

# **Image and Video Watermarking for Wireless Multimedia Sensor Networks**

Noreen Imran

School of Electronics and Electrical Engineering,

Auckland University of Technology,

Auckland, New Zealand.

A thesis submitted to Auckland University of Technology

in fulfillment of the requirements for the degree of

*Doctor of Philosophy*

September, 2013

## Declaration

I, Noreen Imran, hereby declare that this submission is my own work and that, to the best of my knowledge and belief, it contains no material previously published or written by another person (except where explicitly defined), nor material which to a substantial extent has been submitted for the award of any other degree or diploma of a university or institution of higher learning.

---

Signature

Noreen Imran

---

Date

*“In the name of Allah, the most Beneficent, the most Merciful”*

## **Abstract**

Enormous technological growth in wireless communications and CMOS-sensor electronics has enabled the deployment of low-cost, battery-powered multifunctional embedded camera sensor at remote locations. These tiny gadgets adheres a major improvement over traditional wireless sensor networks (WSNs) by accumulating multimedia data into the system. Wireless multimedia sensor networks (WMSNs) are expected to be the solution of many stimulating range of applications such as critical infrastructure surveillance, person locator services, theft control, environmental, health monitoring systems and much more. Many of these applications have mission-critical tasks, which may process the received multimedia content for decision making purpose and thus require that security be considered as a vital concern. Inadequate use of multimedia content that involve tampering or forgery may cause manipulated information distribution that leads to unwanted consequences.

Watermarking, a more flexible and lower complexity solution, is able to ensure that the semantic meaning of the digital content has not been modified by illegitimate sources, while being sustainable against wireless channel errors, lossy compression and other signal processing primitives. Attributed to our literature review, we found that there still exists a considerable research gap in designing security solutions for WMSNs. Therefore, the aim of this thesis was to develop a digital watermarking system for visual information (image/video), complaint with the design requirements of WMSNs based applications. The resource-limited camera sensor nodes need to encode their own raw video data before transmitting them to the sink (or intermediate relay nodes), which makes it essential for sensor nodes to have a video coding paradigm that exploits simple encoding and complex decoding architecture.

The thesis is structured into seven chapters. The first three chapters provide the introduction, background, and literature analysis. The fourth chapter deals with comparison, analysis, and the selection of an appropriate DVC video codec for a given WMSN application. It provides an insight about the computational (encoding/decoding) complexity, energy consumption, node and network lifetime, processing and memory requirements, and the quality of reconstruction of these video codecs. In chapter five, we propose an enhanced semi-oblivious energy-aware adaptive watermarking scheme for WMSNs, which considered key characteristics such as the embedding capacity, security, imperceptibility, computation, and communication energy requirements. We evaluated the distortion in cover image due to watermark redundancies, the number of embedding locations with respect to two channel adaptive parameters, and the impact

of compression of cover image on the correctness of extracted watermark. In addition, we investigated the robustness of the scheme against statistical analysis attacks. Chapter 6 presents a novel, energy-efficient, low-complexity, blind, and imperceptible video watermarking scheme based on transform domain Wyner-Ziv (WZ) coding, which builds on the principles of DVC. It gives an insight about the practical implementation of the proposed scheme on a fully functional WZ codec and its evaluation using real video sequences captured from embedded video camera sensors. In addition, the derived analytical models are used to examine the energy consumption, feedback requests, and rate-distortion performance, embedding capacity, imperceptibility, node and network lifetime in reference to the proposed scheme. Finally, we concluded our work in seventh chapter.

## Acknowledgement

The endless thanks goes to Almighty Allah for all the blessings He has showered onto me, which have enabled me to write this last note in my research work. During the period of my research, as in the rest of my life, I have been blessed by Almighty Allah with some extraordinary people who have spun a web of support around me. Words can never be enough in expressing how grateful I am to those incredible people in my life who made this dissertation possible and because of whom my graduate experience has been one that I will cherish forever.

My first and sincere appreciation goes to *Dr. Boon-Chong Seet*, my primary supervisor, for all I have learned from him and for his continuous help and support in all stages of this dissertation. I would also like to thank him for being a person open to ideas, and for encouraging and helping me to shape my interest and ideas. Each meeting with him added valuable aspects to implementing and broadening my perspective. He has guided me with his valuable suggestions, lightened up the way in my darkest times and encouraged me a lot in our academic life. He kept me motivated at all times. It has been a great pleasure for me to have had a chance to work with him. I personally think that he has made more contribution to this work than I have and if this thesis is worth anything at all, it is due to his endless efforts, concern and commitment. He was the best choice I could have made for a supervisor. *Thank you Sir!*

Special thanks to my second supervisor, *Dr. Alvis Fong* for his support, guidance and helpful suggestions throughout this research work. His guidance has served me well and I owe him my heartfelt appreciation.

I would like to thank my loving husband *Imran Alam* and our kids for their great patience and unequivocal support throughout, for which my mere expression of thanks likewise does not suffice. Imran has supported me in each and every way, believed in me permanently and inspired me in all dimensions of life. I am blessed with two wonderful kids *Saad* and *Tehreem*, who knew only to encourage and never complained about anything even when they had to suffer a lot in my absence over these years. I owe everything to them: without their everlasting love, this thesis would never have been completed.

Finally, I would like to acknowledge the financial, academic and technical support of *AUT University* and its staff, particularly in the award of a Postgraduate Research Scholarship that provided the necessary financial support for this research.

*Dedicated with Extreme Affection and Gratitude to..*

*My Mother Mumtaz Jahan Khanum*

*My Loving Husband Imran Alam*

*My Kids Saad and Tehreem*

*&*

*My Primary Supervisor Dr. Boon-Chong Seet*

# Table of Contents

<b>Abstract</b>	iii
<b>Acknowledgements</b>	v
<b>Table of Contents</b>	vii
<b>List of Publications</b>	xii
<b>List of Figures</b>	xiv
<b>List of Tables</b>	xvii
<b>List of Symbols and Acronyms</b>	xix
<b>1 Introduction</b>	
1.1 Introduction	1
1.2 Motivation and Scope	4
1.3 Contributions	6
1.4 Thesis Organisation	8
<b>2 Background</b>	
2.1 Introduction	9
2.2 Design Challenges for WMSNs	10
2.2.1 Bandwidth	12
2.2.2 Video Encoding Techniques	12
2.2.3 Application Specific QoS	13
2.2.4 Resource Constraints	13

2.3	Video Coding	13
2.3.1	Information Theoretic Depiction of PVC and DVC Paradigm	14
2.3.1.1	PVC Methodology: Joint-Encoder, Joint-Decoder	15
2.3.1.2	DVC Methodology: Independent-Encoder, Joint-Decoder	15
2.4	WMSN Security	18
2.4.1	Cryptography and Watermarking	19
2.5	Digital Video Watermarking	23
2.5.1	Design Goals for Video Watermarking	25
2.5.2	Classification of Video Watermarking Schemes	25
2.5.2.1	Implementation Domain	25
2.5.2.2	Perceptibility	26
2.5.2.3	Detection	26
2.5.2.4	Application Area	27
2.6	Chapter Summary	27
<b>3</b>	<b>Literature Review</b>	
3.1	Introduction	29
3.2	Review of State-of-the-Art DVC, DCVS Architectures and H.264/AVC	29
3.2.1	DVC in WMSNs	30
3.2.2	DVC Architectures	31
3.2.2.1	Power-efficient, Robust, hIgh-compression, Syndrome-based Multimedia (PRISM) Architecture	33
3.2.2.2	Pixel Domain Wyner-Ziv (PDWZ) Video Coding Architecture	36
3.2.2.3	Transform Domain Wyner-Ziv (TDWZ) Video Coding Architecture	38
3.2.2.3.1	DISCOVER (DIStributed Coding for Video sERVICES)	41
3.2.3	Comparison and Analysis of DVC Architectures	42
3.2.4	DCVS Architectures	44
3.2.4.1	Distributed Compressive Video Sensing (DCVS)	46
3.2.4.2	Distributed Compressed Video Sensing (DISCOS)	48

3.2.4.3	Dynamic Measurement Rate Allocation for Distributed Compressive Video Sensing (DMRA-DCVS)	49
3.2.5	Comparison and Analysis of DCVS Architectures	51
3.2.6	Conventional Video Coding – H.264/AVC	53
3.3	Review of WMSN Security Mechanisms based on Digital Watermarking	55
3.3.1	Wavelet based Resource-Aware, Adaptive Watermarking for WMSN	57
3.3.2	Image Watermarking Technique Based on Wavelet-Tree	59
3.3.3	Video Watermarking Technique Against Correlation Attack Analysis in WSN	62
3.3.4	Watermarking Technique Based on Distributed Video Coding	65
3.3.5	Comparison and Analysis of Watermarking Techniques for WMSNs	68
3.4	Chapter Summary	70
<b>4</b>	<b>Analysis of Video Codecs</b>	
4.1	Introduction	71
4.2	Comparison of DVC, DCVS and H.264/AVC Coding Architectures	72
4.3	Experimental Setup	73
4.4	Results and Discussion	80
4.4.1	Encoding and Decoding Computational Complexity	80
4.4.2	Energy Consumption	82
4.4.3	Node and Network Lifetime	85
4.4.4	Quality of Reconstruction	88
4.5	Analysis of Codecs	89
4.6	Chapter Summary	90
<b>5</b>	<b>An Energy-Aware Adaptive DWT Based Image Watermarking</b>	
5.1	Introduction	93
5.2	Related Work	94
5.3	Energy-Aware DWT Based Watermarking for WMSNs	95
5.3.1	Embedding Algorithm	98

5.3.2	Hashing Algorithm	100
5.3.3	Detection Algorithm	101
5.4	Experimental Setup	103
5.5	Results and Discussion	107
5.5.1	Distortion in Cover Image Object and Capacity of Embedding	107
5.5.2	Thresholds and QSWTs	111
5.5.3	Energy Consumption and Data Rate	112
5.5.4	Effect of Compression on Extracted Watermark	114
5.5.5	Statistical Analysis Attacks	117
5.6	Chapter Summary	118
<b>6</b>	<b>A Novel DVC Based Video Watermarking</b>	
6.1	Introduction	119
6.2	Related Work	120
6.3	Proposed Scheme	120
6.3.1	Key Frame Watermarking Inside Reconstruction Loop of H.264/AVC - Intra Mode	120
6.3.2	Key Frame Watermarking and H.264/AVC – Intra Mode	123
6.3.3	Watermark Embedding at Encoder	126
6.3.4	Watermark Extraction at Decoder	129
6.4	Experimental Setup	131
6.5	Results and Discussions	133
6.5.1	Capacity and Imperceptibility	133
6.5.2	Rate Distortion Performance	136
6.5.3	Impact of Watermarking on Feedback Channel	137
6.5.4	Encoding and Decoding Complexity	137
6.5.5	Energy Consumption	140
6.5.6	Node and Network Lifetime	148
6.5.7	Robustness	150
6.6	Chapter Summary	154
<b>7</b>	<b>Conclusion and Future Works</b>	
7.1	Introduction	155
7.2	Conclusion	156

7.3 Future Work	157
<b>References</b>	<b>159</b>

# List of Publications

## Journals

1. N. Imran, B.-C. Seet, and A. C. M. Fong, “A Semi-Oblivious Energy-Aware Adaptive Watermarking for Wireless Image Sensor Networks”, ACM SIGMM/Springer Multimedia Systems Journal, USA, DOI: 10.1007/s00530-013-0320-6, 2013..
2. N. Imran, B.-C. Seet, and A. C. M. Fong, “A Comparative Analysis of Video Codecs for Multihop Wireless Video Sensor Networks”, ACM SIGMM/Springer Multimedia Systems Journal, USA, Vol. 18, No. 5, 2012.
3. N. Imran, B.-C. Seet, and A. C. M. Fong, “A Review of the State-of-the-Art Distributed Compressive Video Sensing Architectures”, International Journal of Computer Applications in Technology, 2013 (accepted).
4. N. Imran, B.-C. Seet, and A. C. M. Fong, “Distributed Video Coding for Wireless Sensor Networks: A Review of the State-of-the-Art Architectures”, Springer Signal and Video Processing Journal, USA (Under Revision).
5. N. Imran, B.-C. Seet, and A.C.M. Fong, “A Novel DVC-based Watermarking Scheme for Wireless Video Sensor Networks”, (Submitted to a Journal).

## **Book Chapter**

6. N. Imran, B.-C. Seet, and A.C.M. Fong, “Security in wireless video sensor networks based on watermarking techniques”, In “Wireless Sensor Networks: Current Status and Future Trends”, CRC Press, USA, pp. 457-482, 2012.

## **Conferences**

7. N. Imran, B.-C. Seet, and A.C.M. Fong, “An Imperceptible and Blind Watermarking Scheme Based on Wyner-Ziv Video Coding for Wireless Video Sensor Networks”, In 1st IEEE/IIAE International Conference on Intelligent Systems and Image Processing, KitaKyushu, Japan, 2013.
8. N. Imran, B.-C. Seet, and A.C.M. Fong, “Distributed compressive video sensing: A review of the state-of-art architectures”, In 19<sup>th</sup> International conference on Mechatronics and Machine Vision in Practice, Auckland, New Zealand, 2012.
9. N. Imran, B.-C. Seet, and A.C.M. Fong, “Performance analysis of video encoders for wireless video sensor networks”, In IEEE Pacific Rim Conference on Communications, Computers and Signal Processing (PacRim), 2011.

# List of Figures

2.1	WMSN architecture	12
2.2	Functional modules of wireless camera sensor node	12
2.3	Video coding framework: a) PVC methodology; b) DVC methodology	16
	Wyner-Ziv logical framework.	17
2.4	Watermarking framework (a) Watermark embedding at encoder; (b)	24
2.5	Watermark detection and verification at decoder	
2.6	Classification parameters of video watermarking schemes	26
3.1	Video transcoder framework.	32
3.2	PRISM: a) Logical architecture; b) Structural framework	33
3.3	Uniform scalar quantization step-size and codeword distribution	34
3.4	PRISM's bitstream packet format	35
3.5	PDWZ architecture	36
3.6	TDWZ architecture	38
3.7	DISCOVER architecture.	41
3.8	Compressive sensing framework	46
3.9	DCVS architecture	46
3.10	DISCOS architecture	48

3.11	DMRA-DCVS architecture	50
3.12	H.264/AVC hybrid video (a) Encoder and (b) Decoder architecture with motion compensation	54
3.13	Conceptual framework for secure communication in WMSNs	55
3.14	Resource-aware adaptive watermarking system for WMSN	58
3.15	Block Diagram for (a) Watermark embedding (b) Watermark Extraction	61
3.16	Block diagram for embedding watermark into intra-frame sequence	63
3.17	Video watermarking scheme based on Wyner-Ziv codec	67
4.1	Test video sequences: (a) Foreman; (b) Akiyo; (c) Carphone; and (d) Hallway	75
4.2	General WMSN scenario	76
4.3	Codec computational complexity (a) at Encoding Site (b) at Decoding Site	81
4.4	Average total energy consumption over different number of hops	83
4.5	Node lifetime at (a) Source node; (b) Relay node; under different duty cycles and video codecs	86
4.6	Network lifetime under different duty cycles and video codecs	87
4.7	Quality of reconstruction	89
5.1	Multi-resolution DWT transformation of an image	91
5.2	Test images	103
5.3	(a) Watermarked Lena image ( $512 \times 512$ ); (b) Watermark composed of letters "AUT" ( $16 \times 16$ ); (c) Watermarked Coffee Cup image ( $512 \times 512$ ); (d) Recovered watermarks ( $16 \times 16$ ) from watermarked Lena; (e) Recovered watermarks ( $16 \times 16$ ) from watermarked Coffee Cup	108
5.4	Distortion of cover image (PSNR) due to watermark embedding	109
5.5	Threshold $\alpha_1$ vs. $\alpha_2$ vs. QSWT trees: (a) Lena; (b) Coffee Cup	110
5.6	Total communication energy consumption versus data rate	112
5.7	Threshold $\alpha_1$ vs. $\alpha_2$ vs. Computational Energy Consumption	114
6.1	Proposed watermarking framework based on Wyner-Ziv video coding	121
6.2		124

6.3	H.264/AVC Intra based key frame watermarking framework	128
	Luminance component of $16 \times 16$ Macroblock and expanded	
6.4	$4 \times 4$ sub-Block	134
	Normalized correlation (NC) and watermarked video frame under	
6.5	maximum embedding capacity	135
6.6	SSIM under different redundancy levels - Hillview	138
6.7	Feedback requests under different redundancy levels - Hillview	139
	Encoding complexity vs. embedding capacity under GOP 2 and 4	
6.8	configurations	140
	Decoding Complexity vs. Embedding Capacity under GOP 2 and 4	
6.9	configurations	141
	Computational energy consumption (encoding + embedding) under	
6.10	different redundancy levels	142
	Overall energy consumption (computation + communication) under	
6.11	different redundancy levels using: (a) GOP 2 (b) GOP 4 - Hillview	150
	Lifetime at (a) Source node; (b) Relay node; (c) Network under different	
6.12	duty cycles	151
	Average normalised correlation using single hop transmission in (a)	
6.13	Indoor scenario (b) Outdoor scenario	153
	Average normalised correlation using two hop transmission in (a) Indoor	
	scenario (b) Outdoor scenario	

# List of Tables

3.1	Comparison of primary DVC architectures	42
3.2	Comparison of primary DCVS architectures	51
3.3	Summary of reviewed video watermarking techniques	56
4.1	TelosB mote energy consumption model	74
4.2	Packet loss model parameters	78
4.3	Computation and communication energy consumption (joules)	83
5.1	Parameters for evaluating transmission energy consumption	105
5.2	Total energy consumption (computation + communication) corresponding to different BERs at frame and block-level hashing	115
5.3	Watermarked and extracted watermark images under different Q factor values.	116
6.1	WZ encoder configuration parameters	132
6.2	Rate-Distortion performance - Hillview	136
6.3a	Overall energy consumption for Hillview sequence using proposed scheme under GOP 2 configuration	144
6.3b	Overall energy consumption for Hillview sequence using Ning's scheme under GOP 2 configuration	145

6.3c	Overall energy consumption for Hillview sequence using proposed scheme under GOP 4 configuration	146
6.3d	Overall energy consumption for Hillview sequence using Ning's scheme under GOP 4 configuration	147

# List of Acronyms

BCH	Error correcting codes by Bose, Ray-Chaudhuri, and Hocquenghem .
BER	Bit Error Rate
CMOS	Complementary Metal–Oxide–Semiconductor
CRC	Cyclic Redundancy Check
CS	Compressive Sensing
DISCOS	Distributed Compressed Video Sensing
DISCOVER	DIStributed COding for Video sERvices
DCT	Discrete Cosine Transformation
DCVS	Distributed Compressive Video Sensing
DFT	Discrete Fourier Transformation
DMRA-DCVS	Dynamic Measurement Rate Allocation for Distributed Compressive Video Sensing (DMRA-DCVS)
DSPs	Digital Signal Processors
DVC	Distributed Video Coding
DWT	Discrete Wavelet Transformation
ECC	Elliptic Curve Cryptography
GOP	Group of Pictures

GPS	Global Positioning System
GPSR	Gradient Projection for Sparse Representation
HVS	Human Visual System
i.i.d	independently and identically distributed
LDPCA	Low Density Parity Check Accumulator
LDPC	Low Density Parity Check Accumulator
LFBSR	Linear feedback shift register
LSB	Least Significant Bit
MAC	Message Authentication Code
MCUs	Microcontrollers
MPEG	Moving Picture Experts Group
MSE	Mean Squared Error
NC	Normalised Correlation
NCFSK	Non-Coherent Frequency Shift Keying
NRZ	Non Return to Zero
OMP	Orthogonal Matching Pursuit
PDWZ	Pixel Domain Wyner Ziv
PICO	Privacy through Invertible Cryptographic Obscuration
PRISM	Power-efficient, Robust, hIgh-compression, Syndrome-based Multimedia coding
PSNR	Peak-Signal-to-Noise Ratio
PVC	Predictive Video Coding
QoS	Quality of Service
QSWT	Qualified Significant Wavelet Tree (QSWT)
RFID	Radio-Frequency Identification
SBHE	Scrambled Block Hadamard ensemble
SCBP	Sparsity-Constraint Block Prediction
SI	Side-Information
SRwDSI	Sparse Recovery with Decoder Side-Information
SSIM	Structural Similarity Index
SMV	Symmetric Motion Vector
TDWZ	Transform Domain Wyner Ziv

TwIST	TWo-step Iterative Shrinkage
VoD	Video-on-Demand
WMSN	Wireless Multimedia Sensor Network
WSN	Wireless Sensor Network
WVSN	Wireless Video Sensor Network
WZ	Wyner-Ziv

# Chapter 1

## Introduction

### 1.1 Introduction

During the past few years, wireless multimedia sensor networks (WMSN) have drawn significant attention from the research community, driven by the enormous scale of theoretical and practical challenges. Such growing interest can be mainly attributed to the new set of applications enabled by large-scale networks of small camera devices capable of harvesting information from the surrounding environment, performing simple processing/compression on the captured data and transmitting it to remote locations (or Base-Station). Today various applications in civil and military use are based on multifunctional wireless sensor nodes that gather scalar as well as audiovisual data. In applications such as emergency response, surveillance, environmental and health-care monitoring, multimedia information (especially video stream) is indispensable. What follows is a brief description of three different yet inter-related topics that form the basis of this thesis:

- A WMSN is a network of spatially distributed sensor nodes, each equipped with a miniaturized camera that captures, compresses, and transmits visual information (image/video) about its surroundings to a sink node or base-station for further content analysis and distribution. The foundation of WMSNs can be understood as the

convergence between the concepts of wireless sensor networks (WSNs) and distributed smart cameras. A WMSN is a distributed wireless system that communicates and coordinates with the physical environment by observing it through multiple media. Moreover, it can perform real-time onboard processing of the retrieved information and respond to it by combining technologies from various diverse disciplines such as wireless communications and networking, signal processing, security, computer vision, control and robotics [1].

- Digital images like any other data require encoding scheme to efficiently utilise the storage capacity, the time and energy incurred for transmission. In general, the image encoding scheme eliminates redundancy within the original image and maps that image onto a bitstream format appropriate for communication on a transmission medium. Image encoding schemes can be broadly classified into two categories, namely, lossless and lossy. Lossless schemes ensure an exact reconstruction of every pixel, but do not work well in term of their compression ratio; they are appropriate for the error-prone WMSN environment. In contrast, lossy techniques permit better compression rates at the expense of a little distortion in reconstructed images.

Thus far, many image compression techniques have been developed based on predictive coding, transform coding and vector quantization [2]. Among these, transform coding and vector quantization based techniques have gained considerable attention. In particular, wavelet transforms for image encoding, has gained wide recognition. In context of image encoding, wavelets can be formulated as mathematical functions that provide high quality compression due to their capability to decompose signals into different scales or resolutions. In WMSN, camera sensors can play a significant role if the power consumption of the node is not considerably increased.

Similar to image encoding, video encoding is often a complex and computation intensive operation that can cause a significant energy drain in the resource-limited camera sensor nodes. Conventional video codecs such as MPEG-x have a complex encoding and simple decoding architecture, where raw video data output from a video source, e.g. a video camera, is not encoded by the source itself, but by a more resourceful machine that can perform complex encoding operations such as motion compensation and estimation. These codecs are intended for applications such as broadcasting or streaming video-on-demand (VoD) where the video data is encoded

once and decoded many times by end-user devices. The ability to decode (playback) the encoded video in a timely manner by end-user devices would be made possible by having a simple decoding process. The video coding frameworks based on this reversed paradigm are as follows:

- Distributed video coding (DVC) is an emerging video coding/compression paradigm that reverses the conventional structure of the video codec by exploiting source statistics at the decoder. DVC-based video codecs are characterized by a relatively simple encoder and complex decoder architecture, which is more feasible for video applications in WMSNs.
- Distributed compressive video sensing (DCVS) is another emerging DVC-based low-complexity video encoding paradigm based on the compressive sensing (CS) theorem, which is a recently developed sampling theorem in data acquisition that enables reconstruction of sparse or compressible data from far fewer measurements than is expected under Nyquist's sampling theorem.
- Lastly, image/video watermarking is an approach which embeds the watermark into visual content at sender site in either a visible or an invisible manner, and is detectable at receiver site to ensure the reliability and authentication of the received content. It is not an autonomous technology; rather it is mainly employed in conjunction with a variety of applications to provide services such as preventing unauthorized copying of visual content in the context of copy control applications; identifying the image/video broadcast over the wide broadcast monitoring applications; environment monitoring and detection of malicious attacks or users in surveillance and military operations; ownership authorization and finger printing applications etc. There have been several works on image/video watermarking, but a very few of them considered the energy cost in terms of the data communication and processing, which is a primary constraint to many WMSN based application. In addition, the watermarked image/video distortion caused by error-prone wireless environments during transmission has not been investigated and resolved in most of the existing watermarking systems.

These findings form the basis of defining the direction and scope of my thesis work.

## 1.2 Motivation and Scope

Developing effective ways of authenticating multimedia data has drawn tremendous attention in recent years due to the fact that digital data can be manipulated easily by extensively available editing facilities. The conventional authentication methods based on cryptographic approach are computationally complex but perform very well when we require bit-by-bit accuracy. However, the scenario for multimedia data may be different where users are more concerned about whether the meaning of data has been intentionally tampered with, instead of binary errors that may be caused by compression/decompression, transmission, storage or necessary data processing.

A wireless communication channel is available to any entity that tunes their radio interface to the frequency of the channel for transmission or reception. This provision offers substantial ease to attackers for disrupting or misleading communication between legitimate parties. Existing strong security solutions for wireless networks may not be easily adapted to the WMSN environment due to their complex processing that is inappropriate for resource limited nodes [3]. Unlike in traditional wireless sensor networks (WSNs) where sensors capture and transmit only simple scalar data such as temperature, pressure and humidity, image/video data comprises rich streaming media generated at a high rate, and thus requires more complex processing and higher network bandwidth and energy for transmission.

From our survey of existing literature, it is evident that security in WMSNs is still a relatively emerging research area with a limited number of works found so far [4]. WMSNs require low complexity, robust and scalable security mechanisms, which will not severely impact on the overall lifetime of an individual node or the entire network.

When WMSN is deployed in an open and hostile environment with the possibilities of encountering attacks, ensuring data integrity and authenticity is an important requirement. For example, a malicious user may intercept the image/video transmission between a sender and receiver (pair and inject) and to subvert or disrupt their communication. Some critical decision making has to be done on the basis of the received information. The application needs to ensure that the received image/video has not been tampered with on its way to receiver and is indeed coming from the intended source. Such validation of data integrity and authenticity is almost obligatory in such scenarios as war zone monitoring and border protection.

Generally in wireless surveillance applications, the digital image/video captured by sensor nodes is used to detect and track some unusual events or activities within the surveillance area, so that necessary actions can be taken if any alarming situation arises. Moreover, the visual data can be examined and then used provided as evidence in the investigation of that particular event. In order to ascertain that the given information is the one captured by the surveillance camera and has not been manipulated, security mechanisms for validating both integrity and authenticity of the visual data must be required.

For surveillance cameras deployed in public places such as airports, train stations, malls or office buildings, the issue of privacy protection may not be a very major concern. However, in more personal environments such as homes or patient wards in hospitals, users will be concerned about their privacy if they get to know that they are being continuously watched. WMSNs are also more vulnerable to denial-of-service attacks due to the high volume of data transmission involved in video streaming applications. The strict node energy, storage and processing constraints makes the implementation of advanced anti-jamming mechanisms such as frequency-hopping and physical tamper-proofing an impractical solution due to their complex design and high energy requirements [1]. In such situations, watermarking is considered an attractive alternative due to its light resource requirement.

Digital watermarking is well-known for providing services such as copyright assertion, authentication, content integrity verification, and copy control in reference to multimedia content along with other types of data as well. However, requirements for digital watermarking may vary across application platforms. Among them blind/semi-blind detection, high security, capacity, low-distortion, and complexity are considered as primary requirements in a WMSN environment. For this thesis, we focused on watermarking for multimedia content (especially image/video) under the design constraints adhere by WMSN platform. From the literature review, it is observed that in contrast to image/video, the majority of the existing watermarking techniques for wireless sensor networks have focused on texts or scalar data. Therefore, the design, implementation and deployment of watermarking schemes across the emerging platform of WMSN is still an open and unexplored research area. Sensor nodes in WMSN need to encode their raw data before transmitting them to the sink, which necessitates a simple encoding process. The majority of existing image watermarking schemes do not consider energy-efficiency as a primary

design requirement and video watermarking schemes are based on conventional video codecs that follow complex encoding and simple decoding framework.

In WMSNs, the encoders operate on tiny and battery-powered camera sensors, which have serious processing and energy constraints. Also, existing image/video watermarking techniques cannot be applied directly to WMSN environment. Either these watermarking techniques need to be adapted, or new watermarking techniques need to be designed for emerging coding paradigms.

Therefore, my research aimed to develop an energy efficient watermarking system/s for wireless multimedia networks (WMSNs), with particular focus on the emerging distributed video coding (DVC) based watermarking.

Video watermarking is logically composed of two different fields: the first one deals with security and protection of the watermark itself, in such a way that it cannot be decrypted/decoded/removed from video sequence by the attacker. The second area is signal-processing which is based on DVC that deals with the modification of original video to embed the watermark. Having conducted a comprehensive review of prior work on video watermarking, it is concluded that video watermarking in the context of WMSNs is a relatively unexplored research area and has plenty of room for innovation. However, during the literature review we did find a few works related to the topic that can be considered as a base reference/benchmark for our proposed scheme.

### **1.3 Contributions**

The primary contributions of this thesis are as follows:

- **Analysis of Video Codecs:** We evaluated and analysed the performance of video codecs based on emerging video coding paradigms such as distributed video coding (DVC) and distributed compressive video sensing (DCVS) for multihop WMSNs. This provides insight into the computational (encoding/decoding) complexity, energy consumption, node and network lifetime, processing and memory requirements, and the quality of reconstruction of these video codecs. Based on the findings, we constructed some guidelines for the selection of appropriate video codecs for a given WMSN application.

- **Image Watermarking for WMSN:** We proposed a semi-oblivious, energy-aware adaptive watermarking scheme for wireless multimedia sensor networks, which considered key characteristics such as the embedding capacity, security, imperceptibility, computation, and communication energy requirements. We evaluated the distortion in cover image due to watermark redundancies, the number of embedding locations with respect to two channel adaptive parameters, and the impact of compression of cover image on the correctness of extracted watermark. In addition, we investigated the robustness of the scheme against statistical analysis attacks. The results have shown that the proposed scheme has sufficient capacity to embed redundant watermarks in the cover image in an imperceptible manner with reasonably low distortion. The scheme is also relatively robust against collusion and middleman attacks.
  
- **Video Watermarking based on DVC for WMSN:** The primary objective of this work was to bridge the gap in research (discussed in Section 1.2) by proposing a novel approach to energy-efficient watermarking based on DVC architecture for WMSNs. The key contributions of this work are as follows:
  - A novel, energy-efficient, low-complexity, blind and imperceptible video watermarking scheme based on WZ coding [5], which builds on the principles of DVC.
  - Practical implementation of the proposed scheme on a fully functional DVC codec and its evaluation using real video sequences captured from embedded video sensors.
  - Derived analytical models from which the overall (computational + communication) energy consumption, number of feedback requests, and rate-distortion performance, encoding and decoding complexity, node and network lifetime of the proposed scheme are estimated.

## 1.4 Thesis Organisation

We have structured the thesis into seven chapters:

- **Chapter 2** – gives a brief overview of three different yet interrelated topics, namely WMSN, video watermarking, and video coding that forms the background of this research.
- **Chapter 3** – reviews and critically analyses the video codecs and the related watermarking schemes for WMSNs.
- **Chapter 4** – provides the experimental analysis of the video codecs in a multihop environment. The results of the analysis helped us selecting the appropriate codecs for the design and implementation of our proposed scheme.
- **Chapter 5** – presents a semi-oblivious, energy-aware adaptive image watermarking for WMSNs and explores energy-statistics in reference to embedding and communication in an application in WMSNs.
- **Chapter 6** – outlines the design and implementation details of the proposed scheme. It also provides the analysis of the results obtained in reference to the related work.
- **Chapter 7** – provides the concluding remarks and discusses the future directions of the research.

# Chapter 2

## Background

### 2.1 Introduction

Wireless multimedia sensor networks (WMSNs) have gained considerable attention in recent years due to their vast application domain, flexible deployment structure, and most of all, the availability of low-cost CMOS sensor modules. The application domain for WMSN spans from surveillance and monitoring to healthcare, traffic and industrial control sectors. The self-organizing, flexible, and easily scalable infrastructure of WMSN is one key factor for its widespread popularity. Secure transmission of video data over radio links is highly desirable for many applications that require data to be shielded from unwanted access, tampering and loss. These requirements motivate the need for new security solutions to be designed for WMSNs as most existing solutions for wireless sensor networks (WSNs) cannot be straightforwardly adapted to WMSNs (to be discussed shortly) [6-7]. Unlike WSNs where sensor nodes capture and transmit only simple scalar data such as temperature, pressure, and humidity, multimedia data is based on rich streaming media generated at a higher rate, and thus requires more complex processing, memory storage, higher network bandwidth, and energy for transmission. At the same time, WMSNs have to deal with Optimisation of performance parameters such as delay, throughput, network lifetime, and quality of service (QoS).

A WMSN is generally comprises of several spatially distributed sensor nodes, each equipped with a miniaturized camera and transceiver that capture, compress, and transmit visual information about its surroundings to a sink or base-station for further content analysis, verification, and distribution. The basic architecture of a WMSN is shown in Fig. 2.1. Sensor nodes are typically deployed in open and unattended areas. Therefore, they are vulnerable to various forms of physical or logical attacks and their transmissions are subjected to radio propagation effects such as fading and shadowing [1].

A wireless camera sensor node is basically comprised of sensing, processing, transceiver and power modules as shown in Fig. 2.2. The mobilizer and location finding modules are optional and deployed in situations where the location of sensor node is not fixed. In such scenarios, these modules are used to track and provide location information using GPS or other positioning technologies. This information is further used to compute and update routing table entries at neighbouring nodes [8].

In contrast to a sensor node that gathers scalar data, a camera sensor node requires:

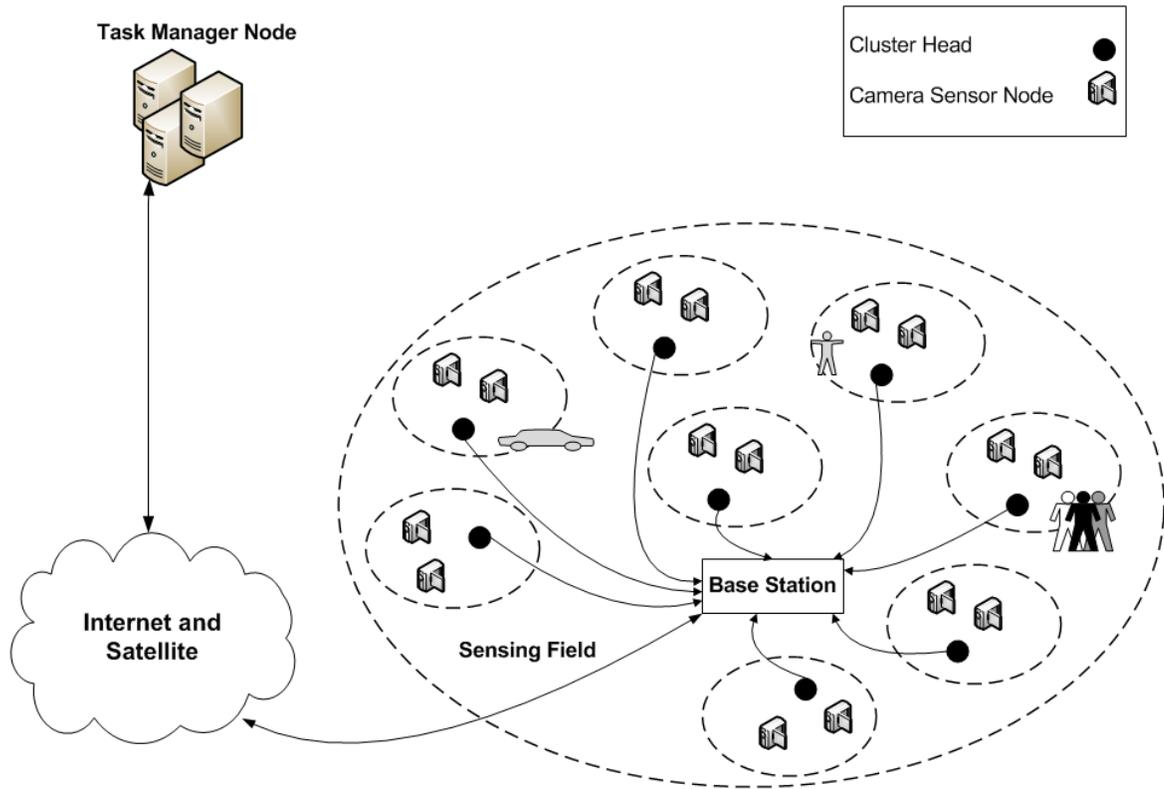
1. More power to process and compress video data.
2. Digital signal processors (DSPs) rather than microcontrollers (MCUs).
3. Higher storage capacity (both for random access and external memory).
4. More bandwidth due to large volume of video data.

The remainder of the chapter is organised as follows: Section 2.2 discusses the design challenges for security mechanisms in WMSNs. Section 2.3 highlights the various security aspects of WMSN. Section 2.4 presents the background on digital video watermarking. Section 2.5 describes the video coding theory in reference to predictive and distributed coding paradigms. Finally, Section 2.6 summarises the chapter.

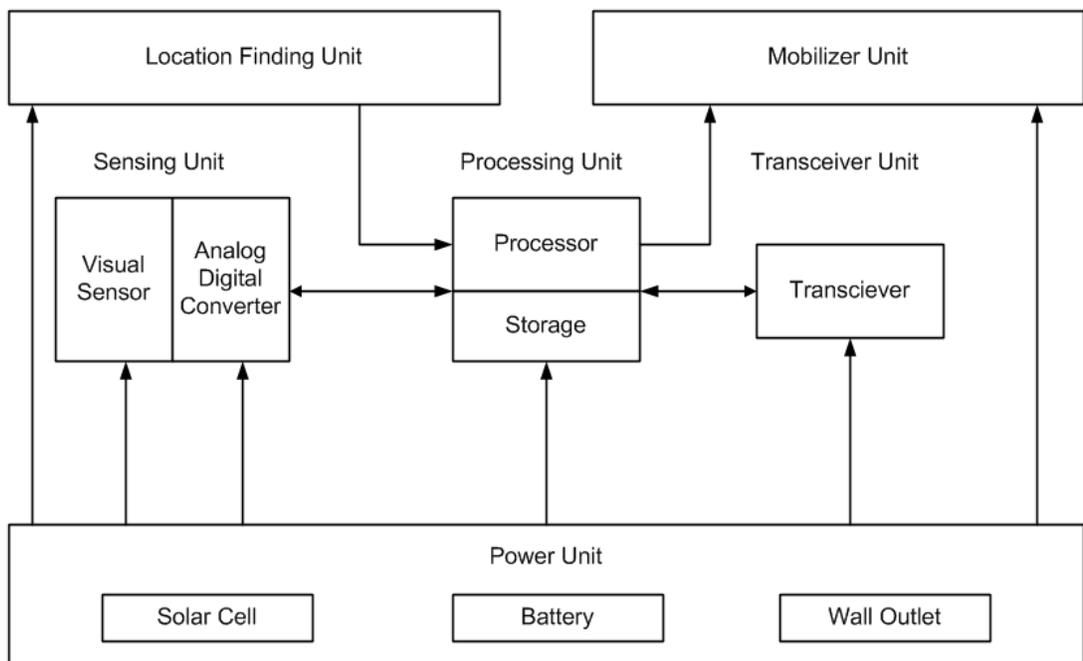
## **2.2 Design Challenges for WMSNs**

The WMSNs have a number of requirements that are unique to their architecture such as high bandwidth for video streaming, low-complexity video encoding techniques, application-specific quality of service (QoS) demands, and resource constraints [9]. These requirements should be taken into account when designing not only the security

mechanisms but also other lower/higher layer protocol mechanisms. A brief discussion of these requirements is as follows:



**Figure 2.1 - WMSN architecture**



**Figure 2.2 - Functional modules of wireless camera sensor node [10]**

### **2.2.1 Bandwidth**

Another related problem for WMSN based applications is the limited bandwidth availability which necessitates the need for more efficient compression algorithms. Video streams are highly bandwidth demanding. Transmitting these streams to the sink via several intermediate nodes over radio links requires much higher bandwidth than one required for transmitting scalar data. The high bit rate and multi-hop transmission make the system prone to congestion due to both intra-flow and inter-flow interference. The congestion gets more serious when there are multiple flows and where traffic exhibits a many-to-one pattern.

Most of the commercially available sensor nodes that comply with IEEE 802.15.4 standard such as TelosB [11] and MICAz [12] have a maximum data rate of only up to 250 kbps, which inevitably limits (depending on actual codec used) the resolution and frame rate of the video that can be transmitted.

### **2.2.2 Video Encoding Techniques**

Video encoding is often a complex and processing intensive operation due to which considerable energy drain can occur at resource-limited sensor nodes. Conventional video codecs such as MPEG-x have a complex encoding and simple decoding architecture, where raw video data output from a video source, e.g. video camera, is not encoded by the source itself, but by a more resourceful machine that can perform complex encoding operations such as motion compensation and estimation. These codecs are intended for applications such as broadcasting or streaming video-on-demand (VoD) where video data is encoded once and decoded many times by end-user devices. In contrast, the resource-limited camera sensor nodes in WMSN need to encode their own raw video data before transmitting them to the sink, which makes it necessary for sensor nodes to have a reversed video coding paradigm i.e. simple encoding and complex decoding. The video coding frameworks based on this reversed paradigm are Distributed Video Coding (DVC) and Distributed Compressive Video Sensing (DCVS).

### **2.2.3 Application-Specific QoS**

Different WMSN applications may have different requirements in terms of bandwidth, processing and compression, among others. Video streaming involves continuous capturing and delivery of data that requires optimized encoding and compression algorithms in addition to efficient hardware to meet the QoS demands pertinent to particular applications [13]. In the context of WMSNs, the approaches for network layer QoS can be based on reliability or timeliness of video delivery [14]. For example, some applications are delay-tolerant, but require reliable and error-free data transmission. It involves packet retransmissions and multipath routing through which a sensor node can inject multiple copies of same packet into different paths so that at least one copy is able to make it to the sink.

### **2.2.4 Resource Constraints**

In contrast to WSN, WMSN requires more resources in terms of processing capability, memory storage (on board and external), operating energy (battery), and transmission bandwidth. Due to the real-time nature of multimedia (especially video) data, WMSN has to employ mechanisms to deal with jitter, frame loss rate and end-to-end delay which consequently require more resources [2].

## **2.3 Video Coding**

The conventional video coding architecture has been challenged by the emergence of wireless sensor networks and the availability of low-cost CMOS video cameras that have made it possible to harvest visual information from the surrounding physical environment, perform compression, and transmit it to various remote locations. Conventional state-of-the-art video coding standards such as H.26x [15], MPEGx [16] are pertinent to the broader class of applications that typically supports encoders with a complexity of at least 5-10 times greater than that of the decoder [17]. These video coding architectures suit applications such as streaming video-on-demand (VoD), video broadcasting, digital home systems, and

multimedia collaboration that requires video to be encoded once and decoded several times by consumers [2, 17-19].

Conventional video coding paradigms are primarily based on hybrid discrete cosine transformation (DCT) and interframe predictive video coding (PVC) frameworks. These frameworks allocate the functionality of codec such that most of the high complexity operations that involve exploiting spatial and temporal correlation, e.g. motion estimation and compensation, are executed at the encoder, while the decoder performs lower complexity operations such as entropy decoding, frame prediction, inverse quantization, and DCT on the bitstream received from encoder [15]. Conventional video coding employs two primary coding modes as follows:

- **Inter-Coding Mode:** Compression in inter-coding mode exploits not only the temporal but also the spatial correlation among video frames and performs the high complexity motion estimation and compensation operations to predict the best matching block for the block under reconstruction. Only the residue between the given block and the corresponding predictor is encoded and transmitted. Therefore, compression efficiency of inter-coding mode is very high at the expense of higher computational complexity and less robustness against packet losses. [20].
- **Intra-Coding Mode:** In contrast, intra-coding mode only exploits the spatial correlation for encoding a block in a given frame. Therefore, the encoding complexity and compression efficiency is lower than the inter-coding mode. However, intra-coding mode does not depend on adjacent frames. It is considered to be more robust against packet losses since it treats each frame as a still image and encodes it separately without exploiting dependencies between adjacent frames [20].

### 2.3.1 Information Theoretic Depiction of PVC and DVC Paradigms

**Problem Definition:** Assume  $X$  and  $Y$  are two statistically correlated, independently and identically distributed (i.i.d) video sequences from two separate encoders that are aware of the existence of each other. Moreover, the decoder has complete information about the encoders. The problem is to determine the minimum encoding (bit) rate for each of the sources such that their joint decoding at the decoder reconstructs each of the video sequence with sufficient accuracy. This problem can be addressed using joint entropy, since

video sequences  $X$  and  $Y$  are statistically correlated. Two different ways to reconstruct them are as follows:

### 2.3.1.1 PVC Methodology: Joint-Encoder, Joint-Decoder

If the two statistically dependent video sequences  $X$  and  $Y$  are encoded together to exploit their statistical dependencies, the minimum lossless rate is  $H(X, Y)$ , which represents their joint entropy decoding is given by:

$$R_{(X,Y)} = H(X,Y) \quad (2.1)$$

PVC is also very sensitive to channel errors; bit errors or packet losses leads to the predictive mismatch which is widely known as drift error and results in a substantial quality degradation of the decoded video. PVC based decoders when employed in an error-prone WMSN environment must follow an error concealment process to constraint the devastating impact of drift error and error propagation in the subsequent frames. The reconstructed video however still contains significant impairments.

### 2.3.1.2 DVC Methodology: Independent-Encoder, Joint-Decoder

On the other hand, if the video sequences  $X$  and  $Y$  are encoded independently, their respective encoding rate is:

$$R_X \geq H(X), \quad (2.2)$$

$$\text{and } R_Y \geq H(Y), \quad (2.3)$$

where  $H(X)$ , and  $H(Y)$ , represents the entropies<sup>1</sup> of  $X$ , and  $Y$ , respectively.

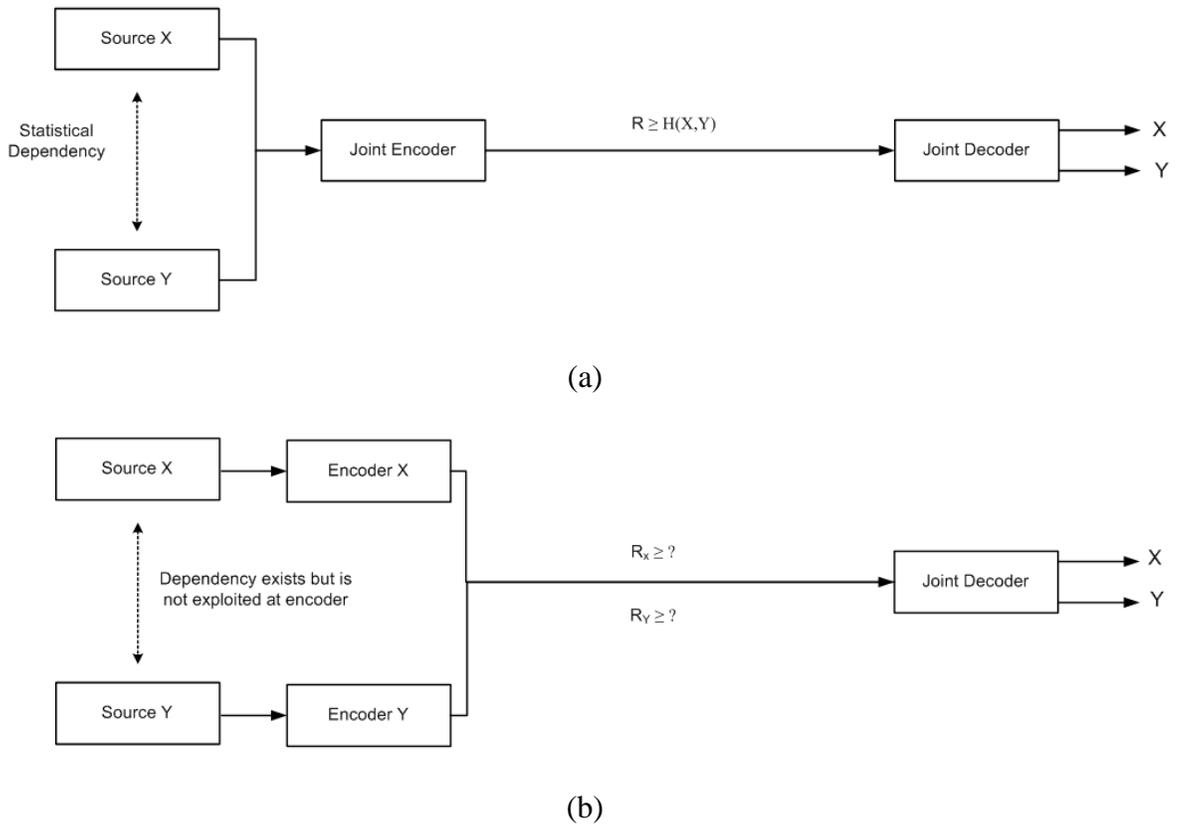
Then the required joint decoding rate is given by:

$$R_X + R_Y \geq R_{(X,Y)} \quad (2.4)$$

The functional block diagram of PVC and DVC frameworks are shown in Fig. 2.3(a), and (b), respectively.

---

<sup>1</sup> Entropy coding determines the minimum bits required to represent one source symbol. The transmission channel is assumed to be error free unless otherwise specified.



**Figure 2.3** - Video coding framework: a) PVC methodology; b) DVC methodology.

One may consider whether it is possible to reconstruct the video sequence with small error probability at encoding rates lower than individual entropies  $H(X)$  and  $H(Y)$ . Distributed source (video) coding provides an answer to this problem as follows [21-22].

- **Slepian –Wolf Theorem for Lossless Compression**

The Slepian-Wolf theorem [21], stipulates that for lossless reconstruction of signals, the encoding rate similar to the one employed in joint encoding of signals  $X$  and  $Y$ , can be attained even if  $X$  and  $Y$  are encoded independently (neither  $X$  nor  $Y$  having access to each other), given that they will be jointly reconstructed at the decoder with an arbitrarily small error. Assume that the minimum encoding rate for lossless reconstruction of signal  $X$  is equal to the signal entropy and is given by  $H(X)$ . To determine the lossless encoding rate between two (or more) related source signals  $X$  and  $Y$ , exploit the statistical correlation between these signals and encode them jointly with the joint signal entropy of  $H(X,Y)$ . The following set of equations represents the individual as well as joint encoding rates of signals  $X$  and  $Y$ :

$$R_X \geq H(X|Y) \quad (2.5)$$

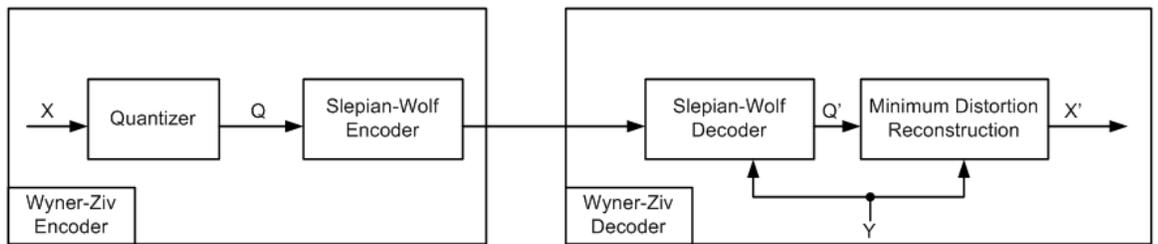
$$R_Y \geq H(Y|X) \quad (2.6)$$

$$R_X + R_Y \geq H(X, Y) \quad (2.7)$$

In practice, the coding performance is determined by the capacity of the correlation channel that approaches the Slepian-Wolf bound used by sophisticated turbo or LDPC codes.

- **Wyner-Ziv Theorem for Lossy Compression**

Wyner and Ziv [22] proposed an extension to Slepian-Wolf theorem by defining the same scenario (as discussed above in Section 2.5.1.2) of independent encoding but in the context of lossy compression. It states that for statistically correlated signals  $X$  and  $Y$ , if encoding of  $Y$  has been performed at the rate  $H(Y)$ , then for joint reconstruction at decoder only the minimum encoding rate of  $X$  needs to be determined with a certain upper bound on the distortion  $D$  in the reconstructed signal. Here,  $Y$  acts as side-information to estimate  $X$ , and the distortion function provides the minimum encoding rate ( $R_X$ ) for reconstruction of  $X$  [22]. Wyner-Ziv theorem is also widely known as Wyner-Ziv rate-distortion distributed video coding theorem, and its logical framework is shown in Fig. 2.4.



**Figure 2.4** - Wyner-Ziv logical framework.

Considering the requirements of video coding for real-time applications, the Wyner-Ziv approach of encoding signals is more realistic and appropriate [23] since it accommodates a certain level of distortion (such as packet losses) during signal reconstruction which is likely in real-time data transmission.

## 2.4 WMSN Security

As discussed in section 2.2, the resource limitations of WMSNs have made it vulnerable to various attacks that vary from video intercepting, tampering, replaying previously stored frames, and injecting fake frames into the network, to disrupting the entire transmission. The fundamental goals of designing security mechanisms for WMSN can be outlined as follows:

- **Privacy:** In certain applications, the video should only capture the behaviour or action of the monitored subjects and not their identity.
- **Integrity:** The video should be received from its intended source without any alteration.
- **Authentication:** The video should be received from an authentic source.

Since sensor nodes can be considered as a soft target for attacks due to their limited resources, it is essential to implement at least some basic level of security at sensor nodes to counteract those attacks. A substantial amount of work has been done in regard to security in WSN, comprising mainly cryptographic (symmetric and asymmetric key cryptography) techniques, hashing, certification, trust management and other advanced techniques [24]. In contrast to WSNs, which mainly process scalar data, addressing security in WMSNs may require a different approach. I now rephrase the privacy, integrity and authentication in the context of WMSN as follows:

- **Privacy:** Privacy protection in video processing environments is defined as a process that requires masking, blurring, or confiscation procedures to hide sensitive information in some regions of the video frame such as human face or vehicle number plate to protect the subject's identity. Therefore, an eavesdropper cannot extract meaningful information by accessing/intercepting the video frames transmitted between the sender and the receiver. Thus far, I have not found many works that deal with privacy issues in WMSN. Kundur et al. [6] proposed a cryptography-based process called PICO (privacy through invertible cryptographic obscuration), which applies a face recognition algorithm to an image and subsequently encrypts the identified facial region using symmetric key cryptography.

- **Integrity:** To ensure the receiver that the given video sequence has not been altered during transit by any malicious user. Yamada et al., [25] devised a watermarking based video integrity verification method, which overcomes the inability of conventional verification mechanisms such as digital signatures, to distinguish between attacks and regular modifications.
- **Authentication:** To ensure the receiver that the given video sequence is originated from the intended (legitimate) sender rather than an impersonator of that sender. Following the low-complexity privacy protection mechanism in [26], which scrambles the frame region containing sensitive information, each frame can further undergo a blind-watermarking process to normalize the scrambling map and embed the authentication information.

Conventionally, cryptosystems used digital signature methodology to address sender authentication and data integrity issues. In WMSNs, the encoders will be the tiny battery-operated video sensors, which have serious computation and energy constraints. Thus, it is not practically possible for the video sensors to perform the processing intensive computation of digital signature for each individual frame and bear associated transmission overheads in addition to the original video. Moreover, WMSNs also suffer from multipath fading effect, path loss, channel interference and other environment factors. Therefore, video security mechanisms pertinent to WMSN have to deal with not only resource limitations and a different video coding paradigm, but also with the environment factors, to address such issues as video data integrity and authentication [27].

#### **2.4.1 Cryptography and Watermarking**

Several analogies have been found in literature [28-30] to define the relationship between cryptography and watermarking. However, it is evident that no single solution is sufficient to deal with all types of security threats. Therefore, a combination of security mechanisms in the same or different protocol layer has been a common solution approach [31]. Prior to making the case for which one (cryptography or watermarking) is more suitable for WMSNs, I will first clearly explain each of them along with their anticipated goals.

- Cryptography deals with secure transmission of a message from sender to receiver through an insecure communication channel, with secure transmission being characterized by three aspects, namely privacy, integrity and authenticity of the

message. It is meant to keep the communication secret primarily by the use of symmetric and asymmetric cryptography, and hash primitives, accompanied by various key distribution and trust management techniques. Only those who have the key(s) are able to access the hidden content.

- Watermarking, on the other hand, is a multidisciplinary field which coalesces diverse areas such as signal processing, cryptography, communication theory, coding, compression, human visual system (HVS) and video quality requirements [27]. It is the branch of information-hiding that is applied to embed watermark (or digital signature) into the digital data signal such that it is hard to remove from the signal without the extraction algorithm.

Watermarking is a promising approach for ensuring authentication, privacy and digital copyrights protection in sensor network environment due to its lightweight processing as compared to the conventional cryptographic approach [28, 32]. In contrast to cryptography, watermark information can be embedded without additional transmission overheads since generally the watermark bits would not add to but replace some bits of the original content at locations determined by a specific watermarking algorithm. In addition, using cryptographic approach to encrypt the entire dataset in WMSN may incur heavy computational overhead at source nodes due to much larger size of the dataset as opposed to the scalar data which consequently induces a greater delay in real-time communication.

However, there have been several works in literature on secure multimedia communication that integrated both approaches to complement the weakness of each other [6, 30, 33-35]. For example, public key cryptography and watermarking approaches have been applied jointly to test and verify digital images/frame in [36]. The owner encrypts the watermark using its public key and embeds it into the least significant bit (LSB) of transform coefficients. On the other hand, the watermark is extracted from the test image/frame and decrypted using the legitimate party's private key for authentication purpose. This scheme protects the watermark against tampering and forging even though the embedding algorithm used is public. A similar approach is proposed in [33-34] where a binary watermark is used to reveal the ownership in the host image/video signal. Visual cryptographic algorithms along with the watermark are exploited to generate the publicly and privately shared watermarked images/videos. While the watermarking algorithm used is

public, only those users who possess the privately shared image are able to retrieve the watermark from the test image and verify the ownership.

Similarly, watermarking and hashing approaches may complement each other to facilitate robust image authentication [37-40]. Watermarking is well-known for its data hiding characteristic while hashing is a promising technique for multimedia authentication. Their astute combination can present versatile and efficient solutions in different network environments. On the other hand, exploiting hash primitives alone for image authentication may not be an energy efficient approach. This is due to the fact that most of the algorithms process the entire image/frame and perform intense computations to generate the hash signature. The hash signature represents an abstract of the entire image/frame which is transmitted to the receiver along with the original image/frame. Upon receiving the image/frame, the hash signature is recomputed and compared with the one transmitted by the source node. If the signatures match, the received image/frame is marked as authenticated or otherwise rejected by the receiver. For resource limited sensor nodes, performing complex processing to generate a hash signature for the entire image/sequence of video frames is not a viable option. Secondly, in the error prone wireless environment, packet losses may cause a mismatch in signatures generated by the source sensor nodes and the receiver, resulting in image rejection and retransmission [41].

Solutions to all three security problems (privacy, integrity and authenticity) have been provided based on cryptography, watermarking, hashing, and sometimes a combination of all three. Rahman et al., [42] presented a privacy protection mechanism to hide sensitive information from the video while providing efficient surveillance. They used a cryptography based data scrambling approach to render regions of video frame that contain privacy-breaching information. Multiple levels of abstraction are used to meet the privacy privileges for various user roles. The scheme is claimed to be computationally efficient and intended for real-time video transmission. Some related works are also proposed in [43-47].

Wei et al., [48] proposed a watermarking based framework to address privacy and authenticity issues in a video surveillance system. The idea is based on monitoring only the unauthorized persons in a given area so that privacy of authorized persons can be ensured. RFID sensors are used to distinguish between authorized and unauthorized persons. From a given video frame, the regions representing the authorized persons can be removed (and restored again if necessary in some circumstances) using a secret key. Watermarking is used to embed and hide the

privacy information (associated with the authorized person) into the video with minimal perceptual changes, while a digital signature is embedded in the packet header for authentication of the watermarked video. There are some other works that address the issues of privacy, integrity, and authenticity of video based on the watermarking approach [25, 49-55].

Video authentication using the cryptographic approach commonly employs a digital signature or message authentication code (MAC) for transmission along with the message to receiver. However, this additional information introduces transmission overhead and there is a probability that the MAC may be corrupted due to format conversion [28]. Secondly, the authentication process using cryptography is content-dependent since digital signature/MAC is computed on the basis of entire content, which makes it a computation-intensive task. On the other hand, this approach provides robust authentication since a few bit inversions/errors will declare the message as corrupted. However, in practice, bit inversions/errors are common in WMSN environment due to lossy nature of the wireless channel.

Watermarking, a more flexible and lower complexity solution, is able to ensure that the semantic meaning of the digital content has not been modified by illegitimate sources, while being sustainable against wireless channel errors, lossy compression and other signal processing primitives [28]. In contrast to cryptography, watermarking can be used to embed a watermark into the content without additional transmission overhead. Generally, the least significant bits (LSB) of the transformed coefficients for an image/video frame are replaced by watermark bits at particular locations/pixels specified by algorithms such as block, edge and corner detection [56].

Nevertheless, watermark security can be complemented by the using cryptographic approach, such that an attacker is not able to detect the watermark within the intercepted frame sequence. Only the legitimate receiver who successfully reconstructed the “encrypted” watermark using a detection algorithm can decrypt it and compare it against the one stored at its own site.

There have been a number of works that recommend the use of symmetric cryptography since its processing requirements are relatively low. However, it requires strong trust management for the distribution of keys, which may be feasible for scalar data based WSN applications, but impractical for those with multimedia transmission [45]. Compared to symmetric cryptography, asymmetric cryptography is a better solution against eavesdropping and compromised nodes

attacks. However, it requires some nodes to have higher processing capability in order to execute signature verification operations [45]. Public key cryptography is a computation intensive approach which is not feasible for resource-limited, battery-powered camera sensor nodes, which also need to perform video processing and transmission operations along with video capturing. Mechanisms such as RSA [57] and digital signatures, which are primarily based on public key cryptography, are complex with significant storage and energy requirements, making them inappropriate for WMSNs [58].

I conclude this section with the remark that the security mechanisms based on watermarking are suitable for WMSNs due to their lower complexity and simplicity. However, cryptographic and hashing approaches can be used to complement the watermark embedding and detection mechanism, whilst considering the environment constraints, and there still exists a considerable research gap in designing security solutions for WMSNs [7].

## **2.5 Digital Video Watermarking**

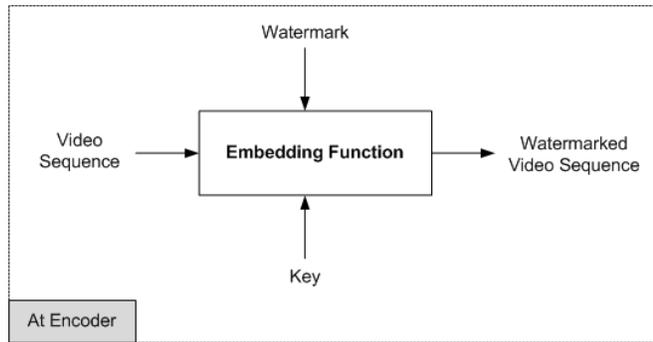
Hui-Yu et al., [59] suggested that watermarking techniques can be made practical for WMSN by exploiting the principles of distributed source coding, which utilize the duality that lies between data hiding and channel coding with side information. Digital video watermarking techniques have been widely studied and implemented in various application domains but hardly at all for WMSN [60]. The video watermarking framework generally comprises of following three modules as shown in Fig. 2.5:

- **Embedding Function**

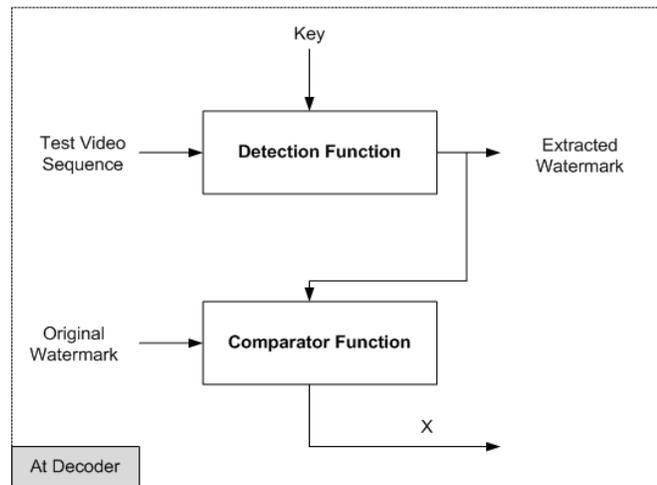
The embedding function resides within the source camera sensor and typically it has three inputs: (i) the information to be embedded as a watermark; (ii) the video sequence for which the watermark is to be embedded; and (iii) a key that is optional and employed to provide an additional level of security to the watermark information.

- **Detection Function**

The detection function is applied at the sink after the video decoder decodes the received video. The detection function extracts the watermark information using the key (if employed) and passes it on to the comparator function to verify its successful recovery and to ensure the authenticity and integrity of the received video.



(a)



(b)

**Figure 2.5 - Watermarking framework** (a) Watermark embedding at encoder; (b) Watermark detection and verification at decoder

- **Comparator Function**

This function generates a real value that indicates the degree of watermark reconstruction by comparing the original watermark  $W$  with the reconstructed watermark. If the value is equal to or greater than the predefined distortion threshold measure, the watermark is taken to be successfully reconstructed and the authenticity of the video is verified. Otherwise, the video frame is considered suspicious and discarded. The distortion threshold measure is usually setup in accordance with the channel conditions. It provides flexibility in situations where a significant portion of the watermark has been recovered but not exactly 100% since data loss is very likely in error-prone wireless environment.

### 2.5.1 Design Goals for Video Watermarking

Besides the underlying approach and structure of video watermarking algorithm, Hartung and Kutter proposed the following design goals for multimedia watermarking schemes [61].

1. *Robustness*: refers to the degree of survival of the watermark against modification in a watermarked video due to attacks such as frame dropping, re-ordering and averaging, compression, noise, cropping, recoding at lower bit rate, and among others.
2. *Invisibility*: refers to the degree to which the watermark seems to be imperceptible in the given watermarked video. A relative measure to the human visual system (HVS).
3. *Capacity*: refers to the tolerable amount of information in the form of watermark bits that can be hidden in the multimedia content.
4. *Security*: refers to the security of the watermarking algorithm using some cryptographic approach.

### 2.5.2 Classification of Video Watermarking Schemes

It is widely believed that the majority of the image watermarking schemes can be applied to video with little or no modification since video can be seen as a periodically-spaced sequence of still images. However, the reality is different from this perception due to reasons such as real-time requirements and complexity in video watermarking, excessive redundancy among frames, and un-normalised balance between active and non-active regions. Watermarking schemes can be classified into four broad categories (Fig.2.6) as follows:

#### 2.5.2.1 Implementation Domain

The majority of watermarking schemes fall into two categories with reference to their implementation domain; namely spatial domain, and transform domain.

1. *Spatial Domain Video Watermarking*: refers to schemes that exploit the correlation among pixels of video frames to embed the watermark. Embedding is performed by carrying out simple operations such as changing pixel location, intensity, pixel replacement at specific regions of video frames to incorporate watermark into

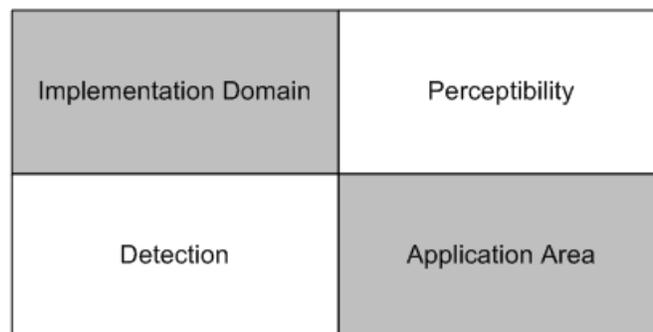
original content. Spatial domain schemes have low complexity in terms of implementation and processing but are more perceptible to HVS and less robust to attacks. Some spatial domain watermarking schemes are proposed in [62-64].

2. ***Transform Domain Video Watermarking***: refers to schemes in which the watermark is inserted into transformed coefficients of video, which provides more information hiding capacity and more robustness against watermarking attacks. Watermarking in transform domain is more robust than in spatial domain because information can be spread out to the entire video frame. The most common transformation methods are discrete cosine transform (DCT) [59, 65], discrete wavelet transform (DWT) [66-67] and discrete fourier transform (DFT) [68].

### 2.5.2.2 Perceptibility

Video watermarking can be categorized into perceptible and imperceptible schemes, having HVS as a classification parameter.

1. ***Perceptible Video Watermarking***: refers to schemes in which the watermark is embedded in a visible manner into host video signal [69-70].
2. ***Imperceptible Video Watermarking***: refers to schemes in which the watermark is embedded in an invisible manner into host video signal [63, 71].



**Figure 2.6** - Classification parameters of video watermarking schemes

### 2.5.2.3 Detection

Watermarking schemes can also be classified on the basis of the information required by the watermark detection algorithm at decoder to detect the watermark from reconstructed video.

1. ***Blind/Oblivious Video Watermarking***: refers to schemes that do not require the original video at the decoder. However, the secret key and watermark bit pattern are required for extraction of watermark from video frames [67, 72-73].
2. ***Non-Blind/Non-Oblivious Video Watermarking***: refers to schemes that require the secret key, watermark bit pattern and the original video at receiver site to detect the watermark from the watermarked video.

#### **2.5.2.4 Application Area**

Video watermarking schemes can also be classified into robust, semi-fragile and fragile in reference to application goals and objectives:

1. ***Robust Video Watermarking Schemes***: refers to watermarking schemes that can endure the majority of video compression and processing operations. Applications such as copyright protection and ownership authentication entail that the watermark have to retain adequate robustness in the context of security against malicious attacks and error-prone network transmission [63-64].
2. ***Semi-Fragile Video Watermarking***: refers to schemes that are particularly premeditated to cope with certain levels of distortion either in the form of noise or attacks [52-53, 55].
3. ***Fragile Video Watermarking***: refers to schemes that are sensitive to all potential modifications in video frames. Generally, fragile watermarking schemes are intended for applications such as content authentication and video integrity verification. Since these schemes are sheltered under cryptographic security along with strong localization characteristics, therefore even minor changes to the frame can be detected [74-75].

## **2.6 Chapter Summary**

This thesis is based on three different yet inter-related topics, the background of which is covered in this chapter, namely WMSN, video watermarking, and video coding. It is expected that most real-life WMSN applications will require adequate security mechanisms, and watermarking is a potential solution as traditional approaches such as

cryptography and steganography have much higher computational requirements. There is a clear need for video watermarking techniques that deal with not only the security issues but also the resource constraints of WMSNs. Achieving optimal use of resources such as node processing capability, power, and network bandwidth while providing level headed protection against various attacks at the same time is the fundamental design goal of security mechanisms in WMSNs. Thus far, research on addressing issues such as privacy, confidentiality, and authentication in WMSNs is still at its infancy and therefore represents an open problem space with much room for innovation.

In the next chapter, we presented a detailed literature review on the video coding architectures and the existing watermarking techniques appropriate for multimedia data in a WMSN environment.

# Chapter 3

## Literature Review

### 3.1 Introduction

This chapter provides an insight into the work related to the DVC and DCVS coding architectures along with various watermarking schemes pertinent to WMSN environment. Section 3.2 presents the detailed review on state-of-the-art DVC and DCVS architectures and their applicability to WMSN environment. Section 3.3 covers the watermarking schemes closely related to WMSN, in the literature up to the present date. Finally, Section 3.4 summarises the entire chapter.

### 3.2 Review of State-of-the-Art DVC, DCVS Architectures and H.264/AVC

DVC is an emerging video coding paradigm for applications that have restricted resources available at encoder. It reverses the conventional video coding paradigm by shifting the complexity of the encoder entirely or partially to the decoder, assumed to be a more resourceful machine than the encoder [19, 76-78]. Therefore, DVC based encoders are much simpler than conventional encoders which makes them a promising candidate for video encoding in WMSNs. Wyner-Ziv is one of the most well-known video coding architectures

based on DVC approach where intra-frame encoding is used to encode video frames, which in turn are conditionally decoded using an inter-frame decoder that exploits side-information obtained by interpolation or extrapolation of previously decoded frames. So far, several different architectures supporting the concept of evolving DVC paradigm [22, 79-93] have been proposed in literature.

Another low-complexity video coding paradigm that came into existence recently integrates the core principles of DVC and compressive sensing/sampling (CS) [94-95]. CS is a relatively new theory that shows how a sparse signal can be reconstructed from measurements far fewer than required by traditional methods based on Nyquist's sampling theory [96-97]. The process of CS video encoding is comprised of two primary steps: obtaining CS measurements of the sparse video signal, and quantizing the compressed measurements to generate the bitstream. On the other hand, the decoder involves complex operations such as decoding of bitstream, de-quantisation and  $\ell_1$  regularization of CS measurement [24, 26].

### **3.2.1 DVC in WMSNs**

Here we discuss the applicability of DVC in wireless sensor network (WSN) and how well it fits under its constrained environment. Almost all of the video coding applications fall within two classes of application models, namely downlink and uplink models. The downlink application model is associated with the broadcasting approach, where a low-complexity decoder is desirable and the complexity of the encoder is not an issue. The encoder of the downlink application model is more like a base-station that does not have constraints on the computational resources. Applications such as video streaming, broadcasting, and telephony are belong to the downlink application model.

On the other hand, the uplink application model, represents the reverse paradigm, where low-complexity encoder is required and the complexity of the decoder is not a major concern. Consider an environment which comprises of several integrated units (wireless camera sensors) that include image/video sensing modules with on-board processing and transmission functionality. These integrated units are interconnected with each other via wireless networking protocol, communicate over radio links, and have limited battery life.

Popular video coding standards such as MPEGx and H.264x/AVC supports only the downlink application model, while the DVC is a solution for applications that follows the

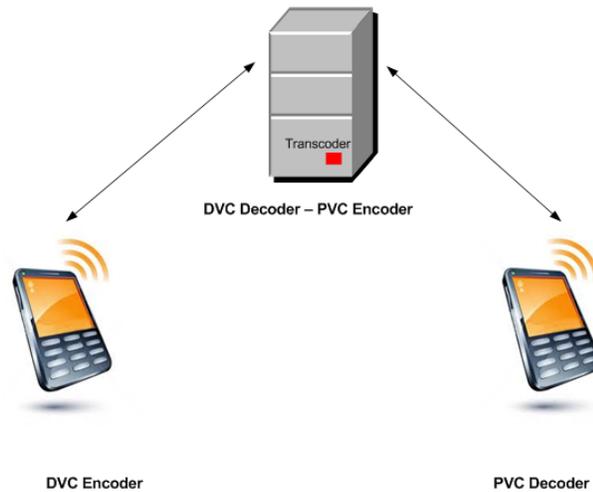
uplink model. Using DVC in the uplink application domain has potential advantages such as flexible allocation of functionality from encoder to decoder (and even transcoder in some cases), low-encoding complexity that in turn leads to low power consumption, longer battery life, and light-weight processing devices [19]. DVC theory came into place with a notion of shifting computational complexity from encoder, which makes it a viable option for applications in WSN domain. The rate-distortion performance of DVC codecs is also comparable to the conventional H.264 Intra coding paradigm [98]. Another important feature of DVC which is desirable in WSNs is that it provides better error resilience. DVC does not employ any prediction loops at encoder and therefore, no estimation and prediction errors are sent to the decoder. Rather, the side-information module in DVC decoder predicts the frame by exploiting statistical and temporal correlation among video frames [19].

Exploitation of correlation in multi-view camera architecture is also a distinguished feature of DVC for WSNs. For example, if multiple wireless video sensors have some overlapping coverage areas, they may not need to communicate with each other to exploit that correlation; rather they encode their video sequences independently and proceed for the transmission to the decoder, which is responsible for processing the correlation among video sequences from these sources [99]. DVC exhibits higher computational complexity at decoder, which consequently makes it a nonviable option for applications that require real-time decoding in WSN. By itself, DVC is not sufficient to support an end-to-end wireless video surveillance system. For this purpose, the use of a transcoder is mandatory to make both ends of the network (encoder and decoder) as simple as possible. However, the middle tier, i.e. the transcoder, requires video to be encoded by conventional PVC based codec. The basic architecture of transcoder is shown in Fig 3.1.

### **3.2.2 DVC Architectures**

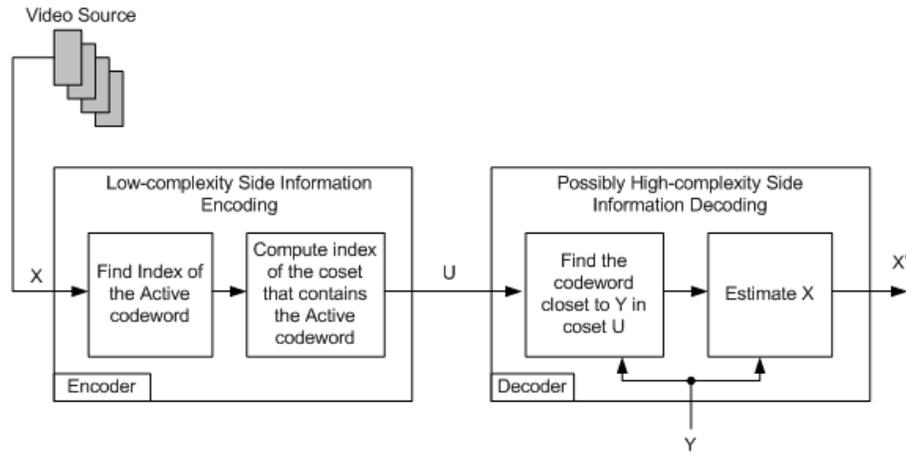
As previously discussed in Section 3.2.1, it is shown that theoretically it is possible to achieve a coding efficiency comparable to that of conventional PVC paradigm when the two sources encoded their correlated video sequences independently and decoded them jointly by exploiting the statistical correlation. However, the theorems have not identified the methodology to attain practically the same compression efficiency. Hence, researchers

have devised their own implementation of DVC codecs with various different side-information generation methods. For example, some codecs use the same DVC encoder while others employ the conventional PVC encoders (Intra mode) to generate side-information. Practically, the compression efficiency of DVC codecs is comparable to PVC codecs (executing in Intra-mode) [19].

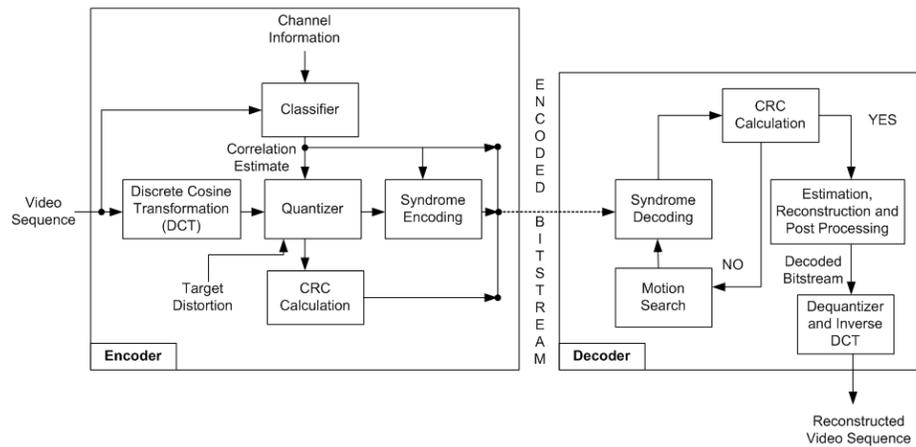


**Figure 3.1** - Video transcoder framework.

Two primary DVC approaches have been proposed in the literature, namely Berkeley [100] and Stanford [77, 80-81] video coding architectures. The Berkeley architecture known as PRISM followed a *block-based* encoding approach with motion estimation module at decoder. On the other hand, the Stanford architecture adopted the *frame-based* encoding approach, which has gained much popularity in the community because of its comparatively better rate-distortion performance than its counterpart. Subsequently, enhancements were made to the original design such as the extension of PDWZ to transform-domain Wyner-Ziv (TDWZ), replacement of turbo codes by low-density parity check (LDPC) channel codes, development of more efficient reconstruction algorithms, and employment of Intra coding mode from state-of-the-art PVC architectures for more efficient generation of a side-information. In the following section, we discuss the three main DVC architectures as mentioned above in more detail, namely PRISM [100], Pixel Domain Wyner-Ziv (PDWZ) [81], and Transform Domain Wyner-Ziv (TDWZ) [80] codec.



(a)



(b)

Figure 3.2 - PRISM: a) Logical architecture; b) Structural framework.

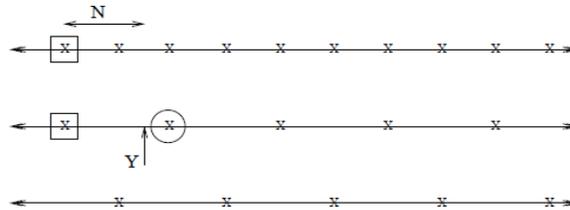
### 3.2.2.1 Power-efficient, Robust, hIgh-compression, Syndrome-based Multimedia coding (PRISM) Architecture

The PRISM (also known as Berkeley DVC) architecture proposed by Puri et al. [100] was designed to achieve compression efficiency comparable to PVC but with lower encoding complexity than its counterpart. The logical architecture and structural framework of PRISM are shown in Fig 3.2 a, and b, respectively.

- Encoder

Each video frame is first decomposed into non-overlapping blocks of size  $n \times n$  where  $n \in \{8, 16\}$ . Thereafter, each macro-block undergoes the following encoding steps.

**Transform Coding:** 2D discrete cosine transform (DCT) is employed to transform each macroblock from spatial to frequency domain with a computational complexity equivalent to that of performing intra-coding. The transformation yields DCT coefficients for each of the macroblock that must be transformed into quantized codewords prior to encoding.



**Figure 3.3** - Uniform scalar quantization step-size and codeword distribution [101].

**Scalar Quantization:** Let  $X$  represents the current macroblock to be encoded,  $Y$  to be its side-information generated from a previously reconstructed frame, and  $N$  is the correlation noise. In Fig. 3.3, the first line represents the quantized codewords for  $X$  while the next two lines represent the corresponding partitions of  $X$ . The observed codeword is surrounded by the square in the first partition. If the quantization step is less than the magnitude of  $N$ , then decoder decodes the codeword (circle on second line) and leads to decoding error. Therefore, the choice of quantization step should be directly proportional to the standard deviation of  $N$ . The scalar quantisation block generates the space of evenly distributed quantized codewords for  $X$ .

**Syndrome Encoding:** Following the quantization step, the quantized codewords are partitioned using Euclidean space Trellis channel code which operates on BCH<sup>1</sup> code [102] for error correction. A lightweight convolution process is executed between the parity matrix of Trellis channel code and quantized codewords to generate coset index (syndrome) for each codeword.

<sup>1</sup> BCH error correcting codes proposed by Bose, Ray-Chaudhuri, and Hocquenghem in 1960. The abbreviation comprises of the initials of the three inventors.

**Refine Quantization:** In scalar quantization, the step size is restricted due to correlation noise  $N$ , which is crucial to avoid decoding errors (Fig. 3.3). Depending on the target reconstruction quality, step size is chosen to be fine or coarse. Therefore, the refined quantization step is applied to coefficients to fine-tune codewords such that the desired target quality is achieved.



**Figure 3.4** - PRISM's bitstream packet format.

**CRC Calculation:** It was observed in [100] that side-information encoding is performed in relation to the motion estimation and the prediction error. Therefore, an error checking module was incorporated in [103] that computes and transmits cyclic redundancy check (CRC) bits of quantized sequence along with the sequence itself. The bitstream format of the associated block is shown in Fig. 3.4.

- **Decoder**

Description of the decoder modules is as follows:

**Syndrome Decoding:** The syndrome bits of a sequence received at decoder are aligned according to trellis channel code to generate the quantized codeword sequence. Thereafter, the decoder employs Viterbi algorithm [104] to determine the closest predictor sequence among the set of predictor sequences.

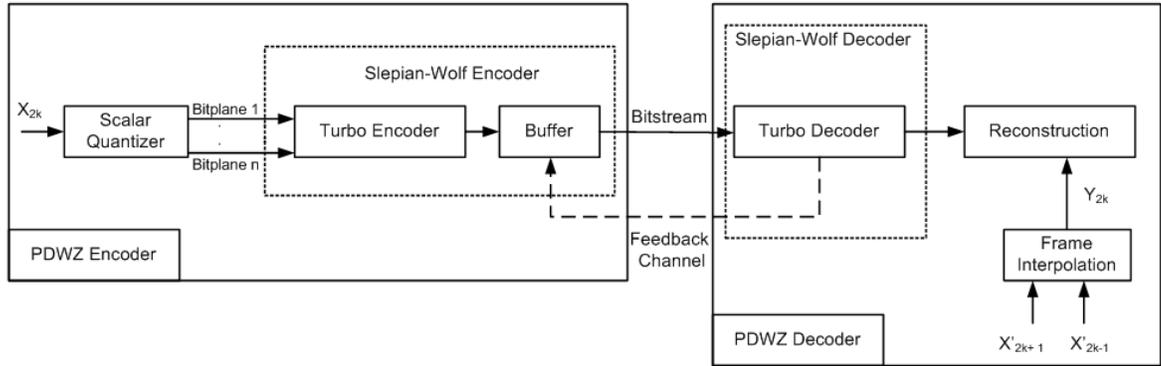
**Motion Search and CRC Calculation:** While performing motion search on decoded syndrome bits, the decoder looks up the entire set of predictors for the best match and compares the bit sequence with the corresponding CRC check sequence, and marked it as a successful reconstruction if an exact match occurs between the two.

**Motion Estimation, Reconstruction and Post Processing:** The quantized codeword sequence together with the predictor is used for the reconstruction (transform coefficients) of the source sequence. The PRISM framework can adopt any of the reconstruction algorithms from spatio-temporal correlation interpolation to efficient post processing algorithms.

**Dequantiser and Inverse Transformation:** Following the reconstruction of the transform coefficients of the predicted source signal, de-quantization and inverse DCT transformation is performed to extract pixel values of the block.

### 3.2.2.2 Pixel Domain Wyner-Ziv (PDWZ) Video Coding Architecture

The Wyner-Ziv (WZ) architecture, also widely known as Stanford DVC architecture, was originally designed based on pixel domain WZ (PDWZ) [81] coding. In later years, it was enhanced to transform domain (TDWZ) [80]. In pixel domain WZ codec, which relies on intra-frame encoding and inter-frame decoding, the video sequence is split into  $X$  and  $Y$ , representing sets of even and odd frames, respectively. Intra-frame encoding is used to encode  $X$  given that  $X$  does not have any knowledge of  $Y$ . The redundancy between successive frames is exploited to determine the side-information ( $Y$ ) at decoder, which in turn employs  $Y$  to conditionally decode  $X$ . The architecture of PDWZ codec is shown in Fig. 3.5.



**Figure 3.5** - PDWZ architecture.

- **Encoder**

Let the frames to be encoded represented by  $X_1, X_2, \dots, X_N$ , and the set of odd numbered frames  $X_{2i+1}$  termed as key frames are available at decoder, where  $i \in \{0, 1, \dots, \frac{N-1}{2}\}$ .

Therefore steps for the compression of even numbered frames  $X_{2i}$  are as follows:

**Scalar Quantiser:** The symbol stream is initially generated by quantizing each pixel of every row of the entire frame at  $2^M$  distinct levels using uniform scalar quantizer. The

resulting quantized symbol stream of length  $L$  for each even numbered frame is then fed into the Slepian-Wolf turbo encoder.

**Turbo Coder and Buffer:** In order to achieve bit rate flexibility, rate compatible punctured turbo (RCPT) coding is implemented, which dynamically adapts to the coding parameters associated with mismatches that occur between the frame to be encoded and its side-information. Each block of input symbols from the quantized stream is assigned a parity sequence and the blocks that have same parity sequence are grouped together in the same coset. Thereafter, the parity sequence is temporarily stored in buffer and transmitted in small chunks to the decoder as and when required. Such an arrangement ensures that the encoder will transmit only a small amount of parity bits to the decoder for reconstruction of quantized bitstream. However, the decoder continues to generate feedback requests until the quantized bitstream has been reconstructed with desired quality parameter.

- **Decoder**

**Frame Interpolation Model and Side-information Generation:** Temporal interpolation between two successive key frames is performed to generate side-information for the current frame to be decoded. However, the decoder design is flexible enough to adopt various interpolation techniques, ranging from simple average interpolation to complex symmetric motion vector (SVM) based motion compensation, which may include multiple frame predictors and intelligent segmentation features. The interpolation technique simply averages the pixel values of successive key frames to predict the pixel value of the non-key frame in between them at the corresponding location. However, SVM interpolates the motion based on the assumption that the motion vector remains the same between the successive key frames. Therefore, the block matching is performed between the successive key frames in order to estimate the symmetric motion vector for the given block of the sandwiched non-key frame.

**Reconstruction:** Each pixel of the frame can be reconstructed provided that its decoded bitstream and the side-information are available at the decoder. Since symbols are grouped together in cosets associated with the levels of quantisation, therefore, if the side-information is close enough to the reconstructed signal resulting from the decoded bitstream, it may fall within one of the coset's bins. Alternatively, the reconstruction

process relies only on the signal to be reconstructed, quantises it to the bin boundaries and ignores the side-information. Such scenarios may happen when there are high motion frames and various occlusions in place.

Several enhancements have been made to the original PDWZ codec design. Some of them are outlined as follows:

- Hyper-Trellis decoding for PDWZ video coding is proposed in [105] to optimize the approach for the reconstruction of WZ frames. A new decoding algorithm is presented which encapsulates and combines various states of the original trellis code. The results show that the proposed approach not only reduces the complexity of the decoder, but also increases the reconstruction quality by 9-10dB.
- In contrast to the original PDWZ codec design, where decoder controls the encoding rate via feedback channel, a low complexity rate-control algorithm that executes at the encoder is proposed in [106-107]. The proposed design shifts the rate-control functionality from the decoder to the encoder, and eliminates the feedback channel which not only reduces the decoding complexity but also the delays.

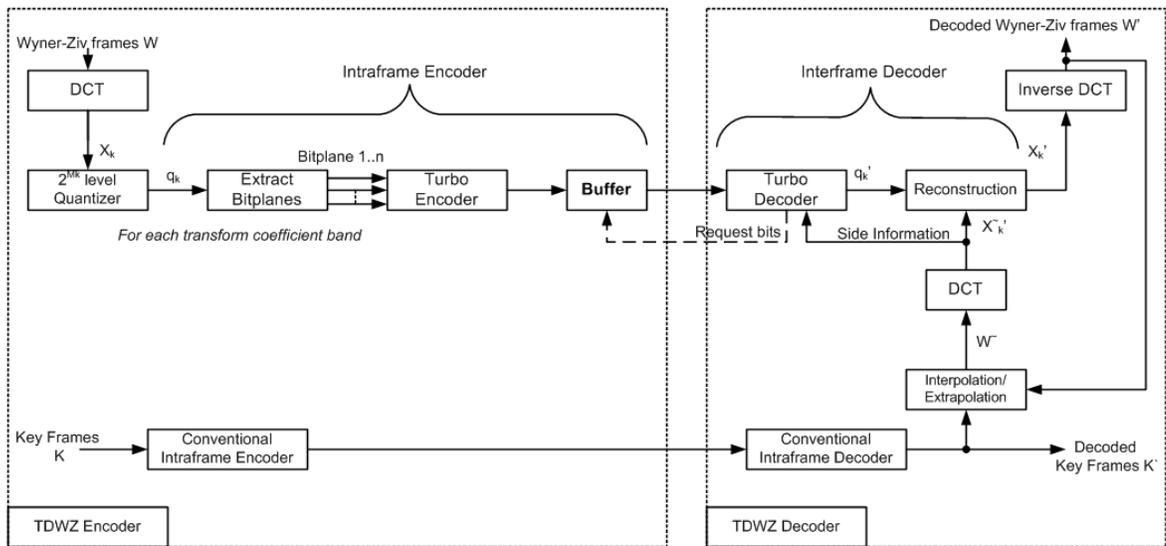


Figure 3.6 - TDWZ architecture.

### 3.2.2.3 Transform Domain Wyner-Ziv (TDWZ) Video Coding Architecture

The pixel domain WZ codec has been extended from pixel-domain [81] to transform domain [80], which exploits spatial correlation within a given frame and temporal

correlation among adjacent frames to achieve better rate-distortion performance is shown in Fig. 3.6. The inclusion of DCT module makes TDWZ a more practical WZ codec, which encodes key and WZ frames with intra-frame conventional, and WZ encoder, respectively. At the decoder site, key frames are reconstructed via conventional inter-frame decoder, whereas decoding of a WZ frame requires side-information generated by the previously decoded key as well as WZ frames. In the following sub-sections, we will discuss only those modules of TDWZ which differ from those in the PDWZ codec architecture.

- **Encoder**

TDWZ codec splits the video sequence into key and WZ frames encapsulated within a group of pictures (GOP).

**Discrete Cosine Transform (DCT):** Each WZ frame is decomposed into sub-blocks which undergo DCT transformation and generate DCT coefficients. These DCT coefficients are assigned to different bands according to their position in the DCT block. Thereafter, each DCT band is quantized into a number of quantization levels via a uniform scalar quantizer.

**Bit-Plane Extraction:** Quantized DCT coefficients (symbols) are grouped together into bit-plane vectors and fed independently to the Slepian-wolf turbo encoder.

**Turbo Encoding:** Turbo encoder starts encoding each bit-plane vector using RCPT and the resulting parity information is temporarily stored in buffer and subsequently transmitted to the decoder in small chunks upon requests received from the feedback channel.

**Conventional Intra-frame Encoder:** Intra-frame encoding mode of conventional video codecs such as H.26x/AVC is used to encode key frames, which upon being received at decoder, are reconstructed via conventional Intra decoder.

- **Decoder**

The decoder processes the video frames according to the GOP configuration and operates conventional intra decoder and WZ decoder in parallel for the reconstruction of key, and WZ frames respectively. However in [80], the GOP size was set to 2, which implies that every alternate frame is a key frame. In later versions [85, 87, 98, 105, 108-109], the decoder was extended to support GOP sizes of 4, 8 and 16 frames.

**Conventional Intra-frame Decoder and Frame Interpolation/Extrapolation:** Key frame decoding is relatively straightforward, since it only exploits the spatial correlation in the given frame. However, the reconstructed key frame also provides an estimate for the WZ frame to be decoded.

**DCT Transformation:** On receiving side-information  $\tilde{W}$ , block-based DCT is performed and the resulting transformed coefficients are aligned to form coefficient bands  $\tilde{X}_k$ , which is an estimate of each decoded bitplane of the received WZ frame  $X_k$ .

**Turbo Decoding and Reconstruction:** The turbo encoder-decoder in PDWZ and TDWZ is utilized as a Slepian-Wolf codec. Each bit-plane vector is turbo decoded, given that the side-information  $\tilde{X}_k$  and the residual statistics are available. However, if the decoder cannot decode a bit-plane, it requests additional parity bits from encoder via feedback channel, and the process continues until a certain acceptable level of bit error rate performance is achieved.

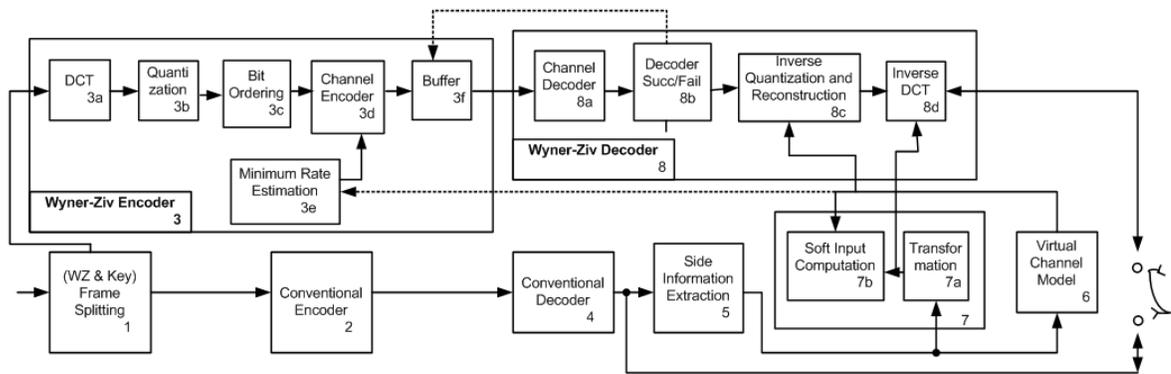
Several improvements have also been made to the original Stanford TDWZ architecture; a few of them are discussed below:

- It is observed that the accuracy of side-information generation has a significant impact on codec's overall performance. Several enhancements have been suggested for the estimation of side-information in TDWZ codec. This includes a progressive side-information refinement framework introduced in [110], which exploits the spatial and temporal correlation among previously decoded frames to improve gradually the side-information as the decoding process progresses. Various approaches for the enhanced side-information module based on progressive side-information refinement, motion searching, resolution-progressive decoding, and extensive motion exploration are presented in [109, 111-114].
- One of the most challenging tasks of the TDWZ codec is to assign an optimal number of bits to encode the WZ frame. Typically, a feedback channel is employed to inform the encoder about the required encoding rate. Since the encoder itself does not have access to motion compensation information of the WZ frame, significant degradation in rate-distortion performance may occur if insufficient bits have been allocated to the

frame. However, for applications that only transmit data in one way such as broadcasting, employing a feedback channel is not possible.

### 3.2.2.3.1 DISCOVER (DISTRIBUTED CODING for Video sERVICES)

Under DISCOVER [98, 115], a European video coding project, several new modules have been introduced and existing modules improved to enhance the overall performance of the TDWZ codec. Notably: i) adaptive GOP selection and encoder rate control mechanisms with input from decoder's virtual channel are introduced; ii) turbo coder is replaced by LDPC [116] coder; iii) correlation noise modelling is performed between the side-information and corresponding WZ frame via soft input computation to enhance the reconstruction quality. The architecture of DISCOVER is shown in Fig 3.7. The following further elaborates on the above key changes.



**Figure 3.7 - DISCOVER architecture.**

The selection of GOP size is made adaptive to varying temporal correlation in the video sequence [117]. By analyzing the video frames, larger or smaller GOP size can be employed for sequences having higher, or lower, temporal correlation among frames respectively. A hierarchical clustering algorithm executed at the encoder to group frames of similar motion activity, is responsible for making the decision about the GOP size. It is observed that the codec exhibits better rate-distortion performance when adaptive GOP is employed as compared to fixed GOP [117].

The LDPC channel codes introduced in [116] have replaced the turbo channel codes not only in DISCOVER, but almost all TDWZ architectures. The LDPC encoder is comprised of syndrome generator and the accumulator which stores the syndrome bits generated from

LDPC codes to form the accumulated syndromes for transmission to the decoder. In contrast to turbo codes, LDPC codes efficiently utilize the capacity of channels under varying communication requirements [116]. Furthermore, a CRC sum of the encoded bit-plane is transmitted to the decoder to perform error checking of its received bits.

The rate-distortion approach proposed in [22] is used to develop the rate-control module at the encoder. It computes the minimum rate employed by the source for a given distortion measure, and enables the encoder to determine the minimum number of accumulated syndrome bits to be transmitted per bitplane for each coefficient band. This enables the DISCOVER to exhibit comparable rate-distortion performance with conventional H.26x Intra codecs. To model the correlation noise between the transform bands of WZ frame and corresponding side-information, DISCOVER uses the Laplacian error distribution [77, 86], which considers the variance in noise statistics pertinent to spatial and temporal correlation and evaluates the distribution parameter online. The resulting noise correlation model aids in transforming the side-information's transform coefficients into soft-input for the LDPC decoder.

### 3.2.3 Comparison and Analysis of DVC Architectures

This section compares and analyzes the primary functional differences among the three video coding architectures discussed in Section 3.2.2. A table summarizing the comparison of the key features of the three primary DVC architectures is shown in Table 3.1.

**Table 3.1** - Comparison of primary DVC architectures

Architectures		PRISM	PDWZ	TDWZ
Features				
Video Coding Unit	Block-Based Coding	√		
	Frame-Based Coding		√	√
Rate Control	Encoder Rate Control	√		√*
	Decoder Rate Control		√	√
Channel Coding	BCH Code	√		
	Turbo Code		√	√
	LDPC Code		√	√

- **Block-Based and Frame-Based Coding:** For exploiting spatial correlation to generate bitstream, various DVC architectures employ either a block-based or a frame-based encoding approach, each of which has its salient characteristics. Block-based coding is relatively more adaptive to spatial variations, thus exploits local features of a video frame more efficiently and has a better reconstruction quality at the decoder side. On the other hand, frame-based encoding has the benefit of being able to deal with larger datasets, and its comparatively lower complexity is a desirable feature for more efficient channel coding. During intermediate processing stages, although a frame-based encoding approach may still offer the advantage of block-based coding, the final bitstream is associated with the entire frame rather than providing support for smaller spatial blocks. Berkeley's PRISM architecture and Stanford's Wyner-Ziv architecture are examples of block-based, and frame-based encoding approaches, respectively.
- **Encoder/ Decoder based Rate Control:** One of the challenging tasks for DVC architecture is to allocate the number of bits associated with each frame/block for transmission to the decoder. In some architectures, this responsibility is placed either on the encoder or decoder (or on both encoder and decoder). If the decoder is involved in taking a decision about the encoding rate, feedback channel is required to provide a more sophisticated and tighter control over the number of bits transmitted from the encoder to decoder. Decoder performs complex processing during frame reconstruction, and depending upon the requirements to achieve predetermined target quality, the decoder may request the encoder to change its encoding rate. For example, parity bits transmitted from the encoder are analyzed by the decoder to determine whether these bits are insufficient to achieve an acceptable level of performance. If not, it will request additional parity bits from encoder, rather than using the decoded frames with well-built traces. Such an approach is applicable to real-time video coding (active communication mode) applications because the decoder can estimate the number of additional bits required only at the time of decoding. Feedback channel also has an impact on decoder's complexity and may introduce latency if used frequently. An improvement to the side-information generation module may involve restricting the number of feedback requests from the decoder. Alternatively, the encoder rate-control approach eliminates the need for a feedback channel (passive communication mode) and estimates the number of bits needed to achieve desired target quality with additional complexity at

the encoder side. Berkeley's PRISM architecture, and Stanford's Wyner-Ziv architecture, are examples of encoder, and decoder based rate-control approaches, respectively.

- **BCH, Turbo and LDPC Channel Codes:** BCH channel codes with ability to correct multiple bit errors are very simple and have tighter control on symbol errors during the channel coding process. Decoding of BCH requires performing simple algebraic operations [102]. BCH encoding and decoding in video coding architecture is also widely known as syndrome coding and decoding, respectively. Turbo codes, on the other hand, are forward error correction codes that can achieve a channel capacity comparable to theoretical bounds. They guarantee reliable communication even at certain noise level in bandwidth or delay-constrained communications [118]. Finally, LDPC channel codes, which are the most sophisticated among the three are the linear error correction codes specifically designed to transmit information under noisy channel conditions. Similar to turbo codes, they can achieve a channel capacity close to the theoretical maximum with a very low bit error rate. LDPC decoding time is dependent on the information block length [119]. Berkeley's PRISM and early Stanford's Wyner-Ziv architectures employ BCH and turbo channel codes, respectively. However, in later years, the turbo codes were replaced with LDPC in both of the Stanford's pixel and transform domain WZ architectures.

### 3.2.4 DCVS Architectures

Generally, video signals are sparse in nature in that they contain a significant amount of redundancy in both spatial and temporal domains and therefore video compression is one of the eminent fields where compressive sampling (CS) can be applied. Following the simple-encoder and complex-decoder framework proposed in DVC theory, the decoder performs more complex computations in order to reconstruct the encoded signal using compressed measurements. Reconstruction of CS based encoded signals can be formulated as an Optimisation problem whose objective function and the associated constraints can be easily customized according to the requirements of the given application.

An epigrammatic description of basic CS principles and notations is as follows: Let  $\mathbf{x} = \{x[1], x[2], \dots, x[N]\}$  denotes the set of  $N$  pixels of an image/video frame and  $\mathbf{s}$  be the transform domain ( $\Psi$ ) representation of  $\mathbf{x}$ . Thus,  $\mathbf{x}$  can be expressed as:

$$\mathbf{x} = \Psi \mathbf{s} = \sum_{i=1}^N s_i \psi_i \quad (3.1)$$

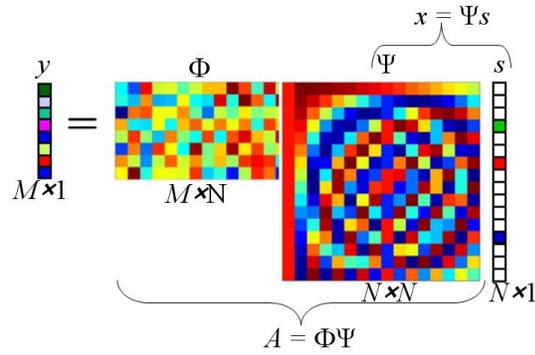
where  $\Psi$  and  $\mathbf{s} = \{s[1], s[2], \dots, s[N]\}$  represents  $K$ -sparse matrix of length  $N \times N$ , and a vector of transform coefficients of length  $N \times 1$  which can be well-approximated using  $K (< M < N)$  linear, non-adaptive measurements respectively, where  $M$  represents the length of compressed measurement matrix taken from  $\mathbf{x}$ . The CS theory states that only  $M = O(K \log(\frac{N}{K}))$  measurements can be sufficient to reconstruct a precise and accurate version of  $\mathbf{x}$  [94]. Fig. 3.8 shows the compressed measurement vector  $\mathbf{y}$  of length  $M \times 1$  and is given by:

$$\mathbf{y} = \Phi \mathbf{x} = \Phi \Psi \mathbf{s} = A \mathbf{s} \quad (3.2)$$

where  $\Phi$  represents the independent identically distributed (i.i.d) Gaussian  $M \times N$  measurement matrix. However, the unique reconstruction of  $\mathbf{x} = \{x[1], x[2], \dots, x[N]\}$  or  $\mathbf{s}$  from measurement vector  $\mathbf{y}$  (of  $M$  samples) is not possible. Therefore, an approximation of solution can be obtained using  $l_1$  minimization:

$$\hat{\mathbf{s}} = \underset{\mathbf{s}'}{\operatorname{argmin}} \|\mathbf{s}'\|_1, \quad (3.3)$$

such that  $\Phi \Psi \mathbf{s}' = \mathbf{y}$ . CS exploits convex Optimisation (such as running gradient projection for sparse representation (GPSR) [120] or orthogonal matching pursuit (OMP) [121] algorithms, etc, iteratively) to reconstruct  $\mathbf{s}'$  from  $\mathbf{y}$ . Finally,  $\hat{\mathbf{x}}$  can be reconstructed using  $\hat{\mathbf{x}} = \Psi \hat{\mathbf{s}}$ . The next section presents a number of representative distributed compressive video sensing architectures in literature.



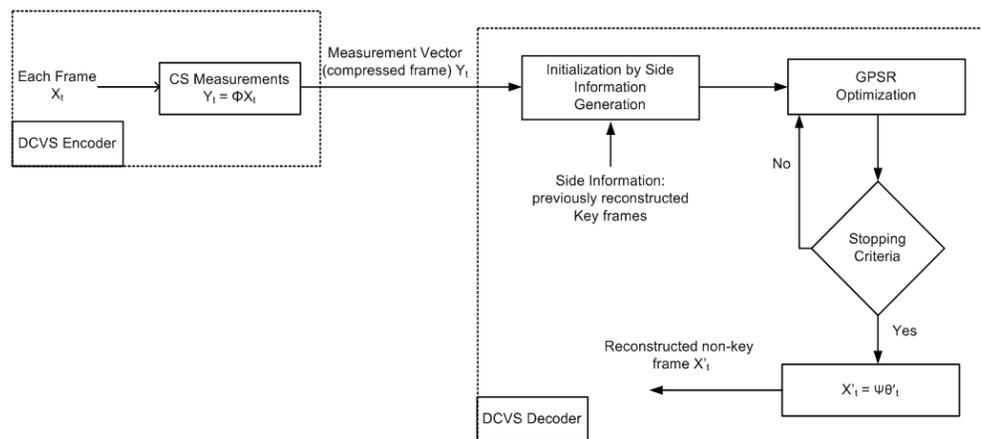
**Figure 3.8** - Compressive sensing framework

### 3.2.4.1 Distributed Compressive Video Sensing (DCVS)

The idea of DCVS was first proposed in 2009 by Kang et al. [122] who exploited both inter- and intra-signal correlation structures for compression of a video signal. Using CS technique, each source signal is independently measured and jointly reconstructed at decoder using correlation among independent compressed measurements. A concise description of this architecture's main modules is as follows:

- **Encoder Modules**

The DVCS encoder classifies video frames as key frames and CS frames, where key frame serves as a reference for the following CS frames in a given GOP. As shown in Fig. 3.9, the encoder encodes each frame independently and generates CS measurements.



**Figure 3.9** - DCVS architecture

**CS Measurements:** The CS measurement acquisition process has already been explained in above. Following the acquisition, the Hadamard block ensemble [123] is used to

compute the measurement matrix  $\Phi$  which is comparatively more efficient in terms of performance, memory requirements and flexibility than i.i.d Gaussian and binary sparse matrices techniques . The only difference between the measurement process of key frame and the CS frame is that the measurement rate of key frame should be greater than that of CS frame in a given GOP. The compressed measurement vector  $\mathbf{y}$  for each frame is then transmitted to the decoder.

- **Decoder Modules**

**Side Information (SI) Generation:** Similar to DVC, a CS frame in DCVS decoder is reconstructed using side information, which is an estimated version of a given CS frame, generated by performing motion compensation and interpolation operations on a previously decoded key frame. CS frame reconstruction follows GPSR [120] based initialization (SI) and iterates until the stopping criterion are met, which are derived from statistical correlation among successive frames.

**Initialization by SI Generation:** In order to reconstruct CS frame with reasonable quality, it is desirable to have a good initialization through GPSR algorithm. Consequently, the decoder needs to perform a lesser number of iterations to obtain the optimal solution.

**GPSR Optimisation:** Key frame reconstruction at decoder is cast as a convex Optimisation problem in which GPSR algorithm is applied and follows the same settings included in Public GPSR code [120] with multiple stopping criteria.

**Stopping Criteria:** Stopping criterion for GPSR iterations are closely related with statistical dependencies between current CS-frame and its SI:

1. *Default Stopping Criterion:* The algorithm will stop if the Laplacian parameter which represents the relative change between the reconstructed CS frame and its SI falls below the certain threshold  $T_{\infty}$ .
2. *Second Stopping Criterion:* It is possible that the default stopping criteria may not provide the optimal solution, but one which is over-sparse and has low visual quality, since it has been evaluated without considering video characteristics. Therefore, a fitness function exploiting the statistical dependencies along with

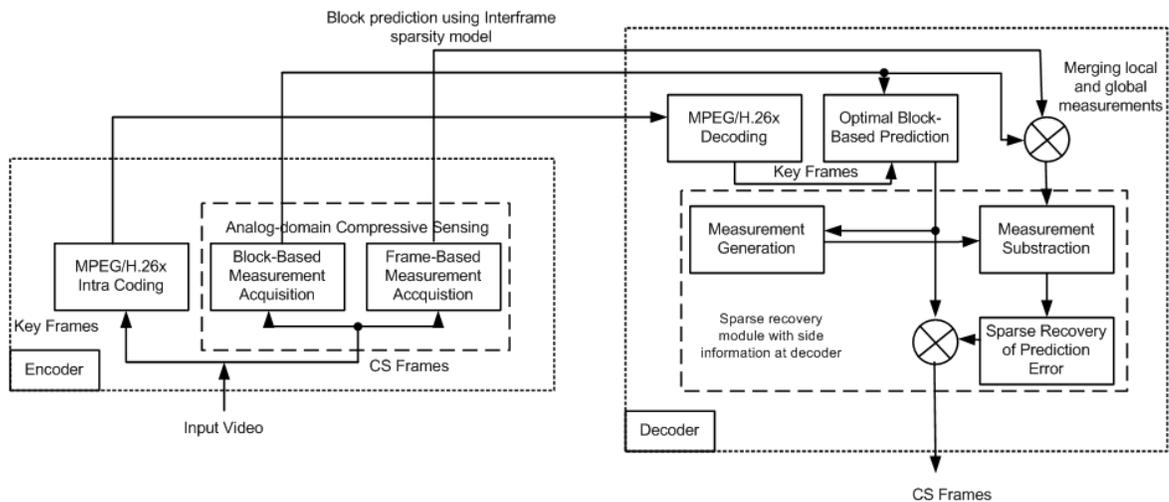
certain visual quality perseverance level has been employed as another stopping criterion.

3. *Third Stopping Criterion:* The algorithm will stop if the relative change between two iterations of the fitness function is below a threshold  $T_f$ .

The efficiency and performance of the proposed architecture using GPSR based reconstruction was reported to outperform the existing two-step iterative shrinkage (TwIST) [124] and OMP [121] techniques.

### 3.2.4.2 Distributed Compressed Video Sensing (DISCOS)

Another distributed compressive sensing based framework known as DISCOS [96] is depicted in Fig. 3.10. Similar to Wyner-Ziv codec[80], DISCOS uses conventional codec for key-frame encoding and CS technique for compression of non-key frames. The description of the architecture's main modules is as follows:



**Figure 3.10 - DISCOS architecture**

- **Encoder Modules**

**MPEG/H.26x Intra Coding:** Analogous to DCVS architecture, encoder categorises the video sequence into key and CS frames. MPEGx/H.26x is used to encode key frames while CS frame were compressively sampled using measurement ensemble. Furthermore, both the frame-based and block-based measurements of CS frames are taken and transmitted to the decoder.

**Frame-Based and Block-Based Measurement Acquisition:** DISCOS employs both block-based as well as frame-based measurements. From the coherence principle of CS-theory, frame-based measurements are known to be denser and more coherent in some sparsifying domains than block-based ones [96]. On the other hand, comparatively less coherent block-based measurements are able to capture local information which is used by the decoder to create more accurate side information (SI).

- **Decoder Modules**

**MPEGx/H.26x Decoding:** This module is responsible for decoding key frames using a conventional MPEGx/H.26x decoder. Intra-decoding mode is selected to decode key frames received periodically after a certain number of CS frames pertinent to GOP size.

**Optimal Block-based Prediction:** Block-based measurements and previously decoded key frames are used to generate sparsity constraint. Consequently, to predict the motion vector of the macroblock (and SI),  $\ell_1$  minimization algorithm follows the sparsity constraint resulting from Sparsity-Constraint Block Prediction (SCBP) algorithm proposed in [96]. Sparse Recovery with Decoder side-information (SRwDSI) is also proposed (which in turn uses GPSR) for the entire CS frame reconstruction followed by the block-prediction module.

**Measurement Generation, Subtraction and Sparse Recovery of Prediction Error:** This module extracts the prediction error by subtracting the measurement vector of currently decoding frame from measurement vector of block-based prediction frame. If frame prediction error is less than predetermined threshold, then the predicted CS frame is considered as a successfully reconstructed frame.

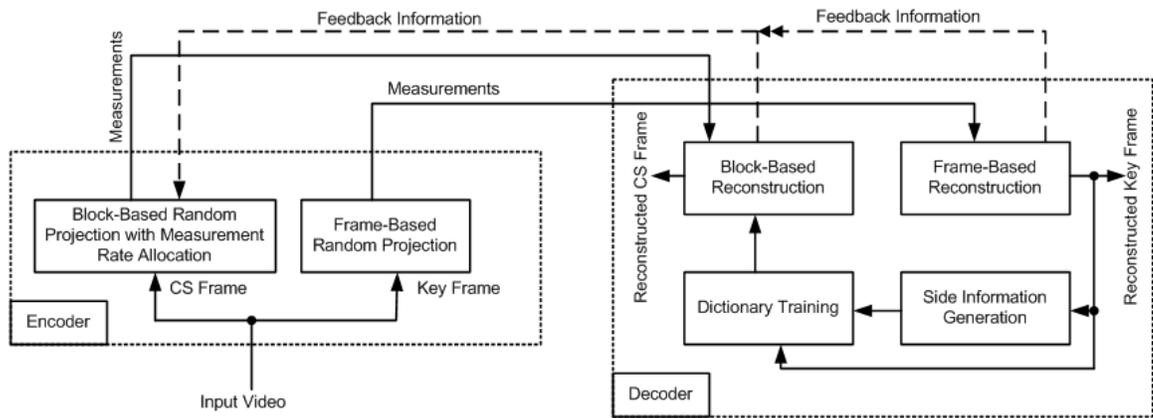
### 3.2.4.3 Dynamic Measurement Rate Allocation for Distributed Compressive Video Sensing (DMRA-DCVS)

A variant of the DCVS architecture [11] presented in [125] that implements dynamic allocation of measurement rate is shown in Fig. 3.11. The encoder adaptively alters the measurement rate by learning the sparsity information of each CS block and key frame via feedback channel. Description of the main encoder and decoder modules is as follows:

- **Encoder Modules**

**Frame-Based Random Projection:** Input video frames are split into key and CS frames depending on GOP size. Key frames are compressed by taking frame-based random projections or measurements and generate the measurement vector. The measurement rate for each key frame has been chosen to be equivalent to the target average measurement rate for the given application.

**Block-Based Random Projection with Measurement Allocation:** Conversely, each CS frame breaks into non-overlapping blocks, and the measurements for each block are individually taken and compressed into measurement vector for transmission to the decoder.



**Figure 3.11 - DMRA-DCVS architecture.**

- **Decoder Modules**

**Frame-Based and Block-Based Reconstruction:** Upon receiving measurement vector, key frame reconstruction requires solving convex unconstrained Optimisation problem via sparse reconstruction by separable approximation (SpaRSA) algorithm using parameters such as sparse coefficients, DWT basis, and scrambled block Hadamard ensemble (SBHE). On the other hand, CS frame reconstruction follows the same steps as in key frame reconstruction but at block level.

**Dictionary Training:** The dictionary (basis) for a CS frame is comprised of a set of spatially correlated blocks of the corresponding previously decoded key frames in a given GOP. Ideally, it should be constructed from the same frame instead of the key frame, but its side-information can be utilized to serve the same purpose since the contents of previously

reconstructed successive frames are generally quite similar. Thus, for a given GOP, CS frame reconstruction employs previously reconstructed CS frames and side-information generated from the key frame to train the dictionary.

**Dynamic Measurement Rate Allocation via Feedback Channel:** It is quite difficult to estimate exact sparse representation of each block of CS frame by using the non-zero CS coefficients resulting from SpaRSA algorithm. Instead, the variance of coefficients in each block is used to perform rate allocation. Target measurement rate for each block of a CS frame is computed using estimated sparse representation and variance of non-zero coefficients for that block.

### 3.2.5 Comparison and Analysis of DCVS Architectures

This section explains and highlights the primary functional differences among the three architectures discussed in Section 3.2.4.

**Table 3.2** - Comparison of primary DCVS architectures

Features \ Architectures		DCVS	DISCOS	DMRA-DCVS
Key Frame Coding	Conventional Intra Coding		√	
	Frame-Based CS Measurements	√		√
	Block-Based CS Measurements			
CS Frame Coding	Frame-Based CS Measurements	√	√	
	Block-Based CS Measurements		√	√
Decoder Rate Control				√
Dictionary Learning				√
Reconstruction Algorithm		GPSR	GPSR	SpaRSA

- Key Frame Encoding:** Various techniques have been employed for key-frame coding namely, conventional Intra coding, frame-based, and block-based CS measurement acquisition. Key frame reconstruction using conventional Intra coding is visually of high quality but more processing intensive than its other two counterparts. On the other hand, the block-based measurement acquisition, as compared to the frame-based approach, is more flexible to accommodate highly non-stationary statistics of video signals but is relatively more complex.

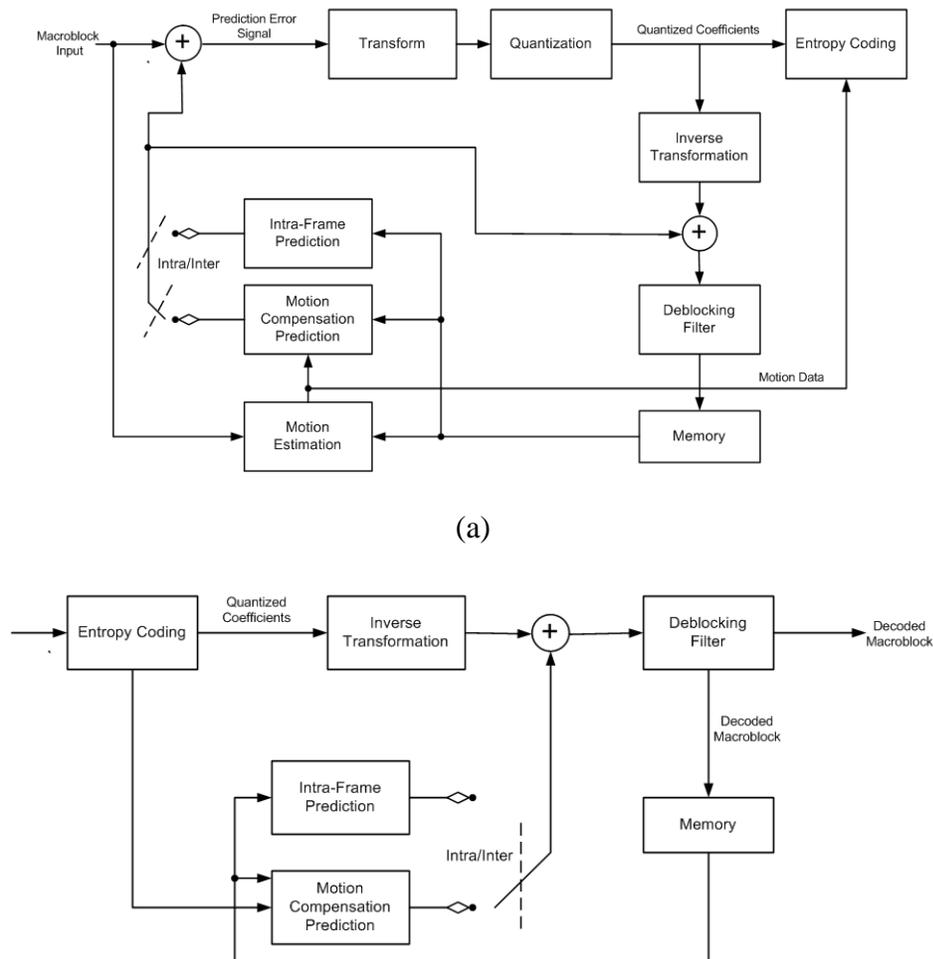
- CS Frame Encoding:** Different methodologies are in use for CS frame coding such as frame-based measurement, a combination of frame-based and block-based measurement, and block-based measurement with quantization and entropy coding. It is observed that efficient quantization and entropy coding of compressed measurements are still unexplored areas. Moreover, we view the DISCOS approach as being more reasonable in terms of frame reconstruction quality since it considers non-stationary regions of each block of the frame along with entire frame statistics. However, we still have concerns about encoding complexity since by employing conventional codec complexity increases.
- Decoder Rate Control versus Encoder Rate Control:** Decoder rate control refers to the approach where the measurement rate allocation to key and CS frames is performed at the decoder. In the encoder rate control, the encoder itself is responsible for allocating measurements rates to both frame types. In the former approach, a feedback channel is required, which restricts its applicability only to real-time (active communication mode) bidirectional applications. DMRA-DCVS support the decoder rate control approach to dynamically allocate channel measurement rate.
- Dictionary Learning based Frame Reconstruction:** The codewords extracted from video frame/image constitute the basis/dictionary for the same frame given that this dictionary should provide sparse representation of the frame. However, at the decoder, it is not possible to reconstruct that dictionary from the video frame/image itself because it is not available until the received encoded video frame/image has been decoded by the decoder. Therefore, the neighbouring frames generate training samples/atoms to produce a good trained dictionary. Dictionary learning improves CS frame reconstruction by enhancing the SI generation process. DMRA-DCVS support the dictionary learning based CS frame reconstruction.
- Reconstruction Algorithm:** There are two main reconstruction algorithms employed by various DCVS architectures, namely GPSR and SpaRSA. GPSR based reconstruction is usually suitable for solving quadratic programming formulation of convex unconstrained Optimisation problem, where the outcome of each iteration is acquired by projecting the negative gradient direction aligned with the feasible solution set. On the other hand, SpaRSA is also related with GPSR but provides a more generalize and efficient framework. It can be adapted to various Optimisation problems

by choosing different regularisers, acceptance values of target function, and stopping criteria, and therefore often leads to faster convergence than GPSR.

DVC and CS principles are widely considered as enabler of low-complexity solutions to several problems that include not only compression but also low-complexity authentication, privacy, biometrics, video fingerprinting and video error concealment applications for battery powered devices. However, the current features of these codecs can be improved and several prospective new features can be explored.

### 3.2.6 Conventional Video Coding - H.264/AVC

Finally, H.264 Intra [15, 126] is a variant of conventional H.264 video codec that encodes each frame independently like an image without performing computational intensive operations such as motion estimation and prediction on correlated frames in the sequence as shown in Fig. 3.12. Therefore, it preserves a lot of computational energy.



(b)

**Figure 3.12** - H.264/AVC hybrid video (a) Encoder and (b) Decoder architecture with motion compensation

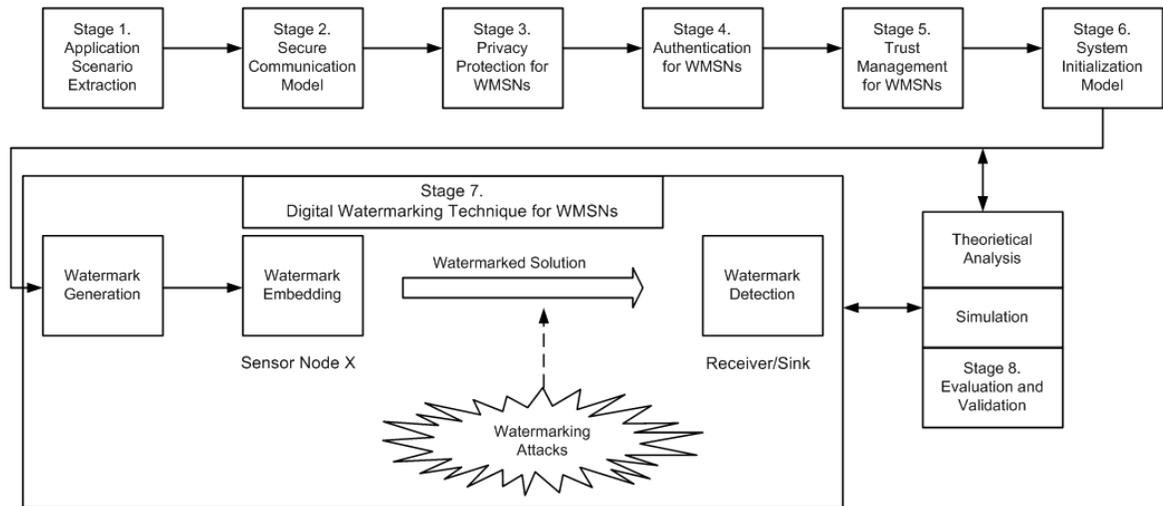
In H.264 video coding, each frame from the video sequence is decomposed into macroblocks, which in turn consist of a luminance component ( $Y$ ), and two chrominance components ( $C_r$  and  $C_b$ ) that provide the brightness and colour information of that macroblock, respectively. These blocks are encoded using Inter (long GOP) or Intra (self-contained) coding mode.

For Intra coding mode, the prediction signal is assumed as zero, which means that the block will be coded independently without using any information from previously sent image information. On the other hand, Inter coding mode encodes macroblock by exploiting motion compensation and prediction by estimating and transmitting displacement vector for each macroblock's corresponding position in previously sent reference image. The spatial and spectral correlation present in the frame is exploited by H.264 Intra coding. The Intra-frame prediction module predicts a block from its spatially adjacent neighbour in order to remove spatial redundancy. Each frame from a given sequence is partitioned into macroblocks of size  $n \times n$ . Subsequently, each macroblock is sampled for luma ( $Y$ ) and chroma ( $C_r$  and  $C_b$ ) components and spatially predicted using the intra-coding prediction module. The resulting prediction residual is transmitted by entropy coding methods. Intra coding works well for smooth images since it performs a uniform prediction for the entire luma ( $Y$ ) component of a macroblock.

For reconstruction of the macroblock, inverse transformation is applied to the quantized coefficients decoded by the entropy decoder, which are further used for motion compensation prediction. The macroblock is stored in memory for future reference predictions after de-blocking filtering. H.264 Intra has been used as the codec for key frames in most DVC systems.

### 3.3 Review of WMSN Security Mechanisms based on Digital Watermarking

There exists numerous works regarding video watermarking on a variety of codecs and platforms but those techniques cannot be applied to the sensor network domain due to their major architectural differences. In literature, digital watermarking has been applied mostly to address copyright protection of digital content, digital fingerprinting, tamper detection, broadcast monitoring and meta-data insertion [127]. Few applications of this technique have been found in wireless multimedia, video or even scalar data sensor networks.



**Figure 3.13** - Conceptual framework for secure communication in WMSNs [7]

In the context of WMSNs, security issues such as privacy, trust management and authentication have a high degree of correlation, and digital watermarking alone is not sufficient to address all of them [60]. Thus, an eight-stage conceptual framework that provides guidelines for secure communication in WMSN based on watermarking is presented in [7], as shown in Fig 3.13. Brief description of key stages is as follows:

- **Application Scenario Extraction:** to identify the application-specific QoS requirements such as timeliness or reliability. For example, surveillance systems deployed in a battle field, traffic or hospital monitoring environments have different types of risks, concerns and resources. Hence, the application settings in terms of network size, node density, topology, software and hardware resources must be known in order to design the security mechanism for it.

- **Privacy Protection:** This stage deals with providing privacy protection to multimedia content. The conceptual framework proposed that privacy protection mechanism should be incorporated into the design of watermarking.

**Table 3.3 - Summary of reviewed video watermarking techniques**

Reference	Classification Parameters			
	Implementation Domain	Detection	Perceptibility	Application
Wang et al.(2010)	Transform	Non Blind	Invisible	Robust
Al-Otum et al.(2010)	Transform	Blind	Invisible	Robust
Ning Zhu et al.(2008)	Transform	Blind/Non Blind	Invisible	Robust
Ju and Jonathan (2008)	Transform	Blind	Invisible	Robust

- **Authentication:** This stage develops a mechanism to identify and authenticate the originator of multimedia content. The framework proposed the use of digital watermarking to ensure correctness and confidentiality of multimedia content. Watermarking based authentication is proposed for WMSNs in [51].
- **Trust Management:** This stage deals with the establishment of trust management to enhance the security and reliability of nodes in WMSN. Since the network is deployed in an uncontrolled environment, there is a high probability that a node may get compromised.
- **Watermarking Technique for WMSNs:** This stage comprises of three sub-stages, namely watermark generation, embedding, and detection. Linear feedback shift register (LFBSR) [128] is used to generate the watermark bit sequence that is further transformed using some constraints under Kolmogorov complexity rule [129] to enhance watermark security. For embedding purpose, multi-modal fusion of watermark sequence bits and multimedia data (captured from the surrounding) is performed at sensor node. Afterwards watermarked solution is obtained by solving the non-linear system equation formed using atomic trileteration process on multi-modal data. Finally, the watermarked solution is transmitted to receiver via wireless communication. Blind watermark detection is then performed at receiver to verify the watermark's existence.

- **Theoretical Analysis and Simulation:** To determine compatibility, benefits, and drawbacks of the proposed architecture in the context of WMSNs.

Overall, the theoretical framework discusses security issues spanning privacy and authentication to trust management in WMSNs. In summary, this section provides an in-depth review of the most relevant video watermarking techniques for WMSNs in literature, which is also summarized as shown in Table 3.3, based on the four classification parameters discussed in Chapter 2.

### **3.3.1 Wavelet based Resource-Aware, Adaptive Watermarking for WMSN**

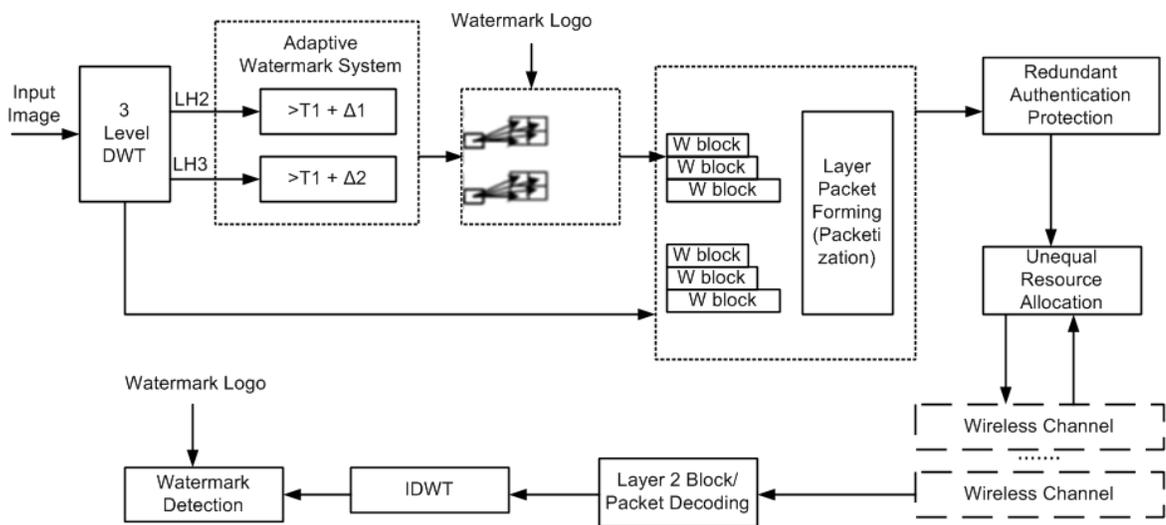
Wang et al., [130] presented a communication resource-aware, adaptive watermarking scheme for multimedia authentication in WMSNs (Fig. 3.14). The primary challenges addressed by the scheme were to embed/protect/extract watermark efficiently in low-cost sensors, and to transmit authenticated multimedia in an energy-efficient manner. The transmission quality of the watermark is maintained by embedding the watermark with adaptive coding redundancies, and allocating network resources adaptively to protect the multimedia packets containing watermark information. Since the watermark is adaptive to network conditions and the processing delay is also reduced by exploiting inter-frame correlation, the proposed scheme is shown to achieve reasonable communication energy efficiency and real time performance.

When watermarked multimedia content is transmitted over WMSN, the multimedia quality could be degraded due to packet loss caused by channel error, and in the worst case makes the watermark undetectable at the receiver site. Therefore, a high-quality and efficient watermarking system in WMSN should be robust against transmission errors along with efficient resource utilization. The scheme embeds the watermark to selective coefficients of the three-level Discrete Wavelet Transform (DWT) middle frequency bands of an image frame, based on the network conditions.

The embedding algorithm in [130] exploits a qualified significant wavelet tree (QSWT) [131] to separate the frequency bands of the multimedia and embed the watermark into the frame's middle frequency bands. The reason behind using the middle frequency bands is that after wavelet transform, low frequency bands contain the most energy of the image frame, while high frequency bands contain noise and the information in those bands is often

lost during compression. The watermark is embedded into host multimedia at chosen locations of LH3/HL3 (middle frequency bands) coefficients.

Two thresholds  $t_1$  and  $t_2$  are selected adaptively based on network conditions and parameters such as packet loss ratio to identify the optimal number of QSWT trees to embed the watermark. Although embedding in a lesser number of QSWT trees maintains the invisibility of the scheme, it would be difficult to detect the watermark in a case of packet losses. On the other hand, embedding watermark redundantly makes use of large number QSWT trees which leads to greater distortion in the multimedia content as well as higher energy consumption at the sensor node.



**Figure 3.14-** Resource-aware adaptive watermarking system for WMSN [130]

The multimedia packets are classified as either WM packets (with watermark information) or NWM packet (without watermark information). The WM packets are protected more by allocating extra network resource to improve watermark transmission quality to facilitate watermark detection and authentication. The network resource is referred to as the average total energy required, computed as a function of the desirable bit error rate (BER), frame size, packet transmission rate, transmission power and retransmission limit. The primary goal of the scheme is to improve the watermark authentication while at the same time maintaining received multimedia quality.

The watermarking algorithm of the proposed adaptive wavelet based scheme is as follow:

1. Apply three-level DWT on host multimedia frame to obtain wavelet sub-bands (LH2 and LH3).

2. Decompose LH2 and LH3 into sub blocks of size  $M$ .
3. Compute mean of sub-blocks in LH3 and LH2, and store them in array  $T_1$  and  $T_2$  respectively.
4. Find the optimal number of QSWT trees for every sub-block. For each sub-block  $m$ , compare coefficient location at LH3 ( $i,j,m$ ) with threshold  $T_1(m) + \Delta_1$ , where  $\Delta_1$  is selected adaptively based on network conditions.
  - a. If LH3 ( $i,j,m$ ) is greater than threshold  $T_1(m) + \Delta_1$ , then look for at least three child coefficients that must be greater than threshold  $T_2(m) + \Delta_2$ , where  $\Delta_2$  is also an adaptive parameter.
    - i. If at least three child coefficients are greater than  $T_2(m) + \Delta_2$ , then LH3( $i,j$ ) is selected as one of the QSWT tree for sub-block  $m$ .
    - ii. Sum the coefficient values from parent to all its children.
  - b. After finding sufficient number of QSWT trees for each block, sort them in the descending order.
5. Watermark embedding decision in current frame is based on QSWT location. If the difference between LH3 coefficients of consecutive frames is beyond some predetermined limit, the watermark will not embedded in current frame.
6. Otherwise embed watermark in each QSWT for each sub-block  $m$ .

The simulation results show that the proposed wavelet based watermarking process achieves a good performance in terms of invisibility and can resist JPEG lossy compression provided that thresholds  $T_1$  and  $T_2$  are chosen carefully. However, the scheme is designed for images, and thus cannot be applied directly to watermark video content, which has much greater requirements in terms of frame rate, inherent redundancy in consecutive frames and complexity than the image watermarking.

### 3.3.2 Image Watermarking Technique Based on Wavelet-Tree

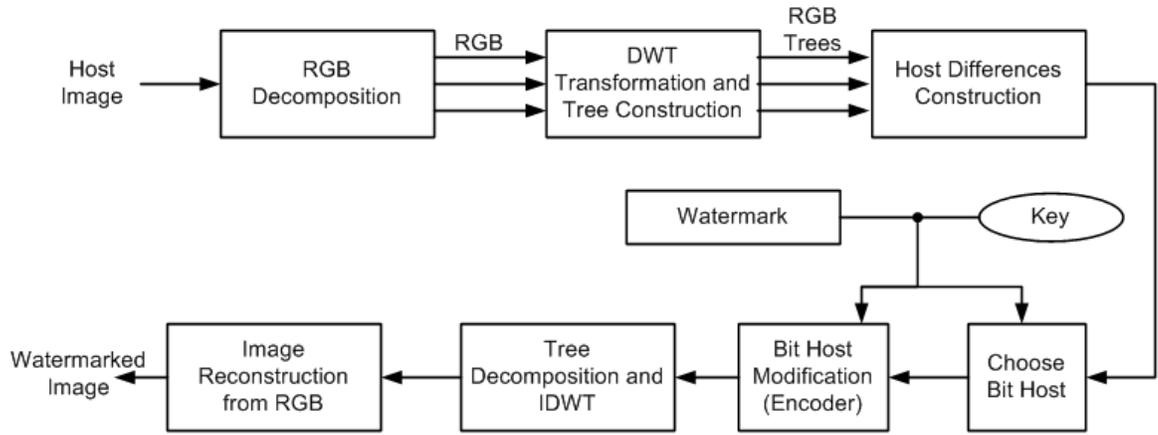
A robust, blind wavelet-tree based image watermarking technique is introduced in [132] that deals with the colour pixels as a unit and exploits pixel components features in wavelet domain for watermark embedding. In colour images, each pixel is made up of 24-bits with

8-bits allocated to each R, G, and B component. The wavelet-trees are generated for each colour component; two trees each from different components are used to embed one watermarking bit. The embedding process modifies the difference between the two selected trees such that it carries the bit sign with sufficient energy to ensure robustness of embedding. The proposed method simplifies the watermark detection process by extracting the sign of difference chosen to represent the embedded bit.

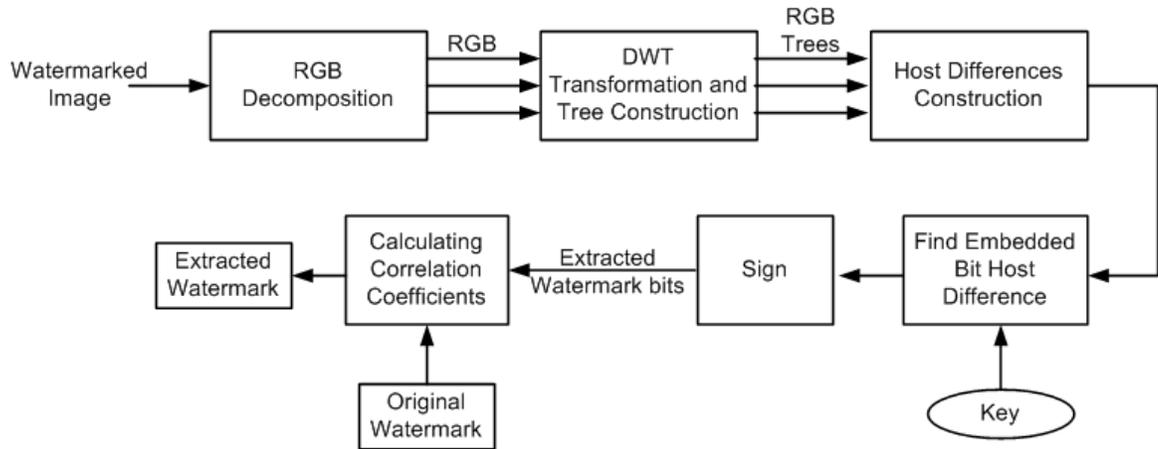
For RGB images, each colour is represented by a unique combination of the RGB values. Any change in these values will eventually change the colour itself. As image transformed into wavelet domain, the RGB combination for each pixel is preserved by maintaining the unique relationship between the wavelet coefficient and the corresponding pixel value. The proposed scheme in [132] uses this uniqueness to construct a robust watermarking technique. The embedding algorithm identifies all available locations that can carry watermark bits, then selects locations for which the introduced modification is as small as possible. It is also noted that each bit may have more than one embedding locations that facilitate the user controllability and scheme's capacity. Consequently, the scheme is claimed to be resistive against several attacks.

The block diagram for watermark embedding and extraction is shown in Fig. 3.15. As illustrated in Fig. 3.15(a), the host RGB image is decomposed into R,G and B components; DWT transformation is applied to each of the components separately that forms their respective wavelet trees. Based on dominant relations, these trees have been searched to identify robust embedding (bit host) locations. Consequently, encoder used these locations to construct robust host differences to embed watermark binary bit sequence. To ensure robustness and minimal distortion to host image, watermark bits are inserted into middle frequency band's wavelet trees. Finally, to restore the RGB watermarked image, wavelet-trees are decomposed and subjected to inverse DWT (IDWT).

Similar to the embedding, the blind watermark extraction algorithm in Fig. 3.15(b) performs RGB decomposition, DWT transformation and wavelet-trees construction. Next, the embedded bit host difference is determined and used for computing the sign of the difference and the watermark bit sequence. Finally, to quantify the extracted watermark, the correlation coefficient is computed between the original and extracted watermarks.



(a)



(b)

**Figure 3.15** – Block diagram for (a) Watermark embedding; (b) Watermark extraction

[132]

Similar to the embedding, the blind watermark extraction algorithm in Fig. 3.15(b) performs RGB decomposition, DWT transformation and wavelet-trees construction. Next, the embedded bit host difference is determined and used for computing the sign of the difference and the watermark bit sequence. Finally, to quantify the extracted watermark, the correlation coefficient is computed between the original and extracted watermarks.

Unlike existing DWT based image watermarking techniques that consider RGB image as three independent layers, the proposed algorithm assumes the colour pixels as a single unit and exploits the significant features and relations between RGB colour components. Most importantly, the watermarking algorithm is based on constructing the so called wavelet-

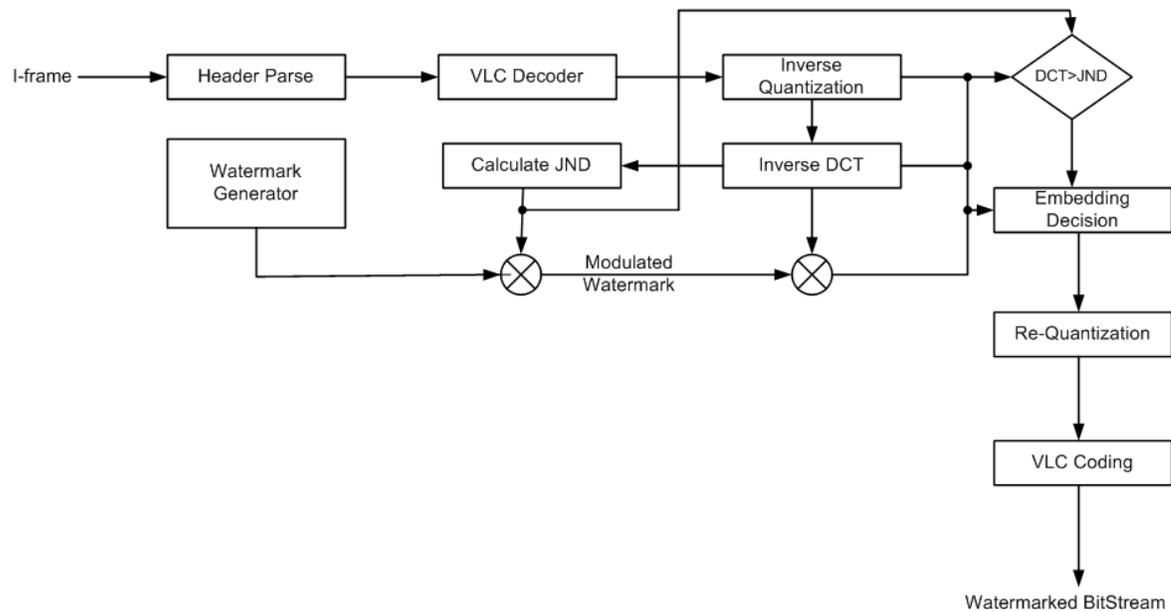
trees (WTs), which correspond to various frequency bands of the same spatial location. Experimental results have shown that the proposed scheme is extremely imperceptible and could attain reasonably good robustness against various common signal processing attacks as well as JPEG compression.

### **3.3.3 Video Watermarking Technique Against Correlation Attack Analysis in WSN**

A blind video watermarking scheme for WMSN using F-modulation method is presented by Ju [25, 48] to deal specifically with content analysis based correlation attacks. The proposed design focused on a cross-layer methodology for addressing the authentication issues for video transmission in WMSN (Fig. 3.16). Traditional watermarking schemes are designed to be resilient to various image processing attacks which attempt to remove or weaken the embedded watermark (i.e., watermark robustness against unintended signal processing is desired). In this scheme, this property is no longer a priority since a video frame without the expected watermark will be considered as a counterfeit. The following problem scenario is used as a case study: A wireless video surveillance system deployed to monitor an area of high security, and a malicious user or attacker accessed the radio communication channel and tried to invisibly destroy/manipulate the original video. An attack is made on the integrity of video data by suppressing the authentic transmission and intercepting the video for intended receiver using highly directional and powerful antennas.

An invisible, low-complexity signature-based authentication mechanism is proposed to thwart these attacks, in which any receiver can validate the integrity of video data, provided that the signatures have been known previously. A test-bed is set up along with a device called VCUmote that comprises of a CMOS camera and two ZigBee transceivers. A blind watermarking embedding process, available in both simple and enhanced versions, is presented. The embedding process replaces the LSB of the selected DCT coefficient with the watermark bits, making the scheme invisible, while the receiver copies the LSB from predefined fixed locations from the frames of received video. The watermark is then embedded at the predefined locations, making the verification process easier. However, it is noted that such approach does not work well against collusion, playback and middleman attacks.

- *Collusion Attack*: refers to attacks in which the attacker uses a statistical method to identify potential watermark locations. Such an attack requires the attacker to have access to a large number of watermarked video frames. For example, the attacker has multiple watermarked video frames where the same watermark is embedded in the same locations. Then for each watermarked video frame, the attacker derives the LSB-plane where the LSB of each DCT coefficient is extracted and forms an image of the same resolution as the original one. The attacker next calculates the average of all LSB-planes to form a grayscale image. After processing a sufficient number of video frames, the colour of watermarked locations becomes either strong white or black, while other pixels approach a grayscale intensity level of 0.5. Therefore, to counter collusion attack, the watermarking scheme must involve non-deterministic watermark.
- *Playback Attacks*: refers to attacks in which the attacker playbacks or uses an old (expired) video frame to fool the system.
- *Middleman Attacks*: refers to attacks in which the attacker exploits certain loopholes in the verification process to make fraudulent video frames.



**Figure 3.16** - Block diagram for embedding watermark into intra-frame sequence [48]

Improvements are made by embedding into each frame a distinct watermark and using a characteristic bit derived from DCT coefficients to modulate the watermark bit. The watermark is considered a time-variant pseudorandom variable that generates a distinct

signature for every frame. Two DCT coefficients are selected to embed each watermark bit. The fixed predefined embedding locations for watermark are vulnerable for correlation and playback attacks. Therefore, the scheme proposed dynamic and random selection of DCT blocks for embedding watermark so that the location of watermark varies over time from one frame to another. Secondly, the communication protocol needs to integrate the intrusion detection mechanism for the watermark security to prevent the channel capturing attacks over a period of time. Additionally, this scheme embeds some information as noise by varying DCT coefficients values to act as a safeguard against content and correlation analysis.

Ju [64] also suggested secure implementation of block selection algorithm at encoder/embedder and decoder/detector. The location generator algorithm identifies the same embedding location at both source and receiver sites by providing identical frame numbers. However, the reliability of the algorithm solely depends upon the synchronization of the frame number between embedder and detector. The watermark detection process comprises of following steps:

- Locate the character bit from DCT coefficients of the  $8 \times 8$  DCT block.
- Extract the watermark bits based on the value of character bit.
- Calculate the detection statistic metrics.
- Compare with the threshold value to decide the authenticity of the frame.

The watermarking scheme discussed above has the following security properties:

1. *Resistant against Correlation Attack:* A distinct watermark is generated for every video frame on the basis of pseudorandom sequence, which makes it difficult for the attacker to judge the watermark location by averaging the LSB plane values.
2. *Resistant against Playback Attack:* The playback attack can be detected by maintaining the record of the frames numbers received at the receiver site. A playback attack may be in-progress if the receiver notices a duplicate frame number. Request for resetting the frame sequence number from receiver to camera sensor node may help to overcome the playback attack.
3. *Resistant against Middleman Attack:* Since the location of the modulation bit is unknown to the attacker, it would not be able to produce the same watermark embedding process by modulating watermark bit. Even if the attacker copies the entire

least significant bits to the injected video frame, the likelihood of passing the watermark verification is negligible because the watermark bits are modulated using characteristic bit.

Correlation analysis is performed between the F-modulation bit and DCT coefficients, with and without block selection algorithm at embedder and detector. The results show that with dynamic block selection, the correlation does not have much variation, which means it is almost impossible to find the exact correlation among the watermark and F-modulation bits. While the proposed watermarking scheme is shown to be robust against correlation analysis and playback attacks, it does not address the energy consumption issue pertinent to WMSNs, as the scheme is built on MPEG-2, a conventional video codec with high complexity encoder and simple decoder architecture. Moreover, the evaluation only demonstrates the resilience of the proposed scheme against correlation and playback attacks. It does not provide any idea about the amount of distortion/noise introduced to the original video due to watermark embedding.

### **3.3.4 Watermarking Technique based on Distributed Video Coding**

Ning et al. [73] proposed the first watermarking scheme for a DVC based Wyner-Ziv video codec. The distributed source coding theorems was established in 1970s by Slepian and Wolf [21] for distributed lossless coding and by Wyner-Ziv [22] for lossy coding with decoder side information. These information-theoretic results revealed that it is possible to encode the frames of a video sequence independently and still achieve efficient compression as long as decoding is performed jointly. This idea formed the basis of a simple video encoder at the cost of increased complexity at decoder. Incoming video frames at the encoder are split into *key* frames, which are compressed using conventional H.264 Intra encoder, and *non-key* frames, which are compressed using a Wyner-Ziv encoder. A configurable setting GOP (Group of Pictures) determines the distance between key frames in the video sequence, and thus the coding efficiency. For example, if  $GOP = m$ , there will be one key frame for every  $m-1$  non-key frames (also referred to as WZ frames). At the decoder side, key frames are decompressed using H.264 Intra decoder, from which side information is generated for decoding of the WZ frames.

To enhance the security and invisibility of the watermarking scheme, the Arnold's image transformation [133] is applied on the watermark image. Arnold's transformation stretches pixels of an  $(n \times n)$  image and wraps the stretched portions in the order to restore the original dimensions. After a certain number of iterations, an image will transform back to the original. Following the transformation, a combined corner and edge detection algorithm [56] is applied to the video frames to identify the points of interest (potential regions to invisibly embed watermark image bits). It is possible that the points of interest identified by the key frames and Wyner-Ziv frames do not exactly match with each other. Therefore, the resulting redundancy of the embedded watermark information leads to robustness of the watermarking scheme.

The watermark embedding process is as follows:

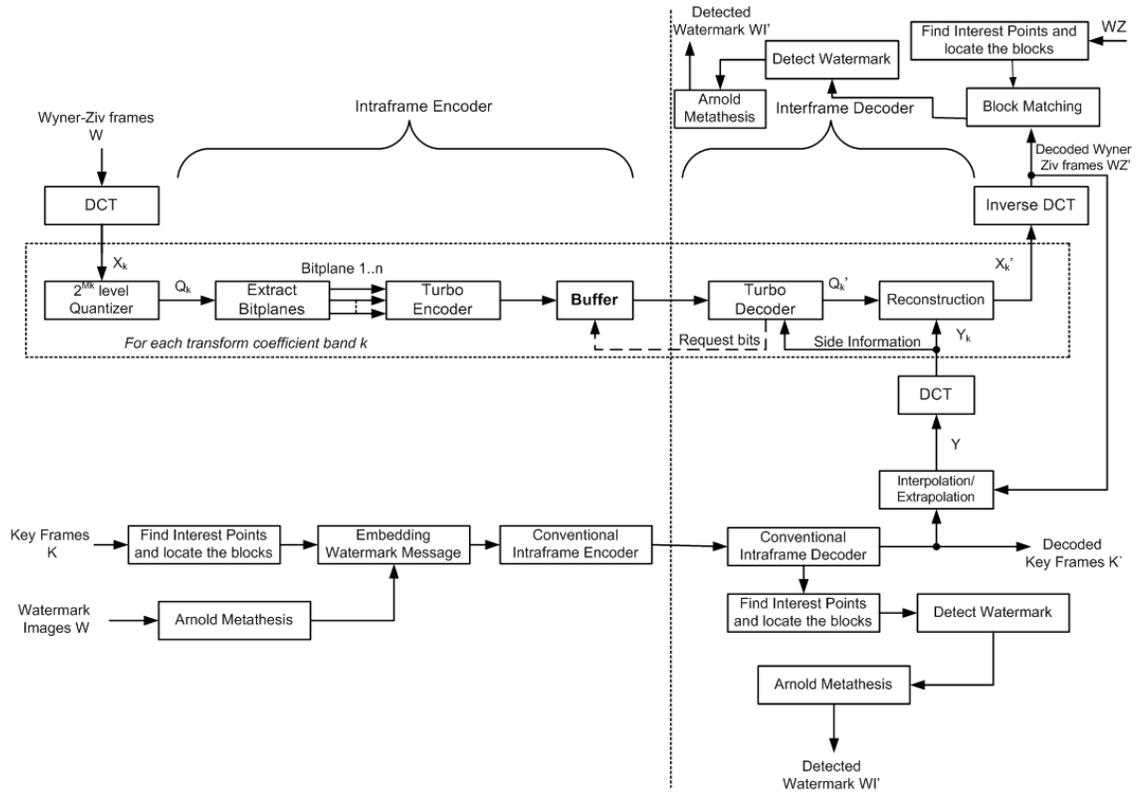
1. Perform Arnold's transformation on watermark image and encrypt it with private key  $K1$ .
2. Identify the points of interest using Harris corner and edge detector in key frames.
3. Locate a block around each interest point to embed whole watermark image
4. For every 8-bit pixel in a block, replace the bit plane with one bit pixel's value in the watermark image matrix

As illustrated in Fig. 3.17, the same watermark can be detected at the decoder from both key frames (KF') and Wyner-Ziv (WZ') frames, with the difference that the detection from KF' is blind, while that from WZ' is non-blind (requires original video frames). Detection from WZ frames requires original frames at decoder because the difference between original WZ frame and reconstructed WZ' frame will enable us to detect the watermark information.

The watermark detection process from key frames (blind) is as follows:

1. Identify the points of interest from the given frame.
2. Locate the blocks around each point of interest, and find the bits in the pixels that were replaced with watermark bits. One complete block around the point of interest constructs the watermark image matrix.
3. Average values calculated from all block to obtain the watermark image matrix.

4. Perform Arnold's transformation on reconstructed watermark to obtain the real watermark.



**Figure 3.17** - Video watermarking scheme based on Wyner-Ziv codec [73]

The watermark detection process from Wyner-Ziv frames (non-blind) is as follows:

1. Identify the points of interest from the given frame.
2. Locate the blocks around each point of interest, and find the bits in the pixels that were replaced with watermark bits.
3. Locate the bits corresponding to pixels of the same blocks in original Wyner-Ziv frame;

$$\text{if } (\text{Reconstructed } WZ' + CV < \text{Original } WZ) \text{ then } WM = 0$$

$$\text{else if } (\text{Reconstructed } WZ' > \text{Original } WZ + CV) \text{ then } WM = 1$$

*else Detect from Key frame*

Where CV is a criterion measure chosen for measuring the difference between original and reconstructed Wyner-Ziv frame,  $0 < CV < 2^{\text{BitPlanes}}$ .

4. Average values calculated from all block to obtain the watermark image matrix.

5. Perform Arnold's transformation on reconstructed watermark to obtain real watermark.

This DVC based watermarking scheme is distinctive since it is the only video watermarking scheme based on Wyner-Ziv video codec, which suits the WSN environment. The simulation results showed that the scheme works well in terms of invisibility, correctness and robustness. However, it is still not ready for application in WMSN because at the receiver site, the watermark extraction from Wyner-Ziv frame is non-blind and thus requires the original Wyner-Ziv frame, which is not possible. Although the watermark can be extracted from key frames in a blind manner, this alone will not be sufficient to authenticate the video. Therefore, the watermarking scheme needs further enhancements to make it operational for WMSN environment.

### **3.3.5 Comparison and Analysis of Watermarking Techniques for WMSNs**

A comprehensive review of the state-of-the-art image and video watermarking techniques for WMSNs has been presented from Section 3.3.1 to Section 3.3.4. Each technique is shown to have its pros and cons, and opens research issues for this relatively unexplored area.

Wang et al. [130] presented a communication resource-aware, adaptive image watermarking scheme for multimedia authentication in WMSNs. Although the scheme addresses efficient watermark embedding and extraction using low-cost sensors, and transmits authenticated multimedia in an energy-efficient manner but it does however require the presence of original image (non-watermarked) at the receiver site in order to extract the watermark from the received (watermarked) frame, which seems to be an unlikely scenario for WMSN applications. The scheme exhibits good performance in terms of invisibility and can resist JPEG lossy compression provided that thresholds  $T_1$  and  $T_2$  are chosen carefully so that they enable redundant embedding of watermark. However, the scheme is designed for images, and thus cannot be applied directly to watermark video content, which has far greater requirements in terms of frame rate, inherent redundancy in consecutive frames and complexity than image watermarking techniques.

Al-Otum et al. [132] presented a blind image watermarking technique based wavelet-trees that exploits the features and relations among the RGB pixel components in the wavelet domain. The watermark is embedded by spreading it through the host image such that the inter-pixel robust relations carry the watermark bit sign with sufficient energy. Based on the

presented results, it can be seen that the proposed technique is considerably imperceptible, robust and able to carry sufficient number of watermark bits. However, it has not taken into account the design constraints of WMSN environment. Therefore, the complexity as well as the energy requirements of the embedding and extraction algorithms are still unexplored.

Ju et al. [25, 48] proposed a blind video watermarking scheme for WMSN using F-modulation method to deal specifically with content analysis based correlation attacks. The proposed watermarking scheme is shown to be robust against correlation analysis and playback attacks; it does not address the energy consumption issue pertinent to WMSNs, since MPEG-2, a conventional video codec with high complexity encoder and simple decoder architecture is used at source site for watermarking of frames. Based on our literature review, we can summarise the two primary design requirements of watermarking schemes for WMSNs as follows:

1. *Energy Efficiency*: Practically, a network of wireless video sensors inherently suffers from constraints such as *limited power/energy supply and computational capabilities* which lead to the requirement of careful design of the watermarking scheme with appropriate selection of compression and signal processing algorithms (preferably lightweight) that efficiently utilize the power not only during video processing but also during their transmission.
2. *Robustness: Data loss or corruption* inherent in error-prone wireless and open operating environment raises the need for more robust watermarking schemes that should survive packets loss or corruption pertinent to the wireless environment and attacks by malicious nodes.

Ning's video watermarking scheme [73] embeds the watermark into least significant bits (LSB) of selected discrete cosine transform (DCT) coefficients of each key frame. Harris corner detector processes each key frame to evaluate the number of interest points which by itself is a complex operation that may cause additional delay in real-time applications. Moreover, the capacity of embedding is dependent on the type of frame itself, while embedding in DCT coefficient increases the bitrate requirement and may cause a drift error in WZ frame reconstruction. In addition, due to watermark embedding inside the selected interest points of the key frame, the redundant watermarks (frame dependent) are possibly

confined only to a specific region (edges/corners) and thus are unable to determine frame tampering within the rest of the frame.

### **3.4 Chapter Summary**

This chapter reviewed and synthesized the promising DVC architectures accompanied by the enhancements made to them in recent years. In addition, it also highlights the significance of DVC in the evolving WSN application domain. Enhancing the target reconstruction quality, and enabling flexible complexity distribution between encoder and decoder, and multi-view DVC coding are still open research issues. Future research directions for DVC may include enhanced side-information generation, rate-control, correlation noise modelling, as well as the design of novel and efficient channel codes. Moreover, a comprehensive review on the state-of-the-art video watermarking techniques for WMSNs has been presented. Energy efficiency and robustness has been identified as primary design requirements of watermarking schemes for WMSNs which elevates the need of novel watermarking techniques based on DVC architecture.

In the next chapter, we have performed a comparative analysis among the video codecs for multihop WMSN. We have selected one codec, each from the DVC and DCVS coding architectures (covered in Section 3.2) and analyze the performance against conventional H.264/AVC Intra codec.

# Chapter 4

## Analysis of Video Codecs

### 4.1 Introduction

In this chapter, we evaluated and analysed the performance of video codecs based on emerging video coding paradigms covered in Chapter 3, namely, distributed video coding (DVC) and distributed compressive video sensing (DCVS) for multihop WMSNs. The main objective of this work is to provide an insight about the computational (encoding/decoding) complexity, energy consumption, node and network lifetime, and the quality of reconstruction of these video codecs. Based on the findings, this chapter also provides some guidelines for the selection of the appropriate video codec for the proposed watermarking scheme as well as various WMSN applications. We investigate the performance of four video codecs on resource-limited camera sensor nodes. Since energy is one of the most critical resources of a sensor node, one significant aspect of our work is the evaluation of the energy consumption by a sensor node due to performing video encoding and transmission of encoded video data to a sink in a multihop scenario. In addition, to investigate potential trade-off between energy efficiency and video quality, we also determined the quality of reconstructed video frames in terms of their peak-signal-to-noise ratio (PSNR) by the respective codecs. We analysed the encoding and decoding computational complexity of the respective codecs at the sensor nodes,

and sink, respectively. Based on the analysis of the above findings, some guidelines are presented for selecting the most appropriate video encoder for a given WMSN application.

The rest of the chapter is organised as follows: Section 4.2 briefly discusses architectural differences between DVC, DCVS, and H.264/AVC. Section 4.3 describes the experimental setup and parameters used to analyze the performance. Section 4.4 presents and discusses the encoder and decoder performance of the respective codecs. Finally, Section 4.5 provides a comparative analysis of all three codecs, and our findings are summarised in Section 4.6.

## **4.2 Comparison of DVC, DCVS, and H.264/AVC Coding Architectures**

In Chapter 3, we discussed DVC and DCVS coding architectures along with H.264/AVC (a candidate from conventional coding). In addition, we also examined several frameworks proposed during recent years based on these architectures. A brief comparison of DVC, DCVS and H.264/AVC is presented below.

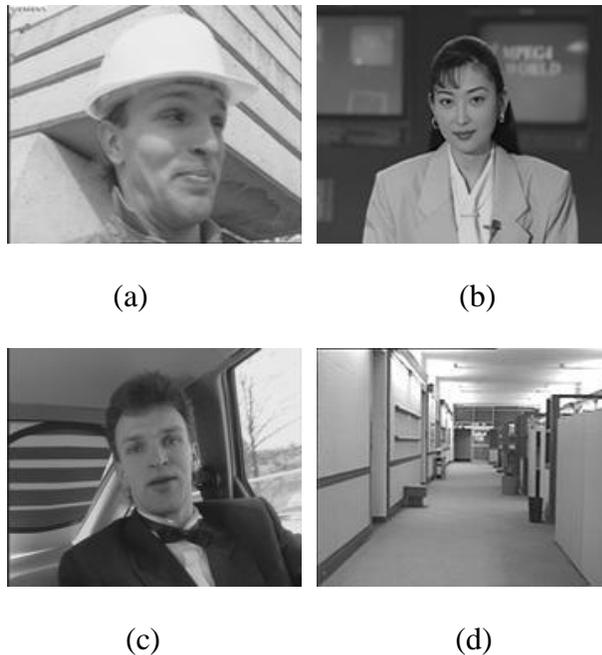
H.264/AVC is generally designed for broadcast applications, where video content is broadcasted to several users, and targets on optimizing the compression performance. It exploits source redundancy (spatial/temporal) at the encoder through predictive video coding. The H.264/AVC encoder generates a prediction of the source video frames and then encodes the residue between the source and its prediction. A motion-compensated prediction operation at the encoder is a key algorithm to obtain high compression. It effectively removes the temporal correlation between successive frames in a sequence, but it is a computationally demanding methodology.

In contrast, DVC architecture exploits source redundancies at the decoder side, leading to separate encoding and joint decoding. In particular, a prediction of the source, called side-information (SI), is generated at the decoder by using the previously decoded frames. By utilizing the statistical dependency among the source frames and the side information, compression can be achieved by transmitting parity (syndrome bits) of a channel code. These parity bits are used to decode the source frames using side information. Hence, computationally expensive tasks such as motion estimation, can be relocated to the decoder; allows a flexible distribution of the computational complexity between the encoder and the decoder, thus enables the design of lightweight encoding architectures.

On the other hand, compressed sensing (CS) is basically a completely new approach to data acquisition which can be applied not only to DVC [96, 122, 134] but also to conventional coding architectures [135-136] due to the sparse nature of video signals. DCVS coding architectures get the benefit of the CS based data acquisition in conjunction with DVC based simple encoding and complex decoding framework. The simplicity of the DCVS encoding process is traded off by a more complex, iterative decoding process. At the decoder, the reconstruction process of CS is expressed as an Optimisation problem which potentially allows the system to adapt the objective function and constraints to the given application. Meanwhile, a lower sampling rate implies less energy required for data processing, leading to lower power requirements for the source coding site.

### 4.3 Experimental Setup

We used the existing implementations of DISCOVER [115], Wyner-Ziv video codec [137] and DVCS [138] with permission from their authors. The DISCOVER comes in an executable format; while Wyner-Ziv and DVCS source codes are provided to us by the authors.



**Figure 4.1** - Test video sequences: (a) Foreman; (b) Akiyo; (c) Carphone; and (d) Hallway

For H.264 Intra codec, we used the H.264/AVC reference implementation (JM 17.2) [139]. The quantisation index used for the DISCOVER and Wyner-Ziv video codec is 4 (from an available

range of 0–8), and while H.264 Intra index set to 28 (from an available range of 1–51). The reason for selecting these values is to compress or map the huge data set resulting from transformation in both codecs, to approximately the same number of quantum values, thereby maintaining a similar quantization level for a consistent comparison among the codecs. On the other hand, DCVS does not use quantization, but it differentiates between key and non-key frames based on their measurement sampling rates. For a reasonable trade-off between video transmission rate and quality of reconstruction, we used measurement sampling rates of 0.5, and 0.2, for key and non-key frames, respectively, same as in [138].

The experiments were performed for the first 150 frames of the Foreman, Hallway, Akiyo, and Carphone video sequences (Fig. 4.1). All the tested sequences have QCIF (176 x 144) resolution using YUV format with a frame rate of 15fps. The sequences were selected to include several variations. For example, the Foreman sequence features sharp motion and low complexity content; the Carphone sequence features non-uniform changing background and significant motion in the foreground; the Hallway and Akiyo sequences contain sharp variation in the foreground but little motion in the background. Such diversity in the test sequences is necessary in order that the behaviour of the encoders in different scenarios can be evaluated.

To measure the computation complexity in terms of number of CPU cycles required by each encoder, we used the Odyssey prediction model [140] to estimate the percentage CPU availability for the encoder. To translate it further into computation energy consumption, and to calculate the energy consumed by a direct transmission of encoded frames from an encoder to a sink/decoder, the energy consumption model based on TelosB mote specifications [11] is used (Table 1). The computation and transmission energy is calculated by multiplying the total CPU cycles with energy consumed per CPU cycle, and total bits transmitted with energy consumed per bit transmission, respectively.

**Table 4.1-** TelosB mote energy consumption model

Per CPU Cycle	$1.215 \times 10^{-9}$ J
Per Bit Transmission	$1.094 \times 10^{-6}$ J

For sensor networks, the transmission energy consumed by a sensor node can be expressed as [141]:

$$E_{tran} = P_{sent} \times P_{length} \times T_B \times I_t \times V \quad (4.1)$$

where  $P_{sent}$  is the total number of packets sent,  $P_{length}$  is the length of each packet in bytes including headers,  $I_t$  is the current drawn by the sensor node in transmit mode,  $T_B$  is the time for sending one byte, and  $V$  is the voltage supply to circuitry. Similarly, the energy consumed for receiving packets over a radio link is referred to as receiving energy, which can be expressed as:

$$E_{recv} = P_{recv} \times P_{length} \times T_B \times I_r \times V \quad (4.2)$$

where  $P_{recv}$  and  $I_r$  are the total number of packets received, and current drawn in receive mode, respectively, with the rest of parameters defined as above for the transmission energy. We adopt the TelosB specifications [11] to determine the values of the above parameters.

Furthermore, we compute the energy consumption due to relaying between the sensor nodes in a multihop scenario with the assumption of a homogeneous sensor network in which nodes are uniformly distributed. We also assume that only sensors have energy constraints, while the final receiver/sink is energy independent. Thus, the overall communication energy consumption  $E$  for a single source-sink pair separated by  $n$ -hops, can be expressed as [142]:

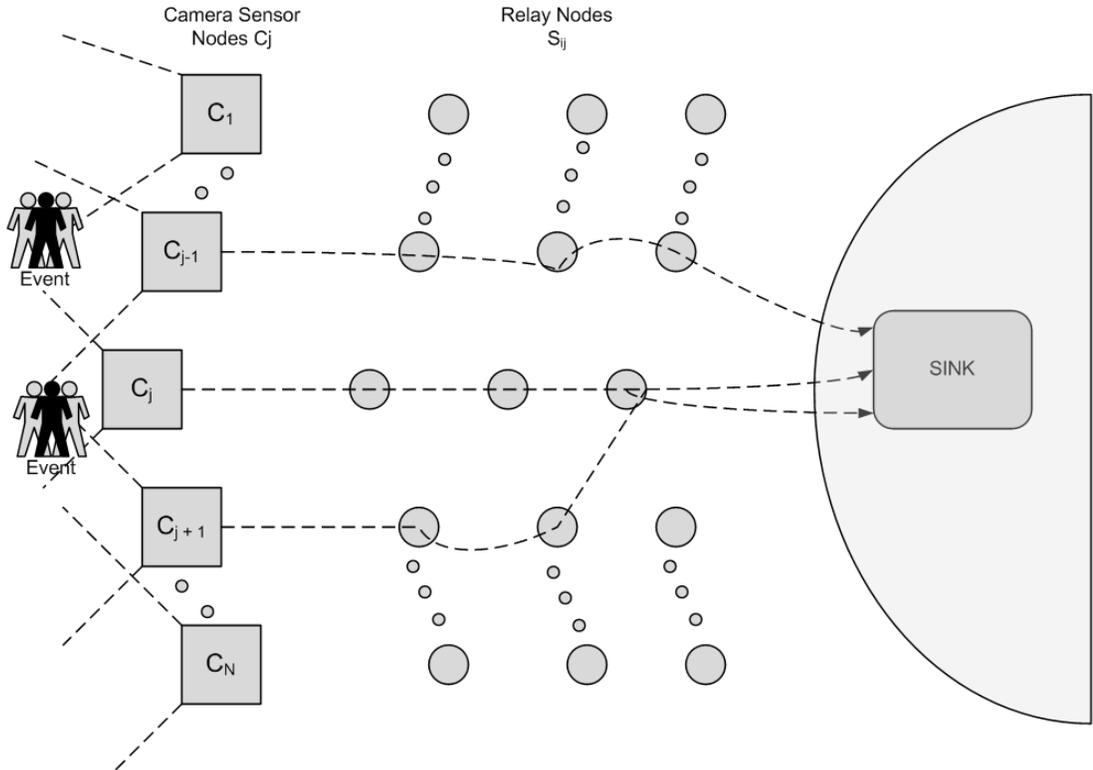
$$E(n) = nE_{tran} + (n - 1)E_{recv} \quad (4.3)$$

Transmission over a wireless channel may cause losses and errors due to reasons such as channel impairments, fluctuations in link quality, antenna characteristics, etc. Empirical studies have been conducted to understand and address channel characteristics in various communication models [143-144]. In this work, we adapted the channel and packet loss models as described in [145] for a multihop WSN consisting of camera sensors and relay nodes as shown in Fig. 4.2. The camera sensors are used to capture video images of events that occur in their surroundings and transmit the captured data to a sink via relay nodes with simple store and forward mechanism.

The radio channel model used is a log-normal shadowing model, where the signal-to-noise ratio  $\gamma$  is defined as a function of distance  $d$  between the nodes [146]:

$$\gamma(d) = P_{out} - PL(d_0) - 10c \log_{10} \left( \frac{d}{d_0} \right) + X_\sigma - P_c \quad (4.4)$$

where  $P_{out}$  is the transmitter output power,  $PL(d_0)$  is the path loss at the reference distance  $d_0$ ,  $c$  is the path loss exponent that generally ranges between 2 and 6 for free-space and shadowed



**Figure 4.2 - General WMSN scenario**

/obstructed indoor scenarios respectively,  $X_\sigma$  is the zero mean Gaussian variable expressed in dB with variance  $\sigma^2$  (also called shadowing variance) and  $P_c$  is the noise floor.

The probability of error  $\rho$  of a transmitted packet over a physical channel is:

$$\rho = 1 - (1 - \rho_e)^{8f} \quad (4.5)$$

where  $\rho_e$  represents the bit error probability and  $f$  is the size of frame in bytes. Assuming non-return to zero<sup>1</sup> (NRZ) and non-coherent frequency shift keying<sup>2</sup> (NCFSK) are employed as the encoding, and modulation schemes, respectively, which is the case in several practical implementations of WSN based systems [147],  $\rho_e$  can be expressed as:

$$\rho_e = \frac{1}{2} e^{-\frac{\alpha}{2}} \quad (4.6)$$

where  $\alpha$  is the energy per bit to noise power density ratio:

$$\alpha = \frac{E_b}{N_0} \quad (4.7)$$

The signal-to-noise ratio is related to the energy per bit to noise power density ratio by the following expression, from which the value of  $\alpha$  can be obtained:

$$\gamma(d) = \frac{E_b}{N_0} \frac{R}{B_N} \quad (4.8)$$

where  $R$  is the data rate in bps, and  $B_N$  is the noise bandwidth in Hz. Substituting the  $\alpha$  obtained from Eqn. (4.8) into Eqn. (4.6), the packet error rate  $\rho$  in Eqn. (4.5) can be updated as:

$$\rho = 1 - \left(1 - \frac{1}{2} e^{-\frac{\gamma(d) B_N}{2 R}}\right)^{8f} \quad (4.9)$$

The packet loss probability for single path transmission over  $n$  hops is then given by:

$$P_n^{Loss} = [1 - (1 - \rho)^n] \quad (4.10)$$

Therefore, the total number of packet transmissions including retransmissions can be evaluated as:

$$N_{total} = N / (1 - P_n^{Loss}) \quad (4.11)$$

where  $N$  is the number of unique packet transmissions.

The parameter values assumed for evaluating packet loss model are specified in Table 4.2. The encoded bit-stream is encapsulated in MAC frames with 9 bytes of frame header for every 93

---

<sup>1</sup> NRZ encoding uses two voltage levels to represent a digital signal. A positive and negative voltage represents a binary one, and zero, respectively. NRZ encoding needs much less bandwidth than Manchester encoding for a given bit rate.

<sup>2</sup> NCFSK is a specialized non-coherent orthogonal modulation technique that has no phase relationship between consecutive elements of signal i.e. phase varies randomly.

bytes of payload [18, 148]. Therefore, the energy results present the sum of energy consumed by the source for encoding its video frames (computation energy), and the energy consumed for transmitting the encoded bit-stream to the sink (communication energy), including any retransmissions due to channel errors and packet losses.

For the video sink/decoder, we assumed that it is a resourceful machine without processing and energy constraints, as is often the case. Thus, we focused on evaluating the lifetime of the source nodes and relay nodes which eventually lