WAVELET BASED IMAGE COMPRESSION INTEGRATING ERROR PROTECTION via ARITHMETIC CODING with FORBIDDEN SYMBOL and MAP METRIC SEQUENTIAL DECODING with ARQ RETRANSMISSION

By

Veruschia Mahomed
BSc. (Electronic Engineering)

Submitted in fulfilment of the requirements for the
Degree of Master of Science in Electronic Engineering
in the School of Electrical, Electronic and Computer Engineering
at the University of KwaZulu-Natal, Durban

December 2009

Preface

The research described in this dissertation was performed at the University of KwaZulu-Natal (Howard College Campus), Durban, over the period July 2005 until January 2007 as a full time dissertation and February 2007 until July 2009 as a part time dissertation by Miss. Veruschia Mahomed under the supervision of Professor Stanley Mneney.

This work has been generously sponsored by Armscor and Morwadi.

I hereby declare that all the material incorporated in this dissertation is my own original unaided work except where specific acknowledgment is made by name or in the form of a reference. The work contained herein has not been submitted in whole or part for a degree at any other university.

Signed: _____

Name : Miss. Veruschia Mahomed

Date : 30 December 2009

As the candidate's supervisor I have approved this thesis for submission.

Signed: _____

Name : Prof. S.H. Mneney

Date :

Acknowledgements

Publications

The following publications are based on the work presented in this dissertation.

V. Mahomed and S.H. Mneney, "*Wavelet Based Compression: The New Still Image Compression Technique*," Pattern Recognition Association of South Africa (PRASA), Cape Town, South Africa, Nov. 2005.

V. Mahomed and S.H. Mneney, "*Wavelet Based Image Compression and Transmission over Error-Prone Channels*," South African Telecommunications and Networking Applications Conference (SATNAC), Cape Town, South Africa, Sept. 2006.

V. Mahomed and S.H. Mneney, "*Robust EZW and SPIHT via Arithmetic Coding with Forbidden Symbol and MAP Decoding*," Pattern Recognition Association of South Africa (PRASA), Parys, South Africa, Nov. 2006.

V. Mahomed and S.H. Mneney, "*Robust SPIHT via Serially Concatenated Arithmetic Coding with Convolutional Coding and Sequential Decoding*," Military Information and Communications Symposium of South Africa (MICSSA), CSIR, Pretoria, South Africa, July 2007.

Abstract

The phenomenal growth of digital multimedia applications has forced the communications industry to re-look at the manner in which multimedia is transmitted and stored. Multimedia technology will in the future produce such high excessive volumes of data traffic that it will exceed its network capacity, thereby prompting greater focus on higher compression techniques coupled with error protection mechanisms. These techniques and mechanisms will provide an efficient multimedia transmission infrastructure needed to sustain the growth whist minimising the impact of network capacity.

This dissertation describes a myriad of compression techniques for image and videos used currently, with particular focus on the industry progression towards wavelet compression. The dissertation then commences through a review of wavelets and wavelet theory fundamentals for the use of compression before proceeding to outline the advanced wavelet coding algorithms developed for efficient image and video compression. Thereafter, evaluations of the wavelet coders are assessed with recommendations of two coders, EZW and SPIHT, for use in the proposed codec as low-bitrate compression coders.

The dissertation reviews and examines the wireless transmission medium as the preferred medium for error protection of the compressed bitstream. The two wireless mediums selected are the additive white Gaussian noise channel, and the Rayleigh multipath fading channel. The channels are modelled to induce errors in the compressed bitstreams whereby the proposed codec can in turn offer protection of the bitstream for successful transmission.

This dissertation presents a codec offering low-bitrate compression via the use of the wavelet coding algorithms of EZW and SPIHT, combined with error protection incorporating error detection and correction to determine and process errors induced by the wireless channels. Error protection is segmented into error detection and error correction, with error detection involving integer arithmetic coding with forbidden symbol and convolutional coding, and error correction using automatic repeat request (ARQ) retransmission and *maximum a posteriori* (MAP) metric sequential decoding. Error detection via arithmetic coding with forbidden symbol, is able to identify errors that have been produced by noisy channel impairments and interferences during the transmission. Error correction is designed to correct, resolve and rectify the identified errors. The MAP metric sequential decoding concept is multifaceted, as it involves sequential decoding that exploits the optimal stack algorithm that uses a greedy tree search and the MAP decoding metric, which is in turn computed using a complex set of *a priori* and *a posteriori* statistical

probabilities. ARQ retransmission is used as a double error correction mechanism in the event that the MAP decoding fails; it is invoked and requests a retransmission of the erroneous bitstream.

The proposed codec is then compared to current three systems arithmetic coding and decoding, convolutional coding with MAP decoding and arithmetic coding, convolutional coding with MAP decoding and arithmetic decoding, through multiple simulations focusing on image quality and erasure performances. Results show that the proposed codec is competitive and its performance surpasses three systems used in the evaluation. The proposed ARQ-MAP scheme proved better and showed greater improvement in error-free decoding than the three systems. The highly successful coupling of error detection, using the forbidden symbol and error correction using MAP metric sequential decoding showed immense potential and ability for error-free compression and transmission of images and video.

Table of Contents

List of Figures

List of Tables

List of Acronyms

LZW : Lempel-Ziv-Welch

JPEG : Joint Photographic Experts Group

ISO/CCIT: International Organization for Standardisation/International Telegraph and Telephone Consultative Committee

MPEG: Motion Picture Experts Group

DCT : Discrete Cosine Transform

DWT : Discrete Wavelet Transform

EBCOT: Embedded Block Coding with Optimised Truncation

ROI : Regions of Interest

ISO/IEC: International Organization for Standardisation/International Electrotechnical Commission

VHS : Video Home System

HDTV: High Definition Television

DVD : Digital Video Decoder

YCrCb: Luminance Chrominance

MB : Macro-Block

I : Image

P : Predictor

B : Bi-directional

FF : Fast Forward

FR : Fast Reverse

ITU-T : International Telecommunication Union Standardisation

CIF : Common Intermediate Format

QCIF : Quarter Common Intermediate Format

AVC : Advanced Video Coding

VCEG: Video Coding Experts Group

IPTV : Internet Protocol Television

CAVLC: Context Adaptive Variable Length Coding

CABAC: Context Adaptive Binary Arithmetic Coding

VMP : Visual Main Profile

HLP : High Latency Profile

ASP : Advanced Simple Profile

MP : Main Profile

QMF : Quadrature Mirror Filter

MRA   : Multi-resolution Analysis

FWT   : Fast Wavelet Transform

EZW   : Embedded Zerotree Wavelet

SPIHT: Set Partitioning in Hierarchical Trees

LSP   : List of Significant Pixels

LIP   : List of Insignificant Pixels

LIS   : List of Insignificant Sets

SR   : Stack-Run

ECECOW: Embedded Conditional Entropy Coding of Wavelet Coefficients

SFQ   : Space Frequency Quantization

AWGN: Additive White Gaussian

PDF   : Probability Density Function

RF   : Radio Frequency

ISI   : Inter-Symbol Interference

BPSK : Binary Phase Shift Keying

2-PSK : 2-bit Phase Shift Keying

DBPSK: Differential Binary Phase Shift Keying

AC   : Arithmetic Coding

FS   : Forbidden Symbol

CC   : Convolutional Coding

MAP : *Maximum a posteriori*

Erfc   : Complementary Error Function

MIMO: Multiple Input Multiple Output

VLC   : Variable Length Coder

MSE   : Mean Square Error

PSNR : Peak Signal to Noise Ratio

dB   : Decibels

bpp   : bits per pixel

BER   : Bit Error Rate

SNR   : Signal to Noise Ratio

S/N   : Signal to Noise Ratio

PER   : Packet Erasure rate

Eb   : Energy per bit

N0   : Spectral Noise Density

R-D   : Rate-Distortion

ME   : Motion Estimation

MC   : Motion Compensation

MD   : Motion Detection

2-D   : Two Dimensional

3-D   : Three Dimensional

[a,b)   : Interval from a to b, including a but not b

CHAPTER 1 -   INTRODUCTION

The last decade has produced a massive yet impressive injection of digital multimedia applications into the digital world. As data intensive multimedia based applications like High Definition Television (HDTV) [88], videoconferencing, Video on Demand (VoD) [90] and IPTV [89] evolve at an explosive rate, the necessity for high bandwidth and increased Quality of Service (QoS) has become the driving force for efficient multimedia deployment.

As new multimedia productivity drives bandwidth demands and a richer experience through increased QoS is needed, the requirement for high compression with efficient error protection coding has become a mandatory step in data intensive multimedia transmission due to the sheer volume of data transmitted and the unreliability of the channel due to failures and interferences. Bandwidth-on-demand schemes premise that network broadband technologies will never keep up with the bandwidth demand, thereby necessitating the development of applications with high QoS in terms of advancing compression and error protection.

Although compression and error coding studied independently produce significant advancements in image and video technology, it is the integration of advanced compression with error protection coding that requires further investigation in order to develop a joint system that efficiently represents the storage and communication of image and video multimedia. The key factors to be considered are; the inclusion of error protection coding to the already reduced low bit rate compressed sequence, will affect the compression attained in what way? And will the image and video quality be retained or can it be improved?

This dissertation aims to present and evaluate the inter-working system of advanced low bitrate compression with error protection coding whilst producing a representation of the multimedia with improved visual quality and maintaining data integrity.

1.1 COMPRESSION

Compression permits the representation of a reduced amount of information with negligible quality loss and minimum distortion. It requires the elimination of irrelevant and redundant information in order to reduce the amount of data necessary to encode, store and transmit information efficiently. There are various types of compression approaches: data, audio, image and video. This dissertation mainly concerns itself with multimedia based compression, namely image and video compression.

The fundamental components prompting the need for image compression are irrelevant pixel information and redundant pixel information [1]. These information methods can be theoretically expressed as redundancy reduction and irrelevancy reduction [1]. Irrelevancy reduction removes or alters information that produces little or no difference to the perception of the image. This type of reduction generally involves perceptibility to the human eye. Redundancy reduction removes duplication and repetition within images and video. There are three types of redundancies [1]:

- Spatial redundancy: correlation between neighbouring pixel values.
- Spectral redundancy: correlation between different colour planes and or spectral bands.
- Temporal redundancy: correlation between adjacent frames in a sequence of images in video applications.

Compression can be further classified as either lossless compression or lossy compression. Both techniques concern the reconstruction fidelity of the compressed information, where lossless compression involves perfect reconstruction without any information loss and lossy compression produces an approximated replica of the original information with some form of information loss.

1.1.1 Lossless Compression

Lossless compression [1] allows the original data to be reconstructed from the compressed data without any loss of information. Lossless compression ensures that the restored data after decompression is identical to the original uncompressed data, allowing no approximations or deviations to occur. Lossless compression is commonly known as error-free compression as the accuracy of the reconstructed image is never in question.

Lossless image compression is based on redundancy reduction as it reduces interpixel redundancies and eliminates coding redundancies in order to compress the image. This method of compression does not involve quantization like the lossy approach, and thus does not reduce the accuracy of the scheme. It is a reversible technique meaning that the original data can be reconstituted.

Some error-free lossless coding approaches are; variable length coding, Huffman coding [87], arithmetic coding [52], [55] and Lempel-Ziv-Welch (LZW) coding [86]. Lossless compression as compared to lossy compression is able to achieve a modest amount of compression.

1.1.2 Lossy Compression

Lossy compression [1] produces reconstructed data that is an approximation of the original data. It can guarantee high compression ratios at the expense of lost information. Lossy compression can affect the fidelity or quality of the data depending on the amount of information discarded. In lossy compression, information is permanently discarded and cannot be recovered during the decompression stage. The reconstruction can produce errors which may or may not be tolerable thereby distorting the image accordingly.

Lossy compression is primarily based on irrelevancy reduction, although it also employs redundancy reduction. The irrelevancy reduction strategy focuses on the characteristics of human visual perception to dispose of irrelevant information. Human visual perception relies on the fact that the human eye is less sensitive to colour (chrominance) than to brightness (luminance) in an image, thus content that is visually redundant is removed. In addition redundancy reduction removes the interpixel redundancies and coding redundancies. This scheme uses a quantizer to reduce the psycho visual redundancies experienced. This operation is irreversible therefore no recovery of discarded information is possible.

Some key lossy compression techniques are JPEG [2] and JPEG 2000 [3] for still image compression and MPEG [6] below, H.263 [8] , H.264 [9], [10] and MPEG4 [7] for video compression. These schemes can reduce the compressed image content to as little as *1%* of the original, although compression less than *10%* can produce significant visual distortion to the compressed image. However, lossy compression is capable of achieving much higher compression than lossless compression. Thus the proposed wavelet coding schemes employ a form of lossy compression as its primary compression technique for high compression.

1.2   WAVELETS

This dissertation concerns various aspects of compression that involve the use of wavelets. Although wavelets are a relatively new concept (approximately 15 years old), it has gained widespread acceptance in the signal processing community and particularly in image compression. The use of wavelets through the Discrete Wavelet Transform offers a more natural description of images than the block-based Discrete Fourier Transform therefore making wavelet compression the preferred choice for image compression.

The discrete wavelet transform is essentially a de-correlating transform that reduces the correlation between the pixels in an image, thereby producing better compression. The wavelet transform also offers greater energy compaction due to its coefficient localization in both the frequency and time domains. In addition, wavelet compression involves advanced analytical mathematical concepts like multiresolution analysis, filter banks, wavelet decomposition and subband coding which facilitates progressive image transmission through the use of robust wavelet based coding schemes like the Embedded Zerotree Wavelet (EZW) [28], [29] and Set Partitioning in Hierarchical Trees (SPIHT) [32] coding schemes.

Wavelet image compression describes pixel regions of varying size, shape and location and performs advanced averaging and differencing operations using the wavelet analysis concepts listed above. Wavelet image compression through the wavelet transform alleviates blocking artifacts, whilst the inherent multiresolution nature of wavelet decomposition produces superior energy compaction, maintaining adequate perceptual quality of the reconstructed image. The wavelet based coding schemes offer substantial improvements in picture quality at higher compression ratios than Fourier-based schemes.

Wavelet-based image compression has developed into a powerful, sophisticated compression technique able to produce superior compression ratios with minimal image degradation. Thus the progress and advancement of wavelet compression has gained momentum and has resulted in the leading compression alternative to the current compression standards.

1.3   ERROR RESILIENCE

Error resilience refers to coding mechanisms that enhance the capability of the compressed bitstream to withstand and resolve channel induced errors during transmission. As a result, error

resilient coding is required for the reliable transmission of images and video over unreliable channels.

Impaired channels produce losses, delays and inject random bit errors into the compressed bitstream during transmission, therefore the proposal of error resilience coding tools are imperative in order to detect and analyse the errors, conceal and then correct them, producing an image with minimal visual defects. Wireless channels are proposed as the transmission medium of choice for the system as they can produce catastrophic error rates, propagation delays and channel losses which constitute a worst-case scenario.

Typical error resilience coding for wireless channels involve error protection in the form of error detection and error correction. Error detection precedes error correction and is designed to permit the detection of bit errors. It determines whether the transmitted bitstream is corrupted by the channel or remains intact. Error detection is simpler to implement than error correction and involves additional redundancy in the transmitted bitstream at the expense of compression. Error detection permits either retransmission of the bitstream or error correction. Retransmission is inadequate as the compression of the system is sacrificed, thus error correction follows.

Error correction involves statistically reconstructing the originally transmitted bitstream through accurate mathematical prediction models. Depending on the errors expected, an error correction coding scheme is chosen for the desired application. There are two types of errors that occur in the communication channel: random bit errors and burst errors. Random bits errors tend to generate isolated bit flips during transmission where the bit errors are independent of each other. Burst errors on the other hand are inclined to produce clumps of bit errors during a single transmission. Error correcting codes are specifically designed to attempt to correct both of these types of errors in order to deliver successful error resilience coding to the system.

The addition of error resilience coding introduces redundancy into the system which in turn diminishes the overall achievable compression of the image. There is a trade-off between the amount of redundancy added and the compression obtained. Thus a balance will need to be achieved for optimum compression and error resilience.

1.4 LAYOUT of DISSERTATION

This dissertation describes wavelet image compression involving transmission over error prone channels whilst providing error detection and correction to corrupted bitstreams producing visually acceptable images. The dissertation begins with a literature review of theoretical concepts like wavelet theory before progressing to wavelet compression coding techniques and finally wireless channel models. Once the background theory of the concepts employed are examined thoroughly, the dissertation then outlines the components involved in the design of the proposed codec before discussing its validation through the results and discussion section.

Chapter 2 provides a general introduction into the compression standards for still image and video compression systems that are currently available. The compression techniques and procedures employed by the various compression standards are explored and outlined in detail. Specific focus is given to the still image compression standards of JPEG and JPEG 2000 compression and the difference between the transforms used in either compression techniques. The video compression standards of MPEG, MPEG4, H.263+ and H.264 are discussed with particular focus given to its compression features and the advanced video coding techniques used. The chapter also includes commonly applied performance metrics used to evaluate and assess the performance of these image processing concepts; namely error rates and Signal-to-Noise ratios. The chapter concludes with a performance evaluation of the still image compression and the video compression standards indicating which standard is superior in comparison to the each other.

Chapter 3 forms the bulk of the literature review concerning wavelet theory, wavelet families and wavelet coding techniques. Wavelet theory examines the wavelet transform theorem and its progression towards multiresolution analysis through a mathematical framework. The main focus is to provide an overview of the development from the wavelet theory to wavelet based image compression. A brief introduction into wavelet families are given where the selection of a wavelet family, can be used to accurately describe an image. A performance comparison detailing the best wavelet family to be used for image compression is described. The final section of this chapter includes the various wavelet compression coding algorithms developed thus far. Features and improvements as well as performances of each algorithm are highlighted clearly illustrating the progression and development of wavelet coding compression. Particular focus is given to the wavelet algorithms of EZW and SPIHT as the proposed codec employs these wavelet coders for its final performance evaluation. Results of these two wavelet

compression algorithms against the JPEG and JPEG 2000 standardised compression algorithms are also described.

Chapter 4 investigates the use of wireless channels to induce errors in wavelet compression in order to observe error propagation as well as assess the error correction technique proposed for the dissertation. The two types of wireless channels investigated are the additive white Gaussian noise (AWGN) and the Rayleigh multipath fading channels. The chapter discusses the channel properties, impairments and various fading characteristics before theoretical channel models are simulated. A key property in the modelling of a channel is the type of signal modulation used to transmit the data. The proposed codec in the dissertation uses binary phase shift keying (BPSK) modulation, a simple yet effective modulation technique that is briefly introduced in this chapter. The bit error rate performance evaluations of the wireless channels are presented in the chapter showing the degradation of the channel. Also included is a detailed performance overview of the EZW and SPIHT compressed images transmitted over these corrupt channels and the resultant decompressed images. These performance evaluations are the basis of the simulated results exhibited by the proposed codec in the following chapters.

Chapter 5 is fundamental as it outlines the conceptual theory behind the subsystems used for the proposed codec. The bulk of the system model theory involving error coding is extensively described in this chapter. Error protection is broken down into two segments; error detection and error correction which involves integer arithmetic coding with forbidden symbol and convolutional coding, and automatic repeat request transmission with m*aximum a posteriori* (MAP) metric sequential decoding respectively. The chapter presents the mathematical and statistical concept of the scheme along with the algorithm approach showing the potential of the system as an efficient error coding codec.

Chapter 6 is the primary focus of the dissertation which involves the system model and representation of the proposed codec used to achieve the error protection system that detects and corrects the errors introduced by the corruptible wireless channels. The chapter begins with a graphical block diagram illustrating the system processes involved in conception of the proposed codec. The proposed codec subsystems involve EZW and SPIHT wavelet compression coding coupled with arithmetic coding with forbidden symbol detection, convolutional encoding and ARQ retransmission with MAP metric sequential decoding. The proposed codec uses either the Gaussian channel or Rayleigh fading channel as the transmission medium. The error correction decoding procedure of the proposed codec is achieved through a double correction mechanism of MAP metric sequential decoding with ARQ retransmission. The finer details of the system including the type of image resolution, wavelet family, the

number of decomposition levels, modulation, convolutional code rate, constraint length and channel SNR range are all explicitly defined and justified within this chapter, clearly describing all the elements involved in the design of the proposed system.

Chapter 7 consolidates and confirms the theory presented in chapter 6, in the form of results and discussion of the proposed system (arithmetic coding with forbidden symbol detection, convolutional coding, and ARQ retransmission with MAP decoding) in relation to currently used error coding systems. Current standard systems used for comparison include; arithmetic coding and decoding, convolutional coding and MAP decoding and arithmetic coding, convolutional coding with MAP decoding, all without forbidden symbol detection. The chapter illustrates in a methodical fashion its performance in terms of PSNR and packet erasure rate (PER) for a test set of images. The performance simulations were executed and implemented in the Matlab simulation environment. The results obtained in this chapter clearly outline the performance of the proposed codec in relation to the above mentioned current systems.

The final chapter, Chapter 8, concludes the dissertation and summarises the important aspects involved in design of the proposed codec and its related results. It reviews each chapter before concluding the dissertation with final remarks and possible future research.

1.5 EXECUTIVE SUMMARY

The aim of this dissertation is to provide error protection to wavelet compressed images transmitted over corrupt channels. The dissertation integrates wavelet compression and error resilience to produce a low bitrate compression codec with superior error decoding. The error decoding mechanism employed is a two-fold error detection and error correction scheme. The compression scheme applied involves two highly convoluted wavelet based coding algorithms.

The dissertation outlines and compares various image and video compression standards, illustrating the progression of wavelet based compression through the comparison of the JPEG standard and the highly performing JPEG 2000 standard. The need for wavelet based compression then focuses on the acclaimed wavelet compression coding schemes of EZW, SPIHT and ECECOW methods amongst others. Two highly destructive wireless channels are proposed for the development of a proposed codec which can withstand error induction during transmission of wavelet compressed media.

The proposed codec is fundamentally designed to detect and correct possible errors induced in the compressed bitstream by channel interference such as noise or fading. Arithmetic coding with forbidden symbol detection, convolutional coding, automatic repeat request transmission protocol and *maximum a posteriori* (MAP) metric sequential decoding are proposed as an alternative method to correct errors. It is found that although the introduction of the forbidden symbol causes additional redundancy in the bitstream, its combination with convolutional coding and MAP metric sequential decoding produces an improved error correction mechanism and with ARQ retransmission strategy providing a double correction mechanism it is able to increase image quality of the decompressed image and decrease overall packet erasures at low SNR's.

This innovative error coding scheme for wavelet image compression produces superior performance results. An intensive and rigorous performance evaluation is conducted on the proposed codec against currently employed error correction standards including variations of the arithmetic encoding, convolutional encoding and MAP decoding stages. The proposed system is able to produce results that are bitrate competitive whilst maintaining efficient error correction and reduced packet erasures for a diverse range of test images.

CHAPTER 2 - CURRENT COMPRESSION STANDARDS

Image and video compression is presently the current focus amongst researchers and is being driven forcefully by the need for international standardisation of multimedia content. Standardisation is critical in facilitating both interoperability and compatibility among various imaging systems as well as in deploying image and video technology throughout the world. However, the pursuit of such technology must be achieved in a cost-effective manner. These standards are indicative of the latest technology researched and are an excellent benchmark with regards to the technology evolution and international current trends.

2.1 STILL IMAGE COMPRESSION

The increase in the use of digitised still images, either through the Internet, digital imaging or digital photography, has resulted in the need to compress such imagery in order to allow for the economical storage and fast data transfers.

2.1.1 JPEG

JPEG (Joint Photographic Experts Group) [2] is the name of a joint ISO/CCITT committee that defined the compression standard for continuous tone still images. JPEG was intended for the image compression of photographic still images. JPEG is extremely popular and has thus become a widely adopted compression standard particularly in the Internet arena.

The JPEG compression standard has two distinct approaches for its image compression [2]: a Discrete Cosine Transform (DCT) based technique focusing on the baseline sequential method for lossy compression and a predictive scheme for lossless compression.

2.1.1.1 Lossy Baseline Sequential Coding

The lossy baseline sequential coding scheme [1], [2] is based on the DCT and involves the following processes as illustrated in the block diagram in Figure 2-1.

29
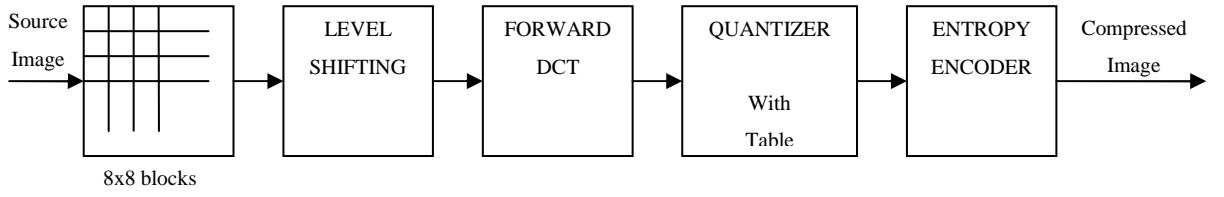
Figure 2-1: Block Diagram of JPEG DCT Baseline Sequential Coding scheme.

The block diagram shows the source image which is first divided into 8x8 pixel blocks. Each 8x8 pixel block is then level shifted from unsigned integer to signed integer by subtracting each pixel value by 2^{n-1} where 2^n is the number of gray levels used and n is the bit precision of the image component. Each 8x8 block is then fed into the forward DCT and a two dimensional DCT of the block is then computed. These DCT coefficients are quantized using a defined quantization table provided by the standard and the following quantization equation [1];

$$F_Q(u,v) = Round\left(\frac{F(u,v)}{Q(u,v)}\right), \qquad (2.1)$$

where $F(u,v)$ is the DCT coefficient and $Q(u,v)$ is the quantizer step size from the quantization table. The quantization coefficients are then reordered into a one dimensional sequence using the zig-zag scanning pattern shown in Figure 2-2 [2]. The coefficients are automatically arranged according to increasing spatial frequencies i.e. low frequencies before high frequencies.
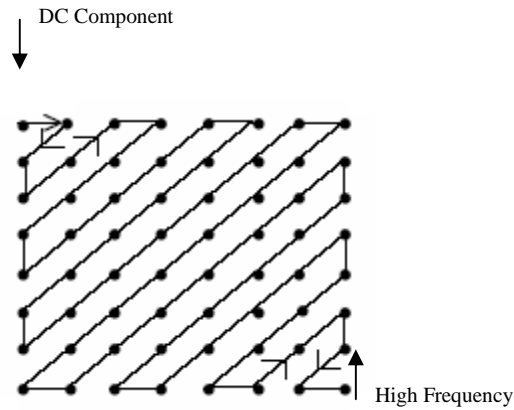
Figure 2-2: Zig-Zag Scanning Pattern [2].

Finally the zig-zag sequence of quantized DCT coefficients are entropy encoded using either Huffman coding or arithmetic coding. For the baseline sequential method, Huffman coding is specific to the entropy coding stage.

2.1.1.2 Predictive Lossless Coding

The predictive lossless coding scheme [1], [2] is an error-free compression approach that does not implement the DCT. It eliminates interpixel redundancy by coding only the new information of the pixel. This new information is the difference between the actual value and the predicted value of the pixel. Hence this scheme utilises a predictor. This system is illustrated in Figure 2-3.
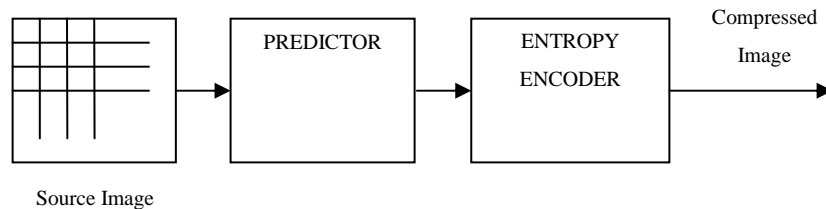
Figure 2-3: Diagram of JPEG Predictive Lossless Coding scheme.

The predictor combines up to three neighbouring samples (A, B, C) to form a prediction (X) as shown in Figure 2-4.

Figure 2-4: Diagram of the prediction neighbourhood [2].

The difference between the actual value and the prediction is then coded using either Huffman coding or arithmetic coding. Lossless codec's normally produce around 2:1 compression rate.

2.1.1.3 JPEG Features

The JPEG compression standard has a few key features that make it one of the most popular and comprehensive continuous tone still image compression standards. These features are highlighted as follows [2]:

- It is state of the art with regard to compression and image quality. JPEG lossy compression usually has a 5:1 compression rate without visible loss for grayscale images and between 10:1 and 20:1 compression rate without visible loss for colour images. For its lossless compression scheme it achieves a 2:1 compression rate.

- It is not restricted to images of certain dimensions, colour spaces, pixel aspect, and scene content therefore making it applicable to any type of continuous tone digital source image.
- It has tractable computational complexity i.e. it can perform on a range of CPU's.
- It utilises sequential, progressive, lossless and hierarchical coding modes of operation.

2.1.2 JPEG 2000

JPEG 2000 [3], [4] is a wavelet based still image compression standard using the Embedded Block Coding with Optimised Truncation (EBCOT) coding scheme. JPEG 2000 provides low bit rate operation with superior rate distortion and image quality as compared to the existing JPEG standard. It was developed as a new still image coding system catering for different types of images (binary, grayscale, colour) with different characteristics (text, rendered graphics, natural images etc.).

JPEG 2000 being a wavelet based coding scheme employs the Discrete Wavelet Transform (DWT) instead of the Discrete Cosine Transform used in the JPEG compression standard. The wavelet transform reduces the amount of information contained in the picture and thus offers greater efficiency than the cosine transform. The DCT expresses a signal in terms of frequency and amplitude at a single instant in time whereas the DWT expresses a signal over the complete time and thus also contributes to the increased efficiency of the standard.

The fundamental architecture of the JPEG 2000 standard is outlined in Figure 2-5. It involves image tiling, DC level shifting, discrete wavelet transformations, quantization and finally entropy coding [4].

Source Image → TILING → DC LEVEL SHIFTING → FORWARD DWT → QUANTIZER → ENTROPY ENCODER → Compressed Image
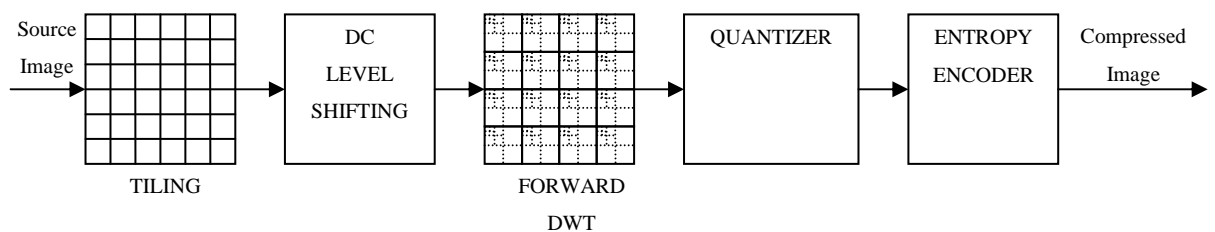
Figure 2-5: Diagram of JPEG 2000 coding scheme.

'Image tiling' is first performed on the source image. 'Tiling' refers to the partitioning of the source image into rectangular non-overlapping blocks or tiles. Each tile is compressed independently. All samples of the image tile are DC level shifted by subtracting the quantity 2^{n-1} from each sample, where n is the bit precision of the image component. The DWT is performed on each tile. The transform coefficients are quantized thereby reducing the coefficients

32

precision. The entropy coding stage is achieved by means of Embedded Block Coding with Optimised Truncation (EBCOT) [5].

2.1.2.1 Embedded Block Coding with Optimised Truncation

EBCOT, created by Taubman [5] is the low level entropy coding framework employed by the JPEG 2000 standard. It uses the wavelet transform to decompose the image into various subbands. Each subband is further partitioned into relatively small blocks called code blocks. The coefficients in each individual code block, are independently coded into an embedded bitstream, thereby implying that each code block generates a separate bitstream without using any information from other code blocks.

The actual coding of the bitplane in each code block into its embedded bitstream involves three critical passes: the significance propagation pass, the magnitude refinement pass, and the cleanup pass. These passes separate the significant information from the insignificant information by identifying the bitplane in code blocks that have non-zero or significant coefficients. This ensures that the coding process need only code the bitplanes that have significant coefficients and can discard the bitplanes that have zero or insignificant coefficients.

After the bitplane coding process each embedded bitstream can then be truncated independently into different discrete lengths. This truncation point as it is known is used to achieve a maximum target bit rate with minimal rate distortion. Hence the algorithm is named Embedded Block Coding with Optimised Truncation. This approach to the JPEG 2000 standard leads to a highly precise coding structure with refined support.

2.1.2.2 JPEG 2000 Features

The significant features of this wavelet transform standard include [3]:

- Superior low bit-rate performance.
- Continuous-tone and bi-level compression – should compress and decompress images with various dynamic ranges (*1* to *16 bits*) for each colour component.
- Lossless and lossy compression.
- Progressive transmission by pixel accuracy and resolution – allows images to be reconstructed with different resolutions and pixel accuracy.
- Random code stream access and processing – allows regions of interest (ROI) in the image to be randomly accessed and/or decompressed with less distortion.
- Robustness to bit errors.

- Open architecture – optimises the system for different image types and applications.
- Real time coding – compressing and decompressing images with a single pass.

2.2   VIDEO COMPRESSION

Video compression deals with the compression of digital video data. Unlike still image compression, video compression exploits the correlation between image frames by discarding redundant information whilst predicting motion. Video compression reduces picture redundancy while allowing video information to be transmitted and stored in a compact yet efficient manner. Digital video data rates are very large and therefore consume a great amount of bandwidth, storage and computing resources, thus the need for video compression becomes imperative.

2.2.1   MPEG

MPEG (Moving Pictures Expert Group) [6] is an ISO/IEC compression standard developed for digital audio and video formats. The MPEG1 and MPEG2 standards are based on motion compensated block-based transform coding techniques while MPEG4 exploits object-based compression techniques.

MPEG1 was designed to achieve VHS-quality video on a regular CDROM. This format is commonly known as the VideoCD. MPEG2 was designed to handle the demands associated with broadcast and entertainment applications like DSS satellite broadcast, HDTV and DVD video. The MPEG1 and MPEG2 standards are layered DCT-based video compression algorithms. Since these two formats are based on similar concepts, they have been commonly referred to as MPEG video.

MPEG video uses block-based coding schemes which divide the picture into blocks of 8x8 pixels. A collection of 6 of these blocks (4Y blocks, 1Cr block and 1Cb block in the YCrCb colour space) is called a macroblock (MB). Each MB can be further represented in 3 different formats when referring to the YCrCb colour space. These are 4:2:0, 4:2:2, 4:4:4 which is illustrated in Figure 2-6. MPEG video operates on the YCrCb colour space as the human eye is more sensitive to changes in luminance or Y.
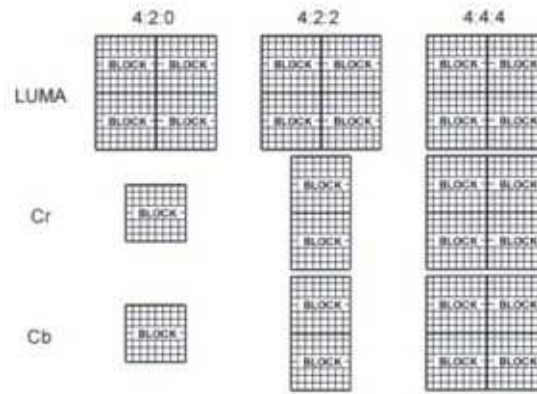
Figure 2-6: Diagram of the YCrCb macroblock format [6].

Since compression involves the removal of spatial and temporal redundancy, the MPEG compression standard focuses on these two basic techniques in its algorithm. The two techniques are commonly referred to as intraframe/spatial coding and interframe/temporal coding [6].

Intraframe coding involves DCT-based coding techniques similar to those applied in the JPEG compression standard that entailed DCT, quantization and entropy coding. This form of coding removes spatial redundancies, which are redundancies that occur within the frame. Interframe coding utilises a technique known as block-based motion compensation prediction using motion estimation. This coding removes temporal redundancies or redundancies that are present between frames.

The intraframe compression portion of the MPEG video produces I (Image) frames which are subsequently used to predict P (Predictor) frames and B (Bi-directional) frames in the interframe section of the standard. P frames can be predicted from either I frames or other P frames immediately preceding it. B frames are either coded based on forward prediction from previous I or P frames or on backward prediction from succeeding I or P frames. This process forms the block-based motion compensation prediction component of the compression scheme and is illustrated in Figure 2-7.

35
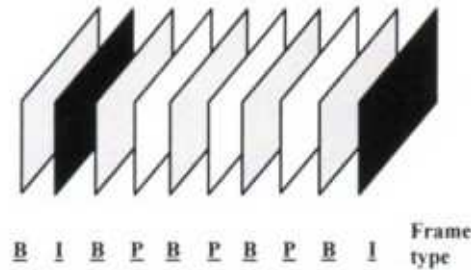
Figure 2-7: Diagram of I, P and B frames in Interframe coding [6].

Since B frames cannot predict future frames, errors generated within a B frame will not propagate further within the video sequence.

As mentioned previously the temporal prediction technique uses motion estimation for interframe coding. Motion estimation is a concept that represents changes between two consecutive video frames. This process involves each MB in the current frame being compared to the previous frame. Once a match is found, motion vectors are assigned to the MB indicating how far horizontally and vertically the MB was displaced. This offset represented by motion vectors forms a prediction. Predictions for all MB's in the current frame are obtained and a prediction frame is constructed. The prediction frame is then subtracted from the current frame resulting in a residue frame. This residue frame is coded using DCT, quantization and entropy coding before being transmitted.

The combination of the above compression techniques, make MPEG video highly scalable. Its block based motion estimation scheme supports important functionalities some of which are: random access, fast forward (FF) and the fast reverse (FR) playback operations.

2.2.2 MPEG4

MPEG4 [7] is a low bit rate multimedia format. MPEG4 supports a mix of media like broadcasting, movie and multimedia applications, thus allowing recorded video images and sounds to coexist with their computer generated counterparts. It provides additional functionalities like bitrate scalability, object based representation and intellectual property management and protection.

MPEG4 provides a standardised method to represent units of aural, visual and audiovisual content. This content is referred to as media objects and can be of natural or synthetic origin. MPEG4 forms audiovisual scenes composed of primitive media objects such as [7]:

- Still images (fixed background)
- Video objects (talking person without background)
- Audio objects (voice associated with that person or background music)

It also provides standardised methodology to describe a scene by [7]:

- Placing media objects anywhere within a given coordinate system.
- Applying transforms to change the geometrical or acoustical appearance of a media object.
- Grouping primitive media objects in order to form compound media objects.
- Applying streamed data to media objects in order to modify their attributes.
- Changing interactively the user's viewing and listening points anywhere in the scene.

There are various aspects of MPEG4 which define the standard, whilst bringing higher levels of interaction with regards to the content. The various aspects include the system component, audio component and the visual component. The visual factor in the MPEG4 standard is critical as it allows coding of natural images and video with synthetic or computer generated scenes.

MPEG4's visual component has the following significant features [7]:

- Efficient compression of textures for texture mapping on 2D and 3D meshes.
- Content-based coding of images and video allows separate decoding and reconstruction of arbitrarily shaped video objects.
- Random access of content within video sequences allows functionalities such as pause, fast forward and fast reverse of stored video.
- Error resilience allows accessing image and video over a wide range of storage and transmission media and includes the operation of image and video compression algorithms in error prone environments at low bit rates.

The basis of the MPEG4 video coding standard is similar to that of the MPEG video in that it is a block based predictive differential video coding scheme and utilises the following techniques to provide greater compression:

- Division of picture into 8x8 blocks of 16x16 MB's.
- Motion compensated prediction.
- Transform coding with DCT.

- Quantization.
- Run length and Huffman coding for variable length codes.

In conjunction to these compression methods a number of motion prediction techniques are used to improve coding efficiency and flexibility, these are listed as follows [7]:

- Standard 8x8 or 16x16 block based motion estimation and compensation with up to quarter pel accuracy.
- Global motion compensation for video objects - which is based on global motion estimation, image warping, motion trajectory coding and texture coding for prediction errors.
- Global motion compensation based for static 'sprites' - a static sprite is a large still image describing panoramic background. Only 8 global motion parameters describing camera motion are coded to reconstruct the object. These parameters represent the appropriate affine transform of the sprite transmitted in the first frame.
- Quarter pel motion compensation - enhances the precision of the motion compensation scheme.
- Shape adaptive DCT - in the area of texture coding, it improves the coding efficiency of arbitrary shaped objects.

MPEG4 embraces a multitude of features which are outlined below. The standard provides solutions in the form of tools and algorithms for [7]:

- Efficient compression of images and video.
- Efficient compression of textures for texture mapping on 2D and 3D meshes.
- Efficient compression of implicit 2D meshes.
- Efficient compression of time-varying geometry streams that animate meshes.
- Efficient random access to all types of visual objects.
- Extended manipulation functionality for images and video sequences.
- Content based coding of images and video.
- Content based scalability of textures, images and video.
- Spatial, temporal and quality scalability - Scalability refers to the ability to decode a part of a bitstream and reconstruct images or image sequences with reduced decoder complexity and thus reduced quality, reduced spatial resolution, reduced temporal resolution and equal temporal and spatial resolution but with reduced quality.

- Error robustness and resilience in error prone environments – where error resilience tools can be divided into resynchronization, data recovery and error concealment.

The MPEG4 standard introduces and supports new concepts for object-based user interactivity. It enables greater reusability and flexibility of content as well as greater interaction with the content. More importantly it provides low bit-rate compression to various multimedia applications.

2.2.3 H.263+

H.263+ [8] is a low bit-rate compression standard developed by the ITU-T. This standard supports video compression for video conferencing and video telephony applications. It combines the features of both the MPEG and H.261 standards. H.263+ allows for the use of five standardised picture formats [8]: CIF (Common Intermediate Format), QCIF (Quarter-CIF), sub-QCIF, 4CIF and 16CIF.

As with MPEG, H.263+ supports block-based motion estimation and motion compensation. The block-based approach involves the division of the image into macroblocks and the motion estimation and compensation involves motion vectors representing the change between frames, as discussed previously in the MPEG section.

The H.263+ compression standard is an extension of its predecessors H.261 and H.263 in that it includes several additional features and modes for improving efficiency and picture quality. These are listed as follows [8]:

- Unrestricted motion vector mode – motion vectors can reference pixels outside the picture boundary.
- Syntax based arithmetic coding – arithmetic coding is used instead of Huffman coding.
- Advanced prediction mode – uses four motion vectors per MB and overlapped block motion compensation.
- PB frame mode – both P frames and B frames are treated as a single entity and are coded together.
- Advanced intra coding mode – improves the efficiency for intra MB coding by using spatial prediction of DCT coefficient values.
- Deblocking filter mode – reduces block artifacts using as adaptive filter.
- Slice structure – improves error resilience by grouping MB's.

- Reference picture selection mode – allows for selection of a previous frame to generate a prediction of the current frame.
- Scalability related enhancements – provides SNR scalability, spatial and temporal scalability.

H.263+ was aimed particularly at video coding for low bit rates. It offers an improvement to the MPEG standard whilst maintaining its distinct features.

2.2.4   H.264 / MPEG4 part 10 AVC

H.264/MPEG4 part 10 Advanced Video Coding (AVC) [9], [10] is the latest video compression standard available. It is a joint collaboration of the ITU-T Video Coding Experts Group (VCEG) and the ISO/IEC Moving Picture Experts Group (MPEG). H.264/AVC or H.264/MPEG4 has become a widely used video compression standard and has been implemented in numerous video applications like, mobile TV, video conferencing, IPTV [89], HDTV [88] etc.

H.264/MPEG4 has a similar structure to the previous video compression standards in that it still encompasses the block-based, motion compensated video compression characteristics. The H.264/MPEG4 standard boasts a few new key features that enhance the performance of the standard. The features that enhance coding efficiency are [9], [10]:

- Variable block-size motion compensation with small block sizes – Supports motion compensated block sizes as large as 16x16 with a minimum block size of 4x4.
- Quarter pel motion compensation.
- Multiple reference frames – Up to 16 different reference frames can be used for inter-picture coding.
- Weighted prediction – Scaling operation by applying a weighting factor to the samples of motion compensated prediction data.
- In-loop deblocking filter – Reduces blocking artifacts by operating on the horizontal and vertical block edges.
- Integer transform – Previous standards use the 8x8 DCT, a new 4x4 integer transform which is derived from the DCT. It reduces blocking and ringing artifacts.
- Quantization – Uses scalar quantization. One of 52 quantizer step size scaling factors is selected for each macroblock, where the step sizes increase at a rate of *12.5%.*
- Entropy coding – Two techniques can be applied, Context Adaptive Variable Length Coding (CAVLC) and Context Adaptive Binary Arithmetic Coding (CABAC).

H.264/MPEG4 initially supported three profiles: baseline, main and extended. A crucial amendment to the standard was the fidelity range extensions that expanded the interoperability of the standard and introduced a fourth profile, high. A profile is a set of coding tools that generates a compliant bitstream. The capabilities of the profiles are listed below [9], [10].

- Baseline profile – Designed for applications that are low-cost and using little computing power. Profile used in videoconferencing and mobile video.

- Main profile – Intended for broadcast and storage applications. Profile used in digital storage media and television broadcasting.

- Extended profile – Primarily for streaming video. It has high compression capability, robustness and server stream switching. Profile used in streaming video.

- High profile – Intended for broadcasting, disc storage, particularly for high definition television application. Profile used for content contribution, content distribution, studio editing, post processing.

H.264/MPEG4 demonstrates significant improvements in terms of picture quality, coding efficiency, increased error resilience and flexibility. It delivers considerable compression efficiency at low bitrates.

2.3 PERFORMANCE METRICS

Performance is a quality measure used to evaluate how well a system functions. A performance metric is a quantifiable measure of a process to be assessed. Performance metrics are used to evaluate and encourage performance improvement and efficiency. Quality performance metrics allow for comparative analysis of varying factors within the system.

Image quality is a key performance metric when dealing with image and video compression. Image quality metrics provide a measure of the correlation between digital images by exploiting the differences in the statistical distribution of pixel values in the images. Image compression is a measure of the amount of insignificant data that has been discarded from the digital image. However, there exists a trade off between image compression and image quality. As the compression increases, the image quality consequently degrades.

There are two quantitative metrics which are commonly used to evaluate the image quality of compressed images, Mean Square Error (MSE) [91] and Peak Signal to Noise Ratio (PSNR) [91]. MSE measures the image difference between the decompressed image and the original

image, whereas PSNR is the ratio of the peak signal power (255^2) against the average noise power (MSE). Essentially PSNR reflects the quality of the reconstructed image and is a standard method used to gauge image fidelity. PSNR is measured in decibels (dB). Their equations are as follows [91];

$$MSE = \frac{1}{MN} \sum_{y=1}^{M} \sum_{x=1}^{N} [I(x,y) - I'(x,y)]^2 \text{, and} \tag{2.2}$$

$$PSNR = 20\,log_{10} \frac{255}{\sqrt{MSE}} \text{,} \tag{2.3}$$

where $I(x,y)$ in (2.2) represents the original image whilst $I'(x,y)$ represents the compressed image. M and N represent the dimensions of the image and the number *255* in (2.3) is representative of the maximum pixel value within the image which is generally 255 for grayscale images.

In addition to the qualitative metrics above, there are two metrics that are used to quantitatively assess the compression of an image, which are compression ratio and bit rate. Compression ratio can be defined as a simple ratio of the number of bits of the original uncompressed image to its compressed reconstructed version. Bit rate is a defined as the average number of bits per pixel for the entire image. Bit rate is extensively used in the dissertation as a measure of compression, where high bit rates represent low image compression and low bit rates represent high image compression. The equations are as follows;

$$Compression\ Ratio = \frac{Number\ of\ bits\ in\ Original\ Image}{Number\ of\ bits\ in\ Compressed\ Image} \text{, and} \tag{2.4}$$

$$Bit\ Rate = \frac{Number\ of\ bits\ in\ Compressed\ Image}{Number\ of\ Pixels} \text{.} \tag{2.5}$$

Two error analysis metrics used to evaluate the transmission of the digital images via error prone channels are, Bit Error Rate (BER) and Signal-to-Noise Ratio (SNR). Bit error rate can be broadly referred to as the measure of data integrity. It is measured empirically as the ratio of the number of erroneous bits received compared to the number of bits transmitted for some duration of time and is given by (2.6);

$$BER = \frac{Number\ of\ bits\ in\ Error}{Number\ of\ bits\ Transmitted}\ . \tag{2.6}$$

Signal-to-noise ratio is a measure of the signal strength relative to the background noise in the channel. It is used to gauge the quality of the transmission channel. Average SNR per bit in terms of digital communication is formally represented by E_b/N_0, which is the ratio of energy per bit (E_b) to spectral noise density (N_0) given by (2.7);

$$average\ SNR\ per\ bit = \frac{E_b}{N_0}\ . \tag{2.6}$$

Since E_b/N_0 is independent of modulation schemes, it is used in the plots against bit error rate and helps compare schemes. E_b/N_0 is typically expressed logarithmically in decibels (*dB*). An E_b/N_0 of zero dB cannot be represented on the logarithmic scale and is thus an indication that the signal is unreadable and impossible to interpret as the noise level severely competes with the signal. The dissertation refers to SNR in the results analysis and must be construed as average SNR per bit.

2.4   PERFORMANCE

2.4.1   Still Image Compression Standards

The performance evaluation of the current still image compression standards, involve the assessment of JPEG and JPEG 2000. The compressed images were generated using the VcDemo Image and Video Compression Learning tool and the PSNR values were simulated using the Matlab simulation engine. The Lena (512x512) [85] and Barbara (512x512) [85] grayscale test images, take from the University of Southern California's Signal and Image Processing Institute (USC-SIPI) database were used to evaluate the JPEG and JPEG 2000 standards. The PSNR values were calculated using the MSE between the JPEG/JPEG 2000 compressed image and the original uncompressed image before finally calculating the PSNR image quality. Figure 2-8 and Figure 2-9 show the improved PSNR image quality for a given bitrate for JPEG 2000 and JPEG.
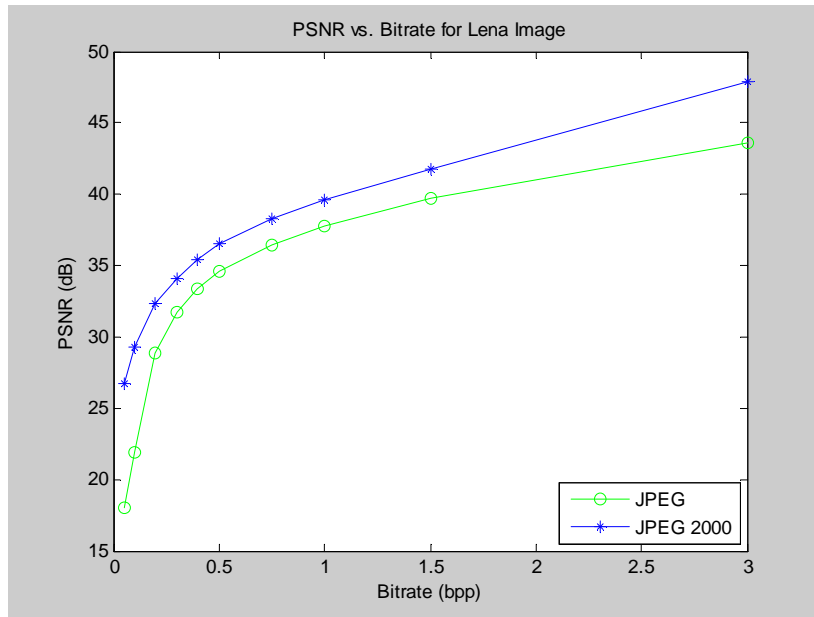
Figure 2-8: Diagram of PSNR vs. Bitrate for JPEG and JPEG 2000 for *Lena* image.
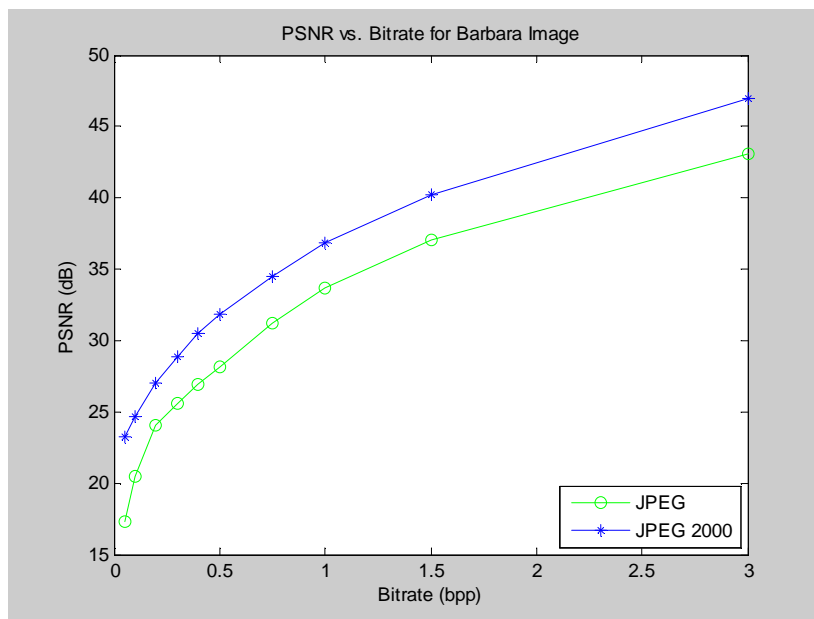
Figure 2-9: Diagram of PSNR vs. Bitrate for JPEG and JPEG 2000 for *Barbara* image.

