

Figure (4-15) size frames for the Music Video a)before b)after

The PSNRs , MSE and SSIM of 40 compress-frames in the MUSIC video are presents on Figures (4-16,4-18) .

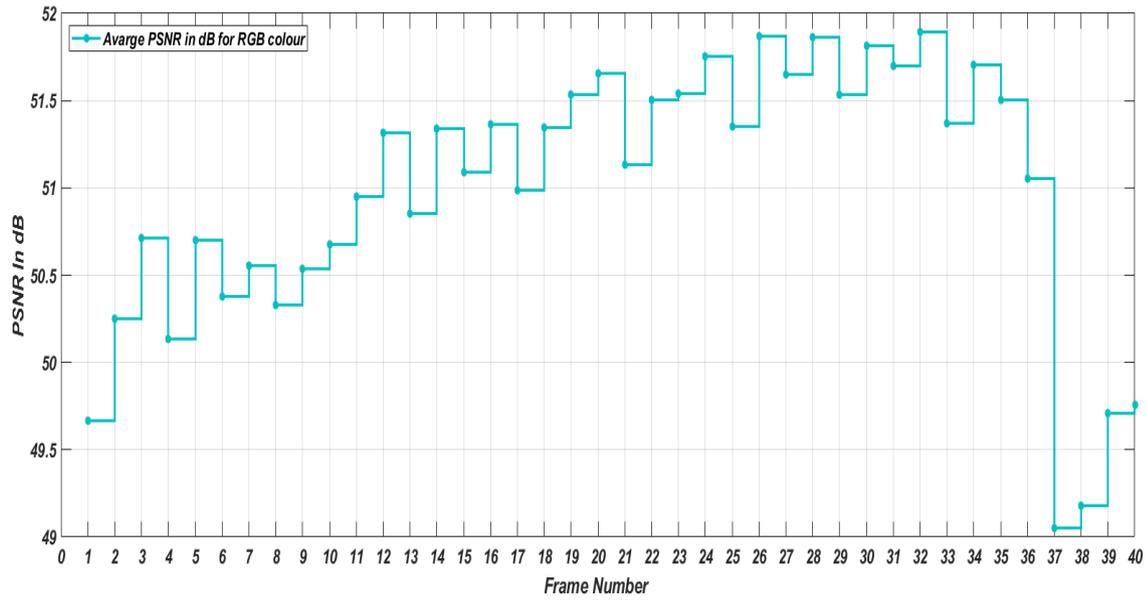


Figure (4-16) PSNR for 40 compress-frames for the MUSIC Video.

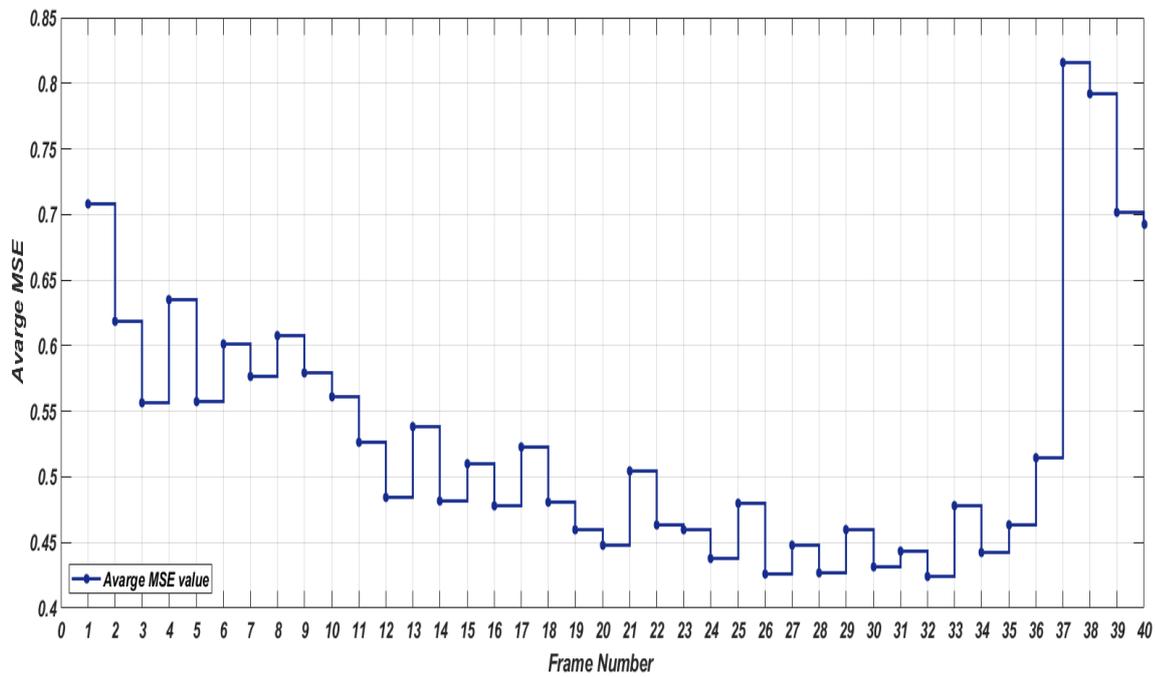


Figure (4-17) MSE for 40 compress-frames for the MUSIC Video.

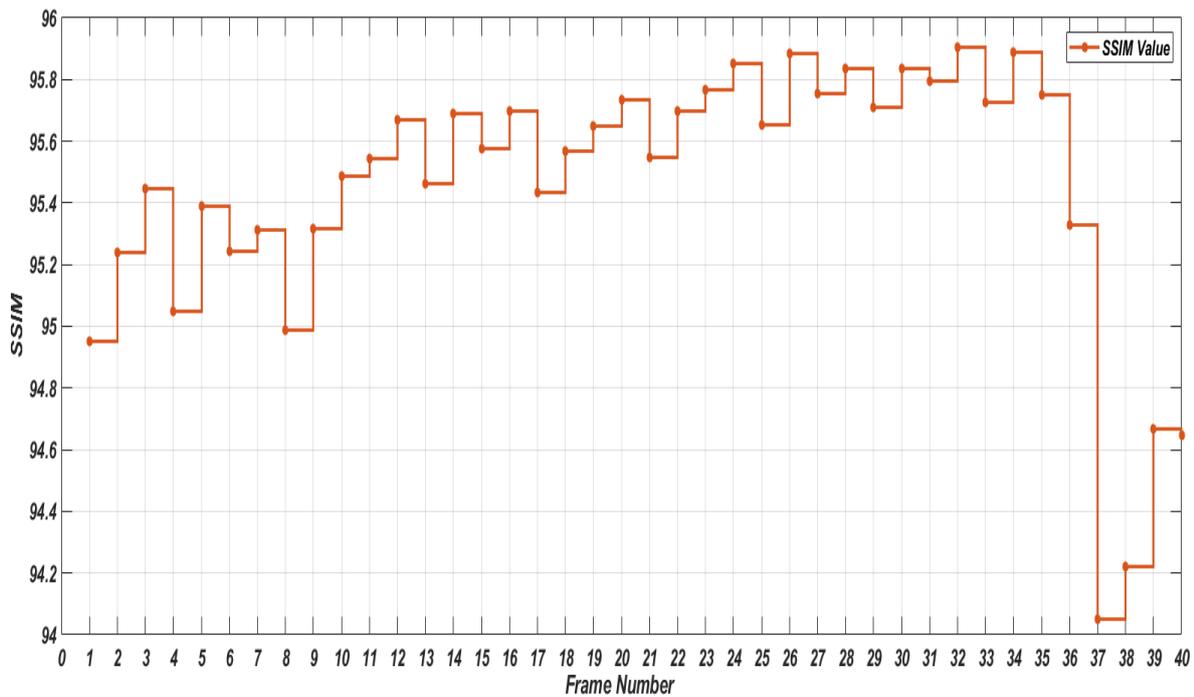


Figure (4-18) SSIM for 40 compress-frames for the MUSIC Video.

4.3 Discussion

The rationale behind this approach lies in that the DWT architectures common used for learned compression (in particular the encoders) are similar to the ones commonly used for inference, and learned image encoders are hence, in principle capable of extracting features relevant for inference tasks. As such, this approach could be extended in the future to simultaneously train and learn for the end-to-end image compression and understanding.

When a high bit rate digital video signal which is taken from a studio camera is to be compressed down to a low enough size so it is need lower bit rate, so that it can make economic use of the available transmission bandwidth, and convert it into a form that can be compared to the standard MPEG-2 format, the transform, decoding, decompression and reformatting have all to be implemented with very fast circuitry that does not overload the whole process with unnecessary delays. So, for these reasons:

- i. The transform is not done for the whole image but rather for an 8×8 blocks of the image.
- ii. The software implementation of the encoder takes much longer time compared to the hardware implementation. This is because hardware implementation involves a big number of parallel DCT/encoding chips that work independently on parallel to encode the different blocks of the image simultaneously.

Having this in mind, the multiwavelet implementation has to be adapted to work on 8×8 or at most 16×16 blocks of the image. A low-resolution image cannot be obtained from the transformed 8×8 blocks. Although, each quarter of a block is a low-resolution version of the block. Then adding up these quarter blocks produces only a very poor low-resolution version of the image. Such an assumption can be justified for the mobile video transport application but is still negotiable for the INTERNET application.

The aim of 3D-wavelet-based MPEG standard is to accomplish a comparable quality (or even less quality) of the video with a much higher compression (by an order of magnitude or so). General remarks can be written about the proposed system-based 3D-wavelets-based MPEG-2 encoder system in comparison to DCT-based system

1. Wavelet's coefficients require fewer bits than the DCT coefficients. This is because the dynamic range of multiwavelets coefficients is much smaller than the DCT coefficients. These can be further decreased by increasing the values of the scaling factor of the quantization matrix.
2. A quantization factor of the orders 2-4 represents a good compromise between quality and compression.

However, the proposed method has two features that make it interesting:

- i. It is possible to support a low-resolution video stream from the original stream. This is a desirable feature for mobile telephone video transmission where the display screen is very small and adapts well to low resolution images.
- ii. The ability to adapt the quality of the required bit rate through the use of the quantization factor.

	Video reslution	Original size	Comp size	Comp ratio
1	320*240	25 MB	1.2 MB	95.20
2	640*480	16 MB	1.4MB	91.25
3	720*480	13.7MB	1.72MB	87.45
4	1080*720	17.5	1.89MB	89.20
5	1280*720	25MB	2.33MB	90.68

Table (4-1) Compression ratio for videos with different resolutions

Through the results shown, we find that the compression ratio is not less than 87 percent, which is a good percentage compared to the rest of the previous codes, and the results for PSNR & MSE are also good.

Table (1-4) shows the compression ratio results for five videos of different resolutions. We note that the higher the video resolution, the lower the compression ratio, as the color focus plays a major role in this process by analyzing the image into the three main colors RGB.

CHAPTER FIVE

CONCLUSION AND FUTURE WORK

CHAPTER FIVE

CONCLUSION AND FUTURE WORK

5-1 Conclusions

Video compression operations are considered one of the important operations at the present time, as video transmission occupies the largest volume of transmission operations in telecommunication networks and the Internet. And one of the most important operations of compression is done through many complex mathematical transformations. Therefore, the selection of the transformation used depends, in particular, on the computational complexity that is measured in terms of the number of complications and additions required to carry out the transformation. In this paper it is facilitated to calculate the 3D discrete wave transformation and inverse transformation which includes the calculation procedure is mainly composed of basic arithmetic operations such as matrix multiplications, permutations, mixing and other simple and easy to check operations.

The proposed 3D-wavelet-based compression framework benefits from the following useful features:

- 1-Low computational complexity.
- 2-Low memory requirement.

Since a significant number of motion vectors in the high frequency subbands can be zero (due to the sparse multiwavelets coefficients in these subbands), no motion vectors are generated for these blocks. The proposed system addressed the transformation variable problem that resulted from motion estimation and compensation in the 3D-DWT field. The color components YCrCb color space are less sensitive to the eye. So, higher thresholds were applied to the coefficients of these components.

1-The results that were presented show that the proposed system gives better visual results and a high compression ratio. The processing time is also reduced because the motion estimate compares two frames less than the frequency (the reference frame and the current frame).

2-The results also showed the improvement of PSNR due to the effective performance of the proposed quantification

3-The proposed video compression system is achieved here using 3D-discrete wavelet transform(3D-DWT) of video frames.

It is obvious, from the results that shown in the table (4-1) , that by using the discrete wavelet transformation the video file size reduces while the resolution of the video remains good quality . the performance of the proposed video compression system has been measured by using Compression Ratio (CR) that considered the best fidelity measurement of then data compression (image ,video ,audio ,.....) ,so that, by using CR, the difference between the original (uncompressed) video and the compressed video can be computed .

4-All figures and examples mentioned previously showed the difference in size between then original video files and the compressed files and then Similarity between then original video and the decompressed video.

5-In this work, DWT used to reduce or remove the redundancy that exists in each frame individually. The coding process performed by using standard function called three dimensional that implemented after the transformation process to remove or reduce the redundancy between video frames . This will result reducing the size of data, Therefore, then compression will occur.

The Quantization process is a more adaptive operation to the nature of the frame as it takes the mean of the frame and according to factor inserted to the quantizer.

Decreasing the factor leads to decrease of the picture quality and increase of CR because of the number of similar coefficients.

6-The CR of the video also depends on the motion that is contained in the movie clip, CR decrease in a movie that has a complex motion and increases in a movie that has simple motion.

5.2 Future Work

1- Using of other types of transformations (Multiwavelet) with different levels.

2- Multi-wavelet can be used instead of DWT and thresholding all the detail bands with two different values of threshold according to the level of decomposition.

3-Applying other lossless transform such as Huffman, arithmetic encoding instead of discrete wavelet transformation method.



REFERENCES

REFERENCES

- [1] Z. J. Mohammad , "Video Image Compression Based On Multiwavelets Transform" thesis, University of Baghdad, 2004 .
- [2] A. Murap Al- Khatap "Digital video Processing", University of Rohester, prentice Hall PTR 1995.
- [3] M. R. Haque and F. Ahmed, "Image data compression with JPEG and JPEG2000", 8th International Confrence on Computer and Information Technology, 2005.
- [4] M. Mozammel Hoque Chowdhury and Amina Khatun " Image Compression Using Discrete Wavelet Transform" Jahangirnagar University Savar,Dhaka-1342, Bangladesh.
- [5] B. Schoner, J. Villasenor, S. Molloy and Rajeev Jain, "Techniques for FPGA Implementation of Video Compression Systems," Third International ACM Symposium on Field-Programmable Gate Arrays, pp. 154-159, 1995.
- [6] Hosam Khalil, " Three-Dimensional Video Compression Using Subband/Wavelet Transform with Lower Buffering Requirements " IEEE TRANSACTIONS ON IMAGE PROCESSING, VOL. 8, NO. 6, 1999.
- [7] V. Rajaravivarma, E. Lord and J. Barker, "Data compression techniques in image compression for multimedia systems," Southcon/96 Conference Record, pp. 624-627, 1996.
- [8] B. Le Bonhomme, M. Preda and F. Preteux, "MyMultiMediaWorld.com: A benchmark platform for 3D compression algorithms," 2008 15th IEEE International Conference on Image Processing, pp. 2700-2703, 2008.
- [9] Chung-Ping Wu and C.J. Kuo, "Design of integrated multimedia compression and encryption systems," in IEEE Transactions on Multimedia, vol. 7, no. 5, pp. 828-839, 2005.
- [10] H. Hakami and Z. Chaczko, "Improve data compression performance using wavelet transform based on HVS," International Conference on Image and Vision Computing New Zealand ,IVCNZ, pp. 1-6, 2016.

- [11] Jaya Krishna Sunkara, E Navaneethasagari, "A New Video Compression Method using DCT/DWT and SPIHT based on Accordion Representation", IJIGSP, vol.4, no.4, pp.28-34, 2012.
- [12] Thomas J. Naughton, "Compression of encrypted three-dimensional objects using digital holography" National University of Ireland, Maynooth, Department of Computer Science, 2004.
- [13] Samir B. Patel, Jeet R. Patani, Nisarg H. Patel "An Implementation of 3 Dimensional DCT for Compression of Video Sequences" , Conference: NUCONE, 2009.
- [14] Eman Abdelfattah and Asif Mohiuddin "PERFORMANCE ANALYSIS OF MULTIMEDIA COMPRESSION ALGORITHMS" International journal of computer science & information Technology (IJCSIT) Vol.2, No.5, University of Bridgeport, Bridgeport, Connecticut, USA, 2010.
- [15] D Pavani, D V Sai Sudheer" A New Video Compression Method using DCT/DWT and SPIHT based on Accordion Representation" International Journal of Image, Graphics and Signal Processing 4(4):28-34, 2012.
- [16] M. Mozammel Hoque Chowdhury and Amina Khatun "Image Compression Using Discrete Wavelet Transform" Jahangirnagar University, 2012.
- [17] K. Sureshraj "Image and Video Compression using Discrete Wavelet Transform Matlab Results" CVR College of Engineering, 2014.
- [18] A. Secker and D. Taubman, "Motion-compensated highly scalable video compression using an adaptive 3D wavelet transform based on lifting," Proceedings 2001 International Conference on Image Processing, Cat. No.01CH37205, pp. 1029-1032 vol.2, 2001.

- [19] Xiaoli Tangab, William A. Pearlmana and James W. Modestinob " Hyperspectral image compression using three-dimensional wavelet coding" ECE Department, University of Miami,2015.
- [20] Divya R. Jariwala, Heta S. Desai "JPEG IMAGE DISCRETE WAVELET TRANSFORM COMPRESSION USING MATLAB" Shri JYT University, 2016.
- [21]Sai Virali Tummala, Veerendra Marni " Comparison of Image Compression and Enhancement Techniques for Image Quality in Medical Images." Master of Science in Electrical Engineering with, Emphasis on Signal Processing, 2017.
- [22] Thirumoorthi , Karthikeyan " A study on discrete wavelet transform compression algorithm for medical Images" Biomedical Research 2017.
- [23] Siwei Ma, X. Zhang, "Image and Video Compression With Neural Networks" in IEEE Transactions on Circuits and Systems for Video Technology, vol. 30, no. 6, pp. 1683-1698, June 2020.
- [24] Ahmed Toman Thahab " Wavelet Transform Based Video Compression" thesis, University of Babylon, 2010.
- [25] D. Babie Zagreb, "Discrete Cosine Transform Algorithm for FPGA Devices", 2003.
- [26] R.C.Gonzalez and R.E.Woods, "Digital Image Processing ", 2nd edition, Printice Hall, New Jersey, 2002.
- [27] G. Strang, "The Discrete Cosine Transform ", IEEE Trans on signal processing, Society for industrial and applied mathematics,1999.
- [28]Guanjun Zhang "Robust Scalable Video Compression Using Multiple Description Coding" A dissertation in partial fulfillment for the degree doctor of philosophy, 2007.
- [29] B.Girod, M.Filer, "Multi frame motion compensated video compression for the digital set-top Box", Information System Laboratory, Dept of Electrical Engineering, Stanford University, 2002.
- [30]Wafaa Hassan Al-Marsomi, "Video Clip Compression Using DCT Technique" , master thesis, University of Babylon, 2006.

- [31] K. S. Thyagarajan "STILL IMAGE AND VIDEO COMPRESSION WITH MATLAB" 2011.
- [32] Video Compression-an Introduction, white paper, Array Microsystems Inc, 1997.
- [33]Geoffery M.Davis and Aria Nosratinia "Wavelet –Based Image Coding" 1998.
- [34] Raymond Westwater and Borko Furht " Real-Time Video Compression Techniques and Algorithms" Florida Atlantic University, 2007.
- [35] Pavankumar Gorpuni, "Development of Fast Motion Estimation Algorithms for video Compression", Department of Electronics and Communication ,National Insitution of Technology, Rourkela, India, master thesis, 2009.
- [36] N. Venkasteswaran and Y.V.Ramana Rao " K-Means Clustering based Image Compression in wavelet domain" college of Engineering, Anna University , Chennai-600 025, India, Information technology Journal 6, 2007.
- [37]Ishraq M.Hassan," Image Compression using Discrete wavelet and predictive Techniques" master thesis, Babylon University, 2006.
- [38] Germán Gómez Herrero "Image Compression Using Wavelets", Tampere University of Technology,2009.
- [39]I. Daubechies, " Orthonormal Bases of Compactly Supported Wavelets" Communication in Pure and Applied Mathematics, 1988.
- [40]N.E.Miner, EPFL and U.Berkerly, " Modern Signal Processing Wavelets versus Fourier ", CERN lectures, 2001.
- [41] S. Suha, "Image Compression –from DCT to wavelets" Crossroads, The ACM Magazine for Students Volume 6Issue, 2000.
- [42] Y. Hao, X.Zhu, "A New Feature in Speech Recognition Based On Wavelet Transform", Proc.ICASSP, 2001.
- [43] M.Sahimi, "Large-Scale porous Media in wavelet Transformation", IEEE computing science and engineering University of southern California, 2003.

- [44] Pinnamaneni, Pujita, Sagar Saladi “3-D Haar Wavelet Transformation and Texture-based 3-D Reconstruction of Biomedical Data Sets.” VIIP (2001).
- [45] K. J. Khouzani, H. S. Zadeh, K. Elisevich, and S. Patel, “Comparision of 2D and 3D wavelet features for TLE lateralization”, Proceedings of SPIE, Vol. 5369, Bellingham, WA, 2004.
- [46] S.Mallat, " A Wavelet Tour of Signal Processing", Academic Press, San Diego, Calif, 1998.
- [47] Jerome M.Shapiro, "Embedded Image coding Using Zerotrees of wavelet Coefficiants", IEEE Transactions on Signal Processing, Vol.4, No.12, 1993.

الخلاصة

أصبح ضغط الفيديو مهمًا بشكل خاص في الوقت الحاضر مع زيادة البيانات المرسلية عبر قنوات الإرسال ، يجب تقليل حجم مقاطع الفيديو دون التأثير على جودة الفيديو. تتم هذه العملية عن طريق قطع الفيديو إلى إطارات ذات أطوال محددة وتحويلها إلى مصفوفة ثلاثية الأبعاد. يستخدم مخطط الضغط المقترح تمثيل الفضاء التقليدي باللون الأحمر والأخضر والأزرق ويطبق تحويل فورييه المنفصل ثلاثي الأبعاد (3D-DFT) أو تحويل الموجة المنفصلة ثلاثي الأبعاد (3D-DWT) إلى مصفوفة الإشارة بعد تحويل دفق الفيديو إلى المصفوفات ثلاثية الأبعاد. يتم ترميز المعاملات الناتجة من التحويل باستخدام خوارزمية التشفير. ثلاثة معايير رئيسية سيتم من خلالها اختبار أداء نظام ضغط الفيديو المقترح ؛ نسبة الضغط (CR) ، ذروة نسبة الإشارة إلى الضوضاء (PSNR) ووقت المعالجة (PT). أظهرت التجارب كفاءة ضغط عالية لمقاطع الفيديو باستخدام التقنية المقترحة مع معدل البت المطلوب ، وهو أفضل معدل بت لضغط الفيديو التقليدي. يتميز التحويل المويج المنفصل ثلاثي الأبعاد بمعدل إطارات مرتفع مع دقة مكانية طبيعية وقابلية التوسع من خلال الدقة المرئية والمكانية وكذلك الجودة مع مزايا أخرى عند مقارنتها بالأنظمة التقليدية الحالية من حيث التعقيد والطاقة المنخفضة والإنتاجية العالية والكمون المنخفض والحد الأدنى من التخزين المتطلبات.

وبالعمل على النظام المقترح تم الحصول على نتائج بنسب ضغط تتراوح بين (95.51 - 87.445).

بالنسبة لـ SNPR ، كانت النتائج بين (51.4-47.78) ديسيل.

وتجدر الإشارة إلى أن قيمة MSE هي (1.08-0.48).

يمكن ملاحظة أن نطاق قيمة التشابه بين (95.3-94.1).



جمهورية العراق
وزارة التعليم العالي والبحث العلمي
جامعة بابل
كلية الهندسة
قسم الهندسة الكهربائية

ضغط الفيديو اعتماداً على خوارزميات التحويل ثلاثية الأبعاد رسالة

مقدمة الى كلية الهندسة في جامعة بابل
كجزء من متطلبات نيل درجة الماجستير في الهندسة /
الهندسة الكهربائية / الإلكترونيك

قدمت من قبل

ضرغام هاني محمد عبيد

بإشراف

الاستاذ الدكتور / ليث علي عبد الرحيم

2021 م

1443 هـ