There is an approximate increase of around *3.65dB* on average in compression improvement for JPEG 2000 over JPEG as the PSNR increases in both Figure 2-8 and Figure 2-9. This increase is calculated by taking the absolute difference or absolute change between each data value of two trends for all bitrates and then calculating the arithmetic mean value of the difference. The results are graphed in dB's using the logarithmic scale and thus only an absolute dB value is needed when stating the amount of change observed between the two trends.

The *3.6dB* increase is indicative of the superior performance of JPEG 2000 against JPEG. These results show that while DCT based coders perform well at moderate compression ratios, at higher compression ratios or low bitrates, the image quality degrades due to artifacts caused by the block-based DCT scheme. DWT based coders provide greater improvement in picture quality at higher compression ratios due to their overlapping basis functions and better energy compaction.

Figure 2-10 illustrates the effect of 'blocking artifacts' for JPEG compression at a low bit rate of *0.3bpp*. This "blockiness" observed in Figure 2-10 is a perceptual measure that results in implementing DCT coding as it is characteristically a block based scheme. The DCT is typically performed on *8x8* blocks where the coefficients in each block are separately quantised. This process consequently leads to artificial horizontal and vertical borders being created throughout the image and it is this feature that is commonly recognised as "blocking artifacts".

Figure 2-10: Diagram of JPEG compressed *Lena* image showing "blocking artifacts".

JPEG 2000 can operate at higher compression ratios without incurring the characteristic "blocking aritifacts" seen in JPEG. The DWT is not block based, thus the transform results in the production of a smoother image. This smoothness can be quantified as a perceptual metric known as "blur". Blur is an artifact of the wavelet based compression technique highlighted in the JPEG 2000 image shown in Figure 2-11. This image was compressed at a bit rate *0.3bpp*.

Figure 2-11: Diagram of JPEG 2000 compressed *Lena* image showing "blur artifacts".

2.4.2 Video compression Standards

Figure 2-12 is the performance results of the current video compression standards available. This PSNR result was obtained from the ITU-T standards documentation for H.264/MPEG AVC [11]. It is evaluated using the *Tempete* test sequence using CIF (352x288) resolution at *15 Hz*. The *Tempete* test sequence has a duration of *8.6s*, and exhibits features that include camera zoom, spatial detail and fast random motion.
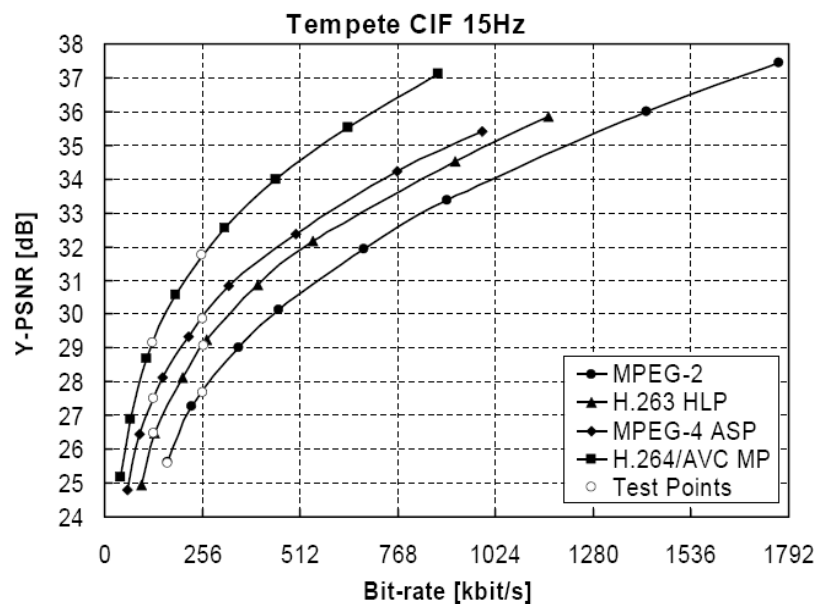
Figure 2-12: PSNR vs. Bitrate for H.263, MPEG2, MPEG4 and H.264/AVC [11].

The four codec's in Figure 2-12 maintain the following profiles.

- MPEG2 Visual, Main profile (VMP)

- H.263 High Latency profile (HLP)
- MPEG4 Visual, Advanced Simple profile (ASP)
- H.264/MPEG4 AVC Main profile (MP)

From the results MPEG2 performs the worst, followed by H.263, MPEG4 and finally with H.264/MPEG4 AVC achieving superior PSNR performance. The results show that at low bit rates H.264/MPEG4 significantly outperforms the other video standards. Table 2-1 denotes the average bitrate savings relative to other standards. The bitrate savings in terms of compression efficiency shows H.264/MPEG AVC achieving a greater percentage than the MPEG4 and H.263 standards. H.264/MPEG4's highly flexible motion prediction and compensation model and its efficient context based entropy coding scheme are the two principal factors that facilitate this superior rate distortion performance.

Coder	Average bit-rate savings relative to:		
	MPEG-4 ASP	H.263 HLP	MPEG-2
H.264/AVC MP	37.44%	47.58%	63.57%
MPEG-4 ASP	-	16.65%	42.95%
H.263 HLP	-	-	30.61%

Table 2-1: Average bitrate savings for H.263, MPEG2, MPEG4 and H.264/MPEG4 AVC [11].

2.5 SUMMARY

This chapter highlights the image and video compression standards currently operating within digital multimedia communication systems. In still image compression, the JPEG 2000 standard exhibits significant image quality, compression and improved rate distortion over the JPEG standard. It was observed that wavelet-based compression when compared with DCT-based compression provided substantial improvements in picture quality at lower bitrates due to the high energy compaction of its wavelet transform.

Four standards were examined within the video compression field, the MPEG, H.263+, MPEG4 and H.264/MPEG4 AVC standards. The most recent standard, H.264/MPEG4 AVC demonstrated superior PSNR and bitrate saving performance as compared to other standards. H.264/MPEG4's numerous improved features make it one of the most widely used particularly efficient standards to date.

CHAPTER 3 - WAVELET COMPRESSION

Wavelet compression has emerged as a powerful compression method that provides significant improvements in image quality and compression ratios. Wavelet based coding systems outperform other coding schemes for example; those based on the Discrete Cosine Transform (DCT) [1], [13], [14].

DCT based coding schemes appear to perform well at moderate bit rates, whereas at low bit rates the image quality tends to degrade rapidly due to the underlying block-based approach employed. Wavelet based coding schemes have achieved far superior image integrity and quality at lower bit rates due to the exploitation of the spatial and spectral redundancies contained within the images and video. Wavelet based coding is more robust with regards to transmission and decoding of errors thereby facilitating the progressive transmission of images and video. Thus image and video compression stand to benefit significantly with the use of wavelet based coding.

3.1 WAVELET THEORY

3.1.1 Fourier Transform

Signals are often represented in the time-domain with a time-amplitude variation. However, this representation fails to reveal the frequency content of the signal. In order to overcome this problem the Fourier Transform was developed. The following equations denote the Fourier Transform and the Inverse Fourier Transform respectively [12], [13];

$$X(f) = \int_{-\infty}^{\infty} x(t)e^{-j2\pi ft}dt \text{ , and} \qquad (3.1)$$

$$x(t) = \int_{-\infty}^{\infty} X(f)e^{j2\pi ft}df \text{ ,} \qquad (3.2)$$

where f is the frequency component and t is the time component. The information provided by the integral in (3.1) corresponds to all time instances as the integration limits extend from minus infinity to plus infinity over time. It follows that the Fourier Transform supplies the frequency information of the signal however, it does not describe when in time this frequency component

exists. This is one of the major shortcomings of the Fourier Transform that subsequently led to the development of the Wavelet Transform.

3.1.2 Wavelet Transform

Wavelets are functions defined over a finite interval and having an average value of zero. Equation (3.3) represents the Wavelet Transform and shows the function $f(x)$ as a superposition of a set of basis functions $\Psi_{s,\tau}(x)$, or wavelets. The parameters s and τ are the scale and translation respectively and denote the dimensions of the wavelet [19];

$$\gamma(s,\tau) = \int f(x)\Psi_{s,\tau}^*(x)dx \ . \tag{3.3}$$

These basis functions or wavelets are generated from a single basic wavelet or the mother wavelet, $\Psi(x)$ by scaling and translation illustrated in (3.4) [19];

$$\Psi_{s,\tau}(x) = \frac{1}{\sqrt{s}}\Psi\left(\frac{x-\tau}{s}\right) \ . \tag{3.4}$$

The concept of scaling and translation of the mother wavelet is graphically illustrated in Figure 3-1. The translation parameter τ denotes the location of the wavelet as it is shifted and thus corresponds to the time information in the Wavelet Transform. The scale parameter s is defined as |1/frequency| and corresponds to the frequency information. Scaling refers to the dilation or contraction of the wavelet and translation refers to the shifting of the wavelet. It must be noted that s cannot equal zero, as division by zero will take place, and s equalling zero implies that no wavelet exists.
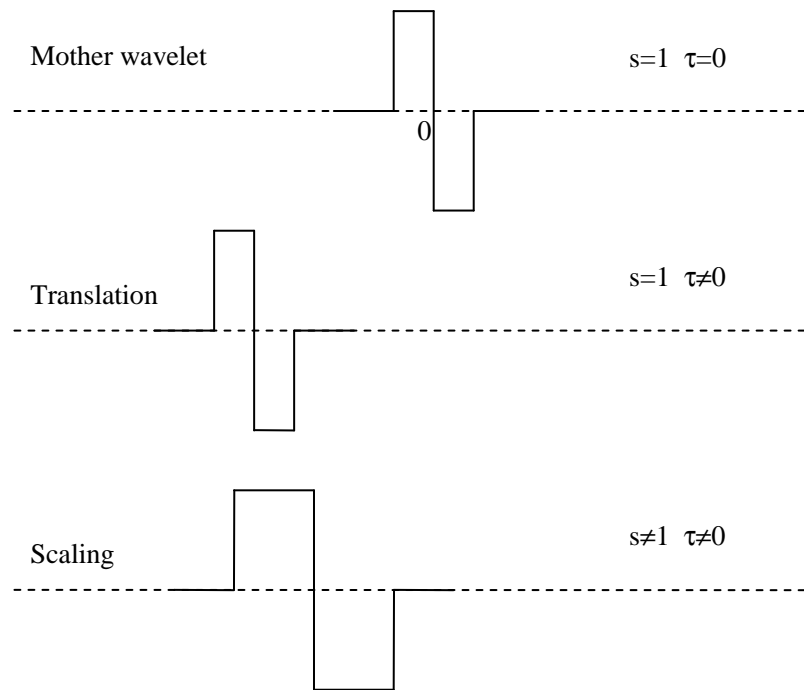
Figure 3-1: Diagram illustrating scaling and translation.

In wavelet analysis, the scale factor is critical as wavelet algorithms are intended to process data at different scales or resolutions. Large scales or low frequencies dilate the signal and provides global information about the signal, whereas small scales or high frequencies compress the signal and provides detailed information that may be hidden in the signal. Figure 3-2 outlines this concept.
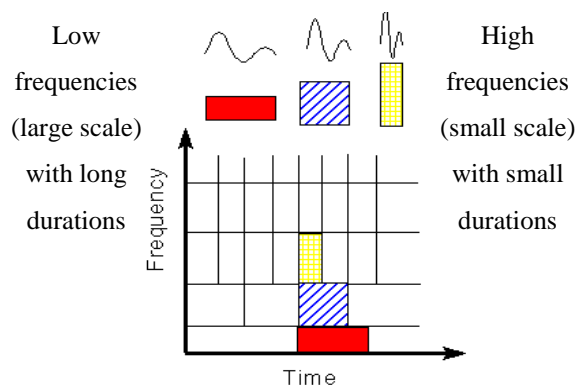
Figure 3-2: Diagram of scale and duration [78].

Wavelet analysis ultimately attempts to combine these two concepts resulting in short bursts or small durations of high frequencies (small scales) or long durations of low frequencies (large scales). This fundamental concept forms the basis of multi-resolution analysis. Figure 3-3 illustrates the merging of the two concepts into a single continuous waveform.
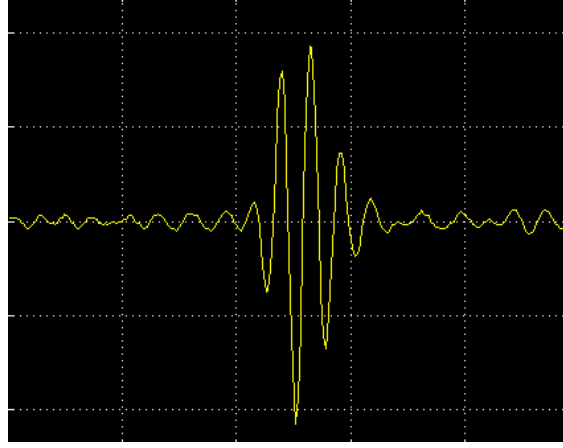
Figure 3-3: Diagram depicting high frequencies with short bursts and low frequencies with long duration [12].

3.1.3 Discrete Wavelet Transform

The Discrete Wavelet Transform (DWT) [1], [13], [19] is an implementation of the wavelet transform using a discrete set of wavelet scales and translations. The Discrete Wavelet Transform exploits the wavelet series expansion which maps a function $f(x)$ into a sequence of coefficients resulting in the DWT transform pair shown in (3.5) [1];

$$W_{\varphi}(\,j_0,k\,) = \frac{1}{\sqrt{M}} \sum_{x}^{M-1} f(\,x\,)\varphi_{j0,k}(\,x\,) \text{ , and}$$

$$W_{\psi}(\,j,k\,) = \frac{1}{\sqrt{M}} \sum_{x}^{M-1} f(\,x\,)\psi_{j,k}(\,x\,) \text{ ,}$$

(3.5)

for $j \geq j_0$. M is the number of samples and j is the resolution or scale and k is the position or integer translation. $W_{\varphi}(j_0,k)$ denotes the approximation (scaling) coefficients and $W_{\psi}(j,k)$ denotes the detail (wavelet) coefficients due to $\varphi_{j0,k}(x)$ representing the scaling function and $\psi_{j,k}(x)$ representing the wavelet function defined in (3.6) [1];

$$\varphi_{j_0,k}(\,x\,) = 2^{j_0/2}\varphi(\,2^{j_0}\,x - k\,) \text{ , and}$$

$$\psi_{j,k}(\,x\,) = 2^{j/2}\psi(\,2^{j}\,x - k\,) \text{ ,}$$

(3.6)

where k denotes the position or integer translation, 2^j denotes the width or the scale of the functions and $2^{j/2}$ controls the amplitude of the functions. These equations are representative of a one dimensional DWT. A two dimensional DWT has the following equations represented by (3.7) [1];

$$W_\varphi(\,j_0,m,n\,) = \frac{1}{\sqrt{MN}} \sum_{x}^{M-1} \sum_{y}^{N-1} f(\,x,y\,) \varphi_{j0,m,n}(\,x,y\,) \ , and$$

$$W^i{}_\psi(\,j,m,n\,) = \frac{1}{\sqrt{MN}} \sum_{x}^{M-1} \sum_{y}^{N-1} f(\,x,y\,) \psi^i{}_{j,m,n}(\,x,y\,) \ ,$$

(3.7)

for $j \geq j_0$ and $i=\{H, V, D\}$ where the wavelet functions are [1];

$$\varphi_{j0,m,n}(\,x,y\,) = 2^{j0/2} \varphi(\,2^{j0} x - m, 2^{j0} y - n\,) \ , and$$

$$\psi^i{}_{j,m,n}(\,x,y\,) = 2^{j/2} \psi^i(\,2^j x - m, 2^j y - n\,) \ ,$$

(3.8)

for $i=\{H, V, D\}$. The two dimensional function $f(x,y)$ can be extended to represent an image where M and N are the dimensions of the image and m and n are the number of pixel rows and columns in the image. The index i identifies the direction of the wavelet either horizontal H, vertical V or diagonal D. $W_\varphi(j_0,m,n)$ denotes the approximation coefficients at resolution j_0 and $W^i{}_\psi(j,k)$ denotes the horizontal, vertical and diagonal coefficients at resolution $j \geq j_0$. $\varphi_{j0,m,n}(x,y)$ represents the scaling function and $\psi^i_{j,m,n}(x,y)$ represents the wavelet function in two dimensional form.

3.1.4 Filter Banks

Filter banks [17], [18], [20] are commonly used to transform an input signal into a time-frequency domain representation. A filter bank uses a number of bandpass filters to isolate different frequency components in a signal as some frequencies are of greater importance than others. This is illustrated in Figure 3-4.
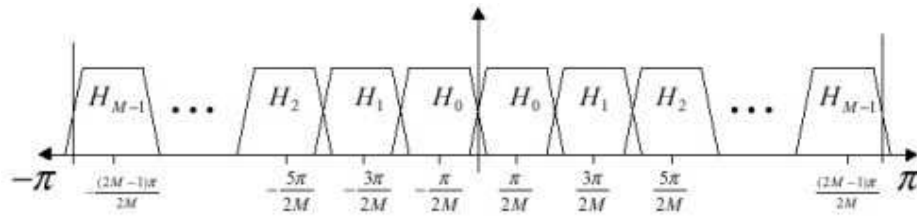
Figure 3-4: Filter Banks [17].

There are two stages involved in the filter bank representation, the analysis stage and the synthesis stage. The analysis stage filters the input signal before downsampling it by the number of filters used in the bank producing subband signals. The synthesis stage involves reconstructing the signal by first upsampling the subband signals, then filtering each signal before adding the signals together. An M-channel filter bank is depicted in Figure 3-5.
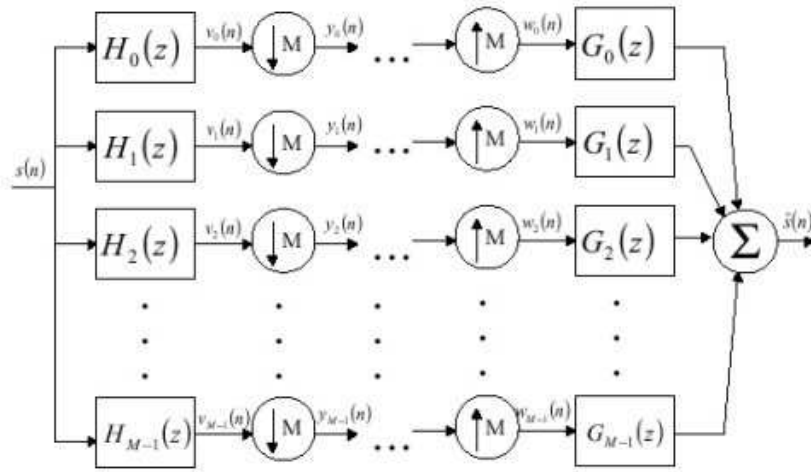
Figure 3-5: M-channel filter bank with analysis and synthesis stages [17].

3.1.4.1 Quadrature Mirror Filters

A quadrature mirror filter (QMF) [18], [20] is a two-channel filter bank. The analysis and synthesis structure of a QMF is shown below where the signal is downsampled and upsampled by a factor of 2.
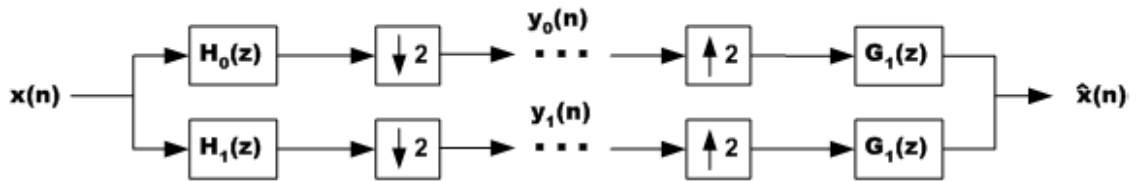
Figure 3-6: QMF analysis and synthesis stages [18].

The frequency response of the analysis lowpass and highpass filters represented by $H_0(z)$ and $H_1(z)$ respectively is illustrated in Figure 3-7.
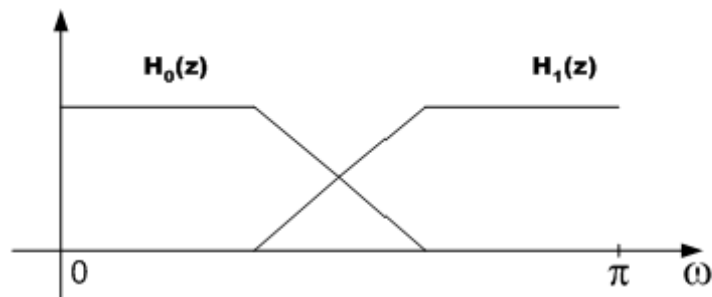
Figure 3-7: Frequency response of analysis lowpass filter $H_0(z)$ and highpass filter $H_1(z)$ [18].

QMF's relates the lowpass filter to the highpass filter by altering the sign whose frequency response is a "mirror" image of itself [18];

$$H_1(z) = H_0(-z) \text{ or } h_1(n) = (-1)^n h_0(n) \,. \tag{3.9}$$

The two-channel QMF is closely related to the wavelet filter bank as it utilises discrete filters. This concept is investigated further in the chapter.

3.1.5 Subband Coding

Subband coding [1], [14] involves the signal being decomposed into a set of band-limited components called subbands. Each subband is generated by bandpass filtering the input signal. Essentially the signal is passed through a series of high pass filters to analyse the high frequencies and it is passed through a series of low pass filters to analyse the low frequencies. Subband coding also involves upsampling and downsampling operations. Since the bandwidth of the resulting subbands is smaller than that of the original signal, the subbands can be downsampled without loss of information. Upsampling is used in the reconstruction of the original image. Ultimately subband coding produces a number of signals which represent the actual signal but correspond to different frequency bands. This concept is illustrated in Figure 3-8.
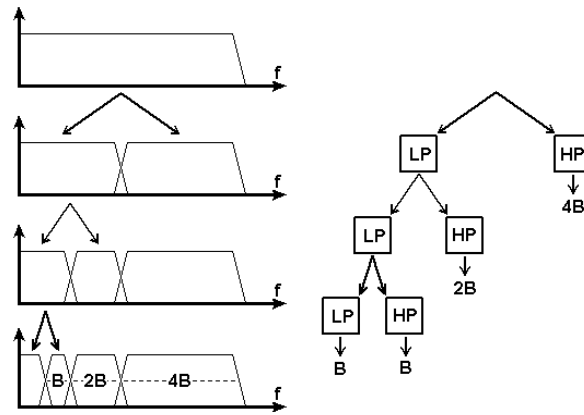
Figure 3-8: Diagram illustrating Subband coding [19].

3.1.6 Multi-resolution Analysis

Multi-resolution analysis (MRA) [1], [18], [21], [22] is a wavelet concept that essentially analyses signals at different frequencies with different resolutions. The fundamental concept behind multi-resolution analysis involves the decomposition of a signal in terms of scaling and wavelet functions.

The scaling function is used to create a series of approximations of the function. In order to choose an appropriate scaling function, fundamental requirements of multi-resolution analysis must be obeyed.

Multi-resolution analysis is defined as a nested sequence of closed subspaces V_j of $L^2(\mathbb{R})$, $j \in Z$, where $L^2(\mathbb{R})$ is the normed finite vector space, j is the resolution level, Z is the set of integers and V_j is a sequence of subspaces with the following four properties [21]:

(1) Nested subspace: The subspaces spanned by the scaling function at low scales are nested within those spanned at higher scales i.e. the subspace V_j be contained in all higher sub spaces:

$$V_j \subseteq V_{j+1} \quad j \in Z \ , \tag{3.10}$$

$$V_{-\infty} \subset \cdots \subset V_{-1} \subset V_0 \subset V_1 \subset \cdots \subset V_\infty \ . \tag{3.11}$$
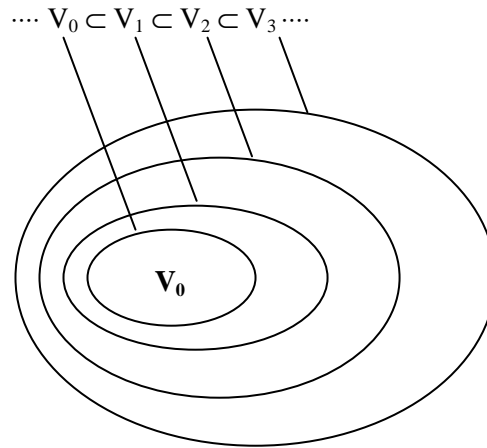
Figure 3-9: Nested subspace.

(2) Scale invariance: The scaling function is orthogonal to its integer translates i.e. all the subspaces V_j are scaled versions of the central space V_0:

$$f(x) \in V_j \ \ \text{if and only if} \ \ f(2x) \in V_{j+1} \ . \tag{3.12}$$

(3) Separation and density: All square integral functions are included at the finest resolution (density) and included in the zero function at the coarsest level (separation):

$$\cap_j V_j = \{ 0 \} \ \text{and} \ \overline{\cup_j V_j} = L^2(\mathbb{R}) \ . \tag{3.13}$$

(4) Shift invariance: There is a function $\varphi(x)$, called the scaling function, such that it translates $\{\varphi(x\text{-}n)\}$, (where n is the integer translation) which forms an orthonormal basis of V_0. Similarly;

$$\varphi_{jn}(x) = 2^{j/2} \varphi(2^j x - n) \ ,$$

<div align="right">(3.14)</div>

forms an orthonormal basis for V_j.

From the above definition and properties, $\varphi \in V_0 \subset V_1$ and there exists constants $h(n)$ such that

$$\sum_n |h(n)|^2 = 1 \ ,$$

<div align="right">(3.15)</div>

which results in the scaling function also known as the multiresolution equation given by:

$$\varphi(x) = \sum_n h(n)\sqrt{2}\varphi(2x - n) \ .$$

<div align="right">(3.16)</div>

Once the scaling function is found, the wavelet function can then be defined. The following diagrams represent the scaling function φ and corresponding wavelet function ψ of the Haar [25], Daubechies 4 and Daubechies 20 wavelets.

$$\psi(x) = \begin{cases} -1 & 1/2 \le x < 1 \\ 1 & 0 \le x < 1/2 \\ 0 & otherwise \end{cases}$$

(3.17)
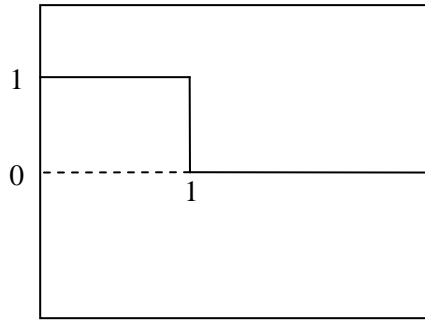
$$\psi(x) = \begin{cases} -1 & 1/2 \le x < 1 \\ 1 & 0 \le x < 1/2 \\ 0 & otherwise \end{cases}$$

Haar Scaling Function

Haar Wavelet Function
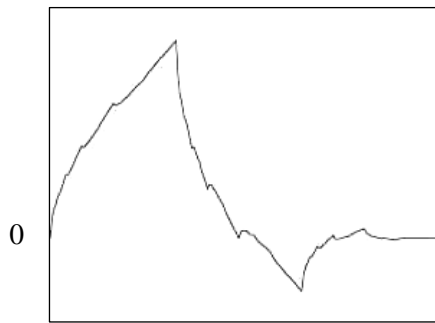
Daubechies 4 Scaling Function
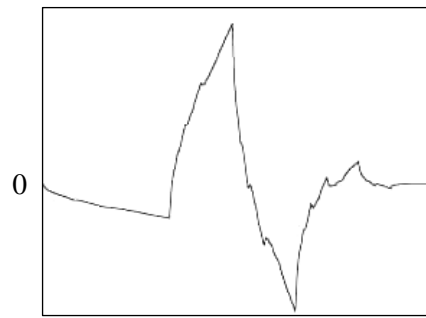
Daubechies 4 Wavelet Function
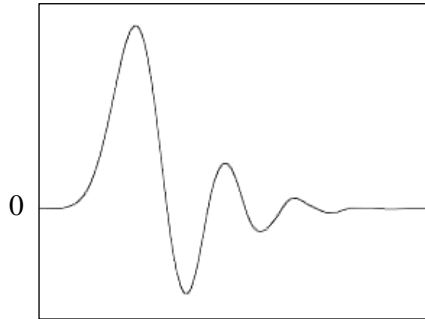
Daubechies 20 Scaling Function
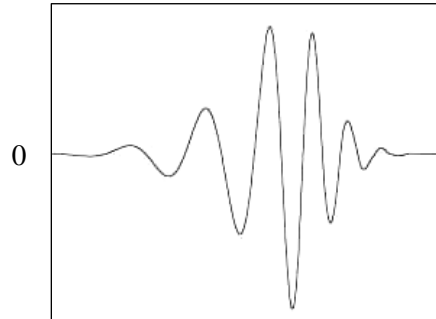
Daubechies 20 Wavelet Function

Figure 3-10: Diagram of Haar, Daubechies 4 and Daubechies 20 scaling and wavelet functions [23], [24].

3.1.7   Fast Wavelet Transform

The Fast Wavelet Transform (FWT) [1], [15], [22] is a computationally efficient implementation of the Discrete Wavelet Transform (DWT) that exploits multiresolution analysis and Mallat's *herringbone algorithm* [22] resulting in the following scaling and wavelet multi-resolution refinement equations [1];

57

$$\varphi(x) = \sum_n h_\varphi(n) \sqrt{2} \varphi(2x - n) \text{ , and}$$

$$\psi(x) = \sum_n h_\psi(n) \sqrt{2} \varphi(2x - n) \text{ .}$$

(3.18)

The Mallat algorithm essentially associates the discrete wavelet transform in (3.18) to discrete time filters. The term $h_\varphi(n)$ is referred to as the scaling function coefficients or the scaling filter and $h_\psi(n)$ is referred to as the wavelet function coefficients or the wavelet filter. These two filters are not independent of each other and can be related by the following equation where L is the filter length or total number of samples [1];

$$h_\psi(n) = (-1)^n h_\varphi(L - 1 - n) \text{ .}$$

(3.19)

These two filters are consequently known as quadrature mirror filters (QMF's). By definition a quadrature mirror filter associates a lowpass filter bank to a highpass filter bank. This concept is illustrated in Figure 3-11 where $h_\varphi(n)$ and $h_\psi(n)$ are half band filters whose idealized transform functions are H_φ and H_ψ.
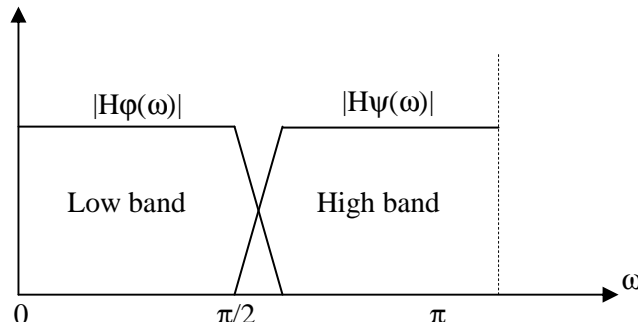
Figure 3-11: Diagram of the Quadrature Mirror Filters showing lowpass and highpass spectra [1].

As a result the scaling filter is associated with a lowpass filter and the wavelet filter is associated with a highpass filter. The highpass filter produces detail information of a signal, while the lowpass filter associated with the scaling function produces coarse approximations.

To conclude the development of the FWT, the scaling and wavelet filters are time reversed becoming $h_\varphi(-n)$ and $h_\psi(-n)$ when applied to the discrete wavelet transform pair denoted in (3.18). The final result is shown in (3.20), where the approximation (scaling function) and detail (wavelet function) coefficients $W_\varphi(j,k)$ and $W_\psi(j,k)$ at scale j is computed by convolving the time-reversed scaling and wavelet filters with the approximation and detail coefficients at scale $j+1$ and finally downsampling the result by 2 [1];

58

$$W_\varphi(\,j,k\,) = h_\varphi(\,-n\,) * W_\varphi(\,j+1,n\,)\Big|_{n=2k,k\geq0} \;\;, \text{ and}$$

$$W_\psi(\,j,k\,) = h_\psi(\,-n\,) * W_\varphi(\,j+1,n\,)\Big|_{n=2k,k\geq0} \;.$$

(3.20)

The filter bank representation and the frequency splitting characteristics of the above FWT equations are illustrated in Figure 3-12 and Figure 3-13:
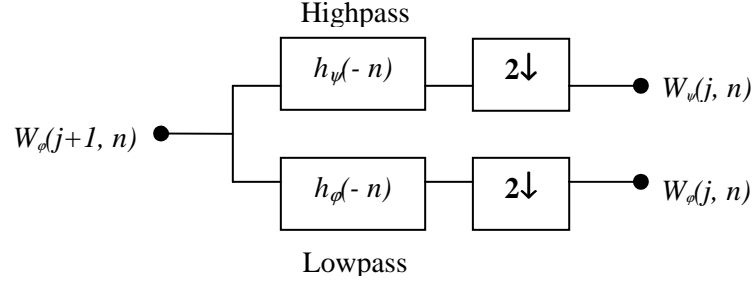
Figure 3-12: Diagram of a One Stage FWT filter bank representation [1].
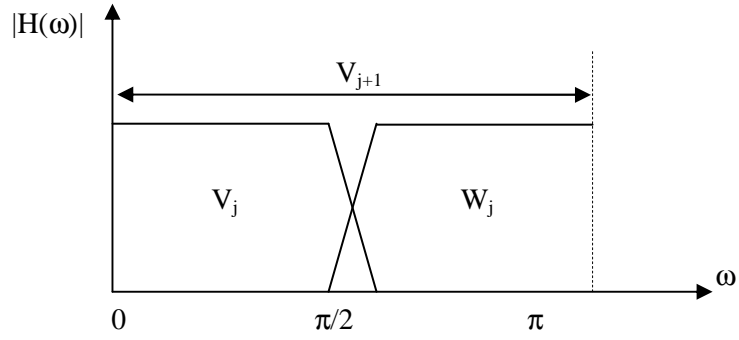
Figure 3-13: Diagram of a One Stage frequency splitting characteristic of the FWT filter bank [1].

An increase in the number of decomposition stages further increases the frequency resolution of the signal. Figure 3-14 and Figure 3-15 are representative of a two stage or two level filter banks. This arrangement of wavelet decomposition levels of a signal resembles a tree structure and is commonly referred to as a wavelet decomposition tree.
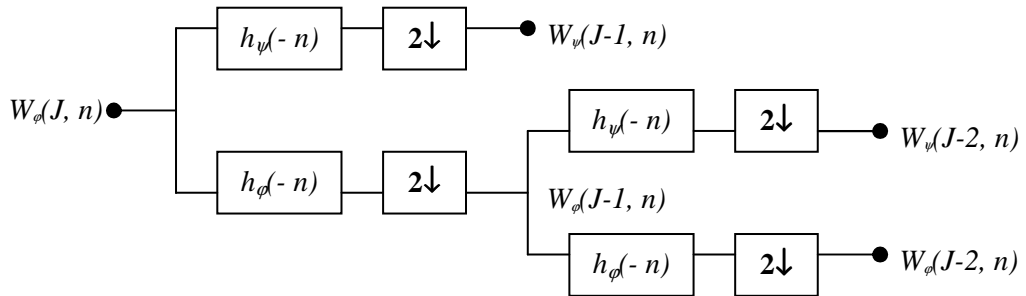
Figure 3-14: Diagram of a Two Stage FWT filter bank representation [1].

59
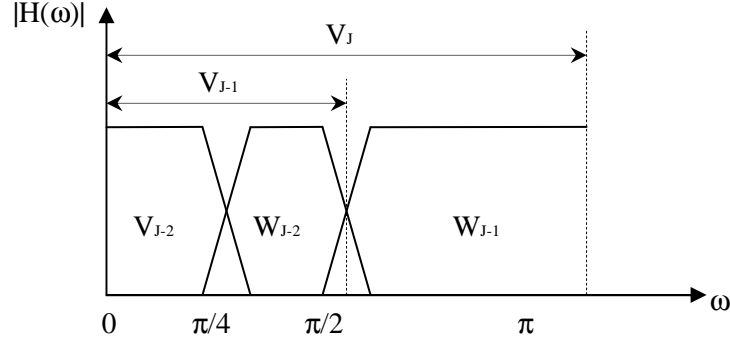
Figure 3-15: Diagram of a Two Stage frequency splitting characteristic of the FWT filter bank [1].

The first stage of the filter bank splits the original function into a lowpass approximation component corresponding to the scaling coefficients $W_\varphi(J\text{-}1,n)$ and a highpass detail component corresponding to the wavelet coefficients $W_\psi(J\text{-}1,n)$. This is graphically illustrated with the scaling space V_J split into the wavelet subspace $W_{J\text{-}1}$ and scaling subspace $V_{J\text{-}1}$. The second stage of the filter bank splits the half-band subspace $V_{J\text{-}1}$ into quarter-band subspaces $W_{J\text{-}2}$ and $V_{J\text{-}2}$ which correspond to the coefficients $W_\psi(J\text{-}2,n)$ and $W_\varphi(J\text{-}2,n)$, respectively.

Computation of the inverse wavelet transform (FWT^{-1}) mirrors its forward counterpart thus instead of downsampling, upsampling is used. Another important observation is that the scaling and wavelet filters $h_\varphi(\text{-}n)$ and $h_\psi(\text{-}n)$ of the FWT must be a time-reversed version of the inverse FWT hence the scaling and wavelet filters are $h_\varphi(n)$ and $h_\psi(n)$. It follows that the FWT^{-1} filter bank equation is [1];

$$W_\varphi(\,j+1,k\,) = h_\varphi(\,k\,) * W_\varphi^{up}(\,j,k\,) + h_\psi(\,k\,) * W_\psi^{up}(\,j,k\,)\big|_{k\geq 0} \;. \qquad (3.21)$$

This concept is graphically depicted in Figure 3-16.
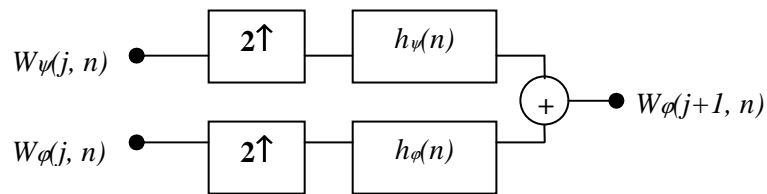
Figure 3-16: Diagram of a One Stage FWT^{-1} filter bank representation [1].

All the above equations and filter bank structures are one-dimensional in nature and can easily be extended to two dimensions thus being able to effectively represent images. As mentioned previously a two-dimensional signal (images) can be described via horizontal, vertical and

diagonal components where the horizontal and vertical components correspond to rows and columns of an image. A two-dimensional filter bank structure has the following arrangement.
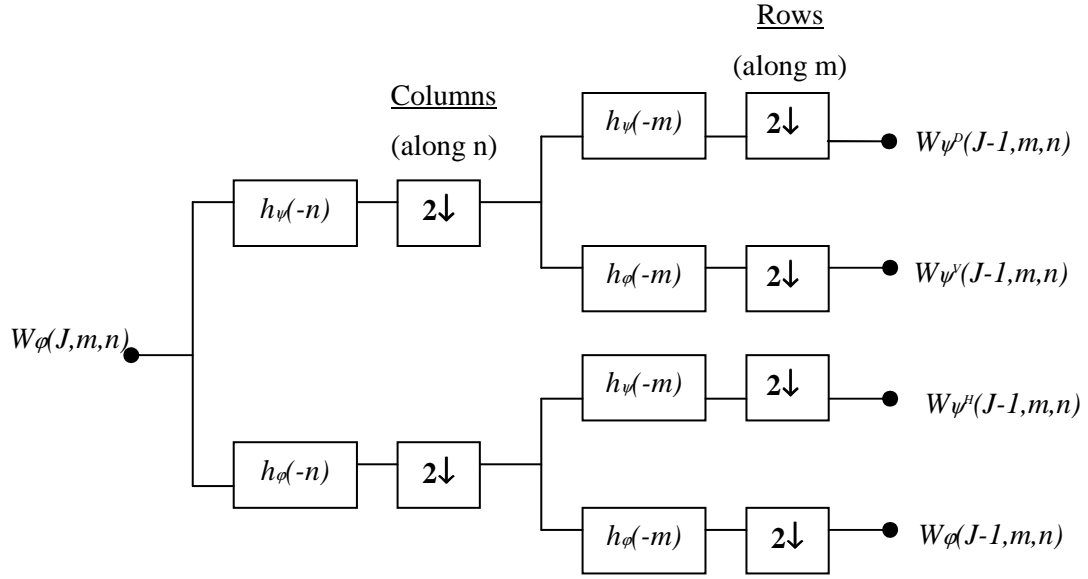
Figure 3-17: Diagram of a Two-Dimensional One Stage FWT filter bank representation [1].

Applying the two-dimensional FWT concept to an image will result in the image being subjected to a one-dimensional FWT, first in the horizontal direction (rows) and then the vertical direction (columns). Essentially the decomposition stages of the two-dimensional fast wavelet transform uses a one-dimensional FWT engine in each iteration. The FWT splits the image into a series of decomposition levels each containing a number of subbands. Each subband contains four bands of data labelled LL, HL, LH and HH which describe the approximations, horizontal details, vertical details and diagonal details of the original image respectively. LL (low-low) corresponds to the low resolution subband, HL (high-low) corresponds to the high vertical and low horizontal resolution subband, LH (low-high) corresponds to the low vertical and high horizontal resolution subband and HH (high-high) corresponds to the high resolution subband. The LL subband at the highest level contains the most amount of image information and thus can be classified as the most important subband. The other detail subbands can be classified as of lesser importance where the degree of importance decreases as the level decreases. The following concept is graphically illustrated in Figure 3-18.

	$W\varphi(J\text{-}1,m,n)$	$W\psi^H(J\text{-}1,m,n)$		**LL₂**	**LH₂**	$W\psi^H(J\text{-}1,m,n)$
(a)	Image $f(m,n)$	**LH₁**		**HL₂**	**HH₂**	**LH₁**
	$W\psi^y(J\text{-}1,m,n)$	$W\psi^P(J\text{-}1,m,n)$		$W\psi^y(J\text{-}1,m,n)$	$W\psi^P(J\text{-}1,m,n)$	
	HL₁	**HH₁**		**HL₁**	**HH₁**	
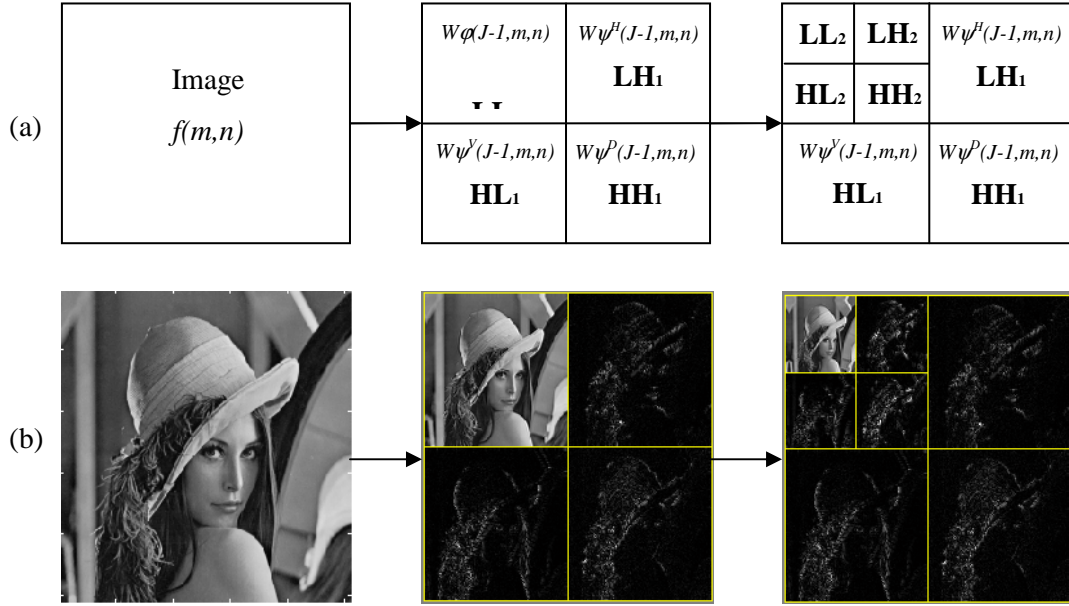
Figure 3-18: Diagram of a Two Stage Wavelet Decomposition showing the subband decompositions and the Lena image decomposition generated in Matlab Wavelet Toolbox [1].

In essence the wavelet transform allows for the de-correlation of the image information by filtering the original image into different frequency components using a filterbank, where it is divided further by means of wavelet decomposition into multiple lowpass and highpass regions representing high detail content and low detail content of the original image. The optimum number of decomposition levels may vary, as the levels improve flexibility, scalability and compression efficiency with various applications.

3.2 WAVELET FAMILIES

A wavelet family can be described as a set of basis functions that is used to accurately represent the signal information. As mentioned previously, basis functions or wavelets are generated from the mother wavelet by scaling and translation. The Fourier Transform has a set of two basis functions, sine and cosine, whereas the wavelet transform has an infinite set of basis functions varying in translation and scaling. Each basis function is associated with a particular wavelet family. The difference between the various wavelet families is exhibited in the smoothness and the compactness of its basis functions. Some of the more commonly used wavelet families are [23], [24], [26], [27]:

- Haar Wavelet
- Daubechies Wavelet
- Coiflet Wavelet

62

- Symlet Wavelet
- Meyer Wavelet
- Morlet Wavelet
- Mexican Hat Wavelet

The various wavelet families may contain a number of wavelet subclasses. These subclasses represent the number of vanishing moments of the wavelet. Vanishing moments constitute the number of coefficients and level of iteration performed on the wavelet. As the vanishing moments increase the wavelet becomes smoother and more regular in nature. Figure 3-19 illustrates the above mentioned wavelet families with some subclasses.
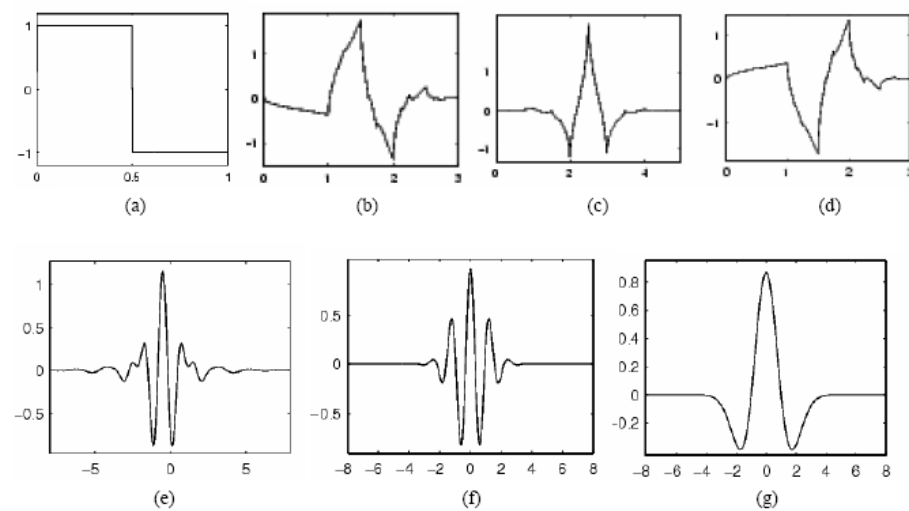
Figure 3-19: Diagram of Wavelet Families (a) Haar (b) Daubechies 4 (c) Coiflet 1 (d) Symlet 2 (e) Meyer (f) Morlet (g) Mexican hat [23], [24].

Given that there are an infinite number of basis functions, the best basis function or wavelet that accurately approximates a given signal representation needs to be selected from the various wavelet families. There are several properties that these basis functions satisfy thus making the selection process simpler. These are [23], [24]:

- Symmetry
- Smoothness
- Orthogonality
- Compact Support

3.2.1 Haar Wavelet

Selection of a wavelet invariably begins with the Haar Wavelet [13], [23], [24], [25] as it is the oldest and simplest wavelet transform. The Haar wavelet is also known as the Daubechies 1 wavelet. The Haar wavelet properties include the following [23], [24]:

- Symmetric scaling function
- Anti-symmetric wavelet function
- One vanishing moment
- Orthogonal
- Compact support

The Haar wavelet provides compact support in that it vanishes outside a finite interval. Haar wavelets are not continuous and therefore not differentiable and thus limited in their application. The Haar wavelet function is illustrated in Figure 3-10.

The Haar wavelet transform generally produces blocking artifacts within the image and is thus not the best wavelet transform to use with regard to smoothing. Majority of Haar's inaccuracy lies in the high frequency content of the image representative of edges and sharp transitions. This demonstrates that the Haar wavelet exhibits best results with low frequency content or areas with uniformity.

3.2.2 Daubechies Wavelet

Daubechies wavelets [13], [23], [24] are the most popular wavelets. Daubechies wavelets satisfy a number of properties. These are [24]:

- Regularity
- Continuity
- Orthogonality
- Compact Support

The property of orthogonality involves the inner products of the Daubechies wavelets equalling zero. The regularity property is satisfied as the Daubechies wavelets can produce linear functions. Daubechies wavelets are continuous but are not differentiable. Daubechies wavelets are known for having a high degree of smoothness. As the vanishing moments increase in the Daubechies wavelet family so does the smoothness of the function. This key property can be seen in the Figure 3-20.
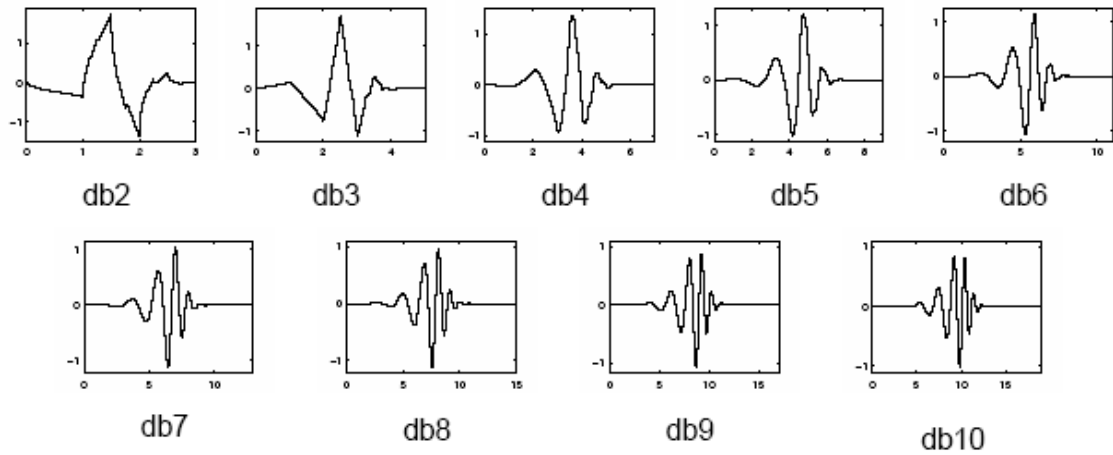
Figure 3-20: Diagram of the Daubechies wavelet family with increasing vanishing moments [23], [24].

The Daubechies wavelet is a much smoother transform than the Haar wavelet. The Daubechies wavelet compression produces a far less lossy image than the Haar wavelet compression scheme. It can also be stated that as the vanishing moments increase, the wavelet compressed image becomes smoother.

3.2.3 Coiflet Wavelet

The Coiflet wavelet transform [13], [23], [24], [26] was built by Ingrid Daubechies at the request of R. Coifman. She designed the Coiflet wavelet to be more symmetric than the Daubechies wavelet as symmetry helps reduce blocking artifacts in the compressed image. It is also another orthogonal, compactly supported wavelet like the Daubechies family of wavelets. Figure 3-21 depicts the nature of the Coiflet wavelet family with increasing vanishing moments. A key property that differentiates the Coiflet wavelet from other wavelet families is that the vanishing moments are equally distributed for both the scaling function and the wavelet.
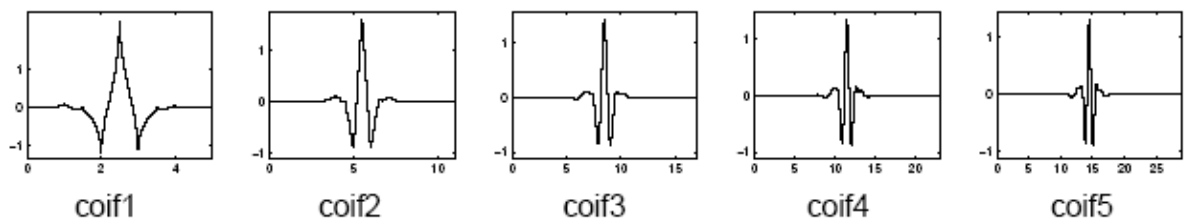
Figure 3-21: Diagram of Coiflet wavelet family with increasing vanishing moments [23], [24].

As the vanishing moments increase, the compressed image becomes less prone to errors. The Coiflet wavelet family has similar compression characteristics as the Daubechies wavelet family.

3.2.4   Symlet Wavelet

Symlet wavelets [1], [13], [23], [24] are maximum symmetry wavelets proposed by Daubechies as a modification to the Daubechies family of wavelets. Symlet wavelets are very similar to the Coiflet wavelets in that they have greater symmetry than the Daubechies wavelets and they are also an orthogonal, compactly supported wavelet family. They were designed to have the least asymmetry and the highest number of vanishing moments for a given compact support. Figure 3-22 shows the Symlet wavelet family with increasing vanishing moments.
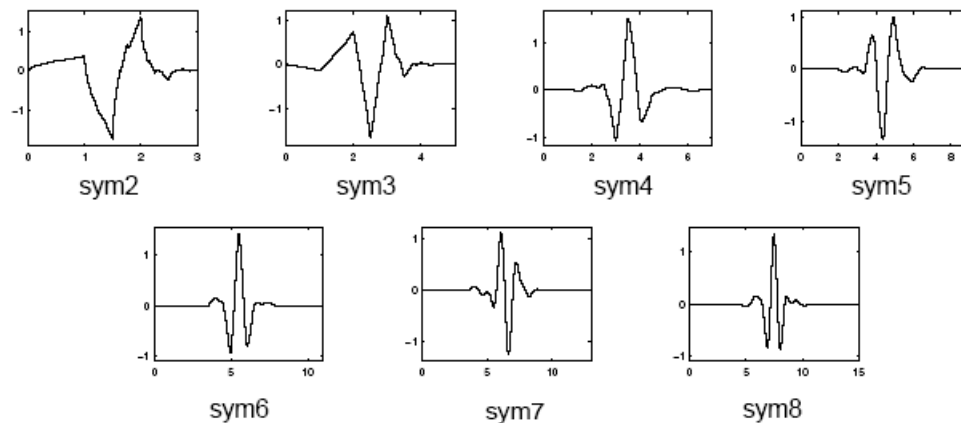
Figure 3-22: Diagram of Symlet wavelet family with increasing vanishing moments [23], [24].

3.2.5   Meyer Wavelet

The Meyer wavelet [13], [23], [24] and scaling function is defined in the frequency domain. Its main characteristic is that it is an infinitely regular orthogonal wavelet. It has no compact support. The Meyer wavelet is symmetric in shape and is capable of perfect reconstruction. Being a continuous wavelet with no compact support, it is therefore infinitely differentiable. Figure 3-23 illustrates the Meyer wavelet.
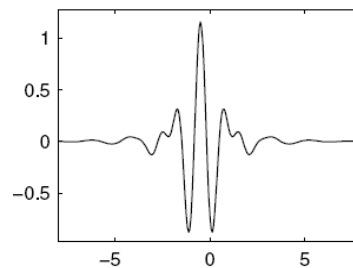
Figure 3-23: Diagram of the Meyer Wavelet [23], [24].

Compression of images only occurs with discrete transforms thus direct compression involving the Meyer wavelet transform cannot take place. There exists a good discrete approximation with FIR (finite impulse response) filters thereby allowing compression.

66

3.2.6 Morlet Wavelet

The Morlet wavelet [13], [23], [24], [27] is the most commonly used continuous wavelet transform. The Morlet wavelet is a locally periodic wave obtained by taking a complex sine wave and localizing it with a Gaussian (bell-shaped) envelope. It is described by the following equation ;

$$\psi(t) = e^{-iw_0 t} e^{-t^2/2} \quad \text{with} \quad w_0 = \pi \sqrt{\frac{2}{ln\,2}} = 5.336 \; . \tag{3.22}$$

Clearly the wavelet is complex and will have a real and imaginary component. Figure 3-24 shows the Morlet wavelet with the solid line representing the real component and the dashed line representing the imaginary component.
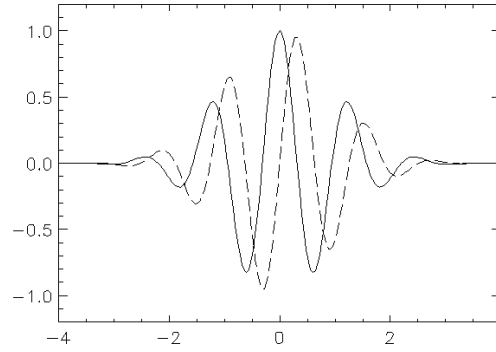
Figure 3-24: Diagram of the Morlet Wavelet [27].

This wavelet has no scaling function and is explicit. It is a symmetric continuous function however, it has no compact support or orthogonal properties. Its complex nature makes it sensitive to frequencies leading to an adequate time-frequency analysis.

3.2.7 Mexican Hat Wavelet

The Mexican Hat [1], [13], [23], [24] wavelet gets its name from its distinctive shape which can be seen in Figure 3-25. It is derived from a function that is proportional to the second derivative of the Gaussian probability density function and is represented by the following equation [1];

$$\psi(x) = \left(\frac{2}{\sqrt{3}} \pi^{-1/4} \right) \left(1 - x^2 \right) e^{-x^2/2} \; . \tag{3.23}$$

67

Its most distinguishing feature is its symmetry. Being a continuous function it has no compact support. The Mexican Hat wavelet has no scaling function and the analysis is not orthogonal. The Mexican Hat wavelet is an admissible wavelet, which implies that the area under the function equals zero. This admissibility condition is mathematically defined in (3.24);

$$\int_{-\infty}^{\infty} \psi(\,x\,)dx = 0 \ \ . \tag{3.24}$$
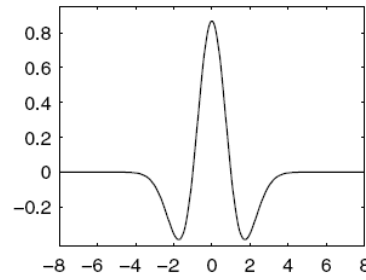
Figure 3-25: Diagram of the Mexican Hat Wavelet [23], [24].

3.2.8 Wavelet Family Properties

A fundamental issue in successful wavelet compression is the choice of the wavelet basis and hence the choice of wavelet family to be used, in order to accurately represent the signal information. Wavelet families have the ability to efficiently represent functions with localized features thus a basis description that exhibits efficiency in the form of minimal expansion terms is able to effectively compress the signal. Therefore selection of a wavelet family function which closely matches the signal to be processed is of utmost importance in wavelet applications such as: compression, signal detection, denoising and interference excision. A summary of the various wavelet family properties contributing to the selection of the correct wavelet basis can be viewed in Table 3-1.

PROPERTY	Haar	Daubechies	Coiflet	Symlet	Meyer	Morlet	Mexican Hat
Infinitely Regular					●	●	●
Arbitrary Regular		●	●	●			
Compactly Supported	●	●	●	●			
Symmetrical	●					●	●	●
Asymmetrical		●					
Near Symmetrical			●	●			
Existence of Scaling Function	●	●	●	●	●		
Orthogonal	●	●	●	●	●		
Continuous Wavelet					●	●	●
Discrete Wavelet	●	●	●	●	●		
Explicit expression	●					●	●

Table 3-1: Summary of Wavelet family properties [23], [24].

Wavelet families exhibiting discrete wavelet properties were tested and compared using a set of six grayscale images (215x215) [85]; Lena, Barbara, Cameraman, Baboon, Goldhill and Peppers, each at a bit rate of *0.75bpp*. The PSNR in dB's for the various wavelet families are displayed in Table 3-2. The results were obtained using the Matlab simulation engine using six wavelet decomposition levels and the SPIHT wavelet compression coding scheme. This was achieved in order to determine the wavelet family that performs best in terms of PSNR or image quality. In Table 3-2, the Coiflet 5 wavelet typically exhibits the best performance with regards to PSNR quality across the sample image set. The PSNR qualities of majority of the wavelet families are similar therefore selection of particular wavelet basis for a specified coder is based on the wavelet basis exhibiting the highest overall PSNR, which is the Coiflet wavelet family.

	LENA	BARBARA	CAMERAMAN	BABOON	GOLDHILL	PEPPERS
HAAR	30.5356	28.1284	31.4245	26.0736	29.2732	30.5800
DB 2	31.3139	28.7962	32.1269	26.2975	29.5812	31.4285
DB 4	31.6725	29.4289	32.4340	26.4586	29.7242	31.5251
DB 8	28.2765	29.5819	28.6871	26.4469	29.6070	31.3560
DB 10	28.2490	29.6464	32.6219	26.4418	29.6222	27.9835

COIF 1	31.3909	28.8330	32.2734	14.1617	29.6038	31.4322
COIF 5	**31.9057**	29.8693	**32.8994**	**26.5274**	29.7758	31.5617
SYM 2	31.3139	28.7962	32.1269	26.2975	29.5812	31.4285
SYM 8	31.8254	29.7269	28.6536	26.5017	**29.7770**	**31.6265**
MEYER	31.8299	**30.0007**	32.6770	26.5244	29.7204	31.3442

Table 3-2: Performance (PSNR in bpp) comparison between discrete wavelet families.

3.3 WAVELET IMAGE CODING

Wavelet based image compression has had great success as of recent due to the fact that its wavelet coding schemes combine excellent compression efficiency with the possibility of an embedded representation. In light of this, a few important coding schemes have emerged, they are:

- Embedded Zerotree Wavelet (EZW) Encoding – by Shapiro [28]
- Set Partitioning in Hierarchical Trees (SPIHT) – by Said and Pearlman [32]
- Space Frequency Quantization (SFQ) – by Xiong, Ramchandran and Orchard [34]
- Stack-Run Image Coding (SR) – by Tsai, Villasenor and Chen [38]
- Embedded Conditional Entropy Coding of Wavelet Coefficients (ECECOW) – by Wu [40]

3.3.1 Embedded Zerotree Wavelet

Embedded Zerotree Wavelet encoding was originally proposed by J. Shapiro [28]. From its distinctive name EZW employs three key concepts; embedded coding, zerotree structure and wavelet transform.

This algorithm was specifically designed to be used in conjunction with wavelet transforms, hence the word 'wavelet' in EZW. Embedded coding is also known as progressive coding and is used to compress an image into a bit stream with increasing accuracy. In other words as more bits are added to the bit stream, the decoded image will contain more detail and thus the accuracy of the encoding will increase. The zerotree structure is based on subband decomposition forming a tree-like hierarchical nature.

This subband decomposition uses the DWT to decompose the image into four different subbands. A DWT coefficient in a lower subband can have four descendants in the next higher

subband. Each of those four descendants then has a further four descendants in the next higher subband and so on. Thus a quad-tree structure emerges from this subband decomposition. Finally the zerotree concept can be formally defined as a quad-tree of which all nodes are equal to or smaller than the root. This definition is illustrated in Figure 3-26.
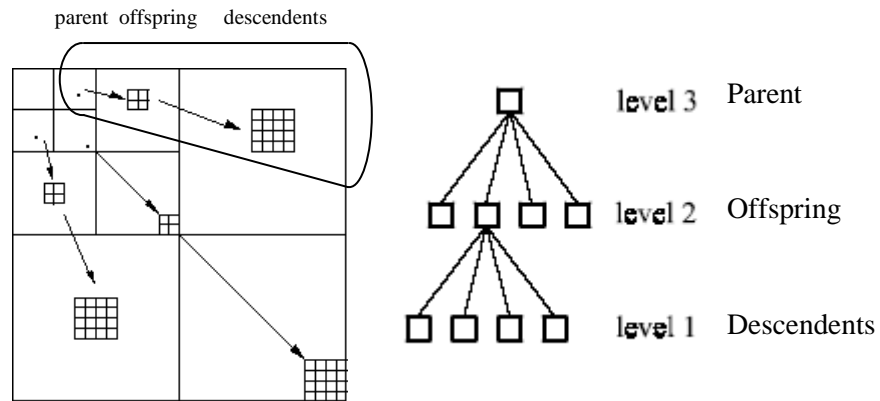
Figure 3-26: Diagram of the Zerotree structure [31].

The zerotree concept [28], [29] is based on the hypothesis that if a wavelet coefficient at a coarse scale (parent) is insignificant with respect to a given threshold T, then all wavelet coefficients of the same orientation in the same spatial location at fine scales (children) are likely to be insignificant with respect to T [28]. Essentially this hypothesis implies that the whole tree need not be encoded and encoding only the root of the tree would provide a fair amount of compression in itself. This parent-child relationship gives rise to the zerotree structure depicted in Figure 3-26.

Embedded Zerotree Wavelet coding is based on two observations of the wavelet transform [28]:

(1) Natural images in general have a low pass spectrum. When an image is wavelet transformed, the energy in the subbands decrease as the scale decreases (low scale means high resolution), so the wavelet coefficients will on average, be smaller in the higher subbands than in the lower subbands. This shows that progressive encoding is a very natural choice for compressing wavelet transformed images, since the higher subbands only add detail.

(2) Large wavelet coefficients are more important than small wavelet coefficients.

The two observations are used to encode the wavelet coefficients in decreasing order in several passes until the target bit rate is achieved.

3.3.1.1 EZW Algorithm

From the above two observations and the zerotree hypothesis, the EZW algorithm was developed. The first aspect of the algorithm involves a simple looping structure where each wavelet coefficient is compared to a threshold value. The second aspect of the algorithm then determines whether the wavelet coefficient is a zerotree root, isolated root or significant root. The third aspect of the algorithm involves two passes with which to code the image; a dominant pass and a subordinate pass [29].

Merging all three aspects the complete EZW algorithm is shown below in Figure 3-27 and functions as follows. The initial threshold is set to $2^{log_2(max)}$ where max is the maximum wavelet coefficient. In the dominant pass the image is scanned through either Raster scanning or Morton scanning and each wavelet coefficient is compared to the threshold. There are three comparison cases in the dominant pass [29]:

(1) If the coefficient and its descendants are larger than the threshold, the coefficient is then declared a significant root and does not need to be coded by lower thresholds and is thus set to zero.

(2) If the coefficient and its descendants are smaller than the threshold, the coefficient is then declared a zerotree root.

(3) If the coefficient is smaller than the threshold but the descendants are larger, the coefficient is then declared an isolated root.

At the end of the dominant pass all the coefficients that are in absolute value larger than the current threshold are extracted and placed without their signs on the subordinate list and marked to prevent them from being coded again. In the subordinate pass, also known as the refinement pass, each coefficient value in the subordinate list is compared to the current threshold. There are two comparison cases in the subordinate pass [29]:

(1) If the coefficient value is larger than the threshold, the current threshold is subtracted from the coefficient value in the subordinate list and a *'1'* is output.

(2) If the coefficient value is smaller than the threshold the output is a *'0'*.

The subordinate list is then re-sorted in order of highest to lowest as the larger coefficients carry more information. The threshold is then decreased by half to improve the accuracy so that a target bit rate can be met. The loop repeats until a minimum threshold is reached, where the minimum threshold controls the bitrate and if specified to *"0"*, lossless compression is

experienced. This minimum threshold represents a target bit rate achieved by the EZW algorithm.

BEGIN

Set the initial threshold to $2^{log_2(max)}$

WHILE *the threshold > minimum threshold possible* **DO**

{

Dominant Pass()

Subordinate Pass()

Decrease the threshold by half to improve accuracy

}

END

Figure 3-27: Algorithm of EZW [30].

As mentioned above the image is scanned using either the Raster scanning method or the Morton scanning method. These scanning methods use a predefined scan order to transmit the coefficients for coding. Both of these methods are illustrated in Figure 3-28. A crucial property of any scanning method is that a child coefficient should never be scanned before a parent coefficient.
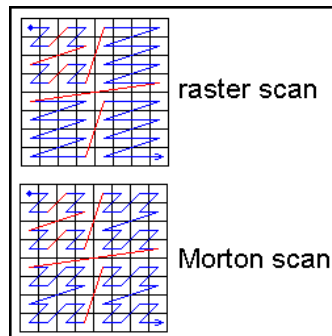
Figure 3-28: Diagram of the Raster and Morton scanning methods [30].

3.3.1.2 Performance of EZW

The EZW algorithm produces excellent results as compared to the JPEG standard, however it is computationally expensive. Though EZW produces improved performance, it must be noted that it was the first wavelet coding scheme developed, which left much room for improvement in wavelet coding. Table 3-3 depicts the PSNR versus bitrate for the *Lena* and *Barbara* test images. An increase in bitrate reveals a proportional increase in the associated PSNR.

	LENA	BARBARA
Bit-Rate (bpp)	**PSNR(dB)**	**PSNR(dB)**
0.25	33.17	26.77
0.5	36.28	30.53
1	39.55	35.14

Table 3-3: PSNR results for EZW [28].

3.3.2 Set Partitioning in Hierarchical Trees

An improved variation of the EZW algorithm was developed by Said and Pearlman known as Set Partitioning in Hierarchical Trees (SPIHT) [32]. This algorithm is considered state of the art with regards to image compression.

SPIHT is a fully embedded progressive wavelet coding algorithm that refines the most significant coefficients. It ensures that the largest coefficients are transmitted first by using various tree searching routines. The SPIHT algorithm uses the partitioning of quad trees to keep insignificant coefficients together. In the implementation of SPIHT, the significant information is stored in three ordered lists [32], [33]:

(1) List of significant pixels (LSP) – contains coefficients that are significant or greater than the threshold.
(2) List of insignificant pixels (LIP) – contains coefficients that are insignificant or less than the threshold.
(3) List of insignificant sets (LIS) – contains sets of coefficients defined by tree structures which are insignificant or smaller than the threshold. The set excludes the coefficients corresponding to the tree or all subtree roots.

The following represents the set of coordinates used with the above lists in the algorithm [32], [33].

(1) $O(i,j)$ – is the set of coordinates of the offspring's of the wavelet coefficient at location (i,j). As each node can have four offspring's (quad-tree), the size of $O(i,j)$ is zero or four.
(2) $D(i,j)$ – is the set of all descendants of the coefficient at location (i,j).
(3) $L(i,j)$ – is the set of all coordinates of the descendants of the coefficient at location (i,j) except the immediate offspring's of the coefficient at location (i,j).
(4) H – is the set of all root nodes.

74

3.3.2.1 SPIHT Algorithm

The SPIHT algorithm consists of two main passes to code the image, a sorting pass and a refinement pass. The LIS and LIP entries are coded in the sorting pass and the LSP entries are coded in the refinement pass. Figure 3-29 shows the outline of the algorithm.

BEGIN

Set the threshold to an initial value

Set LIS, LIP, LSP accordingly

WHILE *the threshold > minimum threshold possible* **DO**

{

    Sorting Pass()

    Refinement Pass()

    Decrease the threshold to improve accuracy

}

END

Figure 3-29: Algorithm of SPIHT.

The initialization of the threshold is the same procedure as that used in the EZW algorithm. The list of significant pixels (LSP) is set to empty or zero and the roots in the similarity trees of lists of insignificant pixels (LIP) and insignificant sets (LIS) are set to H and D respectively. The sorting pass begins by examining each coordinate in the LIP for significance. There are two comparison cases in the LIP, they are [33]:

(1) If the coefficient is significant a *'1'* is transmitted, followed by a bit for the sign of the coefficients to the LSP. The bit is *'0'* for a positive sign and *'1'* for a negative sign.
(2) If the coefficient is not significant a *'0'* is transmitted.

After the LIP is examined, the LIS sets are then examined. There are four comparison cases that make up the LIS component and they are as follows [33]:

(1) If the set at location *(i,j)* is not significant a *'0'* is transmitted.
(2) If the set at location *(i,j)* is significant a *'1'* is transmitted.
(3) If the set is confirmed significant and if it is a set of type D, the offspring coefficients are then individually checked. If the offspring coefficient is significant a *'1'* is transmitted, followed by a bit representing the sign of the coefficient (*'1'* for a positive sign and *'0'* for a negative sign). Next the coefficient is moved to the LSP. If the offspring coefficient is not significant a *'0'* is transmitted.

75

(4) If the set is confirmed significant and if it is a set of type *L*, each coordinate in *O(i,j)* is appended to the LIS as the root to a set of type *D*. These new entries in the LIS are examined during this pass. Thereafter the coordinate *(i,j)* is removed from the LIS.

Once each set in the LIS is processed a refinement pass then takes place. The refinement pass involves examining the coefficients of the LSP and transmitting the nth most significant bit of the coefficient at location *(i,j)*. The remaining stages of the algorithm involve the same procedures described in the EZW algorithm. The SPIHT algorithm does not use scan coefficients like the EZW algorithm however, it is able to output and code descendants immediately.

3.3.2.2 Performance of SPIHT

The SPIHT algorithm offers a more efficient and effective implementation than the EZW algorithm. The results show an impressive improvement in performance as compared to the EZW algorithm. SPIHT exhibits great performance with less computational complexity making it one of the most widely used wavelet coding schemes. Table 3-4 numerically illustrates the performance of the SPIHT coder in terms of its PSNR versus bitrate. Focusing particularly at low bitrate compression, SPIHT exhibits improved performance compared to the EZW coder for the low bitrate of *0.25bpp*.

	LENA	**BARBARA**
Bit-Rate (bpp)	**PSNR(dB)**	**PSNR(dB)**
0.25	34.13	27.57
0.5	37.24	31.39
1	40.45	36.41

Table 3-4: PSNR results for SPIHT [32].

3.3.3 Space Frequency Quantisation

Xiong et al. proposed the Space Frequency Quantization (SFQ) wavelet coding scheme [34]. This algorithm may be viewed as the rate-distortion optimized variant of the EZW algorithm. It is a joint application of zerotree quantization and scalar quantization. The zerotree quantization mode exploits spatial grouping of coefficients in tree structures and the scalar quantization mode exploits frequency grouping of coefficients in subbands hence the name Space Frequency Quantization.

3.3.3.1 SFQ Algorithm

Essentially SFQ uses zerotree quantization to identify a pruned subset of significant wavelet coefficients to be scalar quantized while discarding the rest. The algorithm aims to optimally select spatial regions for applying zerotree quantization and to optimally select the scalar quantizer step-size for quantizing the remaining coefficients.

The algorithm itself consists of two phases. The first phase is called the "tree pruning algorithm", which involves searching for an optimal pruned subtree at a particular quantizer step-size and rate-distortion slope, λ. The second phase called "predicting the tree" involves choosing an optimal quantizer step-size by searching through a finite list of admissible step-sizes and finding the one that minimizes the rate-distortion function shown in (3.15):

$$D + \lambda R \, , \tag{3.15}$$

where D is the distortion and R is the bitrate and the optimal rate-distortion slope λ is then searched using a bisectional algorithm in order to determine the target bitrate R.

3.3.3.2 Performance of SFQ

The SFQ algorithm shows that high performance coding depends on exploiting both frequency and spatial compaction of energy of wavelet coefficients. Using a simple intuitive algorithm, SFQ demonstrates its competitiveness with other wavelet image coding algorithms like EZW and SPIHT. However this scheme is computationally expensive due to its iterative zerotree pruning stage. Table 3-5, shows reasonably similar results to those produced using the SPIHT algorithm. This further illustrates the competitiveness of the SFQ algorithm with SPIHT.

	LENA	BARBARA
Bit-Rate (bpp)	**PSNR(dB)**	**PSNR(dB)**
0.25	34.33	27.20
0.5	37.36	31.33
1	40.52	36.96

Table 3-5: PSNR results for SFQ [34].

3.3.4 Stack-Run Image Coding

The Stack-Run (SR) image coding algorithm was developed by Tsai et al. [38]. It is a conceptually simple algorithm and is computationally inexpensive, yet it remains competitive with other well established wavelet coding algorithms.

3.3.4.1 SR Algorithm

The Stack-Run coding algorithm partitions the quantized wavelet coefficients into two groups containing zero valued and non-zero valued or significant coefficients. The algorithm performs raster scanning within subbands generating stack and run pairs of the form (run, stack) where *run* is the number of zero-valued coefficients encountered before the next coefficient and *stack* is the magnitude and sign of the significant coefficient. This representation is similar to run-length coding used in the JPEG algorithm.

A symbol set containing four symbols {*0, 1, +, -*} were developed to distinguish between the level values and the runs of zeros. *"0"* and *"1"* is used to signify the binary bit values of *0* and *1* respectively in encoding of significant coefficients in the run portion of the stack-run pair. *"+"* and *"-"* is used to represent *0* and *1* respectively in the stack portion of the stack-run pair. The {stack} coefficients and {run} coefficients within the pair are independently arithmetically coded in order to enhance the performance of the algorithm.

3.3.4.2 Performance of SR

The SR image coding algorithm is essentially a low complexity adaptive arithmetic coder. Given this low complexity design it still manages to maintain reasonable performance with respect to the EZW algorithm. It is slightly inferior when compared to other wavelet coding schemes however it does boast lower computational overhead than other coders. The numerical performance results in Table 3-6 were obtained for the SR algorithm. Its low bitrate of *0.25bpp* achieves a slightly higher outcome than EZW nonetheless, it is still performs below the famous SPIHT algorithm.

	LENA	BARBARA
Bit-Rate (bpp)	**PSNR(dB)**	**PSNR(dB)**
0.25	33.63	27.39
0.5	36.79	30.98

Table 3-6: PSNR performance for SR [38].

3.3.5 Embedded Conditional Entropy Coding of Wavelet Coefficients

A wavelet coding technique called Embedded Conditional Entropy Coding of Wavelet Coefficients (ECECOW) was proposed by Xiaolin Wu [40]. ECECOW is a scheme for context modeling and entropy coding of quantized wavelet coefficients. It differs from the EZW and SPIHT algorithms as it is not a zerotree-based method but rather a sample-by-sample bit plane coding technique.

Wu outlined that the zerotree is a high-order context model of small wavelet coefficients and imposes an artificial structure on the wavelet coefficients. In other words the zerotree only uses modelling contexts of square shape in the spatial domain whereas statistically dependent wavelet coefficients may form arbitrary shaped regions. Thus ECECOW adaptively shapes the modelling contexts to the statistics of wavelet coefficients.

3.3.5.1 ECECOW Algorithm

The ECECOW algorithm only focuses on entropy coding of quantized wavelet coefficients and thus does not delve into the wavelet transform or quantization processes. The ECECOW algorithm begins with bit-plane coding the quantized wavelet coefficients into a binary symbol stream. This bit stream is then compressed by an adaptive binary arithmetic coder. The ECECOW algorithm estimates the conditional probability of a wavelet coefficient based on past coded bits and then uses this estimate to drive the arithmetic coder. This estimation is called modelling context and determines the bit rate of the compression algorithm. In fact this statistical context modelling in the form of probability estimation lies at the heart of this compression scheme.

Wavelet coefficients of similar magnitudes statistically cluster in frequency subbands and spatial locations. Large wavelet coefficients in different frequency subbands tend to register at the same spatial locations. Using these observations ECECOW models a coefficient c by its neighbours in the current subband and by the spatially corresponding coefficients p in the parent subband.
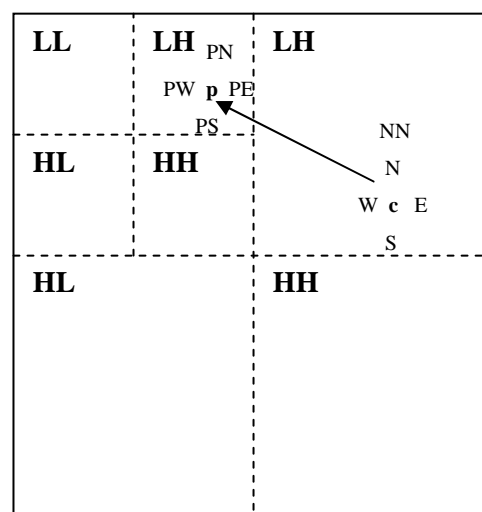
Figure 3-30: ECECOW Context Modelling [40]

Figure 3-30 illustrates the different orientations of the modelling contexts used in different subbands. Thus the modelling contexts used in the LH subbands exhibit predominantly horizontal sample structures and the HL subbands exhibit predominantly vertical sample structures.

Once this adaptive context selection based on subband orientations is completed, quantization of the modelling event takes place. This context quantization helps reduce the number of conditioning states for the entropy coding stage. The essence of context quantization is to merge different conditioning states that have similar symbol probability distributions.

3.3.5.2 Performance of ECECOW

The excellent performance of ECECOW is solely due to high order adaptive context modelling. ECECOW is embedded like the EZW and SPIHT algorithms and yet still manages to maintain higher coding efficiency than these two wavelet coders. ECECOW exhibits superior compression performance thus demonstrating the benefits of using high order statistics in wavelet coefficient coding. This algorithm presents a convincing argument that context modelling and conditional entropy coding of wavelet coefficients are extremely important and particularly effective in wavelet coding. Table 3-7 depicts the PSNR results for the ECECOW algorithm. The results indicate that this particular algorithm achieves impressive PSNR performance as compared to other wavelet coders.

	LENA	BARBARA
Bit-Rate (bpp)	PSNR(dB)	PSNR(dB)
0.25	34.81	28.85
0.5	37.92	32.69
1	40.85	37.65

Table 3-7: PSNR performance for ECECOW [40].

3.4 PERFORMANCE

3.4.1 Various Wavelet Coding Schemes

The above wavelet coding schemes were compared in terms of its PSNR versus bitrate (image quality versus compression) with the standardised compression techniques of JPEG and JPEG 2000. This comparison was demonstrated using the performance values of PSNR and bitrate in Table 3-3 to Table 3-7 which use the *Lena* (512x512) and *Barbara* (512x512) grayscale test

images. The results were generated using the Matlab simulation engine. Both the graphs show the ECECOW wavelet coding scheme producing the best rate-distortion characteristics of all the coders. The SFQ, SPIHT and JPEG 2000 schemes all have similar rate-distortion graphs.
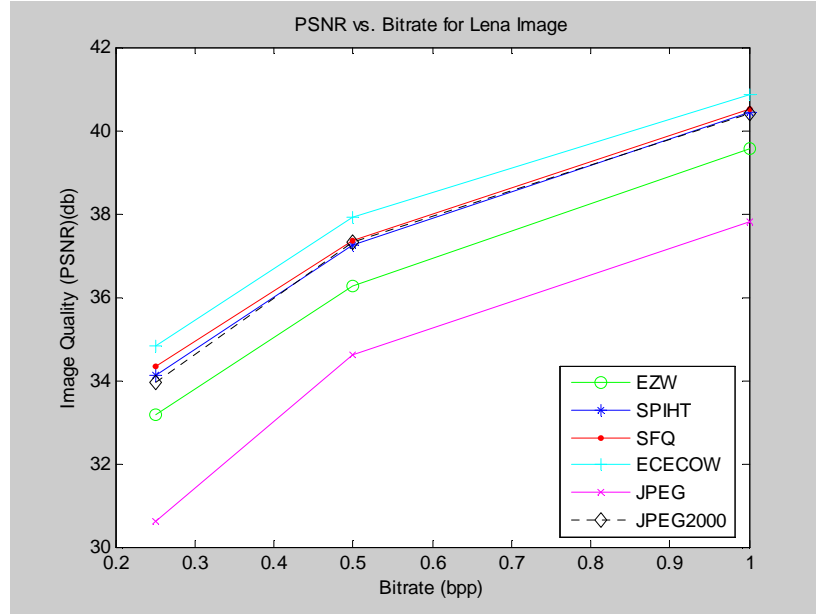
Figure 3-31: PSNR vs. Bitrate for Lena Image.
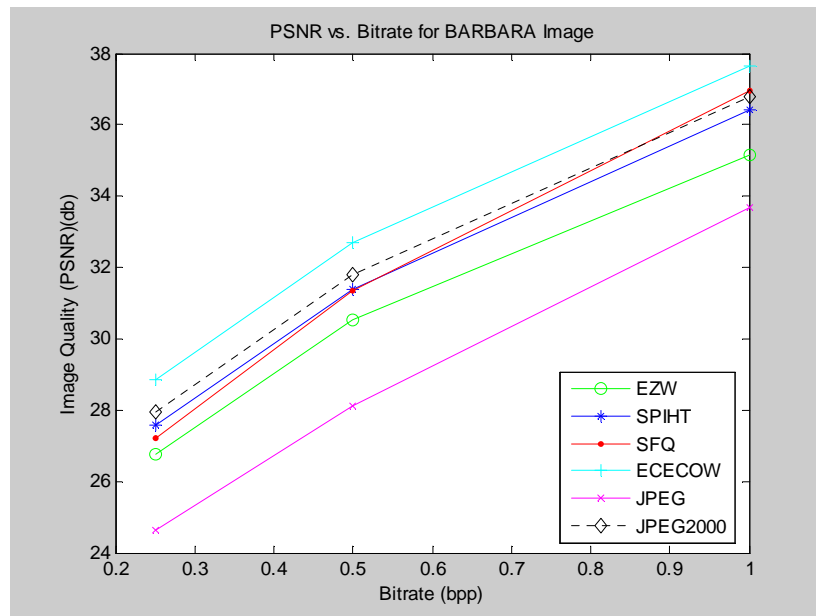
Figure 3-32: PSNR vs. Bitrate for Barbara Image.

3.4.2   EZW and SPIHT

The main focus of the performance consideration concerns the EZW and SPIHT wavelet coders, as these wavelet coding schemes are used in the proposed coder scheme discussed later. The ECECOW coder will not be included as the algorithm itself incorporates adaptive binary

81

arithmetic coding as its wavelet compression technique. Arithmetic coding is used as a joint error detection and entropy coding stage for the wavelet coding schemes which is discussed in Chapter 5. An adaptive binary arithmetic coder is already utilised in the ECECOW algorithm as an entropy coding stage and wavelet coding algorithm, thus manipulation of the current adaptive binary arithmetic coder for error detection will not succeed as it will change the composition of the algorithm producing unstable results. Thus only the EZW and SPIHT wavelet coding algorithms will employed for the proposed coder.

Performance results of the two wavelet compression coders against standardised compression schemes of JPEG and JPEG 2000 are demonstrated. The compressed images are generated using the VcDemo Image and Video Compression Learning Tool and its PSNR calculations are simulated using the Matlab simulation engine. The PSNR values are calculated using the MSE of the compressed images against the original uncompressed image for a series of bitrates ranging from *0.05bpp* to *3bpp*.

The two wavelet coders and the two standard schemes were systematically evaluated using a sample set of three grayscale test images, *Lena, Barbara* and *Cameraman*, each having a resolution of 256x256. The use of these various test images present a range of challenges such as reproduction of fine detail and textures, edges and sharp transitions and uniform regions when compressed by the wavelet algorithms, providing a comprehensive performance evaluation of these algorithms.

Both the EZW and SPIHT coders outperform the DCT-based JPEG standard, with SPIHT performing fairly well across all test images. This can be seen in Figure 3-33 to Figure 3-35. Both these schemes are on par with the performance of the standardised JPEG 2000 scheme. This is due to JPEG 2000 being based on the Embedded Block Coding with Optimised Truncation (EBCOT) wavelet coding scheme, similar to the embedded structure of the SPIHT scheme.

The results are obtained by calculating the difference between PSNR values at a given bitrate for the two sets of data and then finding the mean across the values. There is an overall *2.8dB* improvement in image quality for the EZW algorithm and a *5dB* improvement is seen for the SPIHT algorithm for *Lena*, *Barbara* and *Cameraman* in Figure 3-33 to Figure 3-35 when compared to the JPEG standard.

When compared to the JPEG 2000 standard there is a *1dB* improvement in image quality for the SPIHT scheme, however the EZW does not perform as well and has an image quality that is

1.2dB less than that of JPEG 2000 for *Lena*, *Barbara* and *Cameraman* in Figure 3-33 to Figure 3-35.

EZW performs better than JPEG at low bit rates as it preserves all significant coefficients at each scale by testing the zero tree hypotheses for all the coefficients. The EZW algorithm overcomes the "blocking artifacts" problem as it transfers the entire image before coding. The SPIHT algorithm achieves higher compression performance than EZW due to its improved zerotree searching routine. This is because SPIHT does not scan coefficients in a predetermined order like the EZW which uses Raster scanning, its scans through lists and encodes significant descendants immediately thus improving rate efficiency.
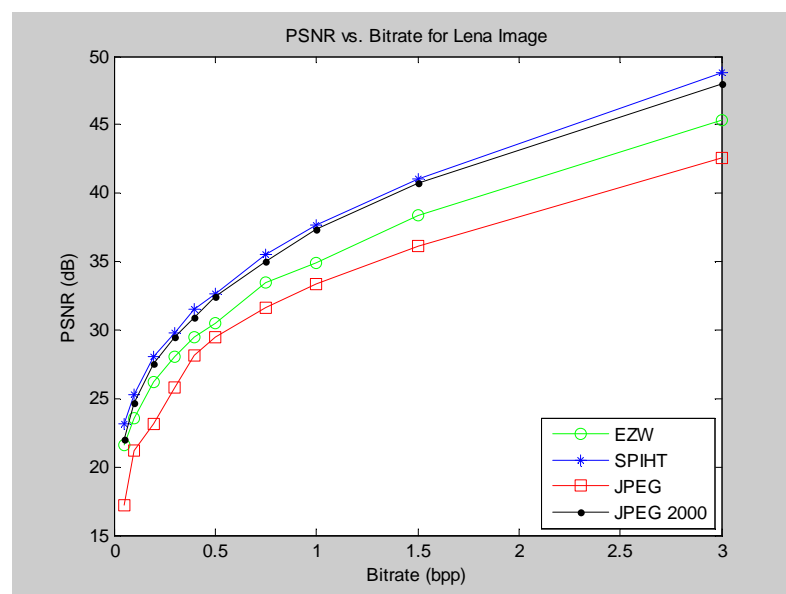
Figure 3-33: Diagram of PSNR vs. Bitrate for EZW, SPIHT, JPEG and JPEG 2000 for *Lena* Image.
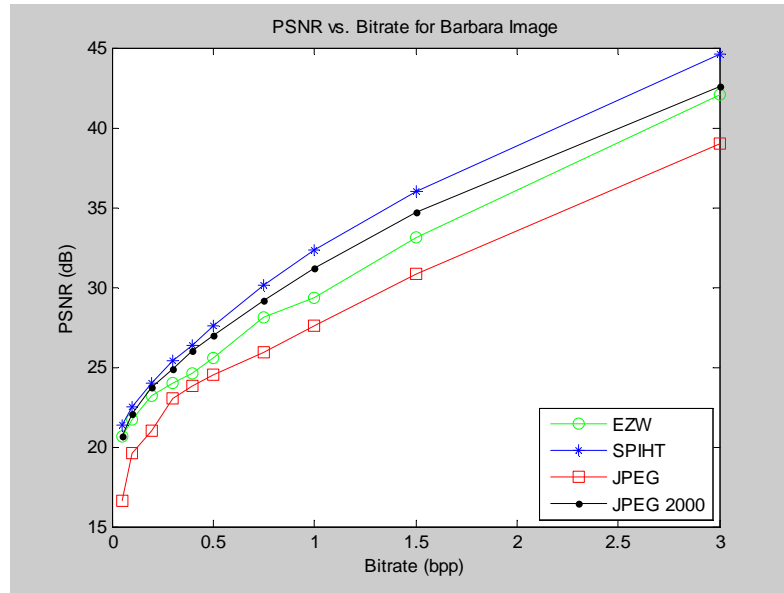
Figure 3-34: Diagram of PSNR vs. Bitrate for EZW, SPIHT, JPEG and JPEG 2000 for *Barbara* Image.
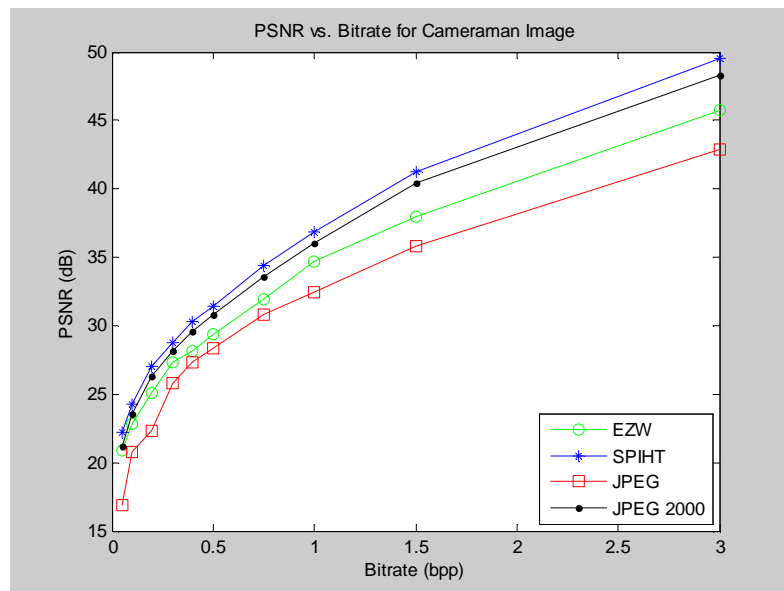
Figure 3-35: Diagram of PSNR vs. Bitrate for EZW, SPIHT, JPEG and JPEG 2000 for *Cameraman* Image.

The image quality (PSNR) for the three test images are visually depicted in Figure 3-36 to Figure 3-38. The images illustrate the image quality for SPIHT coding at various bitrates. As the bitrate decreases or compression increases, the image quality becomes more degraded causing blurring of the image.

Figure 3-36: Lena image for SPIHT coding for bitrates of *0.05bpp*, *0.2bpp* and *0.5bpp* with PSNR of *23.1dB*, *28dB* and *32.7dB* respectively.

Figure 3-37: Barbara image for JPEG 2000 for bitrates of *0.05bpp*, *0.2bpp* and *0.5bpp* with PSNR of *20.7dB*, *23.7dB* and *29.2dB* respectively.

Figure 3-38: Cameraman image for JPEG for bitrates of *0.05bpp*, *0.2bpp* and *0.5bpp* with PSNR of *16.9dB*, *22.3dB* and *28.4dB* respectively.

3.5   SUMMARY

This chapter introduced the concept of wavelets and its associated transforms. The theory behind multiresolution analysis, filter banks and subband coding provided the basis for the Fast Wavelet Transform, which replaced the Discrete Wavelet Transform as a computationally efficient substitute.

85

The various wavelet families were evaluated and the Coiflet 5 wavelet provided the best results. However, any of the recommended wavelet families would still provide fairly good performance with the exception of the Haar wavelet family.

A number of wavelet coding schemes were presented and evaluated. These wavelet coders outperformed the DCT- based JPEG standard verifying the impact wavelet-based coders have in image compression. The EZW algorithm, which was the original wavelet coder, surpassed the JPEG scheme. The ECECOW and SPIHT coder's performance exceeded expectation as they showed superior rate-distortion. SPIHT performed fairly well when measured against JPEG 2000, which is based on EBCOT, a similar embedded structure to SPIHT.

CHAPTER 4 -   WIRELESS CHANNELS

The rapid growth in interactive multimedia has resulted in the spectacular progress of wireless communication systems. However, there still exist many obstacles in efficient multimedia communication over wireless channels, some of which are high error rates and stringent delay constraints caused by severe wireless channel conditions as well as limited bandwidth availability and complex time-varying wireless channel environments.

Some of the critical wireless channel impairments experienced are: path loss, multipath fading, interference and noise disturbances. These impairments consequently affect the transmission of image and video over a wireless channels. Thus reliable multimedia transmission has become essential due to the challenges posed by the highly varying wireless channel conditions.

4.1   ADDITIVE WHITE GAUSSIAN NOISE CHANNEL MODEL

The most prevalent problem in any communication system is noise. Noise corrupts the signal in an additive fashion and can be described by a Gaussian random process. It is generally modelled using the additive white Gaussian noise (AWGN) channel [1], [47], [79], [80]. Noise can be produced by either: thermal noise, atmospheric noise or random interference. White Gaussian noise is noise with its power spectral density distributed over all frequencies and having an amplitude described by the Gaussian probability density function (PDF). The Gaussian PDF is given by [1], [80];

$$p(z) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\left(\frac{(z-\mu)^2}{2\sigma^2}\right)} \, , \qquad (4.1)$$

where μ is the mean, σ is the standard deviation and σ^2 is the variance of the signal z. A graphical representation of this function is depicted in Figure 4-1 where $p(z)$ at the peak equals $\frac{1}{\sqrt{2\pi\sigma^2}}$ at a mean of $z=\mu$. The Gaussian PDF is normally described as bell shaped and is commonly referred to as the "bell shaped curve" as it exhibits symmetric features.
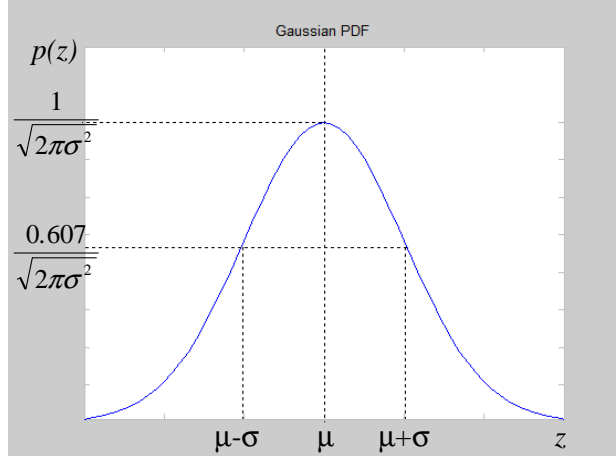
Figure 4-1: Diagram of Gaussian probability density function [1].

The following diagram represents the volume under a two-dimensional plot of the Gaussian PDF with a mean of *0* and a variance of *1*, which is essentially a three-dimensional plot.
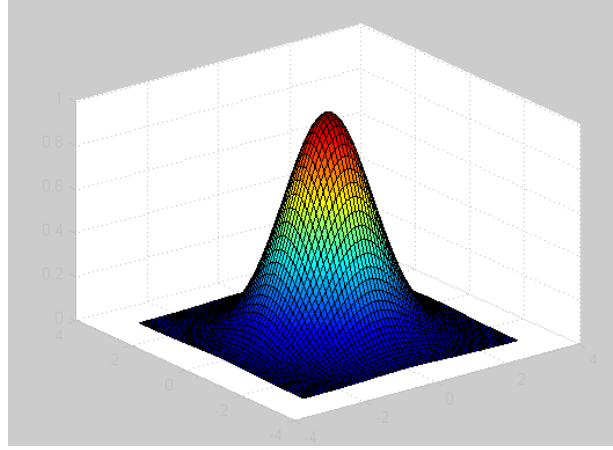
Figure 4-2: Diagram of 2D Gaussian probability density function.

Due to its simple mathematical tractability in both the spatial and frequency domains, Gaussian noise models are perhaps the most frequently used distribution. The additive white Gaussian noise channel model is one where there exists a linear addition of zero mean white Gaussian noise to the transmitted signal. This concept is mathematically defined in equation (4.2) [48];

$$r(t) = s(t) + n(t), \qquad (4.2)$$

where *r(t)* denotes the received signal, *s(t)* is the transmitted signal and *n(t)* is the zero mean white Gaussian noise with a two sided power spectral density of $N_0/2$. The AWGN channel is crucial in defining the noise added to the transmitted signal but is inadequate in characterising

signal transmissions over channels whose transmissions vary with time. Therefore the model cannot account for fading, frequency selectivity, interference, nonlinearity or dispersion.

4.2 MULTIPATH FADING CHANNELS

The multipath phenomenon occurs when a signal arrives at the receiver via multiple propagation paths with various delays due to obstacles and reflections. The multipath channel [43], [45], [46], [47] can be described as having a dominant line-of-sight component with or without non-line-of-sight components (Rician channel model) or having only non-line-of-sight components (Rayleigh channel model) where line-of-sight is the direct connection between the transmitter and receiver. Non-line-of-sight is the path determined after reflections. This concept is illustrated in Figure 4-3.
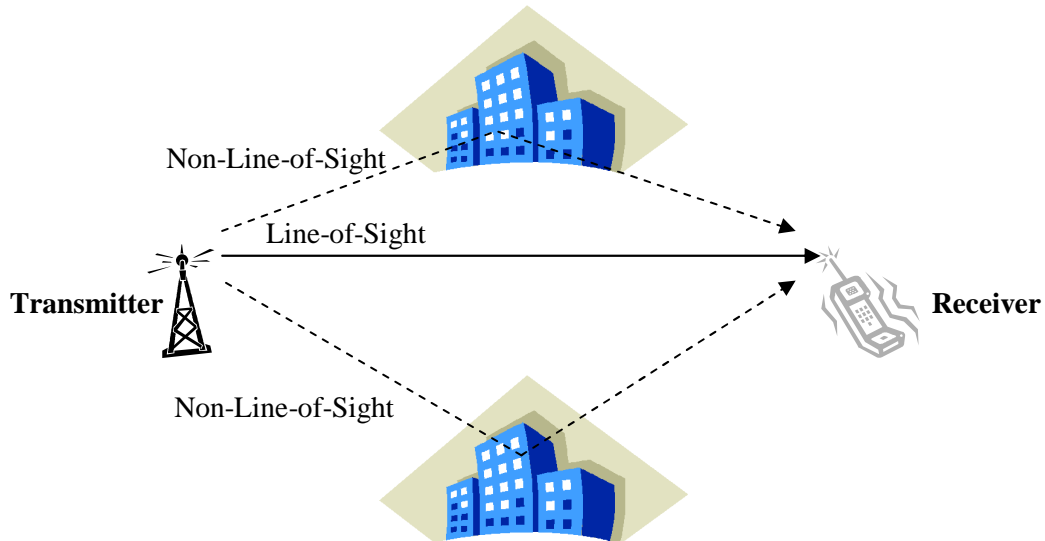
Figure 4-3: Diagram of Multipath Non-Line-of-Sight and Line-of-Sight paths.

There are three types of mechanisms that affect signal propagation; reflection, diffraction and scattering [43].

- Reflection occurs when a propagating radio wave impinges on a smooth surface with very large dimensions compared to the RF (radio frequency) signal wavelength (λ).
- Diffraction occurs when the radio path between the transmitter and receiver is obstructed by a dense body with large dimensions in comparison to λ, causing secondary waves to be formed behind the obstructing body. Diffraction accounts for RF energy travelling from transmitter to receiver without a line-of-sight path between the

two. It is often termed *shadowing* as the diffracted field can reach the receiver even when shadowed by an impenetrable obstruction.

- Scattering occurs when a radio wave impinges on either a large rough surface or any surface whose dimensions are in the order of λ or less, causing the reflected energy to spread out (scatter) in all directions.

The effect of the multipath channel can cause fluctuations in the received signal's amplitude, phase and angle of arrival giving rise to the idea of multipath fading. The delay of the reflected path is known as delay spread.

4.3 PATH LOSS

Path loss [43] is the attenuation of a signal as it propagates from a transmitter to receiver. Path loss may be due to reflection, refraction, scattering, free-space loss, distance between antennas, terrain etc. Path loss can also be modelled using the log-distance path loss equation given by (4.3) [43];

$$P_L(d) = P_L(d_0) + 10\gamma log(d/d_0) \ , \tag{4.3}$$

where P_L is the path loss in dB's, d is the distance between the transmitter and receiver in metres, d_0 is the reference distance, $P_L(d_0)$ is the mean path loss and γ is the path loss exponent. The value of the path loss exponent γ [43] depends on the frequency, antenna height and propagation environments and can range between 2 and 6. It is equal to 2 for free space, and equal to 4 for lossy environments and specular reflection from the earth's surface. When obstructions are present like buildings or irregular terrain the path loss is generally between 2 and 4, in other instances like indoor environments (reflection, refraction, scattering) the path loss can reach values between 4 and 6. A tunnel may act as a strong wave-guide and the path loss value can drop to below 2. Essentially path loss involves the signal power decaying as a function of distance, as the signal propagates through free space in outdoor environments or indoor environments.

4.4 SHADOWING

Shadowing is caused by the presence of obstacles in the propagation path of a signal. Shadowing occurs if the transmitted signal is obstructed or absorbed due to the environment.

This then causes the attenuation of the signal at the receiver. Shadowing is commonly referred to as log-normal shadowing as it is modelled by a log-normal distribution of the mean signal power. The shadowing component is included in the following log-distance path loss equation [43];

$$P_L(d) = P_L(d_0) + 10\gamma log(d / d_0) + X_\sigma \; ,$$

(4.4)

where X_σ is the shadowing component with is modelled as a zero mean Gaussian random variable with a standard deviation σ given in dB's. The standard deviation is dependent on the environment of the receive antenna. Typically urban areas experience a standard deviation of about *6dB* to *8dB*, while rural areas have between *10dB* to *12dB*.

4.5   FADING CHANNELS

The fluctuations in the received signal's amplitude, phase and angle, can be characterised by two main manifestations: large-scale fading and small-scale fading. These two broader types of fading give rise to further specific types of signal degradations. This is outlined in Figure 4-4 showing the breakdown of the fading manifestations and its associated degradations.
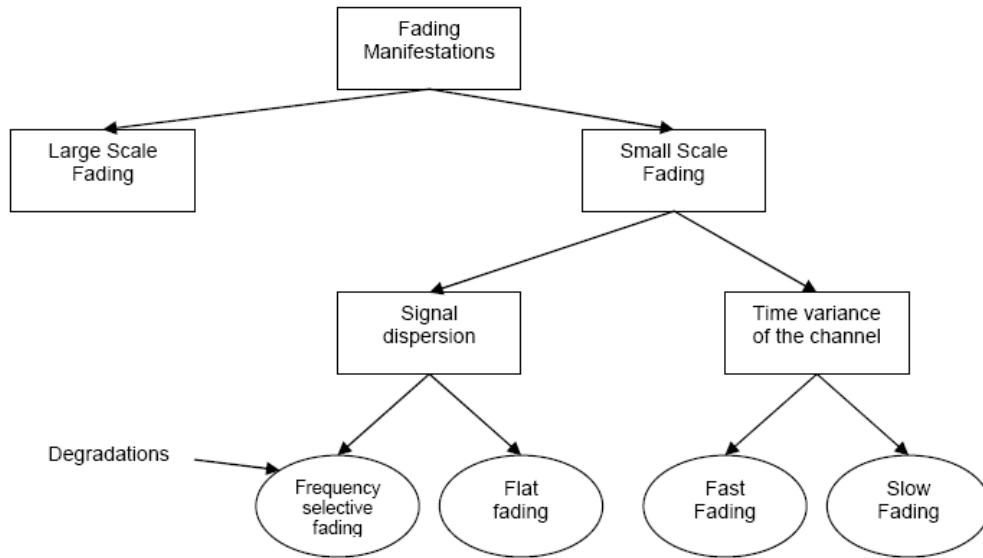
Figure 4-4: Diagram of Fading manifestations and associated degradations [45].

4.5.1   Large Scale Fading Channels

Large-scale fading [43] represents the average signal power attenuation and path loss due to motion over large areas. Large-scale fading losses are generally a result of large physical objects

between the transmitter and receiver, like prominent terrain contours specifically hills, forests, buildings etc. The receiver can be characterized as being "shadowed" by these obstructions.

4.5.2 Small Scale Fading Channels

Small-scale fading [43], [49] refers to the significant changes in the signal amplitude and phase as a result of small changes in spatial positioning of the transmitter and receiver. These changes may be caused by mobility of the transmitter or receiver, or obstructions in the path of the signal. There are two manifestations of small-scale fading, signal dispersion and time variance of the channel.

Signal dispersion involves the time spreading of the signal. This may occur due to multiple scatterers at different delays. The signal dispersion manifestation is based on the multipath delay spread theory. Delay spread is defined as the largest of the delays among various reflected and scattered propagation paths in the channel.

Time variant channels involve the mobility of the transmitter and receiver or obstructions in the path of the signal resulting in propagation path changes ultimately producing fading impairments. These changes may include variations in the relative delays of the signals from multiple scatterers. Time variant channel manifestation is based on the Doppler spread. Doppler spread is defined as the largest of the frequency shifts of various paths.

Both of the fading manifestations, signal dispersion and time variance of the channel can be characterized in the time and frequency domain by the various degradation types. Signal dispersion produces frequency-selective fading and flat fading, whereas the time variant channel manifestation produces fast and slow fading.

4.5.3 Flat Fading Channels

Flat-fading [43], [44], [46], [48], [49] occurs when the channel frequency response is flat or constant as well as linear in phase over the whole signal bandwidth. In flat-fading channels the bandwidth of the signal is less than the channel coherence bandwidth, where the coherence bandwidth is defined as the range of frequencies or bandwidth which the channel can be considered flat or non-distorting. This description is expressed in the frequency domain context of flat-fading channels and can also be described in the time domain context. The time domain context involves delay spread where the delay spread is less than the symbol duration. Figure 4-5 depicts the time and frequency domain flat fading signal representations.
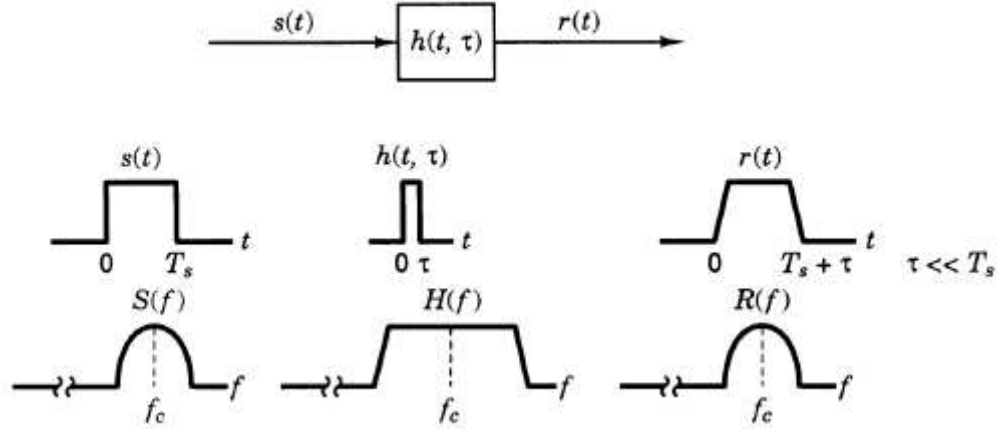
Figure 4-5: Diagram of Flat fading channel in the time and frequency domains [43].

The signal dispersion of a fading channel is equivalent to the signal spreading of a filter. In Figure 4-5 the filter represents a flat fading channel where $s(t)$ is the transmitted signal, $h(t, \tau)$ represents the flat fading channel (wideband filter or narrow impulse response) and $r(t)$ denotes the received signal in the time domain case. T_s is the duration of the transmitted signal and τ represents the delay spread. In the frequency domain, f_c is the carrier frequency with $S(f)$, $H(f)$ and $R(f)$ denoting the transmitted signal, the flat fading channel and the received signal respectively. As discussed above the delay spread is less than the symbol duration or transmitted signal duration and given by $\tau < T_s$. The flat fading channel yields an output free of distortion.

4.5.4   Frequency-Selective Fading Channels

Frequency-selective fading [43], [44], [46], [48], [49] has a reciprocal relationship to flat fading channels. Hence frequency-selective fading arises when the signal bandwidth is greater than the channel's coherence bandwidth. This results in different frequency components of the transmitted signal undergoing different degrees of fading. Its time domain characteristics show the delay spread exceeding its symbol duration causing inter-symbol interference (ISI).
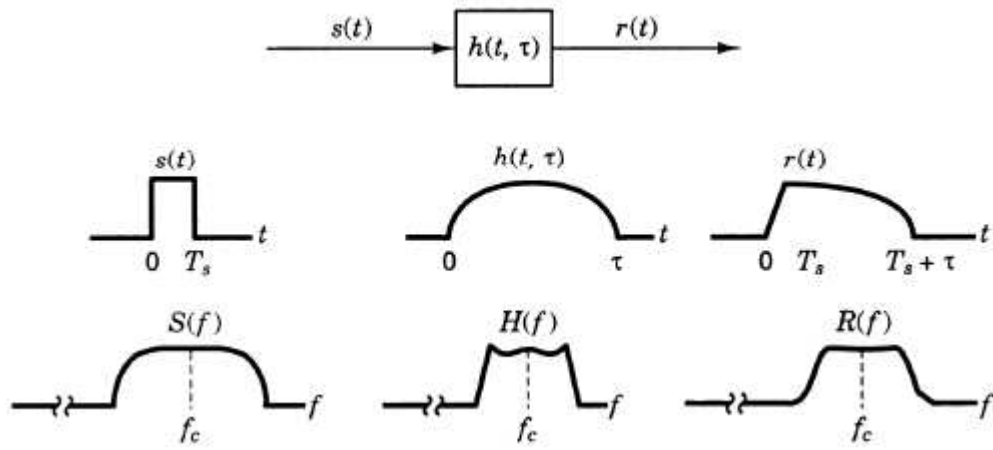
Figure 4-6: Diagram of Frequency-selective fading channel in the time and frequency domains [46].

Discussion concerning the symbols used in the flat fading diagram holds true for this frequency-selective fading diagram with the exception of the filter. The filter in frequency-selective fading is a narrowband filter or a wide impulse response representing a frequency selective channel. In frequency selective fading the delay spread exceeds the symbol duration or transmitted signal duration and this is illustrated with $\tau > T_s$. The output of a frequency selective filter suffers significantly from the distortion as seen in Figure 4-6.

4.5.5 Fast Fading Channels

Fast fading [43], [44], [49] takes place when the channel response changes faster than the symbol duration of the transmitted signal. In other words the coherence time of the channel is less than the transmitted signal duration, where the coherence time refers to the time duration over which the channel response can be considered stable. The frequency domain employs the Doppler spread [43], [49] which is the reciprocal of coherence time. In the frequency domain interpretation of the signal distortion due to fast fading is greater when the signal bandwidth is less than the Doppler spread. Essentially the frequency shift due to the Doppler spread has significant impact on the signal spectrum. In addition fast fading occurs at low data rates and can also occur jointly with the flat fading and the frequency-selective fading channel.

4.5.6 Slow Fading Channels

The slow fading channel [43], [44], [49] has a reciprocal nature to the fast fading channel. Thus when the channel response changes at a much slower rate than the transmitting signal slow fading is experienced. The channel can be assumed static or stable for several symbol durations. Therefore the signal duration is less than the coherence time. In terms of its frequency domain characterisation the Doppler spread is much less than the signal bandwidth. The slow fading channel may also be used in conjunction with the signal dispersion degradations.

4.6   RAYLEIGH MULTIPATH FADING CHANNEL MODEL

The Rayleigh multipath fading channel [1], [43], [46], [49] is a good approximation of a realistic channel where in a wireless communication scenario a receiver is in relative motion to a transmitter with no line-of-sight path between them. Rayleigh fading generally represents the worst fading case scenario due to a no line-of-sight path. The signals arriving at the receiver represent multiple independent random variables with mean and variance constraints typical of a Gaussian process. The Rayleigh fading channel can be characterized mathematically by the following equation;

$$r(t) = s(t)h(t) + n(t) \ , \tag{4.5}$$

where $r(t)$ is the received signal, $s(t)$ denotes the transmitted signal, $h(t)$ is representative of the Rayleigh multipath fading channel and $n(t)$ corresponds to additive Gaussian noise. The Rayleigh multipath fading channel, $h(t)$ is modelled as a zero-mean complex Gaussian random process where it has the following form;

$$h(t) = x(t) + jy(t) \ ,$$
$$= z(t)e^{j\phi(t)} \ , \tag{4.6}$$

where $x(t)$ and $y(t)$ represent real Gaussian random processes which are stationary and statistically independent. The amplitude $z(t)$ can be statistically described by the Rayleigh probability distribution function and $\phi(t)$ is uniformly distributed over the interval $(0,2\pi)$. The fading amplitude $z(t)$ and the fading phase $\phi(t)$ can be further represented in terms of the zero-mean Gaussian process and is indicated in the following equations;

$$|z| = \sqrt{x^2 + y^2} \ , and$$
$$\phi = arctan\left(\frac{y}{x}\right) \ , \tag{4.7}$$

The Rayleigh multipath fading channel is characterized by a Rayleigh probability density function. The Rayleigh PDF is commonly associated with the envelope of a narrowband Gaussian process. It is the most widely used distribution function and its PDF is given in (4.8) [43], [80];

$$p(z) = \begin{cases} \dfrac{z}{\sigma^2} e^{-\left(\frac{z^2}{2\sigma^2}\right)} & \text{for } z \geq 0, \\ 0 & \text{for } z < 0, \end{cases} \qquad (4.8)$$

where z is the envelope amplitude and σ^2 is the variance in the distribution. A graphical interpretation of the PDF is shown in Figure 4-7 where $p(z)$ at the peak equals $\dfrac{0.607}{\sigma^2}$ at $z=\sigma$, the standard deviation. The Rayleigh probability density function envelope is displaced from the origin and is skewed to the right unlike the symmetrical nature of the Gaussian PDF.
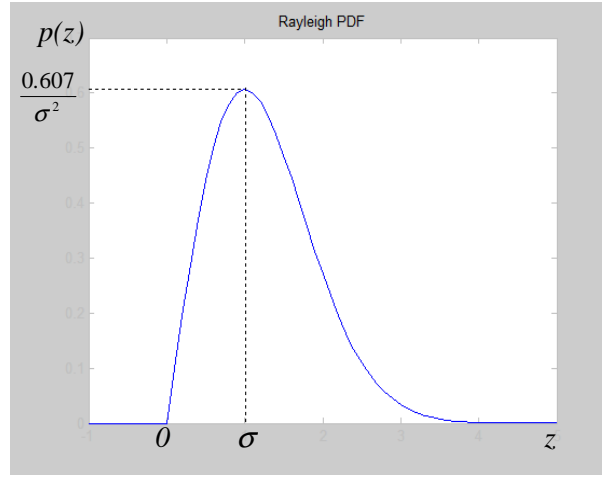
Figure 4-7: Diagram of Rayleigh probability density function [1].

Rayleigh multipath fading is a result of constructive and destructive interference between several versions of the signal via several paths at the receiver leading to attenuation of the signal power or amplitude. Therefore deep nulls can be experienced in the received signal due to significant destructive interference resulting in little or no signal received.

Rayleigh multipath fading channel is a good, simple, mathematically tractable channel model to implement in order to characterise a real wireless channel environment. Fading has been the primary cause of performance degradation and thus demands great attention when trying to model efficient communication systems.

4.7   PERFORMANCE

This performance chapter evaluates the wavelet compression coding schemes of EZW and SPIHT transmitted across two degrading channels; AWGN and Rayleigh fading. The

performance looks at the visual impact the channel has on the wavelet compressed images for a range of SNR's. The performance evaluation is used to determine an acceptable average SNR band that will produce visually acceptable wavelet compressed images.

The two channel models AWGN and Rayleigh fading simulation parameters require the use of binary phase-shift keying (BPSK) [80] also known as 2-PSK, modulation to illustrate its error performance in this chapter. BPSK is the simplest form of PSK as it uses two phases separated by 180°. The BPSK constellation can be seen in the Figure 4-8. The BPSK modulation is the most robust modulation with regard to errors thus making it suitable for use in the proposed coder.
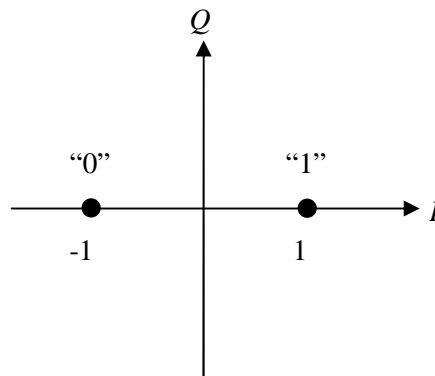
Figure 4-8: Diagram of BPSK constellation.

Occasionally the communication channel can introduce an arbitrary phase shift making it difficult for the demodulator to distinguish the constellation point; -1 or 1. The signal can then be differentially encoded prior to BPSK modulation in order to account for the phase shift. Matlab is able to model a channel that introduces phase shifting by using differential BPSK (DBPSK) modulation [80] which is easier than differentially encoding BPSK signals. Essentially DBPSK eliminates the ambiguity concerning whether the demodulated data is inverted or not. It is more suitable to implement DBPSK modulation for the channel instead of BPSK modulation; this is in the event that the signal begins phase shifting and DBPSK is able to process the signal in during demodulation as it does not require coherent demodulation as BPSK. However the use of DBPSK modulation comes at the expense, as BPSK modulation is able to produce a *3dB* advantage over DBPSK modulation. DBPSK modulation is implemented for the proposed wireless channel results as a phase shift was specifically observed for the Rayleigh fading channel. Therefore in order to maintain consistency throughout the proposed results, DBPSK is implemented for both channels to eliminate the noisy phase reference induced with the Rayleigh fading channel. Implementation of DBPSK over differentially encoding BPSK signals is beyond the scope of this dissertation, and DBPSK is used in order to model the channel for simulation results to induce channel errors.

The channels were modelled using the Matlab simulation engine using the pre-coded channel model tools for the AWGN and Rayleigh fading channels. The Rayleigh channel model uses a sampling period of *100,000Hz* or *0.1μ* seconds and a Doppler shift of *130Hz*. The simulation results for the EZW and SPIHT algorithms were also generated using the Matlab simulation engine using the *Lena* [85] sample image as a visual impact guide.

4.7.1   Theoretical AWGN Channel

The error performance of the AWGN channel using DBPSK modulation is illustrated in Figure 4-9 below. The bit error rate (BER) decreases with an increase in the Signal-to-Noise-Ratio (SNR) for the DBPSK modulated signals demonstrating that less channel errors are produced at higher SNR values. The graph shows the typical signal decay of the channel modelled by equation (4.9):

$$BER_{AWGN} = \tfrac{1}{2}\, erfc(\, SNR\,), \qquad\qquad (4.9)$$

where *erfc* is the complementary error function, *SNR* is the average Signal-to-Noise-Ratio in dB's equalling E_b/N_0 and *BER* is the bit error rate. It can be seen that the AWGN channel is a simple tractable model that is generally used as a benchmark for other fading channels, as the model does not account for fading, frequency selectivity, interference, nonlinearity or dispersion. This is because the AWGN channel's received signal is a form of the transmitted signal with some proportion of Gaussian white noise added to it.
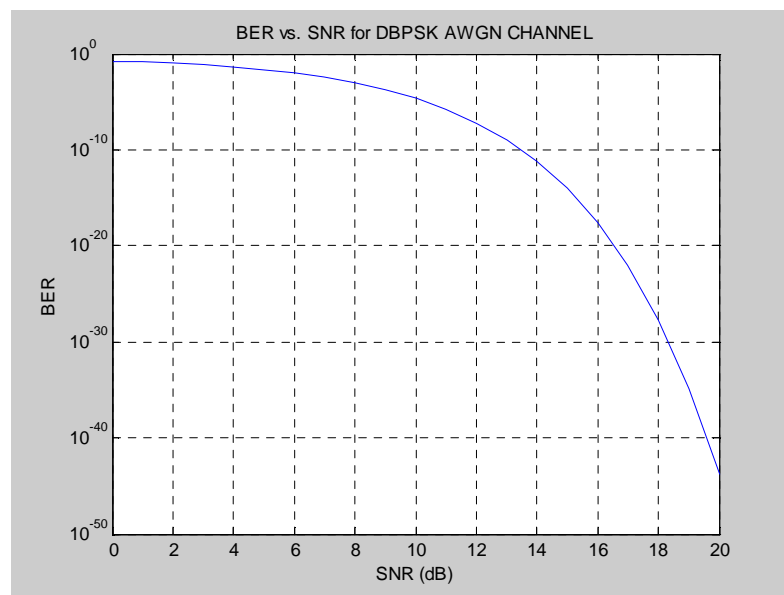
Figure 4-9: Diagram of BER vs. SNR for DBPSK AWGN channel.

4.7.2 Theoretical Rayleigh Multipath Fading Channel

The error performance of the Rayleigh multipath fading channel is depicted in Figure 4-10. DBPSK modulation was used for the fading channel where the signal decay can be described by the following equation:

$$BER_{RAYLEIGH} = \frac{1}{2}\left(1 - \sqrt{\frac{SNR}{1 + SNR}}\right), \qquad (4.10)$$

where *SNR* is the average Signal-to-Noise-Ratio in dB's and *BER* is the bit error rate. The average SNR is defined in (4.11) in terms of the fading variance σ^2:

$$SNR = \sigma^2\left(\frac{E_b}{N_0}\right). \qquad (4.11)$$

As compared to the AWGN channel the bit error rate is far higher in the Rayleigh multipath fading channel, this is attributable to the AWGN channel having no fading properties. The Rayleigh multipath fading model represents a "worst case" scenario in signal fading as the received signal strength can experience deep fades.
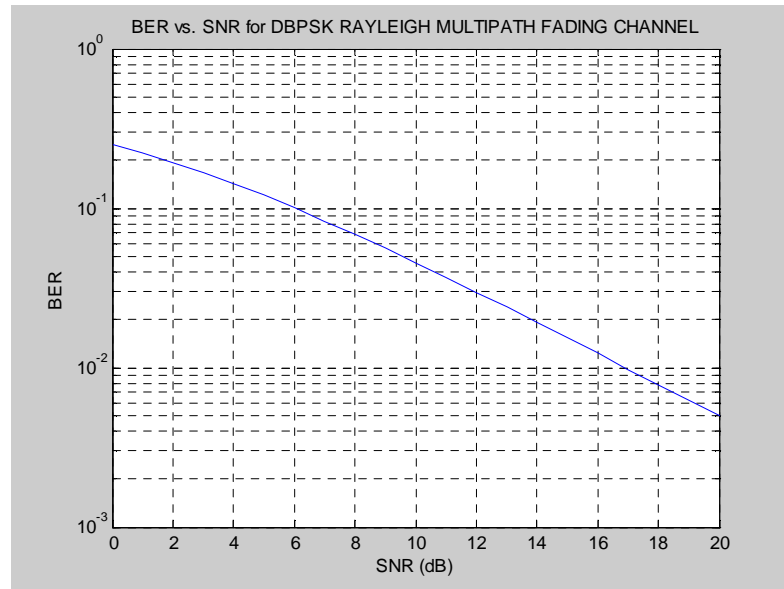
Figure 4-10: Diagram of BER vs. SNR for DBPSK Rayleigh Multipath Fading channel for fading variance σ^2=0.5.

4.7.3 EZW and SPIHT over AWGN Channel

Figure 4-11 and Figure 4-13 show the impact channel effects have on compressed bit streams employing EZW and SPIHT wavelet coding, transmitted across an AWGN channel. Both the

schemes track the theoretical AWGN DBPSK channel model trajectory. These results are determined prior to compression decoding and show the channel degradation. These results confirm the channels affect on the compressed bitstreams as each models the theoretical channel trajectory. Images of the channels degradation in image quality are displayed over a range of SNR's to illustrate which average SNR produces acceptable visual results.

From Figure 4-11 and Figure 4-13, both the wavelet schemes produce minimal errors at higher SNR's. This is due to the BER value equalling zero when the SNR is high signifying that no errors were generated at an average SNR for a specific sample set. Since the scale is logarithmic, a BER of zero cannot be plotted therefore the results end abruptly. Figure 4-11 depicts the EZW DBPSK demodulated data that has undergone AWGN channel effects.
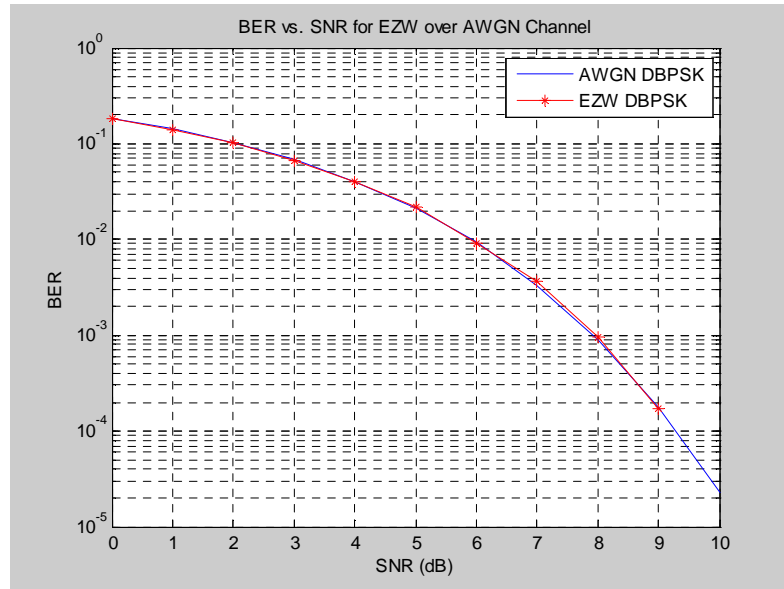
Figure 4-11: BER vs. SNR for EZW (DBPSK) over AWGN channel.

Once the demodulated data at an average SNR of *5dB*, *8dB* and *10dB* is decoded by the EZW wavelet coding scheme, the compressed *Lena* images (256x256) [85] in Figure 4-12 is produced. The transmitted bitstream is free of errors (BER of zero) for *Lena* at an average SNR of *10dB* with a PSNR image quality of *26.23dB* at a bitrate of *0.3bpp*. This is sufficiently excellent performance in terms of image quality. The image is still visually clear, indicating the modest noise effect of the channel at the applied SNR. At an average SNR of *5dB* the channel exhibits degradation causing errors in the decompressed image.

Figure 4-12: EZW (DBPSK) compressed image transmitted over AWGN channel for SNR of *5dB*, *8dB* and *10dB* with BER of *0.0219*, *0.0009* and *0*.

Figure 4-13 shows the BER vs. SNR for the SPIHT scheme. The same channel conditions used in the EZW scheme were applied to the SPIHT scheme. It exhibited similar bit error rate distortion as the EZW wavelet coding scheme but greater visual distortion across the SNR range.
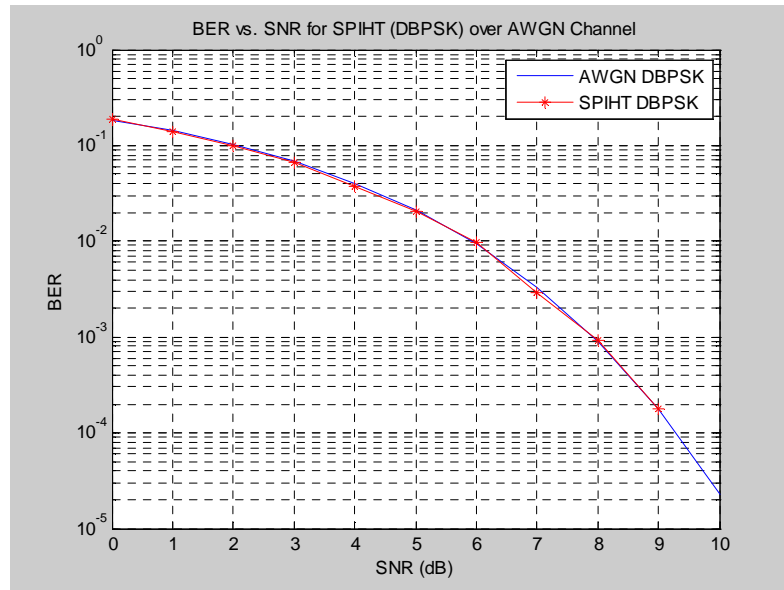
Figure 4-13: BER vs. SNR for SPIHT (DBPSK) over AWGN channel.

Figure 4-14 is representative of the SPIHT compressed *Lena* images (256x256) [85] via the AWGN channel. The BER is zero at an SNR of *10dB* producing a PSNR image quality of *29.85dB* at a bitrate of *0.43bpp*. This image quality metric is higher than that produced by the EZW scheme however, at lower average SNR's the visual quality is more degraded than the EZW scheme. The compression (bitrate) is the same and the number of errors (BER) produced are the similar however, more catastrophic errors are seen in SPIHT images than EZW. This is due to the greater inter-pixel correlation found in the SPIHT algorithm than the less-correlated

pixels in the EZW algorithm. This higher inter-pixel correlation combined with the destructive channel effects cause greater error propagation in the decompression stage of the SPIHT algorithm which thereby produces greater visual image distortion.

Figure 4-14: SPIHT (DBPSK) compressed image transmitted over AWGN channel for SNR of *5dB*, *8dB* and *10dB* with BER of *0.0206*, *0.0009* and *0*.

The results indicate that irrespective of the channel induced errors to the compressed bitstreams, the EZW and SPIHT algorithms are robust enough to decompress the picture to produce a visually acceptable image at medium SNR values. These results provide analysis into the channel errors produced and the effective error resilience integrated into the wavelet coding schemes.

4.7.4 EZW and SPIHT over Rayleigh Multipath Fading Channel

The Rayleigh fading channel used for the channel model is a frequency-selective fading channel with a sampling period of 0.1μ seconds. The Rayleigh fading channel as mentioned previously is based on the Doppler spread. The rate at which the channel fades is affected by the relative motion of the transmitter and receiver as in mobile communication. This motion causes Doppler shifts as the receive antenna which is in motion, experiences shifts in frequency that is dependent on the angle of arrival of the incoming signal as well as the speed of motion. The rate at which the channel fades is affected by the Doppler shifts. The Rayleigh multipath fading channels used in the following figures applied a maximum Doppler shift of 130Hz which created faster more aggressive fading. A fading variance $\sigma^2=0.5$ was used to simulate the average SNR range.

Once again both the EZW and SPIHT wavelet schemes transmitted across the Rayleigh multipath fading channel followed the same trajectory as the theoretical Rayleigh multipath fading channel model, highlighting the BER performance of the channels under wavelet compression. The BER is considerably higher for the SNR's as compared to the AWGN channel thus a wider SNR range is needed to produce image with acceptable image quality and

minimal distortion. This is due to the Rayleigh multipath fading channel representing a "worst case" scenario by exhibiting all the fading properties associated with the channel.

Figure 4-15 is the performance of BER vs. SNR for the EZW (DBPSK) scheme transmitted across the Rayleigh multipath fading channel. The Rayleigh channel exhibits erroneous properties in terms of fading, interference, diversity, nonlinearity etc. throughout the range of SNR averages. It is one of the most aggressively error prone channels, thus the EZW decoder is not able to produce a bitstream free of errors even a relatively high SNR's. The Rayleigh fading channel produces significant errors throughout the bitrate spectrum.
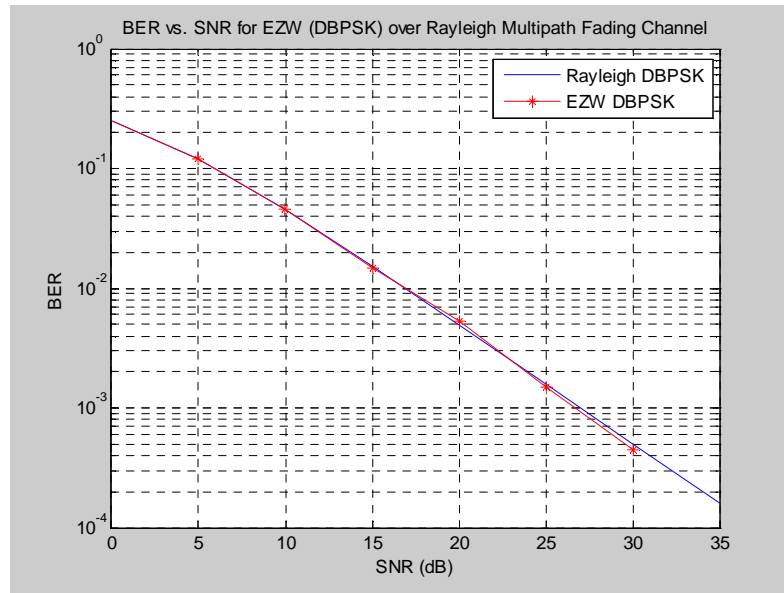
Figure 4-15: BER vs. SNR for EZW (DBPSK) over Rayleigh multipath fading channel for fading variance $\sigma^2=0.5$.

Figure 4-16 is the *Lena* images subjected to EZW wavelet coding, DBPSK modulation as well as the Rayleigh fading channel. The transmitted sequence free of channel errors or degradation occurs at an average SNR of *35dB* and produces an image with an image quality PSNR of *26.23dB* at a bitrate of *0.4bpp*. This image has the same image quality as the AWGN but occurs at a SNR of *3.5* times higher than the AWGN channel. This indicates the extent of the image degradation involved in the Rayleigh channel as compared to a noisy AWGN channel. The image quality of a *10dB* image in the Rayleigh fading channel is significantly visually distorted as compared to the *10dB* image produced by the AWGN channel in Figure 4-12.

Figure 4-16: EZW (DBPSK) compressed image transmitted across Rayleigh multipath fading channel for SNR of 10dB, 20dB and 35dB with BER of 0.0454, 0.0053 and 0 for fading variance σ^2=0.5.

Figure 4-17 is the SPIHT wavelet coding algorithm subjected to the fading effects of the Rayleigh fading channel. The SPIHT algorithm decays in the same manner as the EZW algorithm. Figure 4-17 illustrates the behaviour of the channel on the transmitted signal and the degree of degradation.
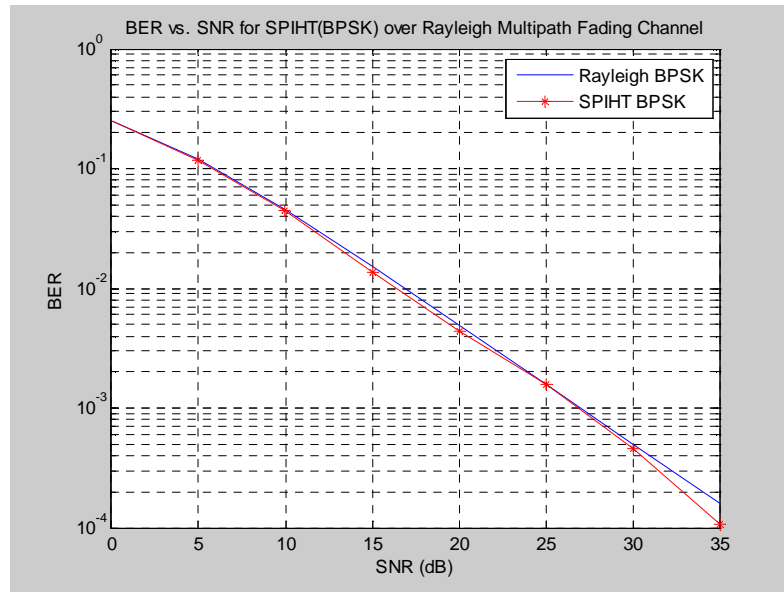
Figure 4-17: BER vs. SNR for SPIHT (DBPSK) over Rayleigh multipath fading channel for fading variance σ^2=0.5.

Figure 4-18 is the SPIHT compressed image after being transmitted through the Rayleigh multipath fading channel. Although the BER curve of the SPIHT algorithm behaves similarly to the EZW BER curve, the decoded images at low SNR are totally degraded. A BER of zero is seen for an average SNR of *35dB* having a PSNR image quality of *27.53dB* at a bitrate of *0.43bpp*. This which is higher than the EZW compressed image in Figure 4-16 and is more sharp whereas the EZW compressed image tends to look more blurry. At lower SNR's greater visual destruction is noticed. This is due to error propagation of the inter-pixel correlation found

in SPIHT than the EZW algorithm. The channel errors produced by the Rayleigh channel have caused irreversible error propagation in the decoded bit stream at *10dB* and *20dB*.
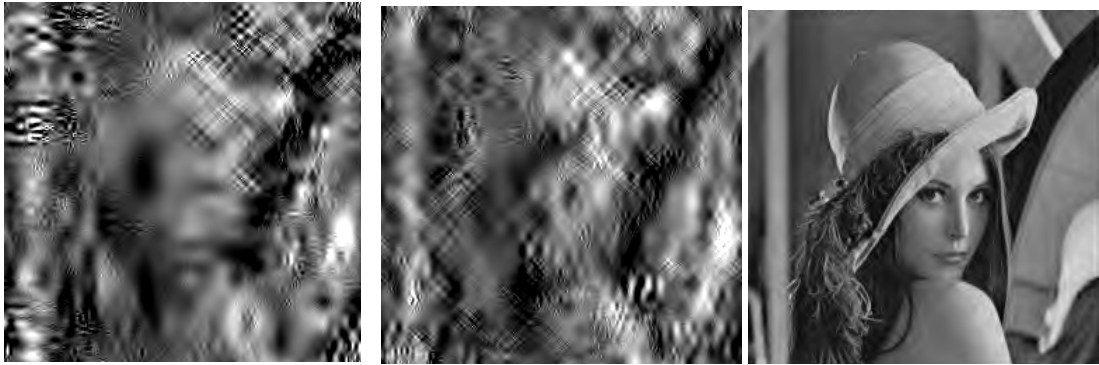
Figure 4-18: SPIHT (DBPSK) compressed image transmitted across Rayleigh multipath fading channel for SNR of *10dB*, *20dB* and *35dB* with BER of *0.0444*, *0.0044* and *0* for fading variance $\sigma^2=0.5$.

Due to the impairments experienced in wireless channels, namely fading, noise, interference, dispersion, nonlinearity, multipath transmission etc, images compressed by the EZW and SPIHT wavelet coding schemes are easily affected by these severe wireless channel conditions in the form of channel errors at lower SNR's. The above performance results are an indication that these wavelet coding schemes are highly susceptible to channel errors and error propagation.

4.8 SUMMARY

This chapter investigated the various types of channels that are involved in wireless communication, focusing specifically on the additive white Gaussian noise (AWGN) channel and the Rayleigh multipath fading channel. The two channels represent the best and worst case scenarios respectively thereby providing a comprehensive view of the channel impairments experienced through these channels.

A thorough performance evaluation of these channels and their effects on the EZW and SPIHT wavelet coding schemes was presented. The EZW and SPIHT demonstrated similar rate distortion when transmitted via the AWGN and Rayleigh fading channels. The image quality in terms of PSNR was typically high for the AWGN channel however, the image quality in the Rayleigh fading channel dropped significantly with both the wavelet coding schemes. The SPIHT compressed images was totally degraded at low SNR's due to the catastrophic error propagation caused by the channel impairments.

CHAPTER 5 -   ERROR PROTECTION

Error protection is a method of providing reliable data transmission over unreliable erroneous channels. Error protection is a dual error detection and error correction approach that is a vital addition in the prevention of transmission errors within images and video. Error resilient multimedia communication exploits error detection and correction in order to maintain the integrity of transmitted data and ensures that the data remains intact when transferred from source to destination across noisy channels.

Robust wavelet image compression is achieved through a concatenated system involving both error detection and correction. An error resilience tool involving arithmetic coding (AC) with forbidden symbol (FS), convolutional coding (CC) with *maximum a posteriori* (MAP) metric sequential decoding and automatic repeat request (ARQ) is used in the reliable transmission of images over noisy corrupt channels. Error detection is achieved through arithmetic coding with forbidden symbol whilst error correction exploits concatenated convolutional coding and MAP sequential decoding with ARQ based packet retransmission for uncorrected packets. This new method of error detection and correction will provide continuous error protection throughout the compression and decompression stages.

5.1   ERROR DETECTION USING ARITHMETIC CODING WITH FORBIDDEN SYMBOL

Error detection is the ability to detect and confine errors that have been produced by noise or channel impairments during transmission. It is used to determine whether the transmitted data has been corrupted. Error detection is the initial procedure within error protection and thus precedes error correction.

The error detection mechanism used in the proposed codec exploits arithmetic coding with forbidden symbol. This technique was proposed by Boyd et al [51] and is able to provide effective error detection within wavelet based image compression.

5.1.1   Arithmetic Coding

Arithmetic coding [1], [52], [53], [54], [55], [56] is a form of entropy coding that produces nearly optimal data compression. Arithmetic coding is a method of statistical lossless coding, as

it encodes source symbols with any given probability distribution and maps it to a code word, which is the binary equivalent of a real number with a probability distribution that lies in the interval of *0* to *1*. Arithmetic coding uses a probability source model to estimate the probability of a source symbol at each point within the data and it is able to achieve compression when it confirms that some source symbols are more likely than others. In essence the probability source model provides a probability distribution of the source symbols and the arithmetic encoder provides compression by transmitting more probable source symbols in fewer bits than less probable symbols. Thus a good statistical model of the source can produce maximum data compression.

Arithmetic coding uses recursive partitioning of the interval [0,1) to encode a message containing source symbols into a real number within the interval [0,1). The partitioning of the interval is based on the source symbol probability distribution as the length of the interval partition is proportional to the probability of the source symbol. The output real number is the refinement of the interval into a unique result that accurately represents the message.

The arithmetic coding algorithm begins by considering an alphabet A={S_1, S_2,…, S_n} of *n* source symbols where S_i are the source symbols and each source symbol has a probability of occurrence p_i ={p_1, p_2,…, p_n} such that $\sum p_i = 1$ [81], [82]. Each symbol with a probability p_i, can be uniquely represented by its own non-overlapping probability range along a probability line, *0* to *1*. The probability range assignment is represented by (5.1) [82];

$$\left[low\_range(\,S_i\,), high\_range(\,S_i\,) \right) = \left[0, p_i \right) \qquad \text{for } k = 1, \text{ and}$$

$$\left[low\_range(\,S_i\,), high\_range(\,S_i\,) \right) = \left[\sum_{k=1}^{i-1} p_k, \sum_{k=1}^{i-1} p_k + p_i \right) \qquad (5.1)$$

$$= \left[\sum_{k=1}^{i-1} p_k, \sum_{k=1}^{i} p_k \right) \qquad \text{for } k > 1,$$

where *k* is the kth source symbol in the alphabet and *i* is the current source symbol. Assigning each source symbol its own probability range ensures that each source symbol can be arithmetically encoded by its range.

Conceptually, the arithmetic coding algorithm operates as follows, where each source symbol encoded is regarded as an event [52], [53], [56]:

- Initialise the current interval [L, H) to [0, 1) where *L* is the lower bound and *H* is the upper bound of the interval.
- For each source symbol in the alphabet:
  - Subdivide the current interval into subintervals according to the probability of each source symbol.
  - Select the subinterval corresponding to the current source symbol and make it the new current interval.
- Output enough bits to distinguish the final current interval from all other intervals.

The length of the final subinterval is equivalent to the product of the probabilities of the sequence of source symbols encoded in the message. The final subinterval is not required in the decoding routine however; a real number within the final subinterval is used as the final encoded value for further decoding. As additional symbols are added to the message, the precision requirements in the final encoded real number used to represent it, increases.

The graphical representation of the arithmetic coding interval subdivision is illustrated in Figure 5-1 [52], [53], [56]. The interval [L, H) is divided into nested intervals according to the probability of the source symbols and a new current interval is chosen based on the source symbol to be encoded.
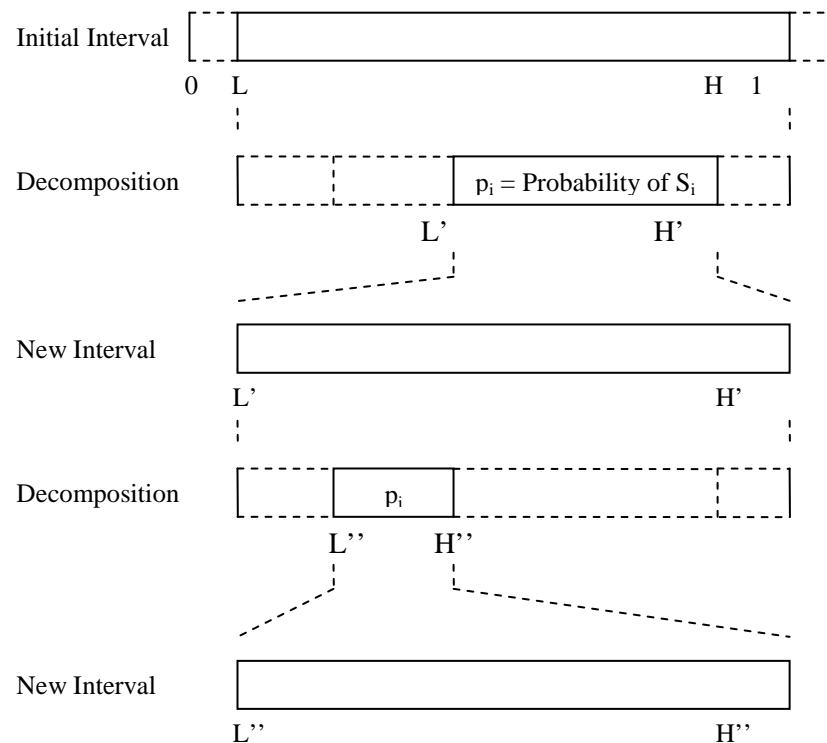
Figure 5-1: Diagram of Arithmetic Coding interval subdivision.

Figure 5-2 and Figure 5-3 [52], are fragments of pseudo code for the encoding and decoding procedures which help facilitate a deeper understanding of arithmetic coding in its entirety. The encoding process begins with an initialisation step where the source symbol probability distribution is determined and the boundaries of the current interval are established. The encoder then recursively partitions the interval into subintervals. This refines the interval to a unique real number that represents the message.

BEGIN

    Get probabilities for each symbol

    Set LOW = 0

    Set HIGH = 1

    WHILE *there are symbols* **DO**

    {

        Get symbol

        RANGE = HIGH - LOW

        HIGH=LOW + RANGE*(Upper bound of symbol)

        LOW=LOW + RANGE*(Lower bound of symbol)

    }

    Output = LOW

END

Figure 5-2: Algorithm of Arithmetic Coding Encoder.

The decoding procedure is the inverse of the encoding procedure whereby the range is expanded in proportion to the probabilities of the source symbols as it is extracted. It begins with the encoded value and recursively outputs the intended symbols. The received encoded value is a binary representation of the floating point number which requires exact mathematical precision.

BEGIN

    Get encoded number

    WHILE *there are symbols* **DO**

    {

        Find SYMBOL whose range straddles the encoded number

        Output SYMBOL

        RANGE = (Upper bound of symbol) – (Lower bound of symbol)

        New Encoded Number = ((Old Encoded Number) – (Lower bound of symbol))/RANGE

    }

END

Figure 5-3: Algorithm of Arithmetic Coding Decoder.

The encoding algorithm encodes the entire message first, before transmission. The same applies for the decoding process; it only begins decoding once the encoded value has been received. Thus the precision of the arithmetic coder deteriorates as the message length increases, as more bits in the output real number is used to represent it. Consequently, if the encoded output real number is inaccurate when received at the decoder, an incorrect decoded message will result. Arithmetic coding theoretically requires infinite precision or else the interval boundaries converge, impacting on the implementation. These precision inaccuracies initiated the desire to use integer arithmetic coding where encoded bits are transmitted once the symbol has been processed producing a more adaptive model.

5.1.2 Integer Arithmetic Coding

Witten, Neal and Cleary [52] developed integer arithmetic coding in order to overcome the infinite precision arithmetic employed in pure arithmetic coding, as well as produce greater practical efficiency for arithmetic coding. Infinite precision arithmetic reduces the current interval considerably and does not produce an output until the entire message has been encoded. Integer arithmetic coding resolves this by first replacing the real interval [0, 1) with an integer interval [0, T) where $T = 2^P$, $P \geq 2$ and P is defined as the bit size or bit precision of the initial interval. It then attempts to output an encoded bit as soon as it is known, followed by renormalisation through doubling of the length of the current interval which prevents the current interval from shrinking too much.

Integer arithmetic coding [52], [53], [56] uses a large initial interval with integer interval boundaries, where the partitioning and selection of the current interval is performed for each symbol encoded. In addition to the interval selection, an output bit is generated for each symbol encoded.

The interval selection and expansion process is detailed as follows, where interval segments of 2^{P-1}, 2^{P-2} and $3*2^{P-2}$ constitute a half, a quarter and three quarter segments of the interval respectively [56], [58].

1. Initialise the current interval [L, H) to [0, $T=2^P$), where L is the lower bound and H is the upper bound of the interval.
2. For each source symbol in the alphabet:
   a. Expand the current interval into a new interval according to the following conditions [58]:
      i.   If $H < 2^{P-1}$, L and H are doubled.
      ii. If $L \geq 2^{P-1}$, L and H are doubled after subtracting 2^{P-1}.

iii. If $L \geq 2^{P-2}$ and $H < 3*2^{P-2}$, L and H are doubled after subtracting 2^{P-2}.

Graphically the interval expansion process for integer arithmetic coding described above is illustrated in Figure 5-4 [53], [56]. The current interval is selected based on the current symbol encoded and the new interval is determined in accordance with the criteria described in the interval expansion process. The length of the current interval is proportional to the current symbol occurrence and is used to encode the current symbol. The new interval represents the renormalisation process that occurs in order to prevent the shrinking of the current interval.
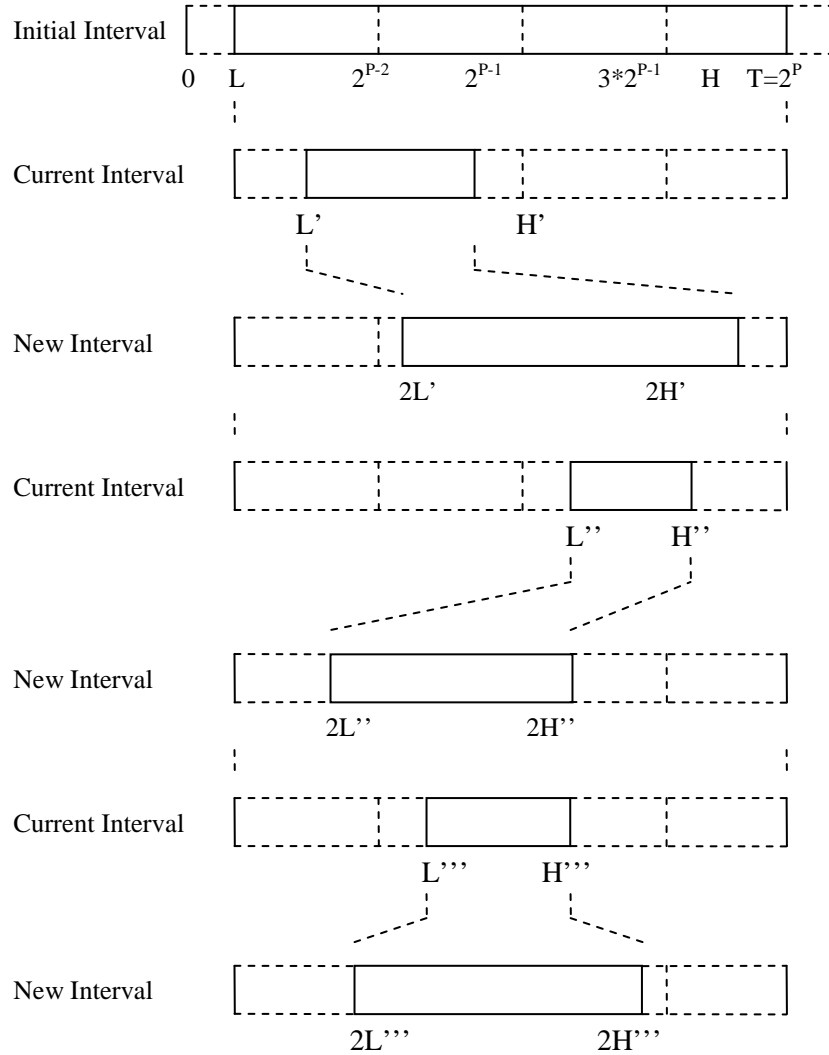
Figure 5-4: Diagram of Interval Expansion process in Integer Arithmetic Coding.

In addition to the interval expansion process, an output bit is required for the encoding and transmission of each symbol. An output bit of either *0* or *1* is generated based on where the current interval is situated, relative to the location of the quarter, half and three quarter interval segments. If the current interval is entirely in the upper half of the interval, a *1* is output.

Likewise if the current interval lies entirely in the lower half, a *0* is output. A *follow-on* procedure [53], [56] is executed if the current interval straddles the half interval segment, as it is impossible to determine which bit the interval corresponds to.

The *follow-on* procedure prevents the current interval from converging about the interval midpoint. It keeps track of the number of times the current interval straddles the midpoint, and each time attempts to renormalize the length of the interval by expanding it, in order to determine which region the interval belongs to. If the current interval enters either the lower half or upper half of the interval region after expansion, then it can be assumed that the interval was situated in the opposite interval region previously. This output bit generation with *follow-on* procedure is described as follows [53], [56]:

1. If the current interval lies entirely within $[2^{P-1}, 3*2^{P-1})$,
   a. No output bit is generated.
   b. A *follow* count is incremented to keep track of future outputs.
   c. Apply interval expansion.
2. If the new current interval lies entirely within $[0, 2^{P-1})$, the lower half,
   a. Output a *0* bit.
   b. Output *follow* bits of *1* from previous events.
   c. Apply interval expansion.
3. If the new current interval lies entirely within $[2^{P-1}, 2^{P})$, the upper half,
   a. Output a *1* bit.
   b. Output *follow* bits of *0* from previous events.
   c. Apply interval expansion.
4. If the new current interval does not lie entirely within one of the intervals: $[0, 2^{P-1})$ or $[2^{P-1}, 3*2^{P-1})$ or $[2^{P-1}, 2^{P})$,
   a. No output bit is generated.
   b. Exit Loop and return.

In order to complete the transmission of output bits, the encoder is *flushed* before the end of the transmission sequence. The flushing of the encoder ensures that the decoder is able to unambiguously decode the last symbol in the message. It involves outputting a few additional bits to guarantee that the decoded sequence falls within the final range.

The combination of interval expansion, output bit generation and flushing of the encoder fundamentally produces integer arithmetic coding. Thus the method of integer arithmetic coding can be conceptually detailed in the fragment of pseudo code given in Figure 5-5 [52].

BEGIN

    Initialise the current interval [L, H] to [0, T=2^P-1].

    WHILE *there are symbols* **DO**

    {

        IF (L ≥ 2^{P-2}) & (H < 3*2^{P-2}) **THEN**

            L=2*L

              H=2*H

            Follow = Follow +1

        IF H < 2^{P-1} **THEN**

            Output a bit 0

            L=2*L

              H=2*H

            FOR 1 to *Follow* **DO**

                Output a bit 1

        IF L ≥ 2^{P-1} **THEN**

            Output a bit 1

            L=2*(L-2^{P-1})

              H=2*(H-2^{P-1})

            FOR 1 to *Follow* **DO**

                Output a bit 0

         ELSE

            Return

    }

    Flush Encoder

END

Figure 5-5: Algorithm of Integer Arithmetic Coding Encoder.

The integer arithmetic coding decoder recovers the source symbols in a similar manner as the encoder. The decoder applies interval reduction, the inverse process used at the encoder. Interval reduction follows the same set of actions around the quarter, half and three quarter interval segments as defined at the encoder. This ensures that the selection and renormalisation steps occur in the same way for both the encoder and decoder.

The decoder begins by decoding the first *P* bits in the received bit stream. These bits are initially stored in a sliding buffer of size *P* bits. The current interval [L, H) is initialised to [0, T=2^P) exactly as the encoder. The decoder then performs the selection and renormalisation

113

procedures before shifting the next transmitted bit into the buffer. The selection of the current interval is based on the buffer of bits which is used to decode the source symbol that represents that specific portion of the interval. Thus the decoder is the inverse process of the encoder.

Integer arithmetic coding solves many of the precision and practical issues surrounding pure arithmetic coding. It has developed into a viable entropy coding option as the demand for the entropy coding stage of arithmetic coding increased within data compression.

5.1.3 Arithmetic Coding with Forbidden Symbol

Arithmetic coding has gained wide recognition as an optimal entropy coding stage for data compression. It exhibits excellent compression efficiency and superior performance when compared to other data compression algorithms like the Huffman encoder. However, arithmetic coding is extremely vulnerable to transmission errors, as a single bit error can cause catastrophic error propagation and loss of synchronisation. This can result in the rest of the bit stream being decoded erroneously. Ironically, it was this vulnerability that led to the development of error resilient tools for arithmetic coders.

Boyd *et al.* [51] developed an effective technique for incorporating error detection into arithmetic coding without compromising its performance. An extra source symbol, the forbidden symbol is introduced into the source symbol alphabet and a small probability is assigned to it. The forbidden symbol is added to the source symbol set but is never included in the message to be encoded by the arithmetic coder and is therefore never transmitted. However, if the forbidden symbol is decoded by the arithmetic decoder, this is an indication that an error has occurred.

The forbidden symbol given by *X*, is assigned a small probability of occurrence equal to epsilon, ε, specified as $p_X=\varepsilon$. Incorporating integer arithmetic coding with forbidden symbol detection [51], [54], [83], [84] produces the alphabet A={S_1, S_2,..., S_n, X} where S_i are the source symbols and *X* is the forbidden symbol and each source symbol has a probability of occurrence p_i ={p_1, p_2,..., p_n} and the forbidden symbol has a probability of occurrence equal to ε such that $\sum(p_i+\varepsilon)=T$ which is equal to 2^P. Thus the initial interval for integer arithmetic coding which includes the forbidden symbol will span the length from *0* to *T=2^P*.

Graphically the arithmetic coding with forbidden symbol encoding process is illustrated in Figure 5-6 [54]. The forbidden symbol is included in the alphabet distribution but is never

encoded, thus its interval subdivision is never selected as a new interval and therefore never partitioned.
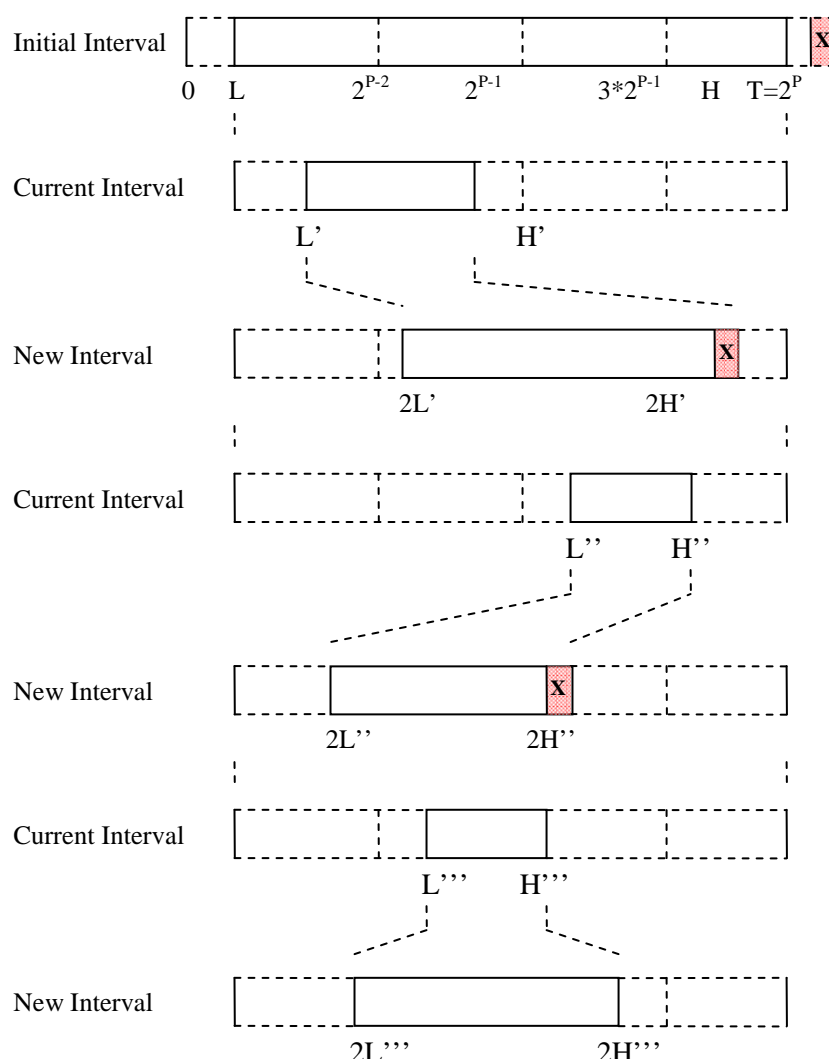
Figure 5-6: Diagram of Interval Expansion process in Integer Arithmetic Coding with Forbidden Symbol.

Arithmetic coders are known for their sensitivity to transmission errors and it is this weakness that is exploited at the decoder for error detection. When an error occurs in an arithmetically coded transmitted bit stream, a loss of synchronization occurs at the decoder resulting in the remainder of the bit stream being decoded erroneously. The introduction of the forbidden symbol at the encoder can guarantee that when an error occurs, the forbidden symbol will be decoded by the arithmetic decoder. The forbidden symbol is never encoded by the arithmetic coder however, if it is decoded, the encoded bit stream and the decoded bit stream are not identical and thus an error has occurred.

The introduction of the forbidden symbol in the symbol alphabet introduces an amount of redundancy [51] based on the probability, ε, associated with the forbidden symbol. Increasing the forbidden symbol probability thereby increases the amount of redundancy embedded in the code stream. Consequently, the greater the amount of redundancy added through the forbidden symbol, the less time it takes to decode the error. However, the redundancy introduced occurs at the expense of compression efficiency. This is proved in [54] where the bits needed to represent the symbol subinterval width γ, is $-\log_2(\gamma)$. By introducing the forbidden symbol the number of bits needed to represent the symbol subinterval width is then given by $-\log_2[(T-\varepsilon)\gamma]$. As a result, the forbidden symbol introduces more bits per symbol encoded and the redundancy R_X, added due to the forbidden symbol becomes [54];

$$
\begin{aligned}
R_X &= \left(-log_2\left[(T-\varepsilon)\gamma\right]\right)-\left(-log_2\gamma\right) \\
&= -log_2(T-\varepsilon),
\end{aligned}
\tag{5.2}
$$

where T is the full interval, γ is the symbol subinterval width and ε is the forbidden symbol interval. Thus the more redundancy added by the forbidden symbol for error detection, the less compression the arithmetic coder can perform.

A significant property of arithmetic coding with forbidden symbol error detection technique is that it is able to provide continuous error detection throughout the compression and decompression stages. The advantage of continuous error detection is that it need not wait for an entire bitstream to be transmitted before an error can be detected, it can determine as each bit is transmitted whether the bitstream is potentially in error. It cannot however, determine exactly where an error has occurred within the bitstream.

Arithmetic coding with forbidden symbol is a technique that is able to successfully integrate error detection with compression via entropy coding. However, there exists a trade off between compression efficiency and the amount of redundancy introduced by the forbidden symbol. The optimal point between the two conditions can be established through the manipulation the forbidden symbol probability, ε.

5.2 ERROR CORRECTION USING *MAXIMUM A POSTERIORI* (MAP) METRIC SEQUENTIAL DECODING AND AUTOMATIC REPEAT REQUEST (ARQ) RETRANSMISSION

Error correction is designed to correct and rectify errors produced during corrupt data transmission. It is more complex than error detection as it uses redundant information to produce an "educated guess" about the original data from the incorrect data received. When coupled with error detection, error correction has the ability to effectively identify and correct errors produced within erroneous transmission.

The error correction scheme employed by the proposed coder makes use of MAP metric sequential decoding of convolutional codes as well as the ARQ retransmission protocol. The MAP metric sequential decoder is used as an error correction tool to obtain the best estimate of the transmitted bitstream in the presence of errors. The ARQ protocol is an error control transmission protocol and is used to request for data retransmission if the MAP decoder does not successfully correct the errors.

5.2.1 Sequential Decoding

Sequential decoding initially proposed by Wozencraft [61] was designed to decode convolutional codes. It attempted to reconstruct the original transmitted sequence by guessing through various paths of a time-expanding tree of possible transmitted sequences. It was later improved by Fano [62] and became known as the Fano algorithm. Fano developed a metric called the Fano metric which was incorporated in the sequential decoding code tree to obtain the correct path based on the largest Fano metric accumulated by the path. Zigangirov [63] and Jelinek [64] independently proposed faster variations of the Fano algorithm using a Stack algorithm approach. This new refined algorithm replaced the code tree structure with a stack, where the branch with the largest Fano metric was extended producing the optimal path.

Sequential decoding is essentially a tree search algorithm that is able to accurately locate the correct path within a code tree which corresponds to the encoded message transmitted. Sequential decoding employs various search strategies for its path selection namely: breadth-first, metric-first or depth-first searches [65]. The sequential decoding algorithm proposed relies explicitly on a metric for directing its search through the code tree. The concept behind sequential decoding is to search paths on a branch by branch basis without having to explore too many branches. The path selection is based on the best accumulated metric obtained along the path. The sequential decoder retains a running metric that is designed to increase along the correct path and decrease along false paths. Sequential decoding allows both forward and backward movement through the code tree, where the search algorithm may give up a path, retract back and follow another path.

The sequential decoder in the proposed coder uses a metric-first search, where the best path selection is based on a greedy approach as it extends branches with the best accumulated metric. The decoder is implemented by means of the optimal Stack algorithm as it offers simple yet fast sequential decoding.

5.2.1.1 Stack Algorithm

The stack algorithm is a metric-first sequential decoding algorithm implemented for the proposed decoder [64]. The stack algorithm uses a stack or ordered list to store all visited paths according to its metric values. The metric value of the paths increase towards the top of the stack, thus indicating the best path location based on the accumulated metric occurs at the top of the stack. However, this structure requires sorting of the paths within the stack at each iteration.

The stack algorithm is based on the code tree structure where the code tree is representative of all possible paths. The code tree uses nodes, associated branches and branch metrics to determine the correct path for sequential decoding. The stack algorithm begins at the root node and only extends node branches with the highest metric. For a code rate of $^k/_n$, each node in the tree extends 2^k branches with each iteration. Each branch is associated with a metric, where false branches have a lower metric and correct branches have a higher metric. Each branch yields a new node which in turn creates more branches and so the decoding continues. Each new extended branch adds its branch metric to the metric of previous branches, thus maintaining a running metric of the path. This code tree concept is illustrated in Figure 5-7, where *BM* is the branch metric and a correct branch has a branch metric equal to *0* and an incorrect branch is *-20*. The correct path is highlighted in Figure 5-7 where the accumulated metric at the final node is the sum of the branch metrics along the traversed path.

Figure 5-7: Diagram of a code tree with nodes, branches, metrics and paths.

The stack algorithm is described below; where the path at the top of the stack is replaced by extended branches [64], [69]. The stack algorithm is described for a code rate of ½, which is the rate designed for the convolutional coding of the proposed coder. This code rate will produce two branches per node extension. The algorithm is briefly outlined as follows [65]:

1. Initialise the stack with the root node and set the metric to zero.
2. Expand the best path, located at the top of the stack by creating two new branches.
3. Replace the best path with the two newly created branches in the stack along with their corresponding metric.
4. Sort the paths in the stack according to the metric, with the highest metric at the top and the lowest at the bottom.
5. Retain the best path at each decoding stage and transfer its information in terms of output bits.
6. If the top path reaches maximum depth of the decoding tree:
   a. Stop iterating.
7. Else loop to step 2.

Graphically the stack algorithm can be illustrated in terms of the stack structure when applied to convolutional coding of a rate of ½ in Figure 5-9. Each path in the stack contains information about the accumulated metric and the state information for sequential decoding. Figure 5-8 is

119

the state information associated with the convolutional codes which is used in the sequential decoding process. The convolutional codes depict the current state, the transition to the next state and the output bits produced by the transition between states. The solid arrow is representative of an input bit *0* and the dashed arrow shows an input bit of *1*.
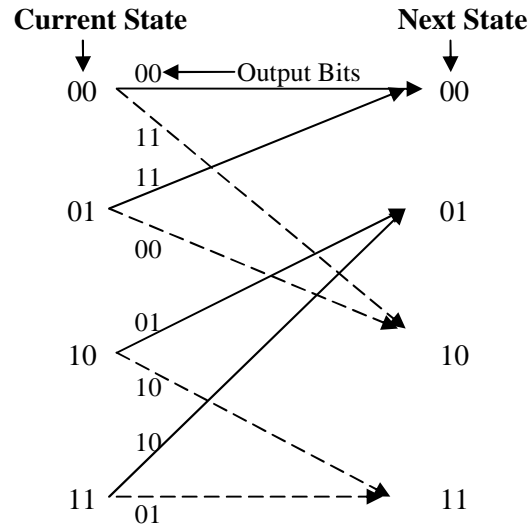
Figure 5-8: Diagram of Convolutional Coding state transition and output bits.

Root Node	Node 2		Node 3		Node 4	
00 (0) [0] *	00 (-20) [1]	11 (0) [1] *	01 (-20) [2]	10 (0) [2] *	10 (-20) [3]	01 (0) [3] *
	11 (0) [1]	00 (-20) [1]	10 (0) [2]	01 (-20) [2]	01(0) [3]	10(-20) [3]
			00 (-20) [1]	00 (-20) [1]	01 (-20) [2]	01(-20) [2]
					00 (-20) [1]	00(-20) [1]

Figure 5-9: Diagram of the Stack structure.

Each column in Figure 5-9 above is representative of the various decoding stages for the stack. The stack algorithm involves two distinct modes of operation: best path expansion and the sorting of the stack. The stack structure begins with the root node or node 1 and a single path with its state information initialized to zero. The path at the top of the stack with the highest metric is extended into two new branches at the next node. The path with an asterisk is the best path to be expanded in the stack. The red highlighted stack columns indicate the resorting of the stack with the highest metric located at the top of the stack. The first two bits in each path represent the decoded output bits for sequential decoding of convolutional codes. The number within the parenthesis indicates the accumulated metric determined from each output bit. In this example if a bit is correct the bit metric is *0*, and if a bit is incorrect the bit metric is *-10*. The square brackets indicate the node at which the path is currently located, thus new branches point to different nodes. However, if there exists, two nodes in stack with the same metric values, the node closest to the top of the stack is extended first and the stack is then resorted. If this node is

incorrect it will result in the second node being located at the top of the stack which will then be extended.

At any decoding stage, branches that did not produce the optimal code path were not extended within the stack. Thus the sequential decoder is able to find the best path without examining too many branches. The stack algorithm only stores visited paths reducing the computation of the sequential decoder.

The stack algorithm establishes the best path traversed through the code tree by retaining the best path determined at each decoding stage. In addition the best path chosen by the sequential decoder is associated with output bits that represent the corrected bit stream. It is these sequential decoded output bits that are used to ascertain whether the erroneous transmitted bits have been accurately corrected by the sequential decoder. When compared to the originally transmitted bit stream, if the output bits are exactly identical, then the stack algorithm has correctly decoded the errors and has performed successfully. However, the success of the stack algorithm for sequential decoding can be solely attributed to the choice of metric applied, as it is of primary importance in the correct design of sequential decoding algorithms.

5.2.2 MAP Decoding Metric

The metric defined in sequential decoding is used to direct the best path selection within the code tree. It minimizes the error probability of explored sequences in sequential decoding. Thus the choice of metric is key in the performance of the sequential decoder. Essentially the metric chosen must measure the correlation between the input and output sequences in order for its decoding success. This correlation forms the basis of the metric derivation.

The sequential decoder is formulated using the classical *maximum a posteriori* (MAP) estimation [70]. The MAP criterion involves minimizing the probability of the error by maximising the *a posteriori* probability defined below. Its mathematical derivation is based on the statistical modelling of the transformation of the bitstream illustrated in the transmission block diagram in Figure 5-10 .

x —————→(CHANNEL)—————→ y

$P(y/x)$
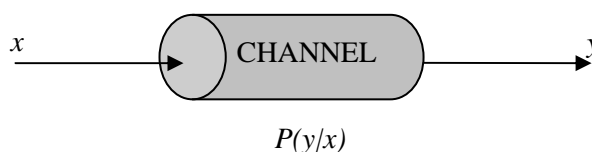
Figure 5-10: Diagram of transmission block diagram.

The *a posteriori* probability is defined as $P(x/y)$, which is the conditional probability of the transmitted sequence x given that the sequence y has been received. The probability of a correct received sequence is equal to the *a posteriori* probability given by;

$$P_C = P(x/y).$$ (5.3)

Thus the probability of an incorrect received sequence then becomes;

$$P_E = 1 - P(x/y).$$ (5.4)

This shows that the minimising of the error requires the maximising of the *a posteriori* probability thus producing the concept which is commonly known as the *maximum a posteriori* estimation. From Bayes theorem the *a posteriori* probability $P(x/y)$, can be further expressed as [72], [73];

$$P(x/y) = \frac{P(y/x)P(x)}{P(y)},$$ (5.5)

where, in terms of the transmission sequence description, $P(x)$ is the *a priori* probability, $P(y)$ is the probability of observing a certain sequence at the receiver and $P(y/x)$ is the channel transition probability depicted in Figure 5-10.

The *a priori* probability $P(x)$ by definition is a marginal probability that describes the probability of a certain hypothesis x, which is known before any data y, is observed. If a certain hypothesis is more probable than others, the *a priori* probability is then higher. In the context of the MAP estimation the *a priori* probability is the probability of the binary bit *1* or *0* in the transmitted sequence before the effect of errors. Essentially it represents a known probability determined at the encoder.

$P(y/x)$ is the likelihood function which describes the probability of the observed data y, assuming the hypothesis x, is correct. The probability $P(y/x)$, represents a certain probability density function (PDF) which accurately describes the measurement of errors within the received data y. In the context of the MAP estimation exercised in this dissertation the likelihood $P(y/x)$ can be described as the channel transition probability as it is a function of the modulation and channel characteristics.

The channel output probability $P(y)$ is the marginal distribution of the observed data y, which essentially serves as a normalising constant such that the *a posteriori* probability $P(y/x)$, can be described as a proper PDF integrating to 1. However, $P(y)$ is an analytically intractable integral and is commonly approximated.

The Bayesian Theorem is more of a general concept and is almost never used directly for estimation purposes due to its complexity. However, it can be optimally approximated thus producing what is commonly known as the *maximum a posteriori* (MAP) estimation. The MAP estimation using the Bayesian approach allows for the exploitation of the *a priori* statistical information for its estimation. The MAP estimation is given by equation (5.6) [71], [74];

$$
\begin{aligned}
\bar{x}_{MAP} &= arg\,\max_{x} P(x/y) \\
&= arg\,\max_{x} \frac{P(y/x)P(x)}{P(y)}.
\end{aligned}
\tag{5.6}
$$

It can further be described in a simpler form where the MAP estimation is decomposed into additive logarithmic terms given by equation (5.7) thereby forming the MAP decoding metric. To convert from the multiplicative to additive form, the monotonicity of the logarithm function is exploited. Thus the MAP decoding metric is [71] , [74];

$$
\begin{aligned}
\bar{x}_{MAP(i)} &= log[P(x_i/y)] \\
&= \sum_{j=0}^{N_i-1} [log\,P(y_j/x_{i,j}) + log\,P(x_{i,j}) - log\,P(y_j)] \\
&= \sum_{j=0}^{N_i-1} \bar{x}_{MAP(i,j)} \quad ,
\end{aligned}
\tag{5.7}
$$

where $\bar{x}_{MAP(i)}$ is the estimated sequenced, i is the number representation for the sequence transmitted, j is the individual bit in the sequence and N_i is the total length of bits in the sequence i. The metric of a path can be defined as the sum of the individual metrics of the branches of which the path consists. Therefore when (5.7) is applied to each branch in a code tree, the MAP decoding branch metric is defined as [71];

$$
\bar{x}_{MAP(i,j)} = log\,P(y_j/x_{(i,j)}) + log\,P(x_{(i,j)}) - log\,P(y_j).
\tag{5.8}
$$

Thus a branch can be extended based on its individual MAP branch metric.

The three MAP decoding metric terms in equation (5.8) have implicit definitions restricted to the context described within this dissertation. The first term in equation (5.8), *log P(y_j/x_(i,j))*, representing the channel transition probability is entirely dependent on the transmission channels employed in the codec design. The channel transition probabilities describe the behaviour of the channel and are generally fixed by the nature of the channels noise distribution. In addition the channel transition probability can be further classified into two decoding methods, hard decision decoding and soft decision decoding.

Hard decision decoding involves the received signal being classified as either a *0* or *1* prior to decoding [75]. A hard decision involves a simple decision threshold between the two binary signals such that if the received signal is greater than the threshold the signal is decoded as a *1* otherwise a *0*. Applying this concept to noisy channels, where the channel transition probability of an erroneous decision can be quantified by the shaded area in Figure 5-11.
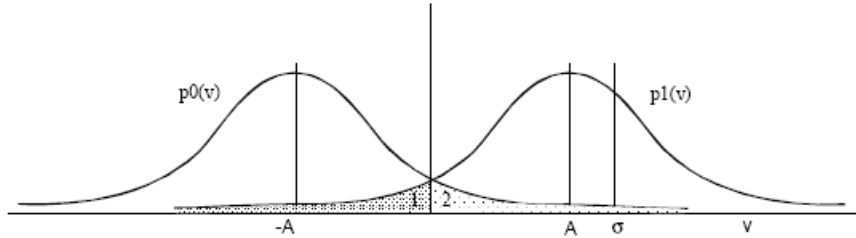
Figure 5-11: Diagram of Hard decision decoding with channel transition probabilities [75].

P(0/1) is the probability of a bit *0* being decoded given that a bit *1* was sent, can be defined as the shaded area *1* and *P(1/0)* is the probability of a bit *1* being decoded given that a bit *0* was sent, is denoted by the shaded area *2*. This shaded area representing an erroneous probability can be mathematically modelled as [71];

$$p_e = \frac{1}{2} erfc\left(\sqrt{\frac{S}{N}}\right).$$
(5.9)

The above equation is a known bit error rate equation where *erfc* is the complimentary error function and *S/N* is the Signal-to-Noise Ratio defined for the channel. When applied to the channel transition probability above, the hard decision channel transition probability is defined as [71];

$$P\left(y_j / x_{i,j}\right) = \begin{cases} 1 - p_e & \text{if } x_{i,j} = y_j \\ p_e & \text{if } x_{i,j} \neq y_j \end{cases}$$
(5.10)

The above decision is taken when the received bit is either in error (p_e) or not ($1-p_e$). The above decision also models a binary symmetric channel shown in Figure 5-12 where the input with respect to the output, transitions in exactly the same manner.
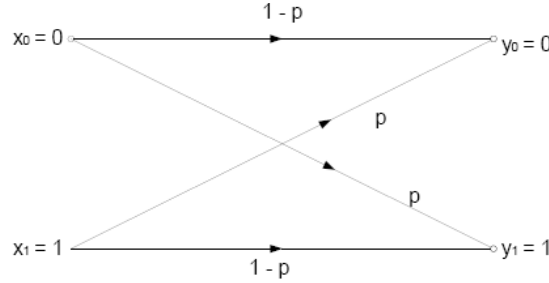
Figure 5-12: Diagram of Binary Symmetric Channel [70].

The hard decision decoding channel transition probability given in (5.10) is relevant for the two wireless channels examined, namely the AWGN and the Rayleigh fading channels. Soft decision decoding of the received bit uses additional side information to generate a decision [75]. Unlike the hard decision decoding where only a *0* or *1* is assigned, soft decision decoding is more flexible as it assigns confidence levels to the binary bits. These confidence levels indicate the degree of certainty that the decision is correct. Hard decision decoding uses two confidence levels, as the decision can either be correct or in error. Confidence levels of three and greater are termed soft decision decoding. Soft decision decoding side information constitutes "soft inputs" with associated "soft outputs", where soft inputs are the *a priori* probabilities and soft outputs are the *a posteriori* probabilities.

For the AWGN and the Rayleigh fading channels, their corresponding channel transition probabilities for soft decision decoding are given in (5.11) [71] and (5.12) which are based explicitly on their PDF's showing the transmitted and received sequences.

$$P\left(y_j \mid x_{i,j}\right)_{AWGN} = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\left(\frac{(y_j - x_{i,j})^2}{2\sigma^2}\right)}, \qquad (5.11)$$

$$P\left(y_j \mid x_{i,j}\right)_{RAYLEIGH} = \begin{cases} \dfrac{(y_j - x_{i,j})}{\sigma^2} e^{-\left(\frac{(y_j - x_{i,j})^2}{2\sigma^2}\right)} & \text{for } y_j \geq x_{i,j}, \\ 0 & \text{for } y_j < x_{i,j}. \end{cases} \qquad (5.12)$$

Essentially the channel transition probabilities use the statistically detectable structure of the source to determine the output.

The second term in equation (5.8), *log P(x(i,j))*, is the *a priori* probability of the binary bit *1* or *0* occurring in the received sequence [74]. The *a priori* probability is defined as P_0, the probability of occurrence of a bit *0* in the transmitted bitstream or P_1, the probability of occurrence of a bit *1*, given by the following equations;

$$P_0 = P(x_{i,j} = 0)$$
$$P_1 = P(x_{i,j} = 1) = 1 - P_0.$$

(5.13)

The final term in equation (5.8), *P(y_j)* representing the channel output probability of the received bit y_j, can be defined more explicitly as [74];

$$P(y_j) = \sum_{i \in B_{N_i}} P(y_j / x_{i,j}) P(x_{i,j}),$$

(5.14)

where B_{Ni} denotes the subset of all possible sequences of length N_i. This term is complex and difficult to evaluate as it involves an exhaustive search which becomes practically infeasible and is beyond the scope of this dissertation. Thus a reasonable approximation is used instead. $P(x_{i,j})$ adopts the approximation of 2^{-Ni} which suggests that there are 2^{Ni} equally likely possible sequences y_i, of length N_i [73], [74], [71], [72]. The approximation in theory is invalid and untrue as the arithmetically coded sequences $x_{i,j}$ are of variable length, however, the assumption is still able to provide satisfactory results for the MAP decoding and can still be used in the metric.

Incorporating the above approximation into the MAP decoding branch metric in (5.8) the metric simplifies to [71];

$$\overline{x}_{MAP(i,j)} = log\, P(y_j / x_{(i,j)}) + log\, P(x_{(i,j)}) + log\, 2.$$

(5.15)

The simplified logarithmic MAP metric still maintains its dependence on channel conditions and error prediction. This metric is applied to each branch in the code tree through the stack algorithm so as to direct the metric-first search with absolute accuracy thereby leading to the overall error correcting capability of the MAP decoding algorithm.

5.2.3   ARQ Retransmission

Automatic repeat request (ARQ) [54], [92], [93], [97] is a communications protocol that is used as an error control mechanism for efficient data transmission. The purpose of ARQ is once transmission errors are detected in the data packet, the packet is discarded and a request for retransmission of the data packet is made. The ARQ protocol being a communication channel requires a two-way channel where the ARQ protocol sends the request for retransmission via the feedback channel.

The ARQ retransmission request makes use of acknowledgments and timeouts in its request protocol. Acknowledgements can either be a positive acknowledgement (ACK) or a negative acknowledgement (NACK), where ACK's indicate correctly received data packets and NACK's indicate erroneous data packets. When an ACK is received the transmitter transmits the next data packet and when a NACK is received the transmitter begins a retransmission of the current data packet. A timeout is when a predefined period of time expires between transmission of the data packet and receiving of the acknowledgment. This prompts a retransmission of the data packet by the transmitter.

The ARQ retransmission scheme has three retransmission protocols:
- Stop and wait ARQ
- Go-back-n ARQ
- Selective repeat ARQ

These protocols attempt to strike a balance between the complexity of the protocol and the throughput of the system, where the stop and wait ARQ scheme involves low design complexity and experiences low data throughput and the selective repeat ARQ scheme involves high design complexity with high data throughput.

5.2.3.1   Stop and wait ARQ

The stop and wait ARQ (SW-ARQ) [93], [95], [96], [97] protocol is the simplest of the three types of ARQ transmission protocols. In this protocol the transmitter sends a data packet to receiver and waits for a positive or negative acknowledgement before the next packet can be transmitted.

If the receiver receives the data packet free of errors, a positive acknowledgement (ACK) is sent back to the transmitter. The ACK is an indicator to the transmitter that the transmitted data packet was unaffected by channel errors and the transmitter can then proceed to transmit the next data packet in the queue. If the data packet was affected by channel errors during

transmission and the errors are detected at the receiver, the receiver discard the data packet and sends a negative acknowledgement (NACK) back to the transmitter. The NACK is a retransmission request which tells the transmitter that the transmitted data packet was erroneous and a retransmission of the current data packet is required. The process continues until the retransmit data packet is accepted by the receiver free of errors and an ACK is sent back to the transmitter acknowledging receipt of the error-free data packet. It is graphically illustrated in Figure 5-14 where an erroneous data packet is transmitted.

Stop and wait ARQ makes use of a timeout in the event that either the receiver has not received the data packet or the transmitter has not received an ACK or NACK. The timeout uses a timer which is set upon transmission of a data packet and counts down from a predefined time. This predefined time is an estimated amount of time determined between transmission of a data packet and receiving of an acknowledgement. If the time expires, the transmitter automatically retransmits the data packet. The use of timeouts for lost data packets requires a data packet sequence number to distinguish between a retransmit packet and the next packet.

The stop and wait scheme showing acknowledgements and timeouts are graphically displayed in Figure 5-13 to Figure 5-15. The illustrations show the three scenarios that may occur in SW-ARQ; a lost acknowledgement, an erroneous data packet or a lost data packet. In SW-ARQ each data packet and acknowledge uses an alternating sequence number of *0* and *1* to keep track of current and next data packets during retransmission events.
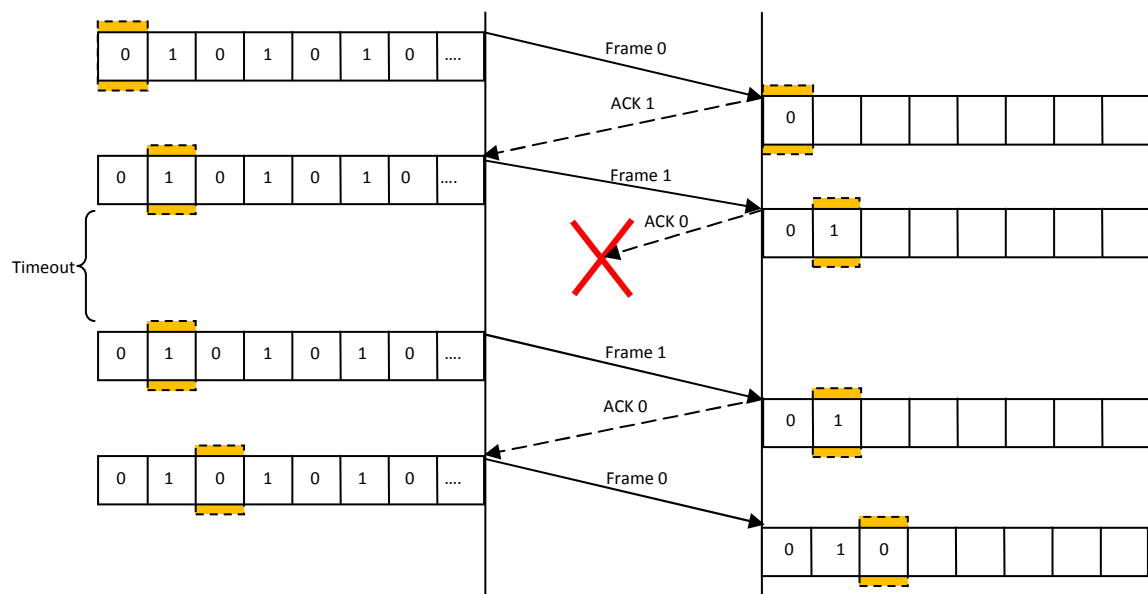
Figure 5-13: Stop and wait ARQ for lost acknowledgement.
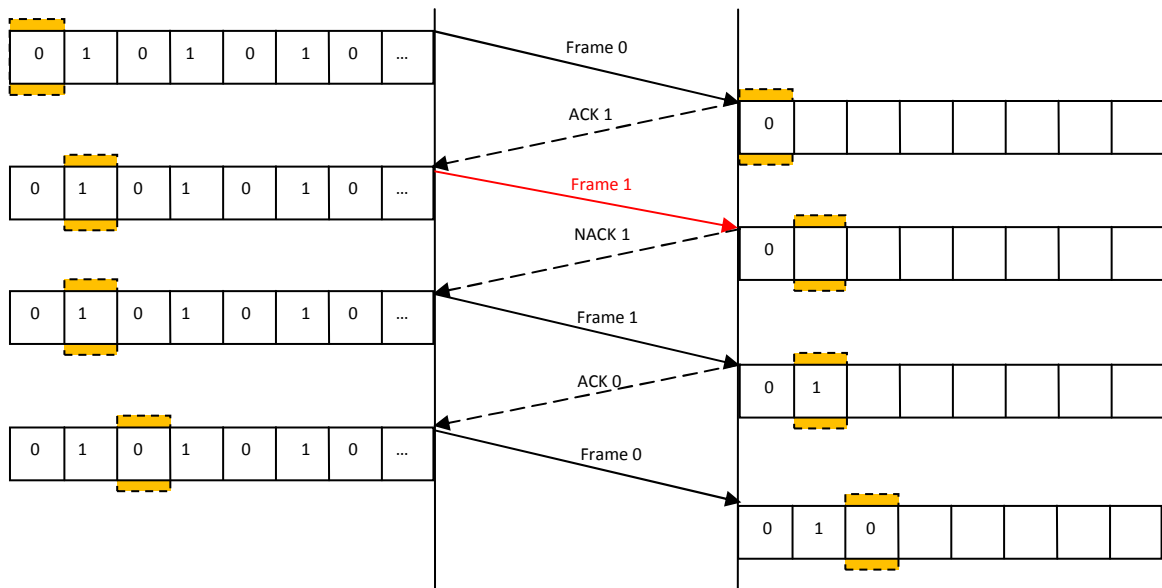
Figure 5-14: Stop and wait ARQ for erroneous data packet (frame).
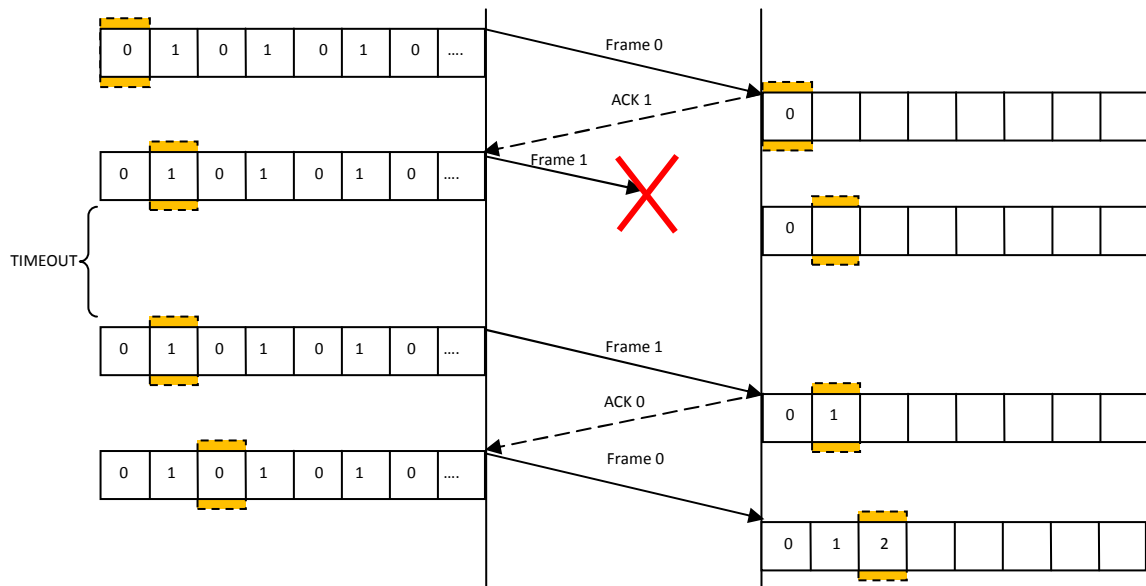
Figure 5-15: Stop and wait ARQ for lost data packet (frame).

The primary benefit of SW-ARQ is that it does not require packet buffering at the transmitter or receiver unlike the other schemes. However, the protocol becomes inefficient due to the round-trip delays and idle times the transmitter spends waiting for acknowledgements.

5.2.3.2 Go-back-n ARQ

The go-back-n ARQ (GBN-ARQ) [92], [96], [97] protocol was developed as an improvement to the stop and wait ARQ protocol where the channel remains busy by transmitting several data packets continuously whilst waiting for an acknowledgement. Go-back-n ARQ attempts to

address the inefficiencies experienced by transmission delays and acknowledgment waiting periods. GBN ARQ uses both flow and error control in its protocol.

The go-back-n ARQ protocol makes use of a sliding window which is a transmit flow control mechanism that allows the transmitter to transmit a specific number of data packets before an acknowledgment is received or timeout event occurs. The sliding window technique is a form of pipelined communication that utilizes the channel more efficiently. Figure 5-16 is representative of the sliding window protocol used in go-back-n ARQ. Sequence numbers are used in the sliding window protocol to keep track of received and lost data packets and acknowledgments.

The sliding window has a window size *n* where n is the parameter that determines the number of successive data packets that is transmitted before there is receipt of an acknowledgment. The data packets are assigned with sequence numbers ranging from *0* to *n-1*. The receiver's buffer is mapped to the transmitters sliding window in order to keep track of the data packets received.

The transmitter will attempt to transmit all the data packets highlighted in the sliding window and will set a timer for each data packet it transmits. The receiver will only accept the first data packet sequence number in its highlighted sliding window before moving on and accepting new data packets. The sliding window protocol is more effective as a cumulative ACK can be sent indicating acceptance of data packets up to and including the ACK's sequence number. This concept is highlight in Figure 5-16 where a cumulative acknowledgment ACK 3 is sent indicating the correct reception of data packets *0* to *2*. Once an acknowledgment is received the sliding window slides over three positions and forms a new window. Figure 5-16 also highlights that if an acknowledgement for an already transmitted packet is received before the transmission is complete, the transmitter assumes the data packet was lost and retransmits the data packet and thus the sliding window does not move until a correct acknowledgement is made.
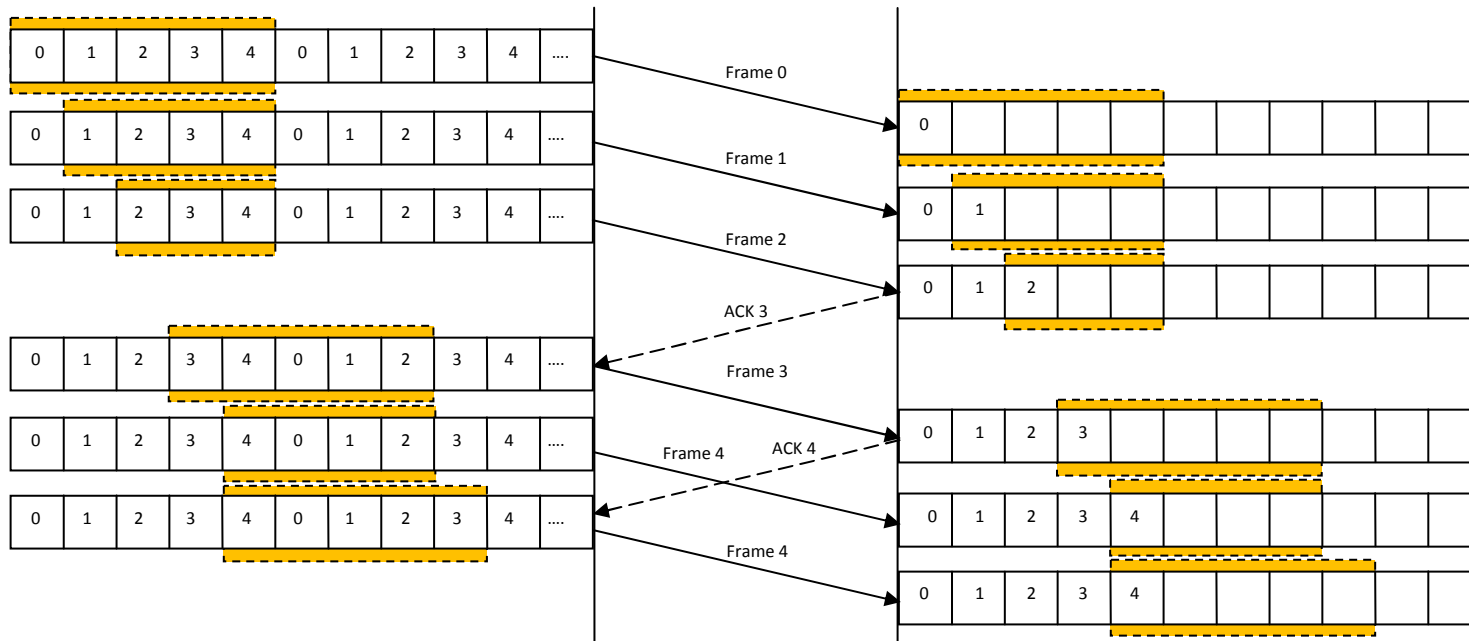
Figure 5-16: Diagram of sliding window protocol.

The go-back-n ARQ protocol uses the sliding window protocol for its communication protocol when no errors are detected within the data packet. It also uses the sliding window protocol as the basis for the three scenarios; a lost ACK, a lost data packet and an erroneous data packet illustrated in Figure 5-17 to Figure 5-18. These three scenarios typically occur for erroneous channel conditions when channel errors affect the data packet.

The GBN ARQ protocol transmits the data packets in a continuous stream. It more efficient as it can send an accumulated ACK instead of sending an ACK for each frame. This accumulated ACK indicates that data packets received prior to the ACK are all correct. In go-back-n ARQ the transmitter allows for buffering by using a buffer space of n packets for the communication protocol. The buffer allows for transmission of n packets without waiting for acknowledgements. Once an acknowledgement for the last packet is received, this is an indication that all transmitted packets up to and including the last data packet were received free of errors and a new buffer of data packets can be created for the next transmission sequence. The receiver only accepts correct data packets in its correct sequence, thus no buffering of data packets is used at the receiver as incorrect data packet that are either erroneous or not in the correct sequence are immediately discarded.

Go-back-n ARQ for a lost data packet frame given in Figure 5-17, shows the buffer with sliding window setup. The buffer is three frames wide and is thus called a go-back-3 ARQ scheme as n represents the buffer/sliding window width. If a frame is lost during transmission, a NACK is sent from the receiver to the transmitter requesting retransmission of the lost frame. All subsequent frames received after the lost frame is discarded whether they are error-free or not. In the go-back-n ARQ protocol upon reception of a NACK with the lost frame sequence number, the transmitter goes back and retransmits the lost frame immediately and then proceeds to retransmit all frames following the lost frame again until an acknowledgement is received. Once an ACK is received the sliding window/buffer moves forward n frames.

Figure 5-18 illustrates the procedure for receiving an erroneous data packet. If the receiver detects an error in the data packet a NACK along with the sequence number of the data packet in error is sent back to the transmitter. Once a NACK is received by the transmitter this indicates a request for retransmission as the received data packet given by the sequence number is damaged or in error. The transmitter stops all current transmissions and then begins to retransmit all the data packets from the damaged data packet onward within the sliding window buffer.
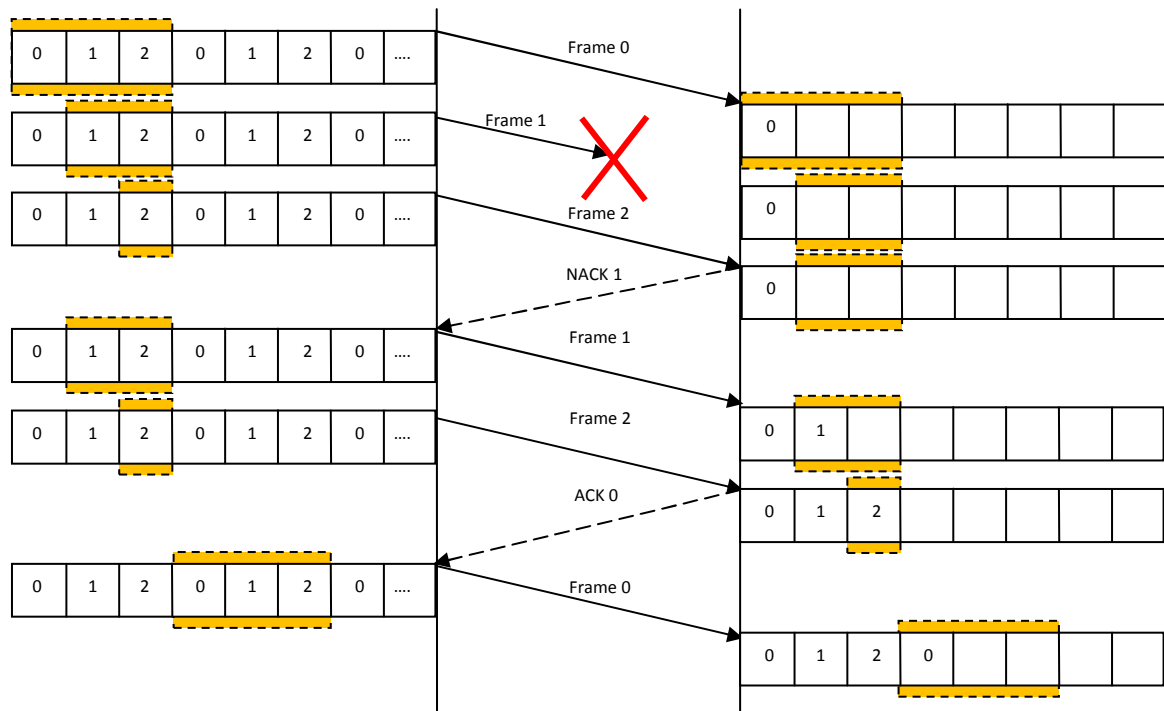
Figure 5-17: Go-back-n ARQ for a lost data packet (frame).
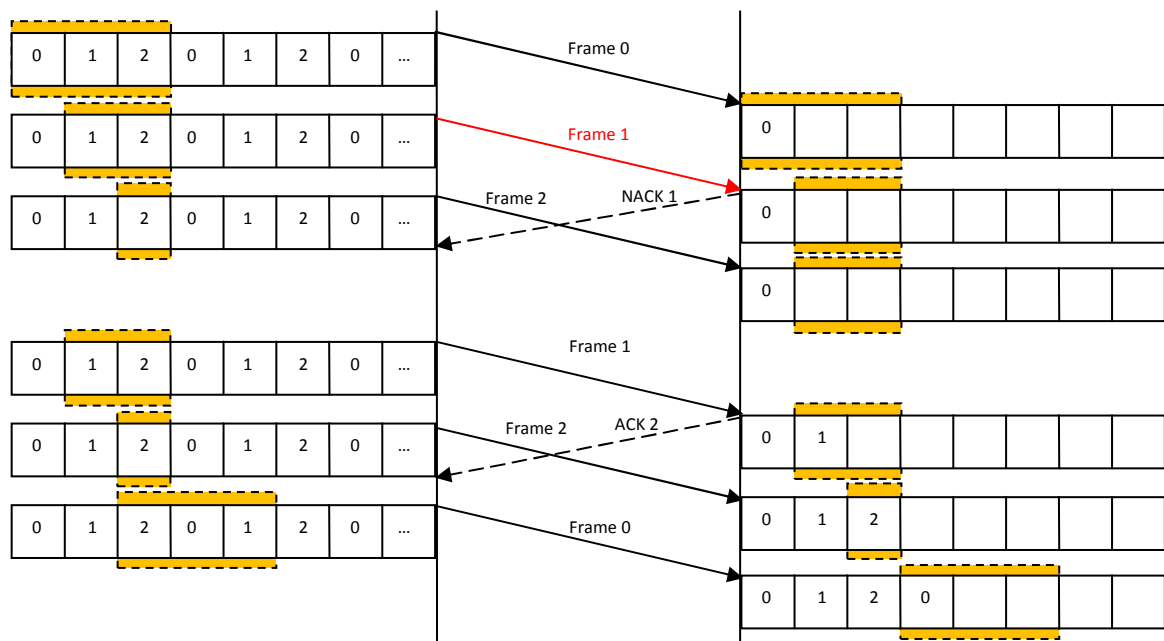
Figure 5-18: Go-back-n ARQ for erroneous data packet (frame).

The lost acknowledgement scenario in Figure 5-19 uses a timeout function, where if a positive or negative acknowledgment (ACK/NACK) does not reach the transmitter within a given amount of time, known as the round-trip delay, retransmission occurs. The timeout function is usually set to an amount of time equivalent to a round trip, which is the amount of time needed to transmit the set of *n* data packets in the buffer. If the timer expires before an ACK/NACK is

received the transmitter stops and retransmits from the last acknowledged frame. In the event that the ACK/NACK is not received and the buffer is emptied by the transmitter before the timer expires, the data packets following the last acknowledged frame in the buffer is retransmitted again until an acknowledgement is received. The receiver will only accept all correctly sequenced frames, thus if the frames have already been received during the first transmission, it will discard the incoming frames and proceed to only accept frames that are next in the queue.
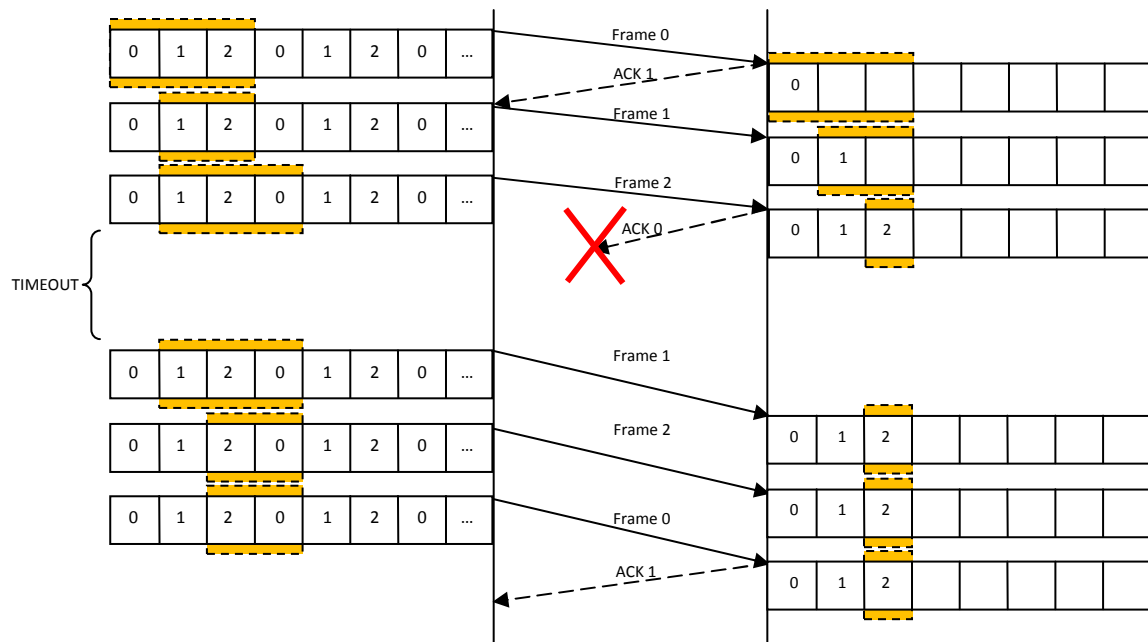
Figure 5-19: Go-back-n ARQ for a lost acknowledgement.

The above scenarios show why the mechanism is named go-back-n ARQ as the transmitter goes back to the lost or damaged data packet and retransmits all subsequent frames within the sliding window.

Go-back-n ARQ is efficient as the channel is kept busy during error-free transmission. Delays are experienced during transmission of lost or damaged data packets and acknowledgements. Another major inefficiency is that when an error occurs the transmitter is required to resend the entire window buffer of data. However, go-back-n ARQ is still much more efficient than stop and wait ARQ as there is a continuous stream of data across the channel.

5.2.3.3 Selective repeat ARQ

The selective repeat ARQ (SR-ARQ) [96], [97] communication protocol is the most efficient amongst the three protocols. The selective repeat ARQ is a variation of the go-back-n ARQ

where the receiver is able to accept data packets that are out of sequence and the transmitter is able to process retransmission requests for erroneous data packets by retransmitting the data packet in question instead of retransmitting the entire window of data packets. This allows for continuous streaming of data packets between transmitter and receiver even after frame loss.

To achieve this efficiency, the protocol requires buffering at both the transmitter and receiver. When an erroneous or lost data packet is detected the receiver buffers the correct data packets until the erroneous packets are received correctly, before storing it in the correct sequential order. However, the addition of a buffer at the receiver introduces greater computational complexity.

As with the previous ARQ protocols, Figure 5-20 to Figure 5-22, graphically illustrates the procedure for a lost data packet, an erroneous data packet and a lost acknowledgement taken by selective repeat ARQ. With a lost data packet, SR-ARQ acknowledges correctly received frames however, if a frame is lost, SR_ARQ is able to recognise that it is an error, as the transmitted frame is out of sequence when it arrives at the receiver. The unexpected frame is buffered and a request for retransmission in the form of a NACK along with the sequence number is sent back to the transmitter indicating that an error has occurred. Any further frames that are received during the transmission of the NACK are buffered until the lost frame has been retransmitted and correctly received. As with GBN-ARQ acknowledged frames in SR-ARQ allow the shift of the sliding window buffer in the transmitter.
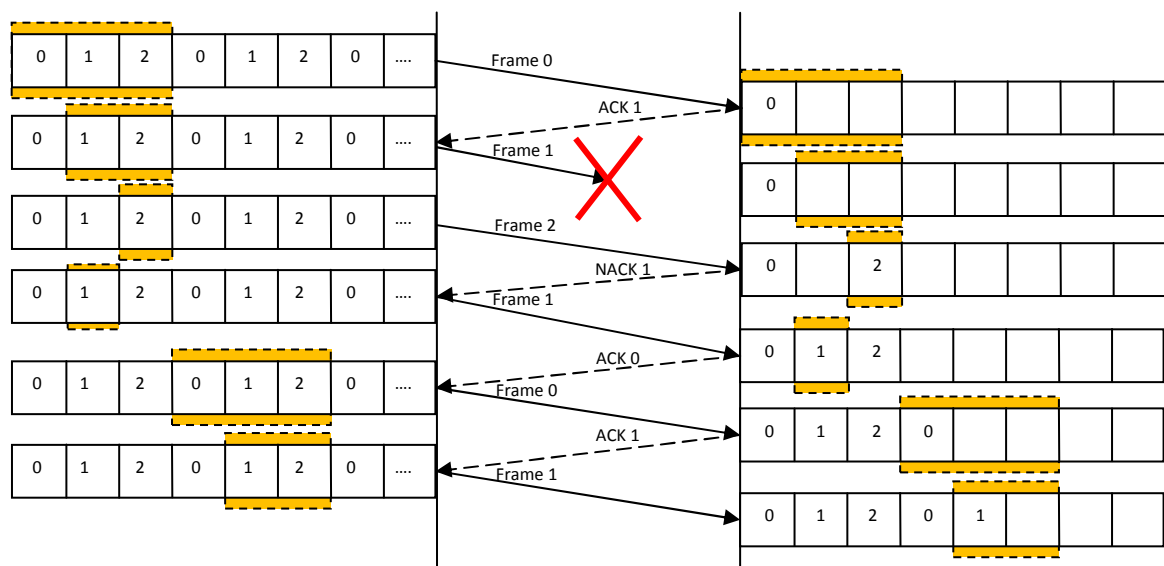
Figure 5-20: Selective repeat ARQ for lost data packet (frame).

The same procedure for a lost data packet also applies for an erroneous data packet. If the receiver detects an error in the data packet a NACK is immediately sent back to the transmitter. The receiver buffers all correct out of sequence frames until the erroneous frames are correctly received. Once all the correctly received frames in its correct order are received, the receiver outputs the packet and proceeds to analyse the next incoming packet.
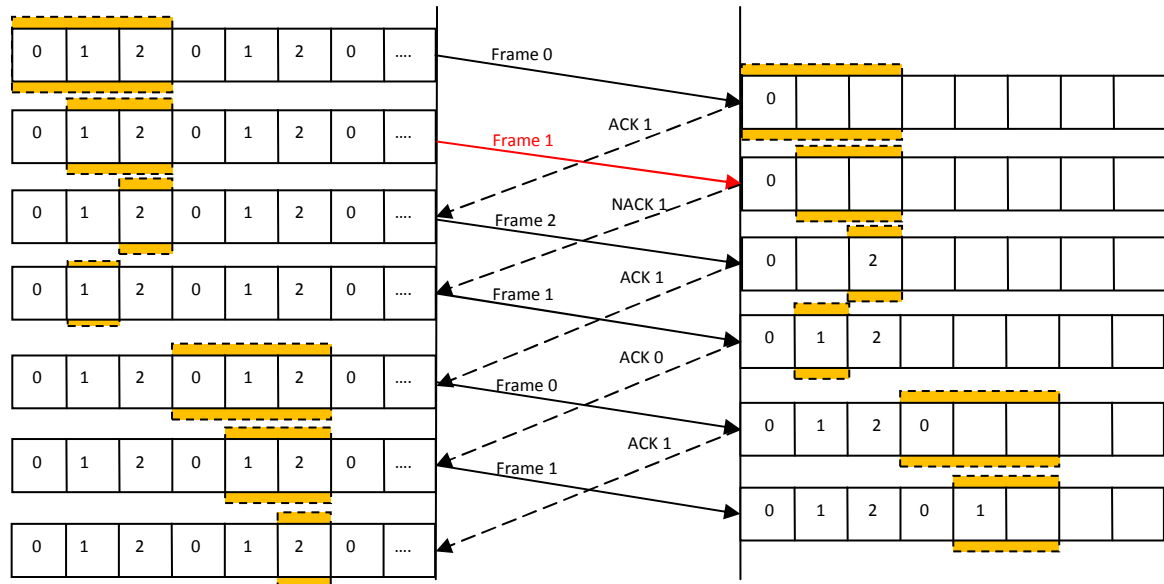
Figure 5-21: Selective repeat ARQ for erroneous data packet (frame).

Selective repeat ARQ uses the timeout function when a lost acknowledgement occurs. Since each data packet transmission requires an acknowledgment, individual timers for each data packet are required to determine whether the acknowledgment has been lost. In the lost acknowledgment scenario, acknowledgements of data packets cause its associated timer functions to terminate however; a lost acknowledgement is not able to terminate its timer and thus causes its timer function to expire invoking a request for retransmission. Once the transmitter is aware of the lost acknowledgment through the expiration of its timer function, it assumes there was an error in the transmission of the data packet. The transmitter stops all current transmissions and attempts to transmit the unacknowledged data packet before resuming back to its previous transmission position. The addition of multiple timers contributes to the increased complexity but is able to produce greater efficiency in the selective repeat ARQ protocol.
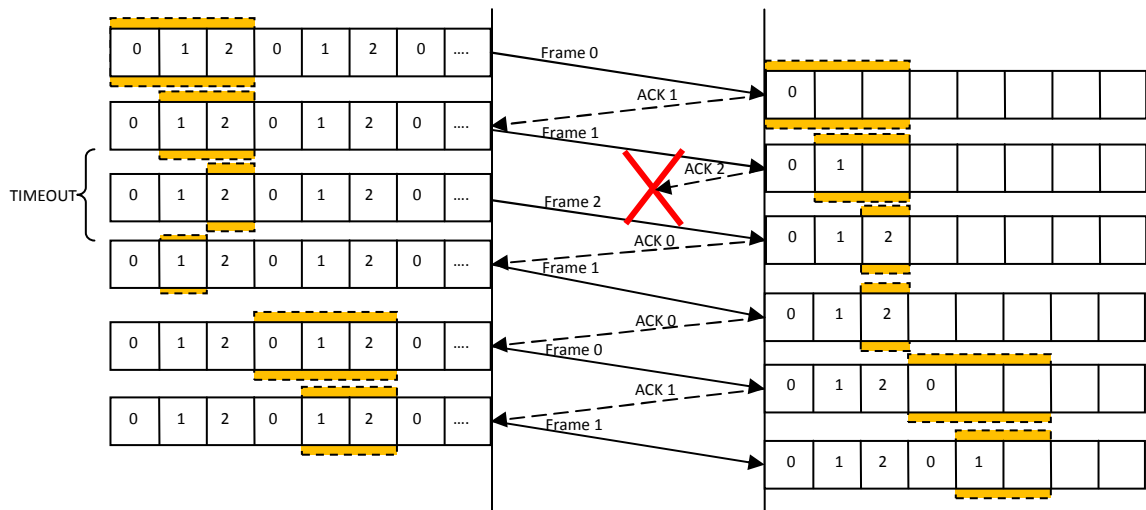
Figure 5-22: Selective repeat ARQ for lost acknowledgment.

Selective repeat ARQ is the most efficient of all the schemes as the channel is kept busy throughout the communication process unlike stop-and-wait ARQ and only erroneous frames are retransmitted unlike go-back-n ARQ which retransmits the entire window.

5.3 SUMMARY

This chapter introduced error resilience tools in the form of error detection and correction in order to preserve the integrity of data in its transmission over error prone channels. Error protection is the term given to error correction coupled with error detection.

The error detection component exploits the entropy coding stage of arithmetic coding by introducing a forbidden symbol as an error marker. The decoding of the forbidden symbol in an arithmetically coded bitstream is a distinct indication that the bitstream is erroneous and will require error correction. Error correction is a multifaceted model involving sequential decoding via the optimal stack algorithm, where the tree search is greedily directed using the MAP decoding metric. The MAP decoding metric is integral in the error correction procedure as it is solely used to correct and rectify errors produced during corrupt data transmission. It uses a complex set of *a priori* and *a posteriori* probabilities to compute the metric. Another subset of error correction is the ARQ based request for retransmission protocol that is only invoked when the MAP decoder is unable to correct the errors in the transmitted bitstream. The integration of the ARQ protocol increases the probability that the retransmitted sequence is less likely to obtain as catastrophic errors as the previous transmission and thus the MAP decoder will be more likely to correct the errors.

This chapter provides a detailed description of the complex subsystems employed by the proposed codec. It highlights each component and its use in the error detection and correction model subsystems. The mathematical theory is presented along with associated algorithm descriptions producing a chapter illustrating the entire overview of the proposed system components.

CHAPTER 6 -   PROPOSED CODEC

The proposed codec is an integral aspect of the dissertation as it consolidates the information, theories and ideas proposed. This chapter shows the progression from the theoretical concepts outlined in previous chapters to the design model of the proposed system. The design of the proposed coder involves wavelet based compression with the integration of error resilience in the form of arithmetic coding with forbidden symbol and MAP metric sequential decoding with ARQ retransmission. The proposed codec clearly defines the various stages within the design of the codec where a thorough description and detailed justification of each stage is provided.

6.1   SYSTEM DESCRIPTION

The proposed codec system description section presents the complete system that is employed in the codec design. The proposed codec combines wavelet image compression with error detection and correction. The codec by definition can be described by its three distinct sections, the encoder, the decoder and the retransmission request protocol. The encoder compresses the image into a bitstream, the decoder decompresses the bitstream into the reconstructed image and the retransmission request protocol requests for the bitstream retransmission over than channel. The system block diagram incorporating the encoder, the decoder and the retransmission protocol is shown in Figure 6-1.
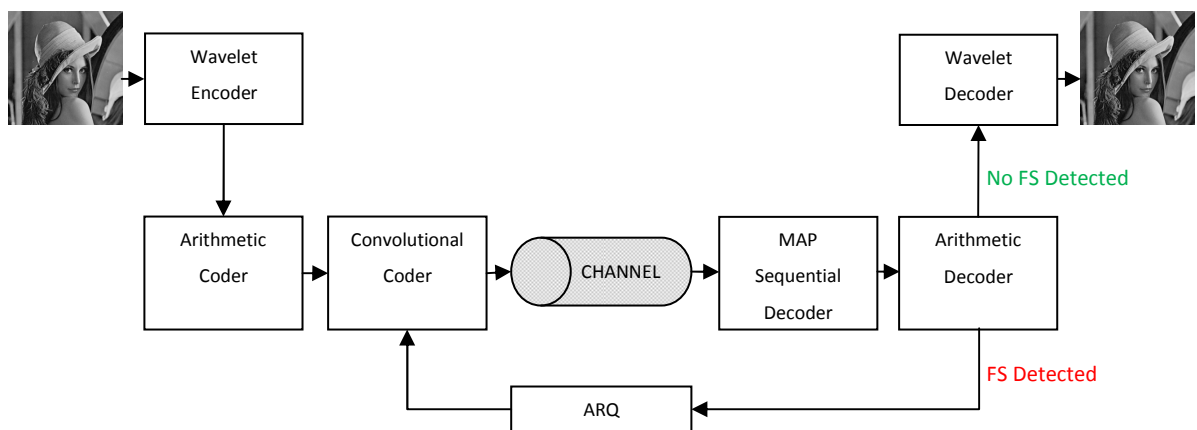
Figure 6-1: System Block Diagram of the Proposed Codec.

The transmission block diagram begins the wavelet compression with the image undergoing wavelet encoding. The arithmetic coder which forms the final entropy coding stage for the wavelet compression then follows. The entropy coding stage is exploited for error detection by

139

incorporating the forbidden symbol into the arithmetic coder source symbol probabilities. The arithmetically encoded stream is then followed by the convolutional coder, which is included to support the error correction capabilities of the system as it encodes the arithmetically coded symbols further.

The convolutionally coded bitstream is transmitted across an error prone wireless channel. The error prone channel corrupts the compressed image by introducing random bit errors into the bitstream. Thus the need for error detection and correction is fundamental in order to reproduce the original bitstream free of errors and hence decompress the image.

The MAP decoder at the receiver is then used to correct any errors that may be present in the convolutionally transmitted stream. Error correction involves the convolutionally coded bitstream possibly affected by errors, the MAP metric sequential decoder and the ARQ retransmission protocol. The MAP metric sequential decoder attempts error correction by decoding the convolutionally encoded bitstream and producing an "educated guess" of the original bitstream.

The new corrected bitstream is then decoded by the arithmetic decoder which exploits the forbidden symbol concept for error detection. If there is no occurrence of the forbidden symbol in the newly corrected bitstream, this is tantamount to the erroneous bitstream being completely corrected by the MAP decoder. However, if it is detected, the MAP error correction has failed to correct the erroneous packet and if decoded by the wavelet decoder the error will propagate resulting in a corrupted, visually impaired reconstructed image unless a request for retransmission of the packet is made.

The ARQ retransmission request is a second error correction mechanism employed to "backup" the MAP decoder in the event of uncorrected or unrecovered packets. If a forbidden symbol is detected the ARQ retransmission request sends a negative acknowledgement (NACK) to the transmitter via the feedback channel indicating that the received bitstream could not be corrected and the convolutionally encoded bitstream requires retransmission.

If a forbidden symbol is not detected two procedures occur; the arithmetically decoded bitstream proceeds to the wavelet decoder for further decompression and the ARQ retransmission protocol sends a positive acknowledgement (ACK) to the transmitter indicating that the received bitstream was free of errors and that the receiver requires the next convolutionally encoded bitstream.

The proposed codec concludes the final decompression of the bitstream by wavelet decoding before reconstructing the original image. The image reconstruction exactness is based on the successful correction of the errors. Thus the image itself can be used as a benchmark to corroborate the accuracy of the proposed codec.

6.2   DETAILS OF THE SYSTEM CODEC

The system description provides an overall account of the sequence of events that occurs in the algorithm. The stage details are explored further where each block in Figure 6-1 is broken down into smaller functional blocks which describe the central processes of the module.

6.2.1   Wavelet Encoding and Decoding

Wavelet image compression is denoted by the wavelet encoder illustrated in Figure 6-2, which applies three key wavelet processes to the image: the two-dimensional discrete wavelet transform, wavelet decomposition and the wavelet coding algorithm.
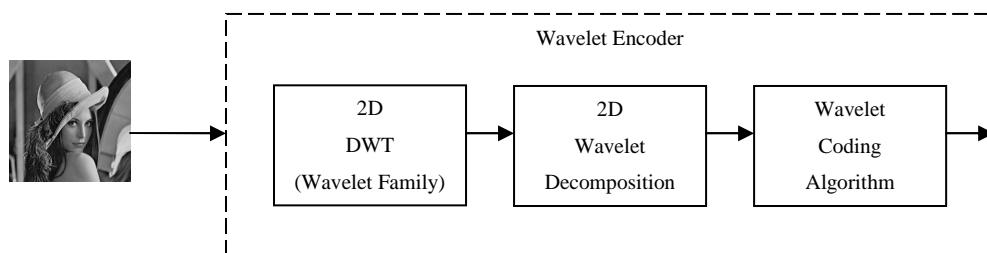
Figure 6-2: Block Diagram of the Wavelet Encoding Stage.

The two-dimensional DWT block computes the discrete wavelet transform by means of a specific wavelet family on the two-dimensional image. The choice of wavelet family for the discrete wavelet transform involves substantial information about the image and the wavelet family basis function in order to accurately represent the image information. There is however, no quantitative or explicit manner in which to choose the optimum wavelet as there are no rules determining the superiority of one wavelet over another. The discrete wavelet family chosen for the DWT is the Coiflet wavelet family; as it exhibits the best performance in terms of PSNR image quality shown in Table 3-2. Specifically the Coiflet 5 will be used for the DWT in the codec.

The two-dimensional wavelet decomposition block performs the multi-level decomposition required for the image. The optimum number of decomposition levels is complex to determine

as it is related to subband coding used by the wavelet coding schemes to achieve the desired compression or bit rate. The desired bit rate is achieved through repetitive iterations of the coding scheme which is directly associated to the number of decomposition levels chosen. There is a distinct relationship between the bit rate and the PSNR image quality, where a higher bit rate yields a higher PSNR and lower bit rate yields a lower PSNR. Thus the number of decomposition levels chosen affects the bit rate and hence the PSNR image quality produced. Therefore through systematic trials the optimum number of decomposition levels will be set to six, as fewer levels are insufficient for compression and more levels become inadequate for PSNR image quality. Nine iterations will be applied producing a target bitrate of *0.55bpp* for the SPIHT proposed codec design and *0.86bpp* for EZW proposed codec design.

The wavelet coding algorithm block defines the wavelet coding schemes employed for the wavelet compression of the image. Two wavelet coding schemes have been chosen for the evaluation of the proposed codec, the EZW and the SPIHT as motivated in Chapter 3.3. The EZW and SPIHT algorithms are the core wavelet encoding techniques incorporated in the wavelet encoder depicted in Figure 6-2, specifically designed to compress the image into a bitstream. Both exhibit reasonably comparable compression performance with regards to wavelet coding algorithms, illustrated in Figure 3-32. The EZW and SPIHT wavelet coding schemes with the addition of the error detection and correction procedures, will demonstrate the trade-off between the error resilience provided and the compression attained by the proposed codec.

The design of the wavelet encoder will produce results based on the various combinations of wavelet processes used. It will involve the two-dimensional DWT based on the Coiflet 5 wavelet family, the two-dimensional wavelet decomposition for three decomposition levels, and the EZW and SPIHT wavelet coding schemes.

The wavelet decoder block in Figure 6-1 controls the inverse of the operations and processes described thus far. Figure 6-3 graphically depicts the inverse operations that occur in order to reconstruct the image: the wavelet decoding algorithm, the two-dimensional wavelet reconstruction and the two-dimensional inverse discrete wavelet transform.
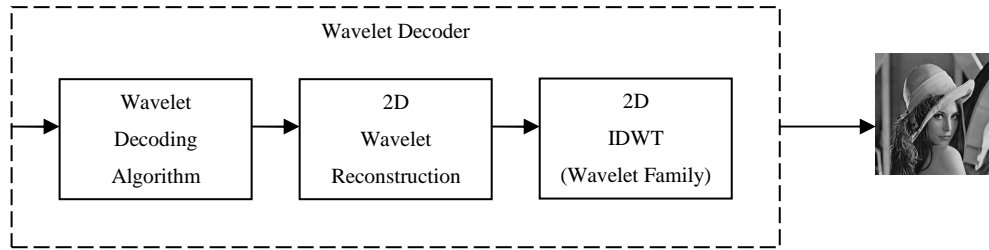
Figure 6-3: Block Diagram of the Wavelet Decoding Stage.

6.2.2   Arithmetic Encoding and Decoding

The arithmetic encoder and decoder blocks form a joint entropy coding and error detection technique for the proposed codec. Entropy coding is commonly used as the final stage of compression as it achieves greater overall compression of the bitstream. Error detection exploits the arithmetic coder by introducing an amount of redundancy in the form of a forbidden symbol. The arithmetic encoder is used as an entropy coder providing compression and the arithmetic decoder is developed for error detection. Thus the arithmetic encoder and decoder confirm its use in the design of the proposed codec as it is able to provide additional compression whilst integrating error detection.

The arithmetic encoder introduces the forbidden symbol into the symbol alphabet, but it is the arithmetic decoder that is used as the error detector. The addition of the forbidden symbol adds an amount of redundancy to the arithmetic encoder at the compromise of compression. In addition to compression, forbidden symbol redundancy at the arithmetic encoder also affects the decoding time or error detection time at the arithmetic decoder. This is the time it takes for the arithmetic decoder to detect or decode the forbidden symbol in the decompression phase. Hence the more redundancy introduced, the less compression occurs and less time it will take to decode the error. Thus a precise balance between compression and error resilience must be maintained at the arithmetic encoder to ensure maximum usage of both the entropy coding and error detection techniques.

The procedure as described in Figure 6-1 shows that if an error is not detected by the arithmetic decoder, the decompressed bitstream then proceeds to the wavelet decoder for further wavelet decompression. However, if the forbidden symbol is decoded, an error has occurred and a signal is transmitted to the ARQ retransmission protocol which begins the bitstream retransmission sequence.

6.2.3   Convolutional Coding and MAP Metric Sequential Decoding

The core error correction procedure presented in Figure 6-1 uses the convolutional coder and MAP metric sequential decoder blocks. These two functional blocks are coupled when error

correction is required. Error correction is designed to correct and rectify errors only once an error has been detected initially. Error correction is complex as it uses redundant information to estimate the original data from the incorrect data received.

The convolutional coder is used in the design of error correction in conjunction with the MAP metric sequential decoder, which when combined is able to correct the erroneous bitstream generated by the arithmetic encoder. The convolutional coder recodes the arithmetically encoded bitstream before it transmits it to the MAP metric sequential decoder. The convolutionally encoded bitstream may be affected by channel errors on transmission. The convolutional encoder implemented has a code rate of ½ and constraint length of 3. The parameterised generator polynomials are $G_0= 101$ and $G_1=111$. The convolutional encoder involves a simple code state design and structure, whereas an optimum convolutional encoder design for the proposed coder is beyond the scope of this dissertation as the focus lies mainly on the MAP metric sequential decoder.

The transmission of the convolutionally encoded bitstream involves the transmission of a packetized bitstream, where each packet is transmitted one at a time across the channel. Once the full transmission of the convolutionally encoded packet is complete, the error correction process begins on this transmitted packet.

The MAP metric sequential decoder uses sequential decoding by means of the stack algorithm and a *maximum a posteriori* (MAP) metric to decode the erroneous convolutionally encoded bitstream illustrated in Figure 6-4. The selection of sequential decoding as the applied convolutional decoder is because it is a classical decoding method for error control decoding of convolutional codes. Sequential decoding offers an alternative to iterative decoding and has now been incorporated in various maximum-likelihood detection systems, like multiple-input multiple-output (MIMO) and inter-symbol interference (ISI) channels as it offers different performance and complexity tradeoffs as compared to other maximum-likelihood decoders like the Viterbi decoder. Sequential decoding is advantageous in that it uses the error probability to direct its search instead of performing a fixed number of calculations for its decoding procedure like other decoders. It is able to backtrack and retrace its path through the code tree depending on the path metric unlike other decoders where all of its paths are followed until two paths converge to a single node, and then the path with the lower metric is discarded. Thus the sequential decoder appears to offer less computational complexity and greater flexibility.

```
┌ ─ ─ ─ ─ ─ ─ ─ ─ ─ ─ ─ ┐
│   MAP Metric Sequential  │
│         Decoder          │
│                          │
│      ┌──────────┐        │
│      │   MAP    │        │
│      │  Metric  │        │
│      └────┬─────┘        │
│           │              │
│           ▼              │
│      ┌──────────┐        │
──┼─────▶│  Stack   │───────┼──▶
│      │ Algorithm│        │
│      └──────────┘        │
└ ─ ─ ─ ─ ─ ─ ─ ─ ─ ─ ─ ┘
```
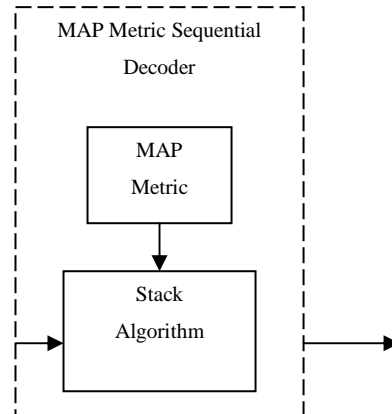
Figure 6-4: Block Diagram of the MAP Metric Sequential Decoding Stage.

The sequential decoder is implemented through the stack algorithm that uses the MAP metric to direct its search path through the codetree. The stack algorithm is a sub-optimal search that offers recursive back-tracking based on the node and hence the path that presents the best accumulated metric. The MAP metric is a statistical model for optimum decoding involving the minimisation of the decoding error probability by maximising the *a posteriori* probability. The stack algorithm, being a depth first search algorithm uses the MAP probability as a metric to select the best path from all possible paths within the codetree.

The MAP metric sequential decoder corrects the erroneous convolutionally encoded bitstream through its probabilistic search model before transmitting the newly decoded bitstream to the arithmetic decoder for further decompression. If the forbidden symbol is detected, the ARQ protocol sends a signal back to the transmitter requesting retransmission of the bitstream. If the forbidden symbol is not detected, the arithmetically decoded bitstream is then sent to the wavelet decoder to reconstruct the image. The image quality portrayed by the detail revealed within the image is a reasonable indication of the accuracy produced by the error correction scheme.

6.2.4   ARQ Retransmission

The automatic repeat request (ARQ) protocol for retransmission was introduced as a failsafe mechanism in the event the MAP decoder is unable to correct an erroneous bitstream. This double correction scenario will attempt to provide better performance results than other performance systems that do not make use of both MAP decoding and ARQ retransmission for error correction.

The ARQ retransmission strategy chosen is stop-and-wait ARQ. Although it performs the least out of the three scenarios given, it is easy to implement, given the already highly complex

wavelet algorithms, the advanced arithmetic coding with forbidden symbol and highly statistical and search intensive MAP decoding. If the performance of the proposed codec with stop-and-wait ARQ supersedes the performance of the compared systems, then it can be concluded that the more efficient schemes like go-back-n ARQ and selective repeat ARQ will produce greater performance.

6.2.5 The Channel

The channel is representative of the transmission medium used to transmit the compressed bitstreams to be decoded. The channel may introduce interference in the form of additive noise, attenuation as well as transmission delays. Thus the channel model is critical in the design of the proposed codec as it is used to facilitate an adequate error correction scheme to counteract the errors produced by the channel impairments.

Two wireless channel models proposed for the codec are an additive white Gaussian noise (AWGN) channel and a Rayleigh fading channel. The preference of wireless communication models for the design of the channel is due to the behaviour and nature of the propagation channel, as wireless channels are non-stationary and typically very noisy due to multipath fading and interference. This is a logical choice in terms of modelling random errors for the injection into the wavelet compressed bitstream.

The channels will feature varying Signal-to-Noise-Ratio's, so that the error correction scheme can attempt to handle and correct several varying degrees of erroneous bitstreams. Thus statistics of average SNR per bit versus performance can be accurately determined. The modulation and demodulation of the compressed bitstream also falls under the channel behaviour. The bitstream will undergo differential binary-phase-shift-keying (DBPSK) modulation before Gaussian noise and Rayleigh fading affects it.

The channel in the proposed codec is used as an error injection and propagation medium, and the error protection is included in the design to mitigate the channel effects from the transmitted bitstream.

6.2.6 Details of the Image to Bitstream Packetisation for Transmission

The encoding process essentially transforms the image into a compressed bitstream via the concatenated wavelet encoding and arithmetic coding processes. The transmission of the bitstream to the decoder involves the bitstream to be packetised prior to the transmission. Packetisation of the bitstream allows for the speedy decoding of the forbidden symbol, faster error correction and quicker retransmission. Moreover, the decoding time for the forbidden

symbol is related to the amount of redundancy, thus the use of packets of data allow for less redundancy to be added, maintaining faster decoding.

The packets of data are individually arithmetically encoded and then convolutionally encoded, and do not require the presence of other packets for the decoding process. The use of packets comes at the expense of compression, as each packet of data involves the flushing of the arithmetic encoder which adds additional bits to each packeted compressed bitstream. Hence a trade-off is established between compression and decoding efficiency. Once again the compression of the system becomes critical when other factors are involved.

6.3   SUMMARY

This is an important chapter as it presents the design and analysis of the proposed codec system employed. The proposed codec forms a hybrid source and channel coding system that attempts wavelet compression with error detection and correction. This hybrid scheme of source and channel coding involves source coding which is associated with compression and channel coding involving error protection, which can be performed sequentially whilst trying to maintain optimality.

The image is initially compressed into a bitstream, packetised and transmitted across an error prone wireless channel before being decompressed. The decompression involves a series of error detection and correction techniques before undergoing image reconstruction. The proposed codec system transmission block diagram is discussed where each block process is examined in detail. The proposed codec illustrates the flow of the processes involved in the design and typically establishes the nature of the process properties employed.

CHAPTER 7 - PERFORMANCE OF THE PROPOSED CODEC

This chapter evaluates and analyses the proposed codec through a series of simulations in terms of its performance metrics using a standard test image sample set and various control parameters. The sample set of test images contribute vastly varying content enabling higher qualitative performance results in terms of recovered image quality. The quantification of the control parameters are characterised by the error propagation phenomenon introduced by the channel as well as the optimisation of the rate distortion trade-off model.

The proposed codec's performance is indicative of the effectiveness of its wavelet compression algorithms and its ability to adequately correct erroneous data packets through its error coding schemes. In addition, the results assess the proposed systems performance when compared to current compression and error coding standards as well as combinations thereof. The proposed codec's simulation results are specific in that: available bandwidth, bit error rates and display resolution, associated with the wavelet compression and error coding, determine acceptable performance criteria. On the basis of these performance measures as well as the associated simulation results, viability of the proposed codec is estimated and the overall performance can be quantified.

7.1 EXPERIMENTAL METHOD

The experimental method describes the basic configuration of the evaluation environment and defines the evaluation parameters when assessing both the proposed and current standardised systems. It aims at maintaining consistency throughout the performance results by specifying the parameters in which the systems should be assessed, measured and quantified. It is meaningless to compare evaluation results obtained under different evaluation environments as both the proposed codec and the current codec's performances are influenced by the evaluation parameters. Thus the experimental method is critical in allowing the performance of the proposed codec to be accurately evaluated as it is impossible to obtain correct and reproducible results if any part of the experimental method is not clearly defined for each performance test.

The platform used for the simulations is an Intel Pentium 4, 1.6 GHz PC with 512MB of RAM, running the Microsoft Windows XP 2002 Professional SP2 operating system. The simulation results were generated using the Matlab simulation engine.

The test image sample set used are standard image processing test images that contain a variety of natural scenery and varying textured detail. They are specifically selected to test the behaviour of wavelet based image compression techniques and error resilience methods of the various codec's. The test image sample set includes three images, each using a resolution of *256x256* frame size with *8-bit* greyscale format.

The evaluation parameters for the simulation tests of the codec's, use both the Coiflet 5 wavelet family, six wavelet decomposition levels and nine iterations for the simulations. The performance simulations include two wavelet compression algorithms, the EZW and SPIHT coding schemes, executed over two wireless channels, the AWGN and Rayleigh fading channels with a fading variance of $\sigma^2 = 0.5$. Performance comparisons between MAP decoding with ARQ retransmission and systems involving the following scenarios are evaluated:

- System 1: Arithmetic coding at the encoder and arithmetic decoding at the decoder.
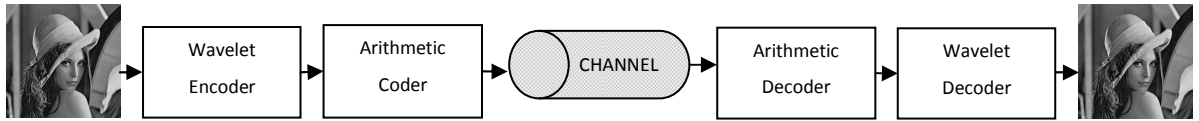
Figure 7-1: System 1 block diagram scenario for performance comparison.

- System 2: Convolutional coding at the encoder and MAP decoding at the decoder.
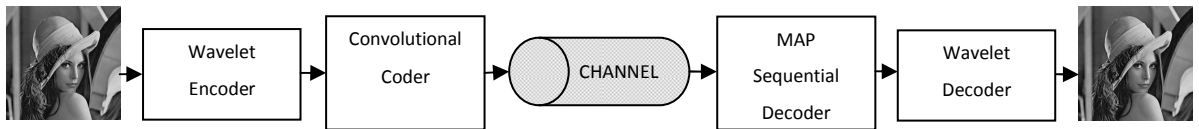
Figure 7-2: System 2 block diagram scenario for performance comparison.

- System 3: Arithmetic coding and convolutional coding at the encoder and MAP decoding with arithmetic decoding at the decoder with no forbidden symbol detection.
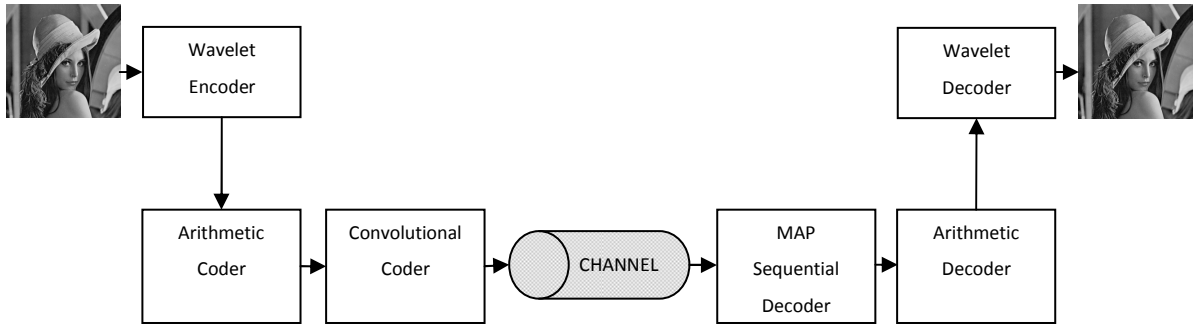
Figure 7-3: System 3 block diagram scenario for performance comparison.

Each system does not use ARQ retransmission nor makes use of the forbidden symbol error detection capabilities found in the proposed codec thus evaluations of the performance of these inclusions in the proposed codec against the systems above are determined. System 1 involving simple arithmetic coding and decoding is included to show how a compressed bitstream without any form of error protection performs against the proposed codec which uses a fully integrated error protection system. System 2 is included in the simulations to illustrate that error correction without the forbidden symbol error detection mechanism does not provide adequate correcting ability as compared to the proposed codec which uses all elements to provide a superior system. System 3 is a combination of arithmetic coding and MAP decoding offering additional compression and error correction but without forbidden symbol detection or ARQ retransmission. This system is compared to the proposed codec to illustrate that even the inclusion of specific elements will still not produce optimal results as two key elements of forbidden symbol detection and ARQ retransmission for additional error correction are not included in the design.

7.2 COMPARISON WITH IMAGE COMPRESSION AND ERROR CODING STANDARDS

This section compares the proposed codec against current systems using the test image sample set and its defined evaluation parameters. The proposed system involving MAP metric sequential decoding system with ARQ retransmission uses the stages illustrated in Figure 6-1. The systems for comparison involve three systems; System 1 which involves arithmetic coding and arithmetic decoding, System 2 which involves convolutional coding and MAP decoding and System 3 which involves arithmetic coding and convolutional coding with MAP decoding and arithmetic decoding.

The arithmetic coding and decoding system does not utilize the forbidden symbol error detection technique nor any other additional error correction techniques. The arithmetic coding behaves as an entropy coding stage in system 1. System 1 which involves arithmetic coding and decoding stages only is compared to the proposed codec which incorporates both error detection via the forbidden symbol and dual error correction through MAP decoding and ARQ retransmission. This comparison attempts to test the error detection and correction capabilities of the proposed system against the simple entropy coding of System 1. The error detecting capabilities of the proposed system of MAP decoding and ARQ retransmission uses the forbidden symbol technique to request for retransmission whereas the comparing system does not and this will highlight the usefulness of this feature in improving overall performance. The lack of error correction in System 1 against the proposed system will show the need for error correction for increased performance improvements when impaired error prone channels are concerned.

System 2 using convolutional coding and MAP decoding without forbidden symbol detection or arithmetic coding, when compared against the proposed system will attempt to correct erroneous bitstreams. System 2 is also free of error detection, but integrates error correction. If the MAP decoder is unable to completely correct a bitstream, the bitstream will remain in error, whereas the proposed codec has the ability to request for a retransmission of the erroneous packet if detected. A performance comparison between the systems will indicate that the use of additional compression via arithmetic coding, error detection due to the forbidden symbol and additional error correction provided by the ARQ retransmission request contributes significantly to the increase in overall performance of the proposed system. The proposed system tests the ability of MAP decoding with ARQ retransmission as an efficient error correcting tool.

The third system, System 3, involves AC, CC, MAP and arithmetic decoding against the proposed system which includes the exact same stages of operation (AC, CC and MAP) however, it includes forbidden symbol detection and ARQ retransmission as an additional error detection and correction mechanism. This performance comparison is used to illustrate that the inclusion of ARQ retransmission for additional error correction improves the overall performance of the system, as bitstreams that could not be adequately corrected by the MAP decoder can be either be free of error or re-corrected in the second transmission sequence. Also the inclusion of the forbidden symbol for error detection provides additional use of arithmetic coding other than an entropy coding stage to compression.

7.2.1 Lena

The Lena [85] test image is the most popular image processing and image compression standard test image in use. It contains a mixture of detail, flat regions, shading and texture, to be able to adequately test the compression coding schemes. It is a diversely detailed image that offers varied detail response and is useful to test the decoder error capability with regard to fine detail, flat regions, shading, contrast and texture. The 256x256 grayscale Lena test image used in the simulation is shown in Figure 7-4.

Figure 7-4: Lena Test Image [85].

The performance simulations executed on the Lena image differ between wavelet compression coding schemes, wireless channels and error detecting and correcting methods. Essentially two performance results are used in the analysis, packet erasure rate (PER) versus average SNR per bit (E_b/N_0) and image quality (PSNR) versus bitrate. PER is used to evaluate the error correcting performance of the systems against the channel's induction of errors. PSNR is used to evaluate the image quality produced by the systems wavelet compression algorithms under channel degradation. Thus the two sets of results produce a broad overview of the various systems performances in comparison with one another.

For the following Lena image performance results, simulations of all four systems using the exact same wavelet coding scheme across a specific channel is generated. These simulations test the performance of the four systems against each other using the same wavelet compression algorithm and the same channel conditions. Essentially these results will determine the best system in terms of error correction and image quality production.

7.2.1.1 EZW Coding over the AWGN channel.

The simulated performances use the EZW wavelet coding algorithm transmitted across an additive white Gaussian noise channel. The difference in the performance simulations for the three systems against the proposed codec is due to the inclusion of the forbidden symbol in arithmetic coding for error detection and ARQ retransmission for additional error correction. In order to analyse the proposed codec's ability to adequately correct channel induced errors performance simulations of packet erasure rate (PER) versus average SNR (signal-to-noise-ratio) per bit (E_b/N_0) in decibels (dB) were generated. The results indicate the AWGN channels effect as well as the correcting abilities of the systems. The packet erasure rate is defined as the number of packets that were not corrected by the system in relation to the total number of packets transmitted across the channel. It can be described by equation (7.1):

$$PER = \frac{\text{Number of Erroneous Packets not Corrected}}{\text{Total number of Packets Transmitted}} \ . \qquad (7.1)$$

The channels average SNR per bit models the amount of errors introduced into the channel during the transmission process. At lower E_b/N_0 the number of errors introduced increases resulting in a higher PER. Figure 7-5 is the performance simulation for the AWGN channel using the EZW wavelet coding scheme for the Lena [85] image using *20-bit* transmission packets. The four simulations show the proposed codec against the three systems; System 1, System 2 and System 3. Figure 7-5 illustrates the number of erroneous packets that could not be corrected by the systems, for the entire process involving compression and transmission of the Lena image. In terms of the number of uncorrected erroneous packets, the proposed system performs best as its packet erasure rate is far less than all other systems throughout the SNR range. The ARQ based system for majority of the SNR's produced impeccable error-free images. A PER of zero constitutes an error-free decompressed image.

The proposed system which is the MAP metric sequential decoding with the ARQ retransmission protocol and forbidden symbol detection for error correction reduces the erasure rate by approximately *85%* against the three comparison systems. This reduction is calculated by (7.2) which is representative of the statistical percent change formula of two data sets; D_1 and D_2. The percentage change between D_2 and D_1 is given as:

$$\frac{\Delta D}{D_1} = \frac{D_2 - D_1}{D_1} \ . \qquad (7.2)$$

The *85%* overall percentage reduction experienced by the three systems uses the above percentage change formula given at each E_b/N_0 and then a mathematical mean of these percentage changes is then calculated to give an approximate overall change across the two trends. Essentially D_2 will be the set of PER results of the proposed system and D_1 is the set of results for System 1 or System 2 or System 3. The percentage decrease formula is used for the PER results due to the linear scale exhibited by the PER results. However, for the PSNR results, which are to follow in the next section, a logarithmic scale is used and the absolute value in dB for the difference between trends is stated. The percentage change between the trends for the PSNR results is equivalent to the absolute change or difference between each value for dB results.

The proposed system which has an ARQ-MAP decoder uses additional forbidden symbol error detection whereas the other three systems do not. Thus the results provide an overview of the impact forbidden symbol detection has on error correction with proposed system versus System 1, System 2 and System 3. The results show that the forbidden symbol detection combined with MAP decoding in the proposed system produces far superior results than simulations without FS detection. The same can be said for inclusion of ARQ retransmission in the proposed system, as System 2 and System 3 do not have a second error correction procedure if the MAP metric decoder fails in any instance. Thus the addition of a second error correction mechanism ensures correction of an erroneous bitstream.

Figure 7-5: Diagram of PER vs. SNR (E_b/N_0) for EZW over AWGN channel for Lena image for the Proposed System against System 1, System 2, System 3.

When evaluating the four systems involving compression and errors, analysis of an images quality produced by the systems is essential in determining the overall effectiveness of the

system itself. Performance simulations showing the image quality of an image are generated by PSNR (peak-signal-to-noise-ratio) in decibels (dB) versus bitrate in bits per pixel (bpp) simulations. Bitrate in the performance simulations defines the degree of compression a system is able to achieve and is formally defined as the number of compressed bits in relation to the total amount of pixels in the image. Bitrates differ according to either the EZW wavelet coding or SPIHT wavelet coding and even arithmetic coding as an entropy coding stage as each scheme produces a different amount of compressed bits and arithmetic coding adds additional compression to the already highly compressed wavelet compression schemes. Low bitrates offer a higher degree of compression and the converse is true for high bitrates.

Image quality is commonly defined in terms of PSNR which demonstrates both the degree of compression applied and the effect of disruptive channel characteristics and error propagation on the image after its has been decoded. Essentially the PSNR versus bitrate results illustrate the extent of a channel's destruction on the image quality during decompression of an image showing the error propagation experienced. If the bitrate for a particular system does not change, it can be observed the impact a channel and error propagation can have on the ultimate image produced. This is essentially the findings these results attempt to examine.

For the image quality performance simulations the channel characteristics in terms of average SNR per bit (E_b/N_0) was kept constant to observe the wavelet's compression effect on the image. An AWGN channel SNR equal to *7dB* was set to show the channels erroneous nature and its impact on image quality and error propagation for the set of results, as well as maintaining consistency whilst evaluating the bitrate impact on image quality.

The proposed system's performance can only be measured through its error correcting capability in the presence of channel errors. Thus the effect of the channels errors at *7dB* will show a performance difference in compression and correction compared to the other three systems that includes all other stages of arithmetic coding and MAP decoding but without any forbidden symbol detection or ARQ retransmission strategies. If no channel errors (i.e. PER equal to zero) were present the PSNR image quality curves would be identical for all the systems as there will be no errors to correct and will in fact only decode the error-free transmitted bitstreams.

The image quality results are used as a performance indication for the comparison of the proposed system against System 1, System 2 and System 3 in terms of error propagation of uncorrected decoded errors. Thus the greater the number of uncorrected errors produced, the greater the chance of catastrophic error propagation, resulting in visually poor images.

Therefore it is essential to correct the channel errors through the use of a precise and efficient error correcting system.

Figure 7-6 illustrates the simulation results for image quality for the four systems; the proposed system, System 1, System 2 and System 3. The proposed system performs extremely well as it maintains relatively high image quality for a range of low to high bitrates. These results indicate the success of the proposed system's error correction technique, in that it is able to correct channel errors for various degrees of compression ratios preventing error propagation and thereby producing remarkable image quality. A higher PSNR or image quality for a given bitrate is always required in image analysis as it signifies the decoder's robustness to error propagation.

System 2 is also able to correct most compression errors (channel errors causing error propagation). System 1 and System 3 perform relatively poorly producing lower PSNR results showing the systems inability to correct channel errors contributing to the high degree of error propagation throughout the bitrates. The additional compression produced by the inclusion of arithmetic coding in System 1 and System 3, when affected by channel errors causes irreversible error propagation producing lower PSNR results. However, System 3's MAP metric sequential decoder is able to improve the image quality of the Lena image as compared to System 1 by correctly decoding more erroneous packets of data which decreases the chance of error propagation thereby increasing the PSNR quality. System 2 does not correct the errors with any success as the proposed system thereby contributing to the increase in error propagation and decrease in PSNR quality depicted in Figure 7-6.

Improved PSNR quality is seen by the proposed system which produces an average of *16dB*, *5dB* and *12dB* increases in image quality over System 1, System 2 and System 3 respectively. These results were calculated by first taking the arithmetic mean of the two trends being compared and then calculating the absolute difference between the mean dB values.
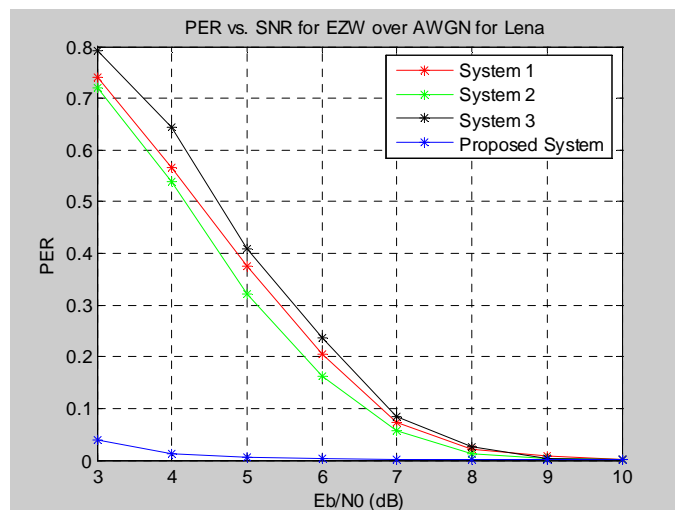
Figure 7-6: Diagram of PSNR vs. bitrate for EZW over AWGN channel for Lena image for the Proposed System against System 1, System 2, System 3.

7.2.1.2   EZW Coding over the Rayleigh Fading channel.

The Rayleigh multipath fading channel is a far more error prone, destructive and aggressive transmission channel than the AWGN channel and thus introduces more catastrophic errors into the channel during transmission. The Rayleigh model used is a frequency selective channel with a fading variance of *0.5*. The channel exhibits destructive errors even at high SNR's (typically above 8dB's) due to the inherent erroneous nature of the channel.

The performance results of the Rayleigh channel characteristics for the associated PER versus E_b/N_0 for a *20-bit* packet is illustrated in Figure 7-7. A significant difference in erasure performance is experienced between the proposed system and the three other systems for the Rayleigh fading channel. This difference can be attributed to the Rayleigh fading channel being an extremely error prone channel, and with a decrease in the channels SNR this has resulted in a greater number of errors being introduced into the three systems bitstreams as the systems cannot adequately correct the errors. Hence, the three systems simulations are fairly correlated in terms of errors unlike the proposed system which corrects the erroneous bitstream throughout the SNR range. The proposed system shows a significant decrease, of approximately 90%, in erasure performance for very low E_b/N_0. At higher SNR all systems actively produce minimal packet erasures as the probability of channel induced errors are relatively low at high SNR's (greater than 8dB's).

Figure 7-7: Diagram of PER vs. SNR (E_b/N_0) for EZW over Rayleigh Fading channel for Lena image for the Proposed System against System 1, System 2, System 3.

The performance simulations for the Rayleigh fading channel under the same conditions as the AWGN channel for PSNR versus bitrate is illustrated in Figure 7-8. From the simulation results the proposed system exhibits greater error correction capability than the three comparison systems. The proposed system involving ARQ retransmission with MAP metric sequential decoding shows improved error coding capabilities even in the more destructive channel as the system is able to correct the channel errors more effectively producing less error propagation in the decompression stages. Thus the proposed system illustrates a *3dB* to *16dB* PSNR improvement in its channel error coding ability, thereby reducing error propagation and increasing PSNR quality more so than the other systems.

The results in Figure 7-8 show the error correction capabilities of System 2 where this system performs relatively well in terms of image quality for the destructive channel. This is validation that the first stage of error correction using CC and MAP metric sequential decoding in the proposed codec works well in correcting channel errors.

The simulations also indicate that both System 1 and System 3 cannot adequately process the additional compressed information from the inclusion of arithmetic coding transmitted via the corrupt Rayleigh fading channel as the image quality remains poor over all simulated bitrates. The integration of arithmetic coding adds compression thereby causing catastrophic error propagation resulting in System 1 and System 3 performing worse than System 2. System 3 does not use the forbidden symbol technique and uses the integrated arithmetic coding as a pure entropy coding stage only, thereby contributing to increased compression and thus greater error propagation.

Nonetheless, the proposed system including ARQ retransmission with MAP decoding exploits the additional arithmetic coding element in its design for forbidden symbol error detection which then requests for a bitstream retransmission via the ARQ protocol if the bitstream is in error. The improved error correcting performance of the proposed system shows a *16dB*, *3dB* and *13dB* PSNR increase in overall image quality performance for System 1, System 2 and System 3. This is significant and verifies that the combination of both the forbidden symbol and ARQ retransmission with MAP decoding offers phenomenal correction when applied together.
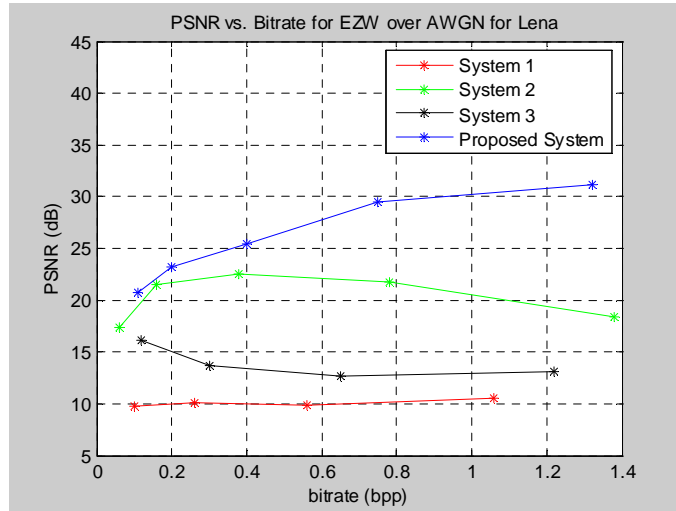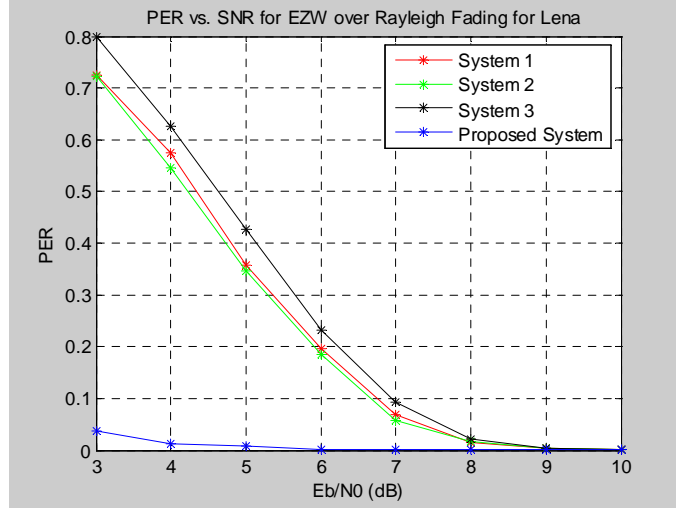

Figure 7-8: Diagram of PSNR vs. bitrate for EZW over Rayleigh fading channel for Lena image for the Proposed System against System 1, System 2, System 3.

7.2.1.3 SPIHT Coding over the AWGN channel.

SPIHT wavelet encoding is a higher compression encoder than EZW coding, therefore uncorrected erroneous packets experience greater packet erasure due to the error propagation experienced during decoding. When investigating erroneous channels, the more destructive the channel, the greater the impact there is of error propagation on uncorrected packets.

The packet erasure rate for the SPIHT wavelet algorithm for the Gaussian channel illustrates the proposed system's improved error correcting capabilities over System 1, System 2 and System 3. Figure 7-9 is the simulation results generated for the PER versus E_b/N_0 for the AWGN channel using SPIHT wavelet compression. For the SPIHT wavelet algorithm, the proposed system out performs the three systems in terms of the number of uncorrected data packets. At high noise ratios ($E_b/N_0= 8, 9$ $10dB's$) the erasure rates of the three compared systems are similar however, there is a significant difference between the proposed system and the three other systems as the proposed system reduces errors by *90%* compare to the other three systems.

The erasure difference between System 1 and System 3's results in comparison to the results produced by the these systems in the EZW AWGN simulations shows that with the increased compression provided by SPIHT coding and additional compression due to the arithmetic coding, error propagation becomes significant in the results as the decoding processes of the two systems (System 1 and System 3) propagates the errors much more in SPIHT coding than EZW coding. Thus higher compression, coupled with increased channel errors causes greater catastrophic error propagation resulting in high erasure rates.

Figure 7-9: Diagram of PER vs. SNR (E_b/N_0) for SPIHT over AWGN channel for Lena image for the Proposed System against System 1, System 2, System 3.

The SPIHT algorithm offers greater compression with improved image quality for a given bitrate. This is depicted in Figure 7-10 where the PSNR values across the spectrum of bitrates produce a high performance. The SPIHT proposed system exhibits increased image quality performance at both low bitrates through to high bitrates. The simulation results indicate that the proposed system responds well in terms of error correction in conjunction with the SPIHT compression technique.

System 2 does not sufficiently correct SPIHT compression and channel induced errors as well as the proposed system resulting in an underperforming system. System 1 and System 3 produce lower quality results as compared to System 2 for these simulations as the increased compression offered by arithmetic coding causes greater error propagation as even one bit error within a packet can propagate further when decoded lowering the image quality.

It can be concluded that the increase in compression does affect the three systems in its correcting abilities, but does not decrease the proposed codec's efficiency in correcting errors thereby minimising error propagation. The average SNR per bit for the AWGN channel was set

160

to *7dB*, a slightly degraded channel consistent with all other PSNR simulations, which is used to illustrate the systems correction capabilities under channel degradation. This result confirms the SPIHT proposed system's ability to handle compression together with channel errors with greater efficacy, producing less error propagation than the other three systems. In terms of PSNR simulations, the critical difference in results occur with the decompression of attempted correcting of channel errors. The image quality directly indicates the decoder's efficiency in correcting results to prevent error propagation.

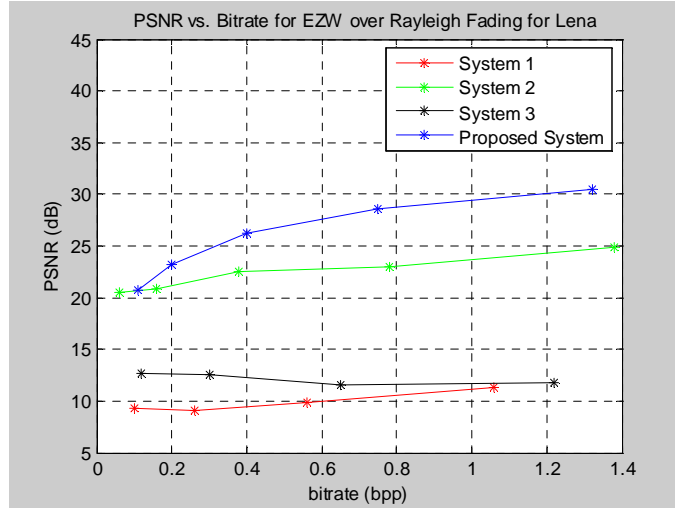Figure 7-10: Diagram of PSNR vs. bitrate for SPIHT over AWGN channel for Lena image for the Proposed System against System 1, System 2, System 3.

7.2.1.4 SPIHT Coding over the Rayleigh Fading channel.

SPIHT coding is known to be more efficient than EZW coding in terms of compression. However, this increase in compression coupled with the destructive nature of the Rayleigh fading channel produces greater channel errors for a given SNR which produces greater erasure performance if the channel errors are not successfully corrected.

The packet erasure performance (PER) of SPIHT wavelet coding across the Rayleigh channel is given in Figure 7-11. The proposed system is able to successfully correct channel errors introduced by the highly destructive Rayleigh multipath fading channel. It is evident from the erasure performance that the proposed system's error correcting capability is superior even when a considerable amount of errors are induced by both SPIHT compression and the Rayleigh fading channel. The addition of the forbidden symbol for error detection and double correction offered by both the MAP metric sequential decoder and ARQ retransmission strategy produces excellent erasure results where the results show the SPIHT coder producing almost no packet

erasures throughout the SNR range. The other three systems exhibit a higher erasure rate at mid to low range SNR's (typically less than 6dB's) as compared to EZW coding. This increased erasure rate is attributed to the increased compression of the SPIHT coding and the increased channel errors of the Rayleigh fading channel.

System 2 manages to perform slightly better than System 1 and System 3 due to the lack of additional compression provided by the arithmetic coding which appears to contribute more significantly to increased error propagation and thus erasure performance. The erasure rates for both System 1 and System 3 are particularly high for this set of simulations, where the proposed system is able to reduce erasures by *90%*. This is due to the arithmetic coding algorithm introducing slightly greater compression, producing a decrease in the total amount of information transmitted. This overall decrease in transmitted information combined with a highly corruptible Rayleigh channel which introduces additional errors into the stream, produces a bitstream of errors which cannot easily be corrected by System 3 as the small amount of compressed information is corrupted beyond recognition. Since no other additional error correction mechanism is used, unlike the proposed system, System 3 fails to correct the error bitstream producing higher packet erasure performance.
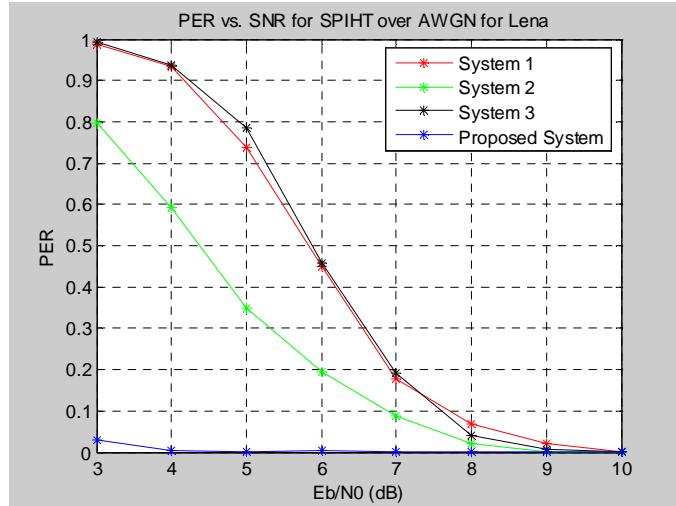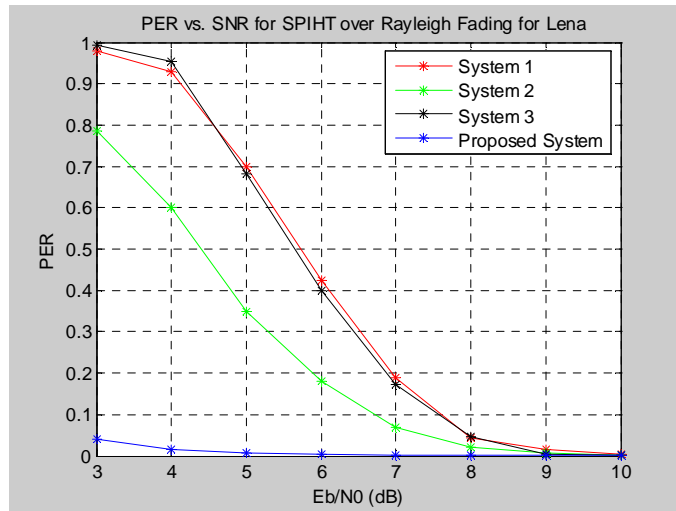

Figure 7-11: Diagram of PER vs. SNR (E_b/N_0) for SPIHT over Rayleigh Fading channel for Lena image for the Proposed System against System 1, System 2, System 3.

The simulated results for the SPIHT wavelet coder with the particularly error prone Rayleigh fading channel represents a worst case scenario for image induced errors. Figure 7-12 depicts this erroneous case for the proposed system and the three other systems using a channel SNR of *7dB*. All systems perform less in terms of image quality showing that the error correction has decreased throughout the bitrate spectrum due to the channel being heavily degraded. Although the three other systems System 1, System 2 and System 3 perform worse, the proposed system

still manages to correct errors producing a high PSNR of *20dB* for the lowest bitrate of *0.1bpp* and even higher PSNR values for majority of the higher bitrates (greater than *0.5bpp*), which implies that its error correction and detection scheme can perform well even with a heavily degraded channel as compression increases and can withstand most errors whether it is compression or channel induced during decompression and decoding.

System 1 and System 3 produce inferior results to System 2 as the integration of arithmetic coding introduces additional compression thereby causing catastrophic error propagation. Nonetheless all three systems still produce lower quality results in relation to the proposed system.
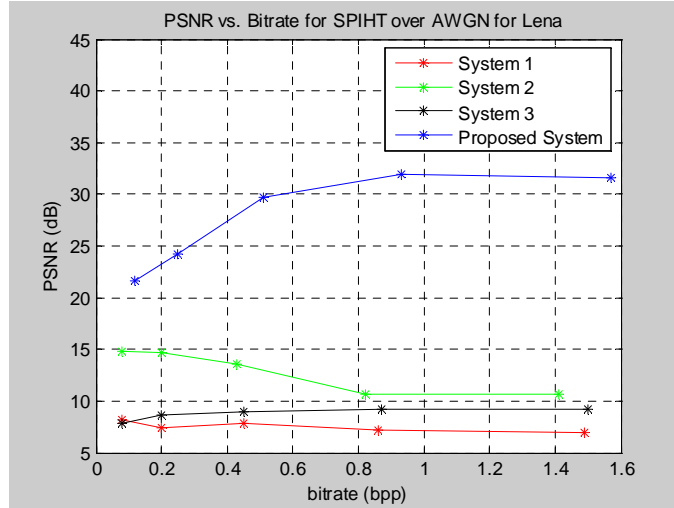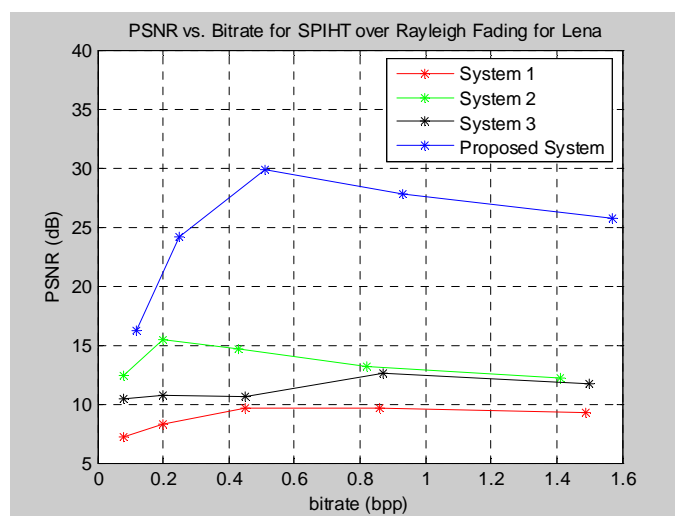

Figure 7-12: Diagram of PSNR vs. bitrate for SPIHT over Rayleigh fading channel for Lena image for the Proposed System against System 1, System 2, System 3.

7.2.2 Barbara

The Barbara [85] test image is generally an excellent test image to use in the testing of image processing and image compression as it contains a significant amount of fine detail. The high frequency detail is contained in the woman's clothing and the table cloth where texture variation is found. Unlike the Lena image, the Barbara image contains a mixture of flat regions, shading with a great deal of emphasis placed on fine detail and texture. The 256x256 grayscale Barbara test image used in the simulation is shown in Figure 7-13.

Figure 7-13: Barbara Test Image [85].

The performance simulations for wavelet compression coding schemes, wireless channels and error decoding methods are executed on the Barbara image and the results follow. The two sets of performance metrics, PER and PSNR are also used in this set of results to evaluate the systems performances.

In the following Barbara image simulations, performances of the systems using both EZW and SPIHT compression algorithms are tested against each other on each of the four systems under both channel conditions. These simulations test the performance of a system under both compression coding schemes and under both erroneous channel conditions. The Barbara image being a heavily fine detailed image may prove more challenging for the correction scheme as error propagation due to compression of fine detail may affect the results more than other images. Thus this image is included in the set of images for testing as it is an excellent image for compression analysis. Essentially these results will determine the best compression scheme in the presence of channel errors and fine detail in terms of packet erasure performance and image quality for a specific system.

7.2.2.1 System 1 over the AWGN and Rayleigh fading channels

SPIHT wavelet coding was designed to be an improvement to EZW coding and therefore applies more compression to a bitstream. The set of results use a slightly higher compression for SPIHT than for EZW, thus higher compression can cause more erasure. The AWGN and Rayleigh multipath fading channels when combined with compression can perform differently, as a combination of channel errors with greater compression can cause more extreme error propagation during decoding like those experienced for SPIHT compared to EZW. The results

in Figure 7-14 show how the channels with the wavelet coding schemes perform against each other in erasure terms using a packet of 20 bits for the packet erasure performance. PER performance is determined by comparing a *20-bit* packet of data before transmission and channel interferences with the same packet after arithmetic decoding which is after decompression. Thus the results will provide a clear view on how the system is affected by errors introduced during transmission, and how the final decompressed bitstream is eventually affected in terms of packet errors.

The System 1 simulations involve only arithmetic coding and decoding, an additional entropy coding stage for compression, which is used to compress the wavelet bitstream further before transmission over the channel. Since System 1 involves no error correction and only compression, the results show the error propagation experienced in the decompression stages due to the combined effect of the wavelet compression and channel conditions.

The EZW coder under both the channels degrade with similar responses and exhibit similar erasure performances as no correction takes place and the results only show how the EZW coder is affected with Gaussian and Rayleigh fading. The same applies to SPIHT coding, where the results of the channels interference is the same however, a difference in erasure is observed between the wavelet coders. This implies that additional compression introduced by the efficiency of the SPIHT coding and the lower bitrate used contributes more significantly to the error propagation. The SPIHT coder at a bitrate of *0.7bpp* performs worse than the EZW coder at a bitrate of *0.83bpp* due to SPIHT coding producing greater compression due to the lower bitrate which is the amount of compressed bits for the entire image as per its design. The addition of arithmetic coding compression to the already increased compression of SPIHT coding produces greater error propagation in the bitstream when exposed to significant channel errors by both the Gaussian and Rayleigh fading channels. Thus the increased compression of the SPIHT algorithm results in a worse erasure performance than the EZW for System 1.

Figure 7-14: Diagram of PER vs. SNR for System 1 for EZW and SPIHT coding over AWGN and Rayleigh Fading channel for Barbara.

The PSNR versus bitrate simulation shows that as the compression increases or decreases with either an AWGN or Rayleigh fading channels destruction. The results show how the system is affected by the errors in terms of error propagation in the final image quality produced. Error propagation is caused when the decoding stage cannot decode or uncompress the bitstream effectively due to the errors affecting the bitstream and changing its bit structure during transmission. Essentially an increase in compression coupled with a channel's destruction can ultimately cause error propagation that results in a decoded image that offers low visual quality due to the decoding of errors. The systems bitrate can be changed according to the arithmetic coding and EZW compression of the system, resulting in more or less compressed bits being transmitted across the channel. Since System 1 only offers additional compression and no error correction, channel errors and error propagation will affect the system more so than the other systems.

The image quality results for the Barbara image are lower than the Lena image owing to the increased fine detail of the woman's clothes which is easily affected by the EZW and SPIHT compression schemes and the additional arithmetic coder. It is essential to allow for each system to be tested on a range of images to address issues such as fine detail with compression and observe how it can impact image quality results. The Gaussian channel performs well in terms of image quality as it affects the EZW and SPIHT decompression stages less, as less destructive channel errors are introduced in the transmissions sequence. The Rayleigh fading channel being more aggressive has a poor image quality influence on the results. The results also show that SPIHT being a more efficient algorithm is able to produce better quality results than EZW for the same channel conditions.
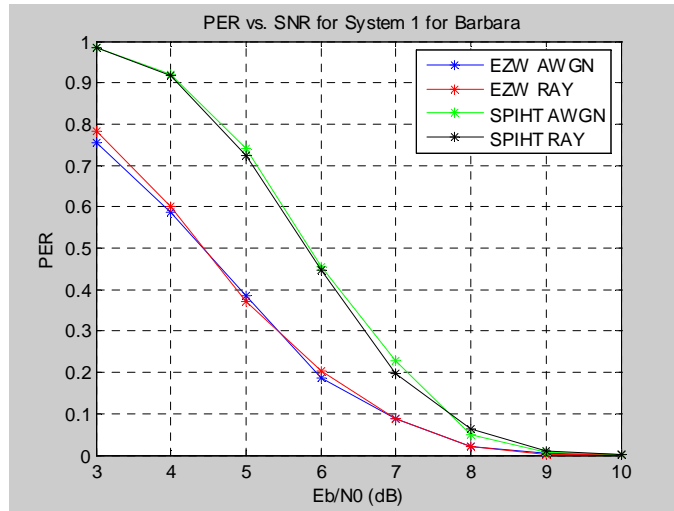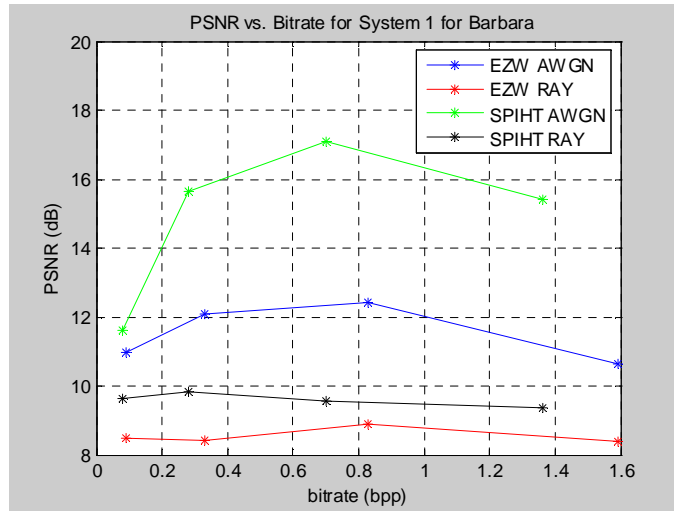
Figure 7-15: Diagram of PSNR vs. bitrate for System 1 for EZW and SPIHT coding over AWGN and Rayleigh Fading channel for Barbara.

7.2.2.2 System 2 over the AWGN and Rayleigh fading channels

System 2 uses convolutional coding and MAP metric sequential decoding for error correction. Both the EZW and SPIHT compression schemes have less packet erasures in System 2 than System 1, due to the correction of erroneous packets. The convolutional coding with *maximum a posteriori* (MAP) metric sequential decoding proves to be competitive as an error correcting scheme as a reduction in erasure performance is seen.

In Figure 7-16 System 2 is evaluated and shows that it is able to process fine detail and high texture variation as found in the Barbara test image with great accuracy and precision as its erasure rates are lower than those produced with System 1. The systems error correction employing prediction of the transmitted stream is able to handle the degradation of the Rayleigh channel competently. The inclusion of the MAP decoder in System 2 shows a higher quality performance for the Barbara test image.

The EZW AWGN and EZW Rayleigh results outperform the SPIHT AWGN and SPIHT Rayleigh results for System 2. The packet erasure results are shown in Figure 7-14. The difference in erasures for EZW and SPIHT show that with increased compression and channel errors the SPIHT scheme cannot perform better than the EZW scheme as this increased compression can cause greater error propagation if even *1* bit is in error in a packet. However, the difference between erasure performances of EZW and SPIHT compared to that seen in System 1 is greatly reduced due to the error correction inherent in System 2. For comparisons of EZW and SPIHT for each set of channel simulations, System 2 reduces the erasures compared

to System 1 by *17%* and *15%* for EZW AWGN and Rayleigh and *40%* and *56%* for SPIHT AWGN and Rayleigh respectively.

The results show that the channel errors have significance in the error correction performance. This can be seen when observing the erasure results at high SNR's versus low SNR's. At high SNR's where there are negligible errors all four simulations perform the same however, at lower SNR's the error propagation takes effect as high compression schemes like SPIHT AWGN for System 2 performs worse as the error correction does not correct all packets and once decoded, causes greater error propagation than a scheme with less compression like the EZW AWGN for System 2.


Figure 7-16: Diagram of PER vs. SNR for System 2 for EZW and SPIHT coding over AWGN and Rayleigh Fading channel for Barbara.

The simulation results for the PSNR versus bitrate for the wavelet coding schemes, for both channels for System 2 is shown in Figure 7-17. All the simulations operate on a channel average SNR of *7dB* for both AWGN and Rayleigh fading as it introduces channel destruction allowing the results to show how the system performs with channel errors.

The SPIHT System 2 results in Figure 7-17 perform very poorly for various bitrates when compared to the proposed system in the next chapter. This is due to the increased compression experienced by the SPIHT algorithm causing greater error propagation in uncorrected packets. Since all results are generated by System 2, the poor results between EZW and SPIHT cannot be attributed to the error correction process per say but rather to the error propagation of erroneous packets. Packet erasure as seen in Figure 7-16 does not take into account erroneous bits in the packet thus if *1 bit* is in error in a packet, the entire packet is considered as being in error. Therefore slight difference in erasure rates of the simulations cannot be a sufficient

representation of the destruction produced in the final image quality. Hence the PSNR will give a view of the entire decoded bitstream in error.

In the results the Rayleigh fading channel produces slightly worse results than the Gaussian channel verifying that the channel is indeed more destructive. These poor results exhibited by the SPIHT algorithm, as mentioned above, can be completely attributed to the small amount of compressed information produced by the SPIHT algorithm combined with the highly degraded and destructive Rayleigh fading channel, producing uncorrected error and irreversible error propagation resulting in extremely poor performance. The EZW wavelet algorithm behaves moderately better with the Rayleigh channel, due to the amount of compression produced by the scheme. The more information (less compression), the greater likelihood the errors can be corrected and less error propagation can then occur. Having greater compression by use of SPIHT coding also causes worse image quality if the system is not used properly in error correction. Thus a well established system integrating error correction for effective correction is necessary in any system design. This is proved later with the proposed system.

Figure 7-17: Diagram of PSNR vs. bitrate for System 2 for EZW and SPIHT coding over AWGN and Rayleigh Fading channel for Barbara.

7.2.2.3 System 3 over the AWGN and Rayleigh fading channels

System 3 uses both arithmetic coding and decoding as an entropy coding stage for compression and convolutional and MAP metric sequential decoding as the error correction mechanism. No forbidden symbol error detection is included in the system or any alternative error correction procedure if the CC-MAP decoding fails due to extreme channel destruction.

System 3's erasure results are shown in Figure 7-21. These results are higher than System 2 and System 1, as System 3 has more bits and packets in its structure as it combines both arithmetic coding and convolutional coding which with a code rate of ½ results in double the amount of bits than the other two systems. This can result in greater erroneous packets identified at lower SNR's as even a *1 bit* uncorrected error in a packet renders that packet erroneous. As with the previous erasure results, the EZW wavelet compression coding technique is able to perform better than the SPIHT compression technique for all noise ratios with the same channel characteristics albeit it's lower compression with a bitrate of *0.96bpp* compared to the SPIHT bitrate of *0.71bpp*. This can be attributed to the less compression involved in the EZW wavelet coding scheme in terms of producing error propagation induced by the wavelet technique over transmission and decoding stages.
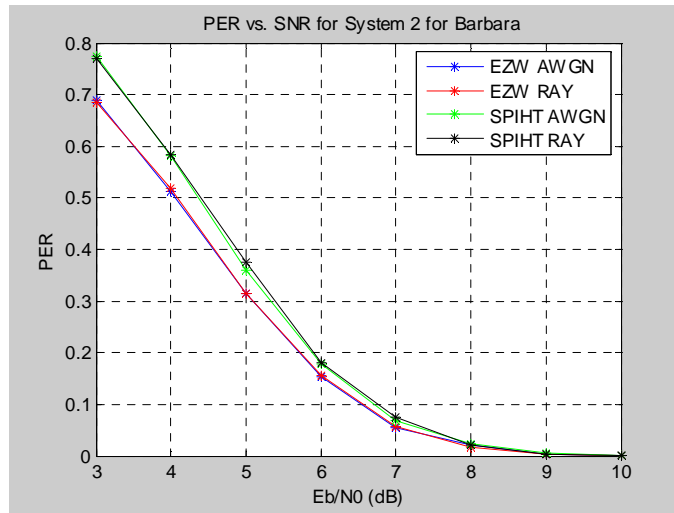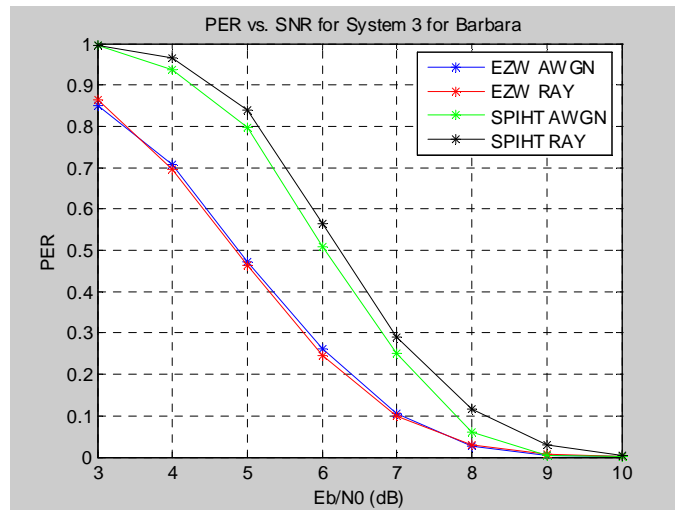

Figure 7-18: Diagram of PER vs. SNR for System 3 for EZW and SPIHT coding over AWGN and Rayleigh Fading channel for Barbara.

The PSNR versus bitrate results are given in Figure 7-19. These results show how the system reacts with decompression and decoding of errors as the bitrate changes. SPIHT AWGN for System 3 performs best, yielding approximately a *3dB* performance improvement over EZW AWGN, EZW Rayleigh and SPIHT Rayleigh respectively. The SPIHT AWGN system does well due to the less destruction experienced by the Gaussian channel. The SPIHT Rayleigh results perform worst as this drop in signal quality is due to the Rayleigh channels erroneous nature and the decoder's inability to successfully process and correct the errors produced thereby causing greater error propagation and less successful image quality results.
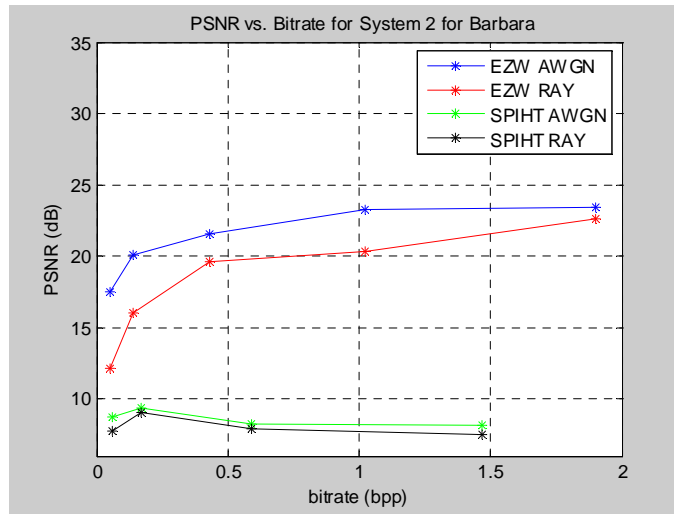
Figure 7-19: Diagram of PSNR vs. bitrate for System 3 for EZW and SPIHT coding over AWGN and Rayleigh Fading channel for Barbara.

7.2.2.4 The Proposed System over the AWGN and Rayleigh fading channels

The proposed system uses arithmetic coding and convolutional coding at the encoder and MAP metric sequential decoding, ARQ retransmission protocol and arithmetic decoding at the decoder. This system exploits arithmetic coding for error detection by introducing the forbidden symbol technique, therefore arithmetic can be used as both an entropy coding stage for additional compression and error detection. The MAP metric sequential decoder is the first option for error correction. If the arithmetic decoder detects a forbidden symbol this is an indication that the MAP decoded packet has failed and the packet is still erroneous. This forbidden symbol detection then invokes the ARQ retransmission strategy which prompts the transmitter to resend the packet in the hope that the retransmitted packet is corrected by the MAP decoder.

The erasure performance sees a marked improvement by the proposed system, as the results in Figure 7-20 when compared to other system's erasures is primarily due to the combination of the ARQ retransmission protocol and MAP decoding technique specifically. However, the FS error detection when combined with the MAP decoder and ARQ strategy provides an error protection mechanism able to withstand highly degraded channels and high compression techniques. The erasure results of the other systems compared to the proposed system drops from an erasure rate of *0.8* to *0.15*, an *80%* reduction in erasure.

As seen with the previous erasure rates, the EZW proposed system performs better than the SPIHT simulations. At high SNR's both the wavelet coders perform the same however, at lower

SNR's EZW performs better as less compression produces less information during transmission, which can be affected less by destructive channels which reduces the error propagation of uncorrected packets. The proposed system performs exceptionally well for the wavelet coding schemes and channel conditions as compared to the previous simulations of the three other systems. Particular attention should be given to the major decrease in overall erasure for all the simulation combinations.
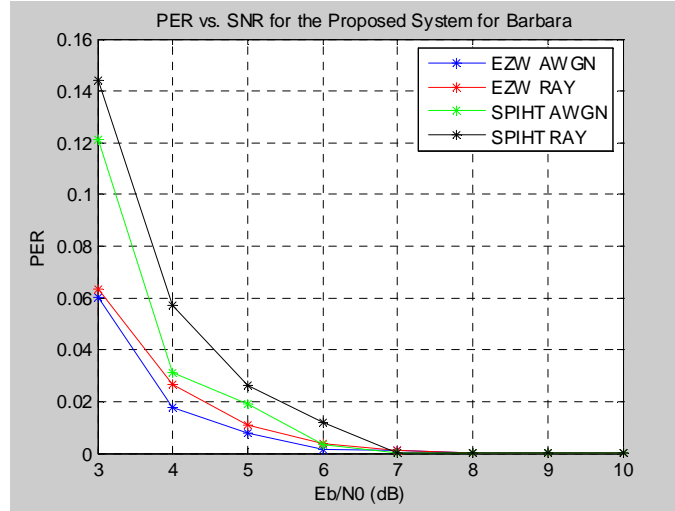

Figure 7-20: Diagram of PER vs. SNR for the Proposed System for EZW and SPIHT coding over AWGN and Rayleigh Fading channel for Barbara.

Simulation results in Figure 7-21 reveal the image quality performance of the proposed system with forbidden symbol detection. The image quality results for the two wavelet algorithms and channels exhibit very similar image quality outcomes with minimal quality differences. The results also show that for the proposed system as the bitrate increases (compression decreases) the image quality increases. This has been highlighted previously where less compression means less information transmitted through an error prone channel, therefore less destruction can take place in terms of error propagation and final image quality produced in the decompression stages.

The SPIHT AWGN results for the proposed system with FS detection performs best in terms of image quality produced for lower overall bitrate or increased compression. The EZW AWGN and Rayleigh fading simulations show similar responses showing the slight difference between channel conditions for this specific wavelet coder. However, the SPIHT Rayleigh results perform worst as high compression offered by SPIHT coding and high channel destruction produced by the Rayleigh fading channel produces a decrease in overall image quality. Compared to the previous systems the PSNR values remain relatively high showing superior results.
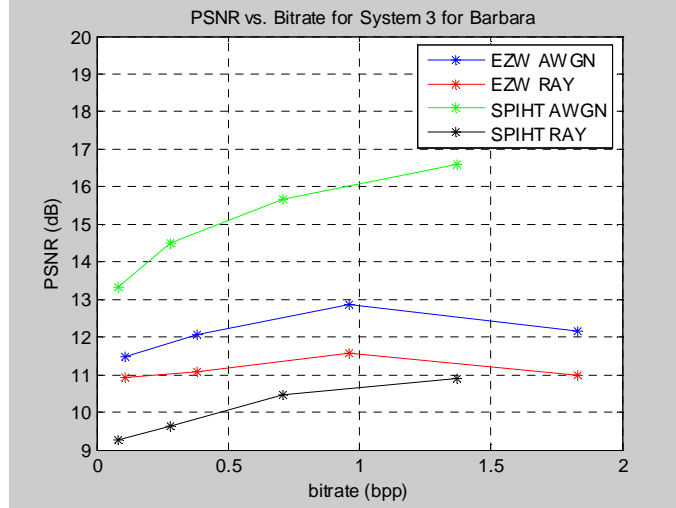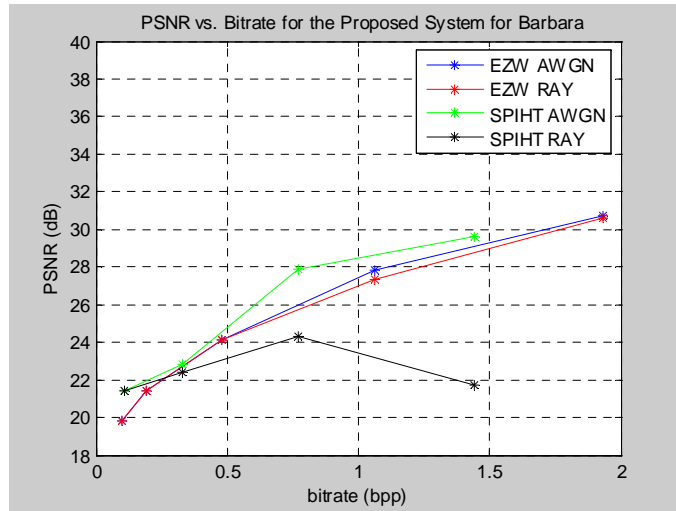
Figure 7-21: Diagram of PSNR vs. bitrate for the Proposed System for EZW and SPIHT coding over AWGN and Rayleigh Fading channel for Barbara.

The image quality differences between SPIHT AWGN and Rayleigh images are visually illustrated in Figure 7-22. The AWGN channel is a less destructive channel than the Rayleigh fading channel and produces less error propagation in its decompression stages. This is visually depicted in the images where an even texture is seen for the image, whereas the Rayleigh channel has more destruction. It can be noted that the proposed system corrects the errors far better than other systems as the woman is still visible whereas other systems produce completely degraded images.
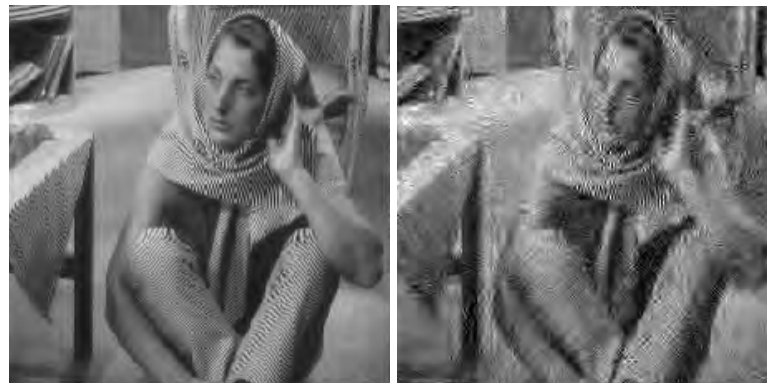
Figure 7-22: (a) Barbara image for SPIHT AWGN channel at a bitrate of *0.77bpp* (b) Barbara image for SPIHT Rayleigh Fading channel at a bitrate of *0.77bpp*.

7.2.3 Cameraman

The Cameraman [85] test image is a standard image processing and image compression test image. Unlike the Lena and Barbara images, the Cameraman image contains a significant amount of flat regions and high contrasting shades. The low frequency detail or flat regions is contained in the sky and cameraman's coat where low intensity and high intensity grayscale

shades are found. The Cameraman image contains a mixture of flat regions, high contrast shading, fine detail and texture. The simulations are executed on the 256x256 grayscale Cameraman test image shown in Figure 7-23.

Figure 7-23: Cameraman Test Image [85].

The cameraman image performance results use simulations of all four systems with the EZW and SPIHT wavelet coding scheme across either the Gaussian or Rayleigh fading channel. These results help determine the best system in terms of error correction in the form of packet erasure and image quality production in terms of PER and PSNR. Images are also used to illustrate the visual impact produced by error propagation and channel degradation in the system.

7.2.3.1 EZW Coding over the AWGN channel.

The packet erasure rate for the cameraman image is similar to the previous erasure rates for the other test images. The proposed system simulation achieves the best erasure rate as its correction scheme performs best. As with the Barbara and Lena test image PER simulations, the proposed system performs best throughout the SNR range. The associated PER versus E_b/N_0 for the cameraman test image is illustrated in Figure 7-24.

The AWGN channel simulations for the EZW algorithm show the proposed system reducing the errors by an estimated *98%*. It is evident the proposed system's ARQ-MAP error correction combination maintains its error performance lead over the variations in the other systems for Gaussian induced noise with test images having a majority of flat regions and low frequency detail. The proposed system maintains a good erasure performance lead against System 1,

174

System 2 and System 3. The results validate the ARQ retransmission protocol and the MAP metric decoder's capability to correct errors with greater accuracy and exactness. It essentially offers greater decoding abilities and is a preferred error coding mechanism.

Figure 7-24: Diagram of PER vs. SNR (E_b/N_0) for EZW over AWGN channel for Cameraman image for the Proposed System against System 1, System 2, System 3.

The cameraman PSNR simulations depicted in Figure 7-25 show similar responses as previously generated simulations. All simulations are generated using a channel E_b/N_0 of *7dB* to introduce channel errors into the transmitted bitstream in order to view how the systems results are affected by channel conditions. The proposed system performs well as observed throughout the set of test images however,

System 3 does not perform as well as System 2 when arithmetic coding is introduced. In the simulations, System 2 only includes error correction, whereas System 3 introduces arithmetic coding which is observed to reduce the image quality. A decrease in performance improvement is experienced for System 3, which involves the MAP decoder with arithmetic coding, as the system does not exploit forbidden symbol detection and is used as a pure entropy coding stage for additional compression. This additional compression is causing greater error propagation in the system resulting in System 3's image quality to be more poor than System 2. The same can be said for System 1 which also uses arithmetic coding that is contributing to greater error propagation as even one bit error in a packet can be propagated further during decoding.
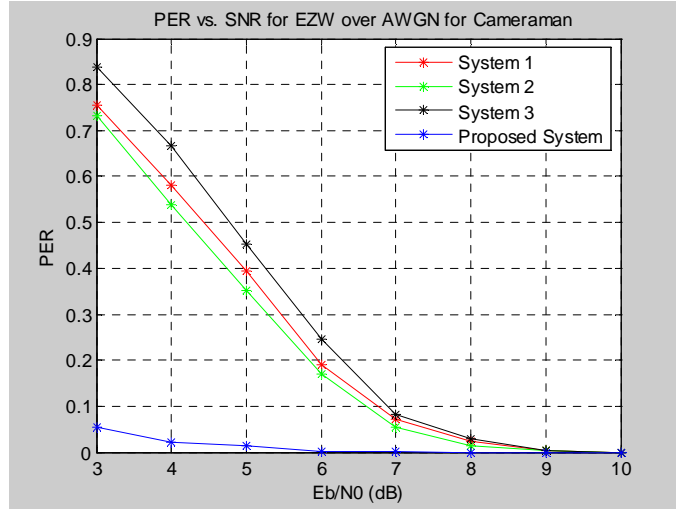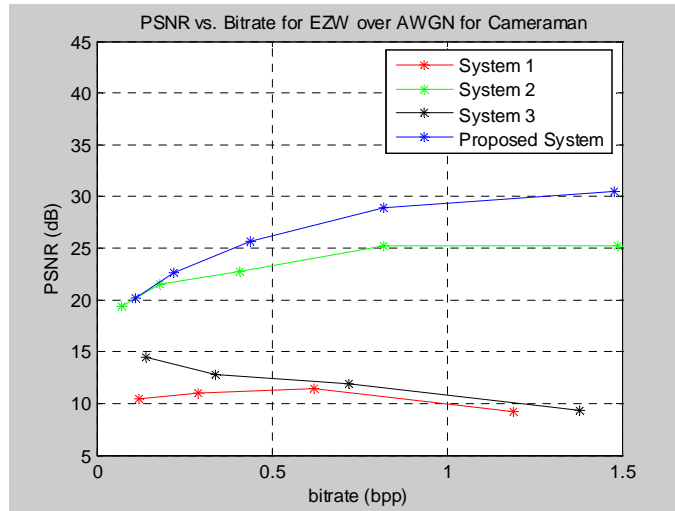
Figure 7-25: Diagram of PSNR vs. bitrate for EZW over AWGN channel for Cameraman image for the Proposed System against System 1, System 2, System 3.

The images shown in Figure 7-26 attempt to analyse and interpret the performance behaviour of the different systems in terms of final image quality produced after decompression. Figure 7-26 (a) – (d) are the image quality results for the four systems; the proposed system, System 1, System 2 and System 3. Visually the proposed system corrects the errors and produces a visually clear and accurate replica of the original image. System 2 has a higher PSNR than System 1 and System 3, and is more visually defined and less corrupted and blurry. This is a result of the Gaussian channel inducing fewer channel errors. Figure 7-26 (c) and (d) are very blurry, corrupted and visually indefinable and is a result of uncorrected channel errors and catastrophic error propagation for System 1 and System 3.

In image processing, high frequency detail in the form of these black and white segments is catastrophic to the image quality as it signifies error propagation in the decompression stages. The image for the proposed system, Figure 7-26 (a), is extremely defined and far clearer than System 3 in Figure 7-26 (c), where the black and white segments suggest a greater deviation from the original image due to induced channel errors resulting in error propagation producing a lower quality image. The shows that the inclusion of the forbidden symbol and ARQ retransmission protocol for the additional error detection and correction of the system produces a vast improvement in performance and image quality.

From the destructed results of System 1 and System 3, the cameraman image is still slightly defined by the dark region. This indicates that low frequency detail in the form of flat regions and shading found in the cameraman image are less affected by errors propagation and thus easier to correct and decode than the high detail seen in the Barbara image. High detail causes greater catastrophic error propagation than flat uniform tone regions.

<p style="text-align:center">(a)   (b)</p>

<p style="text-align:center">(c)   (d)</p>

Figure 7-26: Cameraman image for EZW AWGN for
(a) Proposed System with a PSNR of *28.95dB* at a bitrate of *0.82bpp*
(b) System 2 with a PSNR of *25.18dB* at a bitrate of *0.82bpp*
(c) System 3 with a PSNR of *11.94dB* at a bitrate of *0.72bpp*
(d) System 1 with a PSNR of *11.4dB* at a bitrate of *0.62bpp*

7.2.3.2 EZW Coding over the Rayleigh Fading channel.

The Rayleigh fading channel represents a worst case scenario for channel degradation as it is a more destructive channel than the Gaussian channel. A fading variance of $\sigma^2=0.5$ is used with a packet length of *20 bits*. The same systems as highlighted previously are simulated using the Rayleigh fading channel in Figure 7-27.

The Rayleigh fading channel simulations show the proposed system performing better than the three other methods. A *98%* erasure reduction of the proposed system over the System 1, System 2 and System 3 was observed for the EZW Rayleigh simulations. However, almost exact performances are seen for the EZW AWGN systems over its Rayleigh channel counterparts. This is unusual as the erasure value for a Rayleigh channel should be higher than the Gaussian channel as it is a less aggressive error channel. But this could be related to the fact the cameraman image has less fine detail and variation and greater areas of flat grayscale regions which is affected less by the channels errors.
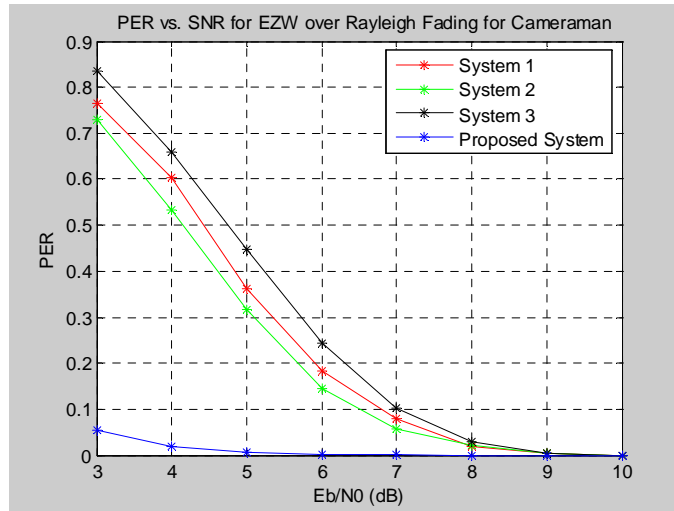
Figure 7-27: Diagram of PER vs. SNR (E_b/N_0) for EZW over Rayleigh fading channel for Cameraman image for the Proposed System against System 1, System 2, System 3.

The image quality results show the competitiveness of the proposed system's ARQ-MAP metric sequential decoder against System 1, System 2 and System 3. The System 2 PSNR values at the lower bitrates (less than *0.5bpp*) are not as poor as seen in the other test images. However, the proposed system is still able to prove its error correcting performance against System 2 in the visual image quality results that follow. System 2 as with the previous test image results performs better than System 1 and System 3 due to exclusion of arithmetic coding. It has been noted previously that the introduction of arithmetic coding produces additional compression in the system which in turn causes greater error propagation during the decoding stages if errors are not corrected.

The EZW Rayleigh fading channel performances are similar to the Gaussian channel and do not vary much. It can be noted that as the bitrate of the systems increase, which constitutes less compression, the error propagation decreases in the decompression stages resulting in better quality images as seen in the gradual incline of the PSNR values in Figure 7-28.

Figure 7-28: Diagram of PSNR vs. bitrate for EZW over Rayleigh fading channel for Cameraman image for the Proposed System against System 1, System 2, System 3.

For PSNR image quality simulations it is always best to view the images visually as this can show how slight erasure differences can impact vastly on an image. Figure 7-29 (a) to (d) show each system performance at a bitrate between *0.5* and *1* as per the PSNR-bitrate simulation above. The proposed system produces a crystal clear image under EZW Rayleigh channel conditions. Fine details like the cameraman's camera, background image and gloves are easily identifiable verifying the error correction capability of the system. System 2 performs adequately well with subtle areas of destruction particularly edges surrounding the cameraman's figure. This is commonly known in image processing as edge detail which is the movement from low frequency to high frequency detail. Both System 1 and System 3 are affected by the channel errors and decompression error propagation. The System 1 image contains areas with massive black patches. This is considered to be high destruction and results in the PSNR value dropping significantly. Areas of either black or white regions affect the quality drastically.

(a)   (b)

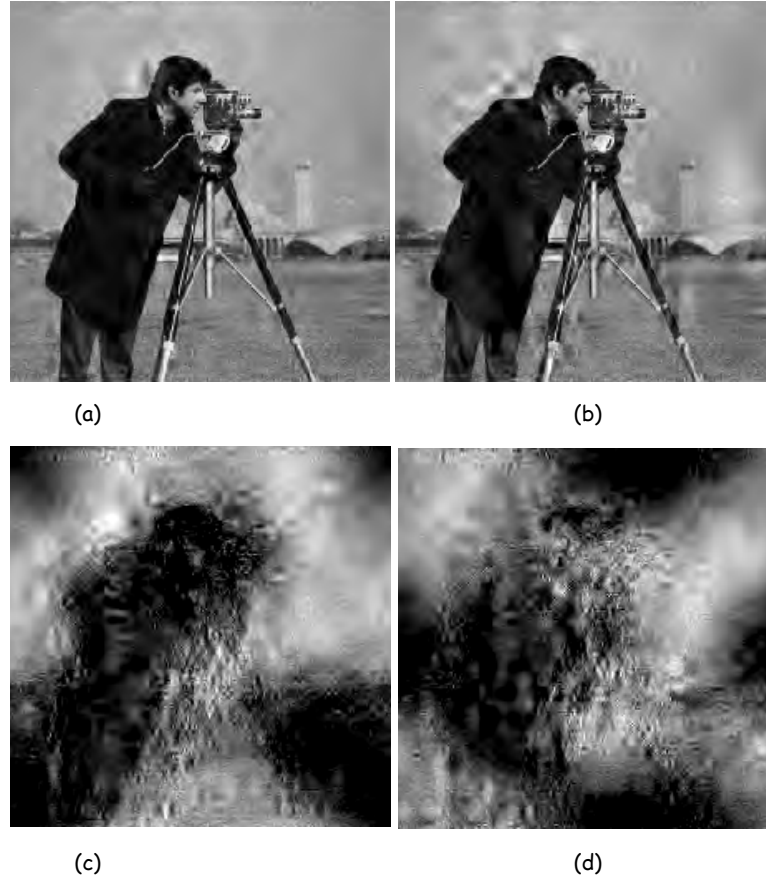<div align="center">(c)   (d)</div>

Figure 7-29: Cameraman image for EZW Rayleigh fading for
(a) Proposed System with a PSNR of *27.82dB* at a bitrate of *0.82bpp*
(b) System 1 with a PSNR of *24.68dB* at a bitrate of *0.82bpp*
(c) System 2 with a PSNR of *11.04dB* at a bitrate of *0.72bpp*
(d) System 3 with a PSNR of *8.94dB* at a bitrate of *0.62bpp*

7.2.3.3 SPIHT Coding over the AWGN channel.

The packet erasure simulations determining the error correcting capabilities of the SPIHT AWGN systems against each other are given in Figure 7-30. The AWGN channel simulations for the SPIHT algorithm show the proposed system reducing the errors by an estimated *97%* compared to the three systems in terms of total error packets produced. The SPIHT AWGN algorithm for System 2 offers a *58%* reduction in erasure packets over System 1 and System 3. This is slightly higher for this set of results than previous test images. It is evident the proposed system maintains its error performance lead for Gaussian induced noise with the cameraman image having a majority of flat regions and low frequency detail.
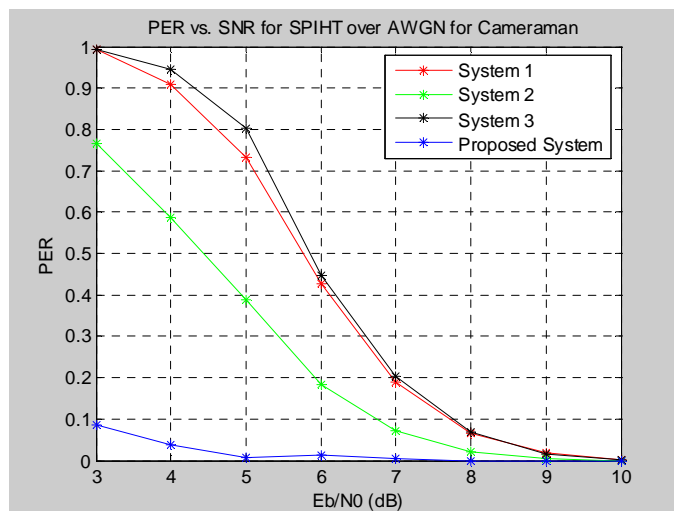
The PSNR quality simulation results for the proposed system with MAP metric sequential decoding with FS detection in comparison with the other systems with no FS detection or ARQ retransmission protocol is illustrated in Figure 7-31. As observed with previous test images, the proposed system performs better than System 1, System 2 and System 3 which unlike the proposed system offers less error correction of its bitstreams and thus exhibits decreased PSNR results for the Cameraman image. It experiences high image quality at low bitrates through to higher bitrates proving that on the whole the proposed system which includes FS detection is a superior system for consistent error correction across the bitrate spectrum. The proposed system for SPIHT AWGN coding with FS detection achieves a *9dB* performance increase against the System 2 without FS detection. System 1 and System 3 without FS detection achieves the same poor PSNR values for the bitrate spectrum.

Once again the results show System 2 performing better than System 1 and System 3 due to the inclusion of arithmetic coding and decoding in the systems. Arithmetic coding introduces additional compression which may cause greater error propagation during decoding. As one error bit in a packet can propagate to numerous error bits if uncorrected during decoding.
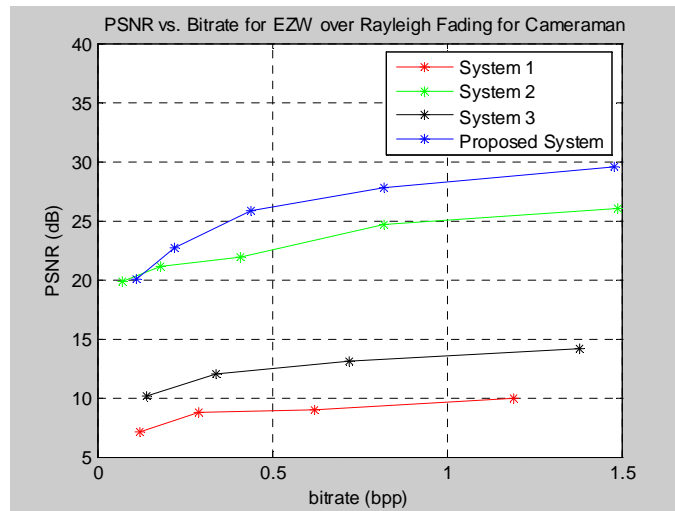

Figure 7-31: Diagram of PSNR vs. bitrate for SPIHT over AWGN channel for Cameraman image for the Proposed System against System 1, System 2, System 3.

The visual images produced by the SPIHT AWGN simulations at a lower bitrate than the EZW simulation are illustrated in Figure 7-32 (a) to (d). As discussed previously a lower bitrate represents greater compression and when combined with an error prone channel can cause greater visual distortion as seen in the following images. Although the proposed system's results

experience high SPIHT compression, additional arithmetic coding compression, added redundancy due to the forbidden symbol inclusion and a destructive channel, the system is still able to produce visually worthwhile results and successful error correction. System 1 experiences extremely catastrophic error propagation. Large areas of black and white regions represent visual degradation that is beyond repair.

<div align="center">(a)</div>

<div align="center">(b)</div>
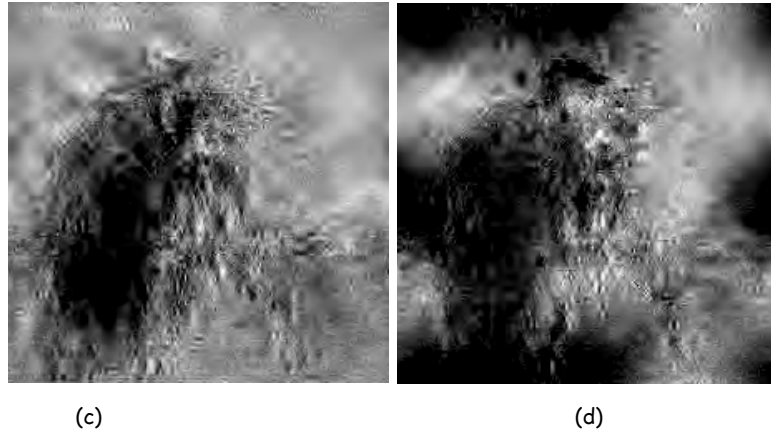
<div align="center">(c)</div>

<div align="center">(d)</div>

Figure 7-32: Cameraman image for SPIHT AWGN for
(a) Proposed System with a PSNR of *27.32dB* at a bitrate of *0.58bpp*
(b) System 2 with a PSNR of *15.18dB* at a bitrate of *0.5bpp*
(c) System 3 with a PSNR of *8.86dB* at a bitrate of *0.52bpp*
(d) System 1 with a PSNR of *7.97dB* at a bitrate of *0.52bpp*

7.2.3.4 SPIHT Coding over the Rayleigh Fading channel.

The simulations for SPIHT coding with the Rayleigh fading channel can be considered as a worst case scenario as seen in the Barbara results for system comparisons of wavelet coders with channels. SPIHT Rayleigh fading simulations performed the worst in the erasure simulations producing the highest erasures experienced per uncorrected packet. Figure 7-34 illustrates the erasure performance for the SPIHT wavelet coding algorithm over the highly error prone Rayleigh fading channel. The proposed system performs best however; the combination of high SPIHT compression having a low bitrate of *0.58bpp* for the proposed

system and the aggressive Rayleigh fading channel produces an erasure rate higher than the previous wavelet encoder simulations. Nonetheless, the proposed system's results prove its performance is superior to the three systems as the results for all the wavelet compression and channel alternatives are illustrated below.

A significant difference in erasures is observed between System 1, System 2 and System 3. This highlights that System 2 which uses no arithmetic coding, produces less erasures in its error correction than System 3. The proposed system proves that by exploiting arithmetic coding for forbidden symbol error detection and using ARQ retransmission as a second error correction mechanism, impeccable erasure results can be achieved. Thus the results prove that the proposed codec which uses each element of the comparison systems for its design is more effective when combined together than used individually.
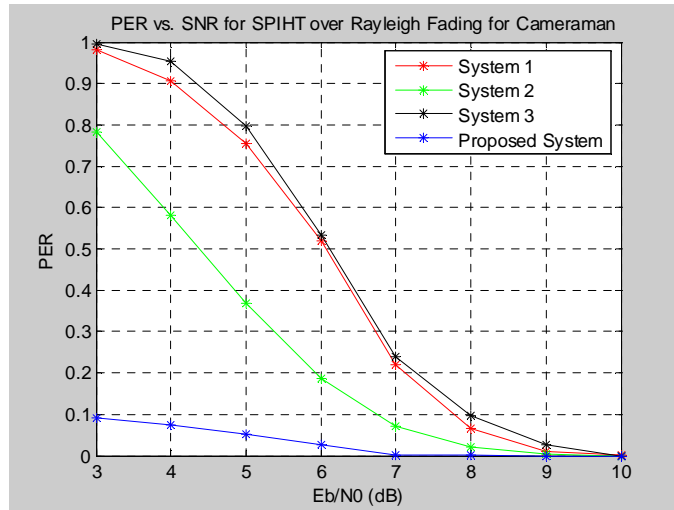
Figure 7-33: Diagram of PER vs. SNR (E_b/N_0) for SPIHT over Rayleigh fading channel for Cameraman image for the Proposed System against System 1, System 2, System 3.

The Rayleigh channel as discussed above achieves less performance improvement than the Gaussian channel. The PSNR results for proposed system across the Rayleigh channel illustrated in Figure 7-34 is slightly less high, unlike the Gaussian channel results. This is a result of the highly contrasted Cameraman image and when compressed more than the EZW encoder and transmitted across the extremely erroneous Rayleigh channel, the systems fail to adequately correct the induced channel errors which propagate throughout the bitstream during decompression producing degraded image quality. The proposed SPIHT Rayleigh system has a PSNR gain of *15dB*, *9dB* and *14dB* against System 1, System 2 and System 3 respectively.

The increased compression included in System 1 and System 3 due to arithmetic coding reduces the system ability to produce high quality images as error propagation becomes evident.
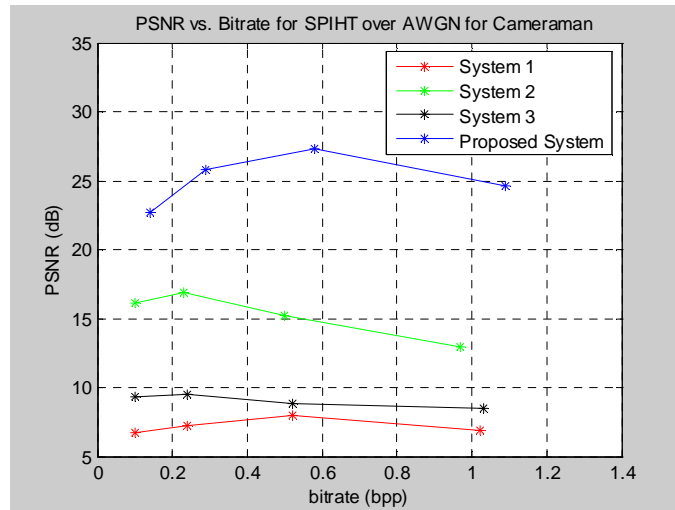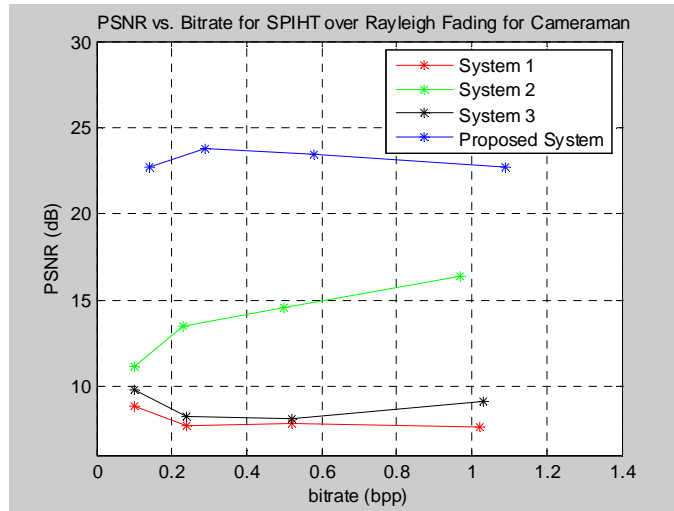
Figure 7-34: Diagram of PSNR vs. bitrate for SPIHT over Rayleigh fading channel for Cameraman image for the Proposed System against System 1, System 2, System 3.

The Rayleigh PSNR results are lower and visually more inferior when compared to the predictable image responses of the Gaussian channel. A visual depiction of the systems is illustrated in Figure 7-35. The proposed system produces an image that is clear and identifiable verifying that its error correction scheme is effective and successful in mitigating channel errors. System 2 and System 3 without forbidden symbol error detection produces extremely poor image quality performance showing the systematic failure of the decoders to process and correct sufficient errors for acceptable results. The difference in the SPIHT Rayleigh proposed system with FS detection results for the Cameraman image compared to the previous Lena and Barbara images is due to the high contrast in the image instead of the fine detail experienced for the previous two images. Errors in images compressed with fine detail experience more catastrophic error propagation and degraded image quality than those with higher contrast.

System 3 performs extremely poorly with visual degradation due to its decoding and lack of error correction. The aggressive Rayleigh channel induces errors that that are too numerous to effectively corrected by the MAP decoder, as the decoder uses a statistical predictive analysis of the channel and metric examination of past bit errors to successfully correct future bit errors. If the quantity of induced errors is too great and the errors occur in a succession, the decoder eventually fails to correct the errors accurately, and the errors propagate throughout the bitstream producing inferior image quality as seen by the SPIHT AWGN System 3 results. The System 1 results also perform poorly. Due to the lack of error correction in the design of the system, very little could be avoided however, this is not the case for System 3. Although System 2 performs well showing that the error correction by itself is effective, the MAP decoding fails as the combination of arithmetic coding and MAP metric error correction does not improve the image quality of the image.
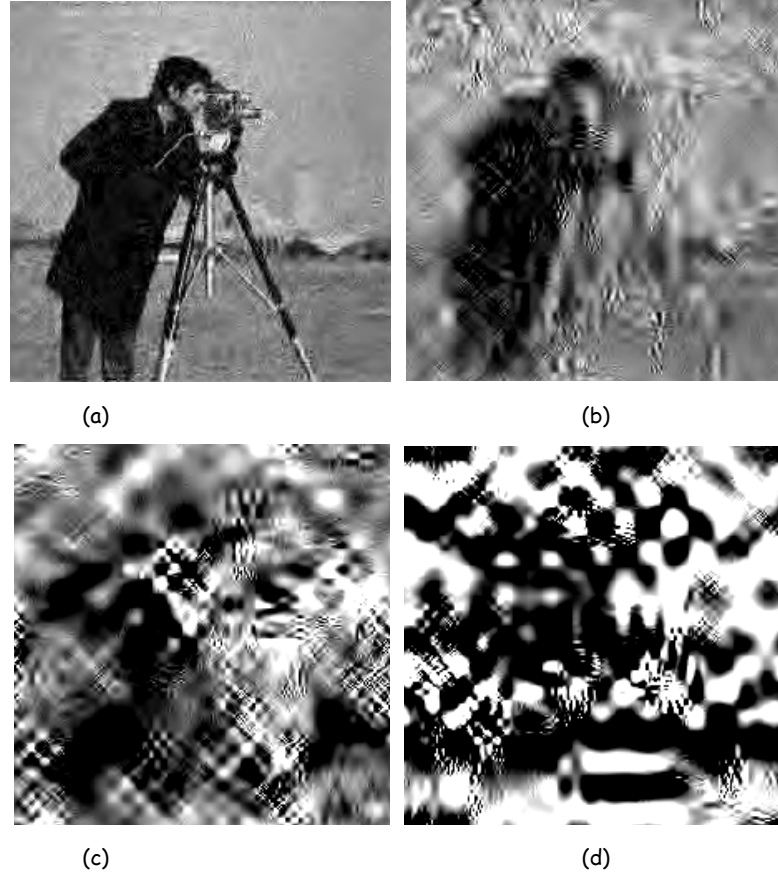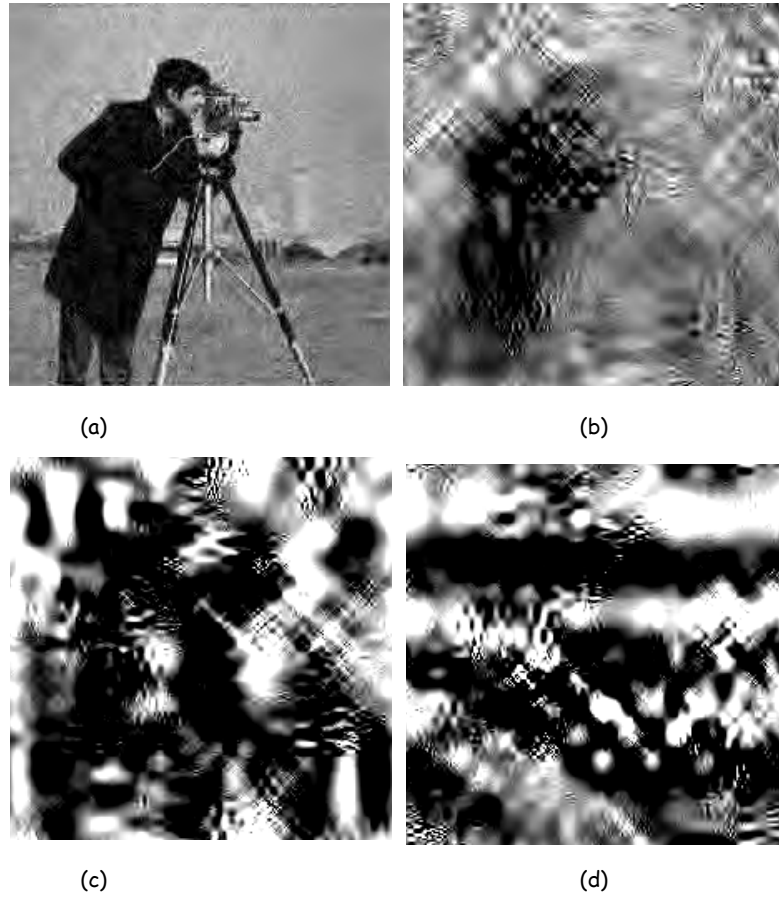
Figure 7-35: Cameraman image for SPIHT Rayleigh for
(a) Proposed System with a PSNR of *23.45dB* at a bitrate of *0.58bpp*
(b) System 2 with a PSNR of *14.57dB* at a bitrate of *0.5bpp*
(c) System 3 with a PSNR of *8.11dB* at a bitrate of *0.52bpp*
(d) System 1 with a PSNR of *7.84dB* at a bitrate of *0.52bpp*

7.3   DISCUSSION OF RESULTS OBTAINED AND CONCLUSION

The simulation results show that the proposed system's ARQ retransmission strategy with the MAP metric sequential decoding algorithm is extremely efficient in solving and correcting the applicable error problems experienced within compression and transmission of images. It also successfully handles the constraints experienced by wavelet based compression and corruptible channels. The superior performance of the error correcting ability of the proposed system using forbidden symbol detection over System 1, System 2 and System 3 without forbidden symbol detection or ARQ retransmission was illustrated through two approaches, the first using wavelet based compression as a corruptible error propagation method through PSNR simulations and the second using destructive erroneous channels to induce errors through PER simulations. The proposed system then illustrated the use of its forbidden symbol error detection technique, MAP metric sequential decoding and ARQ retransmission on performance through simulations

against System 1 which only involves arithmetic encoding and decoding, System 2 which uses convolutional encoding and MAP decoding, and System 3 which uses both arithmetic encoding, convolutional encoding and MAP decoding and arithmetic decoding without the use of the forbidden symbol.

The proposed codec's performance showing the error correcting ability for wavelet compression induced errors, were illustrated using the PSNR versus bitrate simulations. These simulations try to keep channel characteristics and impairments constant while varying the degree of compression induced; through manipulation of the bitrate in order to view the impact error propagation has on the final decoded decompressed image. The greater the compression induced, the less information is transmitted, and thus any errors that occur during transmission through the channel can propagate as a result of inaccurate decoding resulting in a visually poor image. Therefore it is imperative that the decoding and error correcting ability of the decoder's are able to process the channel errors accurately in order to reduce error propagation in the PSNR results. The proposed system with FS detection is able to achieve higher image quality through its error detection and correction method than the three systems that use variations of the proposed system without FS detection and ARQ retransmission for its PSNR simulations.

The Gaussian channel results show fairly consistent performance for all the wavelet encoders and decoding schemes. Typically if channel errors are introduced into either the proposed system or System 1, System 2 and System 3 for the Gaussian channel, the PSNR results will show huge image quality differences between the proposed system and the other three systems. This can be attributed to the error correction of the proposed system being successful, producing no error propagation occurrences in the resultant PSNR images. Differences in the three other systems observed PSNR performances are due to various methods experiencing a greater number of channel errors, which are in turn not decoded correctly and consequently results in error propagation and thus decreased PSNR performance. Variations between the systems can also be attributed to the inclusion of the FS detection in the proposed system versus the exclusion of the forbidden symbol in the other systems. The inclusion of the FS can add an amount of redundancy to the proposed system thereby increasing its compression slightly. This redundancy introduced by the forbidden symbol reveals its significance at lower bitrates but helps the proposed system produce unparalleled results.

Through all the PSNR simulations the Rayleigh channel results exhibit highly poor, inferior, degraded and erratic performances especially when combined with the SPIHT wavelet algorithm. In some instances the erratic PSNR behaviour is due to the channel errors along with high contrast and fine detail compression producing catastrophic error propagation. Other

instances the combination of SPIHT wavelet compression coupled with Rayleigh channel errors produce a transmitted bitstream far too erroneous, that the decoders cannot satisfactorily correct the error to produce distinct PSNR results.

The proposed system shows a significant improvement for the PSNR simulations as compared to the comparison systems without error detection or an additional error correction mechanism if MAP decoding fails. The improvement in results lie in the proposed system's forbidden symbol error detection approach and the error correcting technique which employs a statistical predictive metric approach coupled with a stack search of possible correct combinations which uses the ARQ retransmission strategy as a second error correction support versus the other systems that use either arithmetic coding or MAP decoding or a combination thereof with no error detection. The search utilised by the MAP decoding method discards irrelevant branches, converging the search rapidly. The direction of the search is based on a metric that uses channel characteristics to statistically analyse the errors and predict the correct solution. The forbidden symbol is then detected in the MAP decoded bitstream, if it exists then the ARQ retransmission strategy is invoked providing a second error correction mechanism. The success of this approach is observed in the PSNR results, where the proposed system surpasses the other system; System 1, System 2 and System 3 without error detection for the Gaussian channel, and even so in the highly destructive Rayleigh channel for both the EZW and SPIHT schemes even though the SPIHT results are not as successful as the EZW.

As a final point for PSNR results, the proposed system outperforms System 1, System 2 and System 3 without FS detection and ARQ retransmission for the PSNR results for both channels and both wavelet compression algorithms, illustrating the significance between error correction, error detection and error propagation, producing visually clear images free of errors.

The proposed ssytem's performance for the channel induced errors approach (erasure rate) clearly illustrates its overwhelmingly significant improvement in error correction over the comparison systems for all erasure rate simulations. The packet erasure rate versus channel SNR simulations are generated to present the decoder's error correcting ability visually in the form of a PER versus E_b/N_0 graph. The typical decay of the PER simulations should show for high E_b/N_0, a low erasure performance and conversely for low E_b/N_0, a high erasure performance.

Through the test images and PER performances, the proposed codec outperforms the other systems in all simulation cases. This exemplifies the remarkably accurate and precise execution of its error correction mechanism as compared to the unconvincing and unsuccessful error

correction employed by the other systems. The channel impairments for both decoders are designed to be consistent as to discard any discrepancies that may occur in producing permissible results. The packet erasure performance is used to specifically demonstrate the error correcting ability of the decoders in the presence of channel errors.

The AWGN channel results for the set of test images show consistently excellent erasure performance for the entire range of simulations. The error correction process employed by proposed system using the AWGN channel generates superior results for all channel SNR's, which validate the competence, correctness and efficiency of the system. The Gaussian channel is able to induce sufficient errors to demonstrate the proposed MAP decoder's enhanced ability to effectively decode and correct the channel errors producing results that are successful.

The Rayleigh multipath fading channel is an extremely highly error prone channel, and produces results which are less than the Gaussian channel's response. The unpredictable nature of the Rayleigh channel errors causes the error correction mechanism of the decoders to fail thereby producing inferior results for System 1 and System 3. However, the proposed system with error detection is able to prove that its statistical error correction processing is in fact better than the other systems which use no error detection or retransmission, and is thus a more appropriate decoding method to employ for reliable image compression and transmission. Another important point to note, is the PER performance for highly fine detailed images transmitted across the Rayleigh channel, produce a more erratic erasure response than with high contrast or monotone images. This is due to the increased fine detail compression coupled with the highly erroneous channel, producing disastrous errors that the decoder is not able to correct with any success.

The erasure simulations are fundamental to the dissertation as they provide a definite and conclusive indication of the performance of the proposed system's error correction in relation to the other systems. The proposed system with the ARQ protocol and MAP metric sequential decoder throughout the series of erasure rate simulations exhibit performances that exceed the other three systems. For the PER results, the proposed codec shows substantial improvement for the Gaussian channel for both wavelet compression coders. For the Rayleigh channel, the proposed system is still able to exhibit greatly improved performance for both compression algorithms however; the improvement is not as great as the Gaussian channel but is still testament to the improved ARQ-MAP decoding error correction. The performance of the proposed system's ARQ-MAP metric decoding technique is directly related to the error correcting decoding procedure utilised in the decoding process, thus improved performance suggests an enhanced error correction and detection mechanism being employed. The proposed

system's ARQ-MAP decoding versus System 1, System 2 or System 3 without FS detection is essentially the comparison of a predictive metric based stack algorithm search against combinations of the proposed system. The proposed system's results surpasses the results of System 1, System 2 and System 3, signifying that the inclusion of the forbidden symbol for error detection and the predictive metric which uses probabilistic channel properties in MAP decoding along with the ARQ retransmission is more powerful and accurate in terms of error correction than these components individually. In addition to the metric computation, the MAP decoding process is superior due to the search algorithm applied. The MAP's stack algorithm based search is able to back track immediately to a branch with the highest metric. This combination of a highly complex metric structure and rapid search pattern makes the proposed MAP metric sequential decoder a powerful tool in error correction. The ARQ retransmission protocol also provides the system with additional support in the event the MAP decoder fails for any reason. This adds to the robustness of the error correction system. The proposed codec's results corroborate and validate the error correction technique designed for wavelet compression and channel induced errors on transmitted bitstreams.

The viability of the error detection mechanism as an aid to the error correction mechanism was shown through the results determined between the proposed system using forbidden symbol detection and the other system algorithms like System 1, System 2 and System 3 without forbidden symbol detection. The arithmetic coding and decoding option proved inadequate in all respects and suggested that the proposed system was far more superior. System 2 that uses no FS detection did present a greater challenge for determination of forbidden symbol error detection as an improvement for image quality. The Rayleigh simulations of the MAP decoding without FS detection failed and posed no threat to the proposed system. The stability of the proposed coder over the entire bitrate spectrum ensured that the proposed system is a better option for maintaining error detection and correction consistency in the processing of the transmitted bitstream.

An additional performance comparison apart from the error correction shown between the proposed system and the comparison systems in the PER simulations, was the degree of compression achieved between SPIHT wavelet coding against EZW wavelet coding. In all the PER simulations for the various test images the SPIHT wavelet encoder generates results using a smaller bitrate than the EZW encoder. A smaller bitrate implies greater compression which results in greater error propagation during decompression if affected by channel errors during transmission. Nevertheless, the SPIHT wavelet encoder is still able to surpass the EZW encoder in terms of PSNR performance. The results confirm the SPIHT design in producing improved compression over the EZW wavelet method for image quality at higher SNR.

Finally, the experimental evaluation suggests that the proposed system is competitive with the arithmetic encoder and decoder, the convolutional encoder and MAP decoder and the arithmetic encoder, convolutional encoder with MAP decoder and arithmetic decoder without FS detection for all wavelet encoder bitrates and channel SNR's, with the use of images with medium to high detail content. The results illustrate the efficacy of the proposed system utilising the ARQ retransmission and MAP method within the context of error correction coding and error detection techniques. The proposed system achieves maximum benefit and consistency on less destructive channels like the Gaussian channel. High compression in the form of SPIHT wavelet coding and high channel degradation like the Rayleigh channel, produce poorer results that offer less information. In conclusion the proposed system's ARQ retransmission with MAP metric sequential codec has proved to outperform other codec's without forbidden symbol detection and ARQ protocol for a set of complex test images.

CHAPTER 8 -   CONCLUSION

The dissertation presented the theory and literature pertaining to image and video compression with emphasis on wavelets. Transmission of wavelet compressed images over wireless error inducing channels were explored which resulted in the proposition of an error protection mechanism for efficient error-free transmission. The error protection mechanism entailed error detection and error correction techniques employed for an optimal error coding system. The proposed error-free coding system with the joint wavelet compression codec's and wireless channels are discussed and results of the system against various standard structures are explored. This chapter outlines in detail the deductions and conclusions attained in the preceding chapters. This chapter concludes by proposing intended future work with associated recommendations for improvement of the proposed system.

8.1   CHAPTER SUMMARIES

8.1.1   Chapter 2 – Current Compression Standards

This chapter highlighted the commonly known image compression and video compression standards currently available today. These included the image compression standards of JPEG and JPEG 2000 and the video compression standards of MPEG, MPEG4, H.263+ and H.264. Details of the compression techniques coding and processing were discussed with particular emphasis on their compression performances. The chapter also discussed the performance metrics used in evaluating compression coding performances thus quantifying the performances exactly.

It was observed that JPEG 2000 produced significant image quality results against JPEG compression indicating the success of wavelets as a method for efficient image compression. H.264/MPEG4 is an improved coding technique for video compression and thus produced enhanced compression performance than its predecessors. The performances stressed the importance of efficient and effective compression techniques for images and video.

8.1.2   Chapter 3 – Wavelet Compression

This chapter examined majority of the theoretical concepts related to wavelets, the wavelet transform, filter banks, multiresolution analysis, subband coding, wavelet families and wavelet coding techniques. A comparison between Fourier transforms and Wavelet transforms were

detailed providing tangible support for the progression towards wavelets as a compression technique.

In addition the chapter included performances of the wavelet families and wavelet coding techniques where the best performing structures were then employed in the proposed codec. Analyses of seven highly varying wavelet families were completed, with the Coiflet wavelet family exhibiting the greatest adaptation to emulate the signal, producing high image quality. The chapter also reviewed five different wavelet coding techniques. The EZW and SPIHT wavelet coders were preferred for the proposed codec as they exhibited stable consistent performance. Performances of the EZW and SPIHT codec's against JPEG and JPEG 2000 compression standards for still image compression were executed with the wavelet coding techniques showing their viability and potential as wavelet compression standards.

8.1.3   Chapter 4 – Wireless Channels

This chapter proposed the use of two wireless channels for the simulations and results. The two channels were the additive white Gaussian noise (AWGN) channel and the multipath Rayleigh fading channel. The AWGN channel proved to be a less aggressive channel than the Rayleigh channel which is shown to be extremely destructive and erroneous. The AWGN induces fewer errors whilst the Rayleigh channel being a worst case scenario channel induces far more errors into the bitstream.

Bit error rate performance evaluations were completed using the selected EZW an SPIHT wavelet algorithms. The wireless channels degradation on the compressed bitstreams were shown and detailed. The image quality produced for the AWGN channel was higher than the more destructive Rayleigh channel. The chapter was able to visually depict the channel induced errors for the wavelet decoders via the reconstructed images.

8.1.4   Chapter 5 – Error Protection

This chapter outlined two error resilience mechanisms namely error detection using arithmetic coding with forbidden symbol detection and error correction using a *maximum a posteriori* (MAP) metric sequential decoder with automatic repeat request (ARQ) retransmission protocol. The chapter presented the theoretical concept employed by the methods using numerous mathematical references.

The error detection theory model progressed from arithmetic coding to integer arithmetic coding to integer arithmetic coding employing forbidden symbol detection. The progression highlighted key features of the algorithm including the forbidden symbol redundancy. The error correction

model produced an extremely critical section in the chapter as the basis of the proposed system and its success relies primarily on the execution of this model. The error correction employed is MAP metric sequential decoding which computes a metric that uses *a priori* and *a posteriori* probabilities in its statistical prediction of the errors and ARQ retransmission protocol as a second error correction mechanism that requests a retransmission of the bitstream if MAP decoding is unsuccessful in correcting the erroneous bitstream. The chapter then outlines the use of a stack algorithm to sequentially decode the errors where the probabilistic metric directs the search.

8.1.5   Chapter 6 – Proposed Codec

This chapter forms the crux of the dissertation where the complete proposed wavelet compression error coding system is discussed and presented. Simulations in the following chapter are executed using the system proposed within this chapter.

It is described as a hybrid source and channel coder where the source coding involves wavelet compression and arithmetic coding and the channel coding includes error correction coding in the form of MAP metric sequential decoding and ARQ retransmission. The proposed codec stages begin with the compression component which entails wavelet encoding followed by arithmetic coding with the forbidden symbol and then convolutional coding. The second stage is the channel which transmits the arithmetically compressed convolutionally encoded bitstream.

The final stage is complex in that the arithmetic and convolutionally encoded bitstream is transmitted is decoded via MAP decoding which automatically corrects the bitstream if errors are present or decodes it normally if no errors are present. Once decoded the arithmetic decoder begins decoding the MAP decoded bitstream. However, if the forbidden symbol is detected, this invokes the second error correction mechanism of ARQ retransmission which prompts the transmitter to resend the bitstream again. A second transmission attempt is made of the arithmetically compressed convolutionally encoded bitstream. The cycle begins again however, if the MAP decoded bitstream is error-free then it is arithmetically decoded. This bitstream is further decoded by the wavelet decoder reproducing the initial image. If no error is detected, the arithmetically encoded bitstream is free of errors and continues to arithmetic decoding and finally wavelet decoding producing the image.

The chapter consolidates the process described above by illustrating it in a system block diagram highlighting the flow of communication.

8.1.6   Chapter 7 – Performance of the Proposed Codec

This chapter evaluates the system proposed in Chapter 6. The evaluation uses a sample set of three image processing test images for a series of simulations to demonstrate the performance of the proposed system's ARQ retransmission and MAP metric sequential decoder with arithmetic coding and forbidden symbol detection. The proposed system is performance evaluated against three standardised systems: arithmetic encoding and decoding without forbidden symbol detection, convolutional encoding and MAP metric decoding without forbidden symbol detection and arithmetic encoding, convolutional encoding and MAP metric decoding without forbidden symbol detection. Image quality and erasure performances highlight the proposed systems retransmission and metric based sequential decoding technique for efficient error correction and the forbidden symbol method for efficient error detection.

The final results indicated the proposed system is competitive as an error detection and correction mechanism and produced comparable results in comparison to the other standards. Through majority of the results, the proposed system proved better and showed greater improvement in error-free decoding than the other algorithms. The highly successful combination of error detection using the forbidden symbol and error correction using ARQ retransmission and MAP metric sequential decoding showed the potential and ability to be utilised for efficient error-free compression and transmission of images and video.

The Gaussian channel shows consistently improved results compared to the Rayleigh multipath channel whose results were slightly less. The proposed system's ARQ retransmission and MAP algorithm with forbidden symbol detection showed improved results for the SPIHT wavelet encoder simulations using the Gaussian channel, against the EZW wavelet encoder whose results were poorer. The Rayleigh channel produced wavelet coder results that were inferior for the SPIHT algorithm, with the EZW algorithm appearing slightly better.

The proposed system's ARQ retransmission and MAP metric sequential decoder using arithmetic coding with forbidden symbol detection outperforms its standardised counterparts across numerous test images, image quality and erasure simulations, and is able to guarantee and offer efficient error-free transmission of images via erroneous destructive channels.

8.2   FINAL REMARKS

The dissertation proposes an efficient system using an error detection and correction mechanism for wavelet based compression transmitted over wireless channels. The error detection and

correction employs a joint technique of arithmetic coding with forbidden symbol, convolutional coding and MAP metric sequential decoding. The choice of wavelet compression is suitable for low bitrate compression requiring a high degree of quality and performance integrity. The use of two different wireless channels, depict various severity of errors which is used to prove that the proposed system is able to control and process the errors adequately.

The proposed system operates efficiently on bitstreams containing moderately intense errors. High compression with moderate intensity errors produces results that are competitive and outperform the standard codec's which are used to benchmark the proposed systems performance. However, high compression with high intensity errors, as displayed by the Rayleigh channel produces results that are inferior across all the codec's. The transmission of an arithmetically encoder bitstream and a convolutionally encoded bitstream appears disadvantageous as it increases the bitrate of the system as two bitstreams are transmitted, but it instead offers reliable image retrieval due to this error correction scheme as shown in the results.

In conclusion the proposed system of arithmetic decoding with forbidden symbol detection and MAP metric sequential decoding, is extensively analysed across multiple wavelet compression algorithms and wireless channels in order to prove that the technique employed to detect and correct errors is successful and competitive with most common error coding standards. Thus the system is able to offer reliable, capable image and video transmission for low bitrate multimedia environments that demand image integrity and quality.

8.3   FUTURE WORK

The wavelet compression and error coding system proposed in this dissertation has great potential to incorporate additional work for improvement of the system and its performance. Recommendations for possible improvements are detailed as follows.

A test set of greyscale images were used in the simulations to evaluate the proposed system's performance. The proposed system can be extended to include colour images as well as video. However, the use of the colour component involves the introduction of greater redundancy to the system and thus an increase in bitrate capacity. Thus colour redundancy requires further investigation for the proposed systems efficient compression and transmission capabilities in the multimedia domain.

The dissertation revealed that the ECECOW wavelet algorithm showed superior performance against all other wavelet codec's. Although the EZW and SPIHT algorithms were chosen as the compression mechanisms for the proposed system, the system can investigate the implementation of the ECECOW wavelet coding scheme. A key observation it that the ECECOW algorithm integrates arithmetic coding in its compression technique unlike the other algorithms which include the arithmetic coding externally for improved compression. Thus it can be utilised to further include the forbidden symbol for error detection in its integrated arithmetic coder to produce a more efficient error detection algorithm.

Improvements in the proposed systems bitrate can be accomplished through the use of a hybrid source and channel coder involving the transmission of the arithmetically encoded bitstream with the MAP decoder, which can improve the transmission bandwidth considerably. The MAP decoder will no longer use the convolutionally encoded bitstream to decode the channel errors. The arithmetically coded bitstream will have to statistically predict the errors of the arithmetically encoded bitstream and correct it without the convolutionally encoded bitstream. Thus the system will exhibit a true fusion of source and channel coding and will improve the bitrate significantly thereby improving the systems performance.

The error correction mechanism employed by the MAP decoder can be improved by moving away from its currently implemented hard decision decoding to soft decision decoding. Hard decision decoding uses a simple decision threshold to examine each received bit and makes a decision whether it represents a *1* or *0 bit*. Soft decision decoding examines additional side information in the form of multibit quantisation of the received bit to decide if it represents a *1* or *0 bit*. Thus soft decision decoding is a more flexible approach than hard decision decoding but harder to implement. It will offer increased error correction performance as it improves the sensitivity of the decoding metric used in the MAP decoder.

Finally a real-time implementation of the proposed system involving transmission and reception of images and video should be analysed. Furthermore, various advanced error coding schemes require investigation, as a thorough performance evaluation of the newly proposed system against more advanced systems will show greater performance. The proposed system has the potential to become excellent error detection and correction scheme for wavelet based compression of images and video.

REFERENCES

[1] R.C Gonzalez and R.E Woods, "Digital Image Processing," 2nd Edition, Prentice Hall Inc., New Jersey, 2002.

[2] G.K Wallace, "The JPEG Still Picture Compression Standard," Comm. ACM, vol. 34, pp. 30-44, 1991.

[3] A. Skodras, C. Christopoulous, and T. Ebrahimi, "JPEG-2000: The Upcoming Still Image Compression Standard," Pattern Recognition Letters, Vol. 22, pp. 1337-1345, Oct. 2001.

[4] A. Skodras, C. Christopoulous, and T. Ebrahimi, "The JPEG-2000 still image compression standard," IEEE Signal Processing, vol. 18, pp. 36-58, Sept. 2001.

[5] D. Taubman, "High Performance Scalable Image Compression with EBCOT," IEEE Trans. on Image Proc., vol.9, no. 7, pp. 1158-1170, July 2000.

[6] J. Wiseman, (2005) An introduction to MPEG Video Compression [Online]. Available from http://www.john-wiseman.com/technical/MPEG_tutorial.htm [Accessed 24/10/07].

[7] R. Koenen, "Overview of the MPEG4 Standard," ISO/IEC JTC1/SC29/WG11 N1730, Luglio, 1997.

[8] D. Turaga and T. Chen, "ITU-T Video Coding Standards," Electrical and Computer Engineering, Carnegie Mellon University, 2000.

[9] G.J. Sullivan and T. Wiegand, "Video Compression – From Concepts to the H.264/AVC Standard," Proc. Of the IEEE, Dec 2004.

[10] T. Wiegand, G.J. Sullivan, G. Bjontegaard and A. Luthra, "Overview of the H.264/AVC Video Coding Standard," IEEE Trans. on Circuits and Systems for Video Tech., vol. 13, no. 7, pp. 560-576, July 2003.

[11] B. Girod, (2004) Overview: Video Coding Standards [Online]. Available from: http://www.stnford.edu/class/ee398b/handouts/lectures/05-StandardsH264AVC.pdf [Accessed 24/10/07]

[12] R. Polikar, (2001) The Wavelet Tutorial [Online]. Available from: http://users.rowan.edu/~polikar/WAVELETS/WTtutorial.html [Accessed 24/10/07].

[13] A.M. Reza, "From Fourier Transform to Wavelet Transform, Basic Concepts," Spire Lab, UWM, White Paper, Oct. 1999.

[14] S. Saha, (2004) Image Compression – from DCT to Wavelets: A Review [Online]. Available from: http://www.acm.org/crossroads/xrds6-3/sahaimgcoding.html [Accessed 12/10/07].

[15] V.V Zaharov and S. Fausto Casco, "Discrete Fast Wavelet Transform Algorithm and its Application to Prediction Problem," Academic Open Internet Journal, ISSN 1311-4360, Vol. 8, pp 1-10, 2002.

[16] O. Rioul and M. Vetterli, "Wavelets and Signal Processing," IEEE Signal Proc. Magazine, Vol. 8, No. 4, pp 14-38, Oct. 1991.

[17] E. Welsh, Filter Banks [Online]. Available from: http://is.rice.edu/~welsh/elec431/filterBank.html [Accessed 24/10/07].

[18] P. Lee, "Wavelet Filter Banks in Perceptual Audio Coding," MScEng Thesis, University of Waterloo, 2003.

[19] G. Kaiser, "A friendly guide to wavelets," Birkhauser, Boston, 1994.

[20] G. Strang and T. Nguyen, "Wavelets and Filter banks," Wellesley-Cambridge Press, Massachusetts, 1996.

[21] Wavelet Explorer Documentation: Multiresolution Analysis [Online]. Available from: http://documents.wolfram.com/applications/wavelet/FundamentalsofWavelets/1.2.99.html [Accessed 24/10/07].

[22] S.G. Mallat, "A Theory for Multiresolution Signal Decomposition: The Wavelet Representation," IEEE Trans. On Pattern Analysis and Machine Intell, Vol. 11, No. 7, pp 674-693, July 1989.

[23] Matlab Help File, Wavelet Toolbox: General Concepts.

[24] I. Daubechies, "Ten Lectures on Wavelets," Society for Industrial and Applied Mathematics, Philadelphia, PA, 1992.

[25] M. Blume and D.R. Ballard, "Image Annotation Based on Learning Vector Quantization and Localized Haar Wavelet Transform Features," Proc. SPIE, Vol. 3077, pp 181-190, 1997

[26] D. Wei "Coiflet-Type Wavelets: Theory, Design and Applications," PhD Thesis, University of Texas at Austin, August 1998.

[27] J. Lewalle, (1995) Morlet Wavelet and Transform [Online]. Available from: http://www.ecs.syr.edu/faculty/lewalle/tutor/node14.html [Accessed 24/10/07].

[28] J.M. Shapiro, "Embedded Image Coding using Zerotrees in the EZW Algorithm," IEEE Transactions on Signal Processing, vol. 41, pp. 3445-3462, Dec. 1993.

[29] J.M. Shapiro, "A Fast Technique for Identifying Zerotrees in the EZW Algorithm," Proc. IEEE International Conference on Acoustic, Speech and Sig. Proc., vol. 3, pp. 1455-1458, 1996.

[30] P. Xiao, "Image Compression by Wavelet Transform," Department of Computer and Information Sciences, East Tennessee State University, Aug. 2001.

[31] C. Valens, (2004) Embedded Zerotree Wavelet Encoding [Online]. Available from: http://perso.wanadoo.fr/polyvalens/clemens/ezw/ezw.html [Accessed 24/10/07].

[32] A. Said and W.A. Pearlman, "A New Fast and Efficient Image Codec Based on Set Partitioning in Hierarchical Trees," IEEE Transactions on Signal Processing, vol. 6, pp. 242-249, June 1996.

[33] Z. Lu, D.Y. Kim and W.A. Pearlman, "Wavelet Compression of ECG Signals by the Set Partitioning in Hierarchical Trees (SPIHT) Algorithm," IEEE Transactions on Biomedical Engineering, vol. 47, pp. 849-856, July 2000.

[34] Z. Xiong, K. Ramchandran and M. T. Orchard, "Space-Frequency Quantization or Wavelet Image Coding," IEEE Trans. Image Processing, vol. 6, pp 677-693, May 1997.

[35] Z. Xiong, K. Ramchandran and M. T. Orchard, "Wavelet Packet Image Coding Using Space-Frequency Quantization," IEEE Trans. Image Processing, vol.7 no.6, pp. 892-898, June 1998.

[36] Z. Xiong and X. Wu, "Wavelet Image Coding Using Trellis Coded Space-Frequency Quantization," IEEE Sig. Proc. Letters, Vol. 6 No. 7, pp. 158-161, July 1999.

[37] S. Brooks and M. Tabbara, "Wavelet Image Compression: A Heuristic Approach to Space Frequency Quantisation," Dept. of Electrical and Electronic Engineering, University of Melbourne.

[38] M.J. Tsai, J. Villasenor and F. Chen, "Stack-Run Image coding," IEEE Transactions on Circuits and Systems for Video technology, vol. 6, pp. 519-521, October 1996.

[39] A.M. Awan, N. M. Rajpoot and S. A. Husain, "Stack-Run Adaptive Wavelet Image Compression," Data Compression Conference, 2003.

[40] X. Wu, "High-order Context Modelling and Embedded Conditional Entropy Coding of Wavelet Coefficients for Image Compression," Proc. of 31st Asilomar conf. on Signals, Systems and Computers, pp. 1378-1382, 1997.

[41] X. Wu, "Compression of Wavelet Transform Coefficients, The Transform and Data Compression Handbook," E.K. Rao et al. Boca Raton, CRC Press LLC, 2001.

[42] R. Sudhakar, R. Karthiga and S. Jayaraman, "Image Compression using Wavelet Coefficients: A Survey," Dept. of Electronics and Computer Engineering, PSG College of Technology, Tamilnadu, India.

[43] B. Sklar, "Rayleigh Fading Channels in Mobile Digital Communication Systems part I: Characterization," IEEE Comms. Magazine, Vol. 35, Issue 7, pp. 90-100, July 1997.

[44] H. Hourani, "An Overview of Diversity Techniques in Wireless Communication Systems," Helsinki University of Technology Communications Lab, 2004.

[45] J. Arauz, "Discrete Rayleigh Fading Channel Modeling," Dept. of Information Sciences and Telecommunications, University of Pittsburgh, March 2002.

[46] S. Gaur, "Analysis of Advanced Diversity Receivers for Fading Channels," MSc. Eng. Thesis, Virginia Polytechnic Institute and State University, Sept. 2003.

[47] J.G Proakis and M. Salehi, "Communication Systems Engineering," Prentice Hall, 2002.

[48] M.J. Manglani and A.E. Bell, "Wavelet Modulation Performance in Gaussian and Rayleigh Fading Channels," Proc. Of MILCOM, 2001.

[49] S. Mathur, "Small Scale Fading in Radio Propagation," Dept. of Electrical Eng., Rutgers University, 2005.

[50] W. Yu, Z. Sahinoglu and A. Vetro, "Energy Efficient JPEG 2000 Image Transmission over Point-to-Point Wireless Networks," Technical Report, Mitsubishi Electric Research Lab. Inc, TR2003-111, Oct. 2003.

[51] C. Boyd, J. Cleary, I. Irvine, I. Rinsma-Melchert and I. Witten, "Integrating error detection into arithmetic coding," IEEE Trans. on Comm., vol. 45, no. 1, pp. 1-3, Jan. 1997.

[52] I. Witten, R. Neal and J. Cleary, "Arithmetic Coding for Data Compression," Comm. of the ACM, vol. 30, no. 6, pp. 520-540, June 1987.

[53] P.G. Howard and J.S. Vitter, "Practical implementations of arithmetic coding," Image and Text Compression, vol. 12, no.7, pp. 85-112, 1992.

[54] J. Chou and K. Ramchandran, "Arithmetic coding-based continuous error detection for efficient ARQ-based image transmission," IEEE Trans. on Comm., vol. 18, no. 6, pp. 861-867, June 2000.

[55] G.G. Langdon, "An Introduction to Arithmetic Coding," IBM J. Res. Develop., vol. 28, no. 2, pp. 135-149, March 1984.

[56] P.G. Howard and J.S. Vitter, "Arithmetic Coding for Data Compression," Proc. of the IEEE, vol. 82, no. 6, pp. 857-865, June 1994.

[57] D. Marpe and T. Wiegand, "A Highly Efficient Multiplication-Free Binary Arithmetic Coder and its Application in Video Coding," Proc. IEEE ICIP, pp. 263-266, Sept. 2003.

[58] S.B. Jamaa, M. Kieffer and P. Duhamel, "Efficient MAP estimation of CABAC encoded data with objective adjustment of the complexity/efficiency tradeoff," Proc. ICASSP, Nov. 2006.

[59] S.B. Jamaa, M. Kieffer and P. Duhamel, "Exact MAP Decoding of CABAC Encoded Data," Proc. ICASSP, Oct. 2006.

[60] S.B. Jamaa, M. Kieffer and P. Duhamel, "Controlled complexity MAP Decoding of CABAC Encoded Data," Proc. ICASSP, Oct. 2006.

[61] J.M. Wozencraft, "Sequential Decoding for Reliable Communication," 1957 National IRE Convention Record, vol. 5, part 2, pp. 11-25, 1957.

[62] R.M. Fano, "A heuristic discussion of probabilistic decoding," IEEE Trans. Information Theory, vol. IT-9, no. 2, pp. 64-74, April 1963.

[63] K. Zigangirov, "Some sequential decoding procedures," Problemy Peredachi Informatsii, vol. 2, no. 4, pp. 13-25, 1966.

[64] F. Jelinek, "A fast sequential decoding algorithm using a stack," IBM J. Res. Develop., vol.13, pp. 675-685, November 1969.

[65] V. Senk, P. Radivojac and I. Stanojevic, "An overview of decoding procedures for trellis codes," Proc. of TELFOR '99, Belgrade, Yugoslavia, pp. 257-264, Nov. 1999.

[66] S.B. Jamaa, M. Kieffer and P. Duhamel, "Improving sequential decoding of CABAC encoded data via objective adjustment of the complexity-efficiency trade-off," Proc. of SPIE, vol. 6077, 60770F, 2005.

[67] P. Orten and A. Svensson, "Sequential decoding of Convolutional Codes for Rayleigh Fading channels," Wireless Personal Communications, Kluwer Academic Publishers, vol. 20, no. 1, pp. 61-74, Jan. 2002.

[68] P. Orten and A. Svensson, "Sequential decoding for wireless communications," Proc. of FRAMES Long term research workshop, Gothenburg, pp. 135-139, Jan. 1998.

[69] Y.S. Han and P.N. Chen, "Sequential Decoding of Convolutional Codes," The Wiley Encyclopedia of Telecommunications, New York, John Wiley and Sons Inc., vol. 4, pp. 2140-2164, 2002.

[70] N.C. Yeong, "The Comparison of Maximum-Likelihood and MAP Decoding Techniques," BEng (Hons.), University of Queensland, Oct. 2002.

[71] M. Grangetto, P. Cosman and G. Olmo, "Joint source/channel coding and MAP decoding of arithmetic codes," IEEE Trans. On Comm., vol. 53, no. 6, pp. 1007-1016, June 2005.

[72] M. Grangetto and P. Cosman, "MAP decoding of arithmetic codes with forbidden symbol," Proc. of ACIVS, Ghent, Belgium, Sept. 2002.

[73] M. Grangetto, G. Olmo and P. Cosman, "Error Correction by means of arithmetic codes: an application to resilient image transmission," Proc. IEEE ICASSP, 2003.

[74] M. Grangetto, E. Magli and G. Olmo, "Robust video transmission over error-prone channels via error correcting arithmetic codes," IEEE Comm. Letters, vol. 7, no.12, pp. 596-598, Dec. 2003.

[75] C. Langton, (1999) Coding and Decoding with Convolutional codes [Online]. Available from: http://www.complextoreal.com/chapters/convo.pdf [Accessed 09/05/2008].

[76] E.S. Jackson, "High Ratio Wavelet Video Compression through Real-Time Rate-Distortion Estimation," MScEng Thesis, University of Natal, July 2003.

[77] Y. Brijmohan,"Codec for Multimedia Services using Wavelets and Fractals," MScEng Thesis, University of KwaZulu Natal, Nov. 2004.

[78] Naval Air Systems Command and Naval Air Warfare Center, (1997) Electronic Warfare and Radar Systems Engineering Handbook [Online]. Available from: http://www.microwaves101.com/encyclopedia/Navy%20handbook/2.12%20wavelet.pdf [Accessed 24/10/2007].

[79] S. Haykin, "Digital Communications," John Wiley & Sons, 1988.

[80] H. Taub and D.L. Schilling, "Principles of Communication Systems," 2nd Edition, McGraw-Hill, 1986.

[81] E. Bodden, M. Clasen and J. Kneis, "Arithmetic coding revealed: A guided tour from theory to praxis," Proseminar Datenkompression, RWTH Aachen University, 2002.

[82] D. Pechyonyj and A. Bogomjakov, (1998) Lossless Image Compression [Online]. Available from: http://cis.cs.technion.ac.il/Done_Projects/Projects_done/VisionClasses/DIP_1998/Lossless_Compression/lossless.html [Accessed 15/01/2008].

[83] S. Ben-Jamaa, C. Weidmann and M. Kieffer, "Asymptotic Error-Correcting Performance of Joint Source-Channel Schemes based on Arithmetic Coding," Proc. of IEEE 8th Workshop on Multimedia Signal Processing (MMSP '06), pp. 262-266, Victoria, BC, Canada, October 2006.

[84] S. Ben-Jamaa, C. Weidmann and M. Kieffer, "Analytical Tools for Optimizing the Error Correction Performance of Arithmetic Codes," IEEE Trans. on Communications, vol. 56, no.9, pp. 1458--1468, Sept. 2008.

[85] University of Southern California, Signal and Image Processing Institute, Image Database.

[86] T.A. Welch, "A Technique for High Performance Data Compression," IEEE Computer, vol. 17, no. 6, June 1984.

[87] D.A. Huffman, "A Method for the Construction of Minimum-Redundancy Codes", Proc. of the I.R.E., pp 1098-1102, September 1952.

[88] J.R. Butler (1992) HDTV Demystified: History, Regulatory Options and the Role of the Telephone Companies [Online]. Available from: http://jolt.law.harvard.edu/articles/pdf/v06/06HarvJLTech155.pdfm [Accessed 02/04/2009].

[89] E. Martinsson, (2006) IPTV the future of television? [Online]. Available from: http://www.cs.chalmers.se/~tsigas/Courses/DCDSeminar/Files/IPTVrapport.pdf [Accessed 02/04/2009].

[90] M. Ko and I. Koo, (1996) Interactive Video on Demand System [Online]. Available from: http://www.ece.ubc.ca/~irenek/techpaps/vod/vod.html [Accessed 02/04/09].

[91] B. Shrestha, C.G. O'Hara and N.H. Younan, (2005) JPEG2000: Image Quality Metrics [Online]. Available from ftp://ftp.ecn.purdue.edu/jshan/proceedings/asprs2005/Files/0161.pdf [Accessed 24/10/07].

[92] J.M. Jacobsmeyer, "Introduction to Error-Control Coding," Pericle Communications Company, 2004.

[93] S. De Vuyst, S. Wittevrongel and H. Bruneel, "Delay Analysis of the Stop-and-Wait ARQ Protocol over a Correlated Error Channel," Proc. Of HET-NETs 2004, pp. 21/1 – 21/11, July 2004.

[94] N. Agarwal," Coded Cooperative ARQ Protocol in Wireless Networks," University of Texas Dallas, May 2005.

[95] N. Jain, R. Asthana and M. Barbari, "New Stop and Wait ARQ Protocol," UbiCC Journal, vol. 3, no. 2, April 2008.

[96] J. Liebeherr, (1998) Data Link Layer: Flow Control [Online]. Available from: http://www.cs.virginia.edu/~zaher/classes/CS457/lectures/flow-control.pdf [Accessed 16/05/2009].

[97] P.A. Chou and M. Van der Schaar, "Multimedia over IP and Wireless Networks," Academic Press, 2000 [Online]. Available from: http://books.google.co.za/books?id=zeLFs3GD0QQC&pg=PA223&lpg=PA223&dq=chou+arq&source=bl&ots=aclLwM61M-&sig=F9bfRzV9Le9uDQnO12Ek66AN08c&hl=en&ei=c-leSofBPI6HmQfuzfF0&sa=X&oi=book_result&ct=result&resnum=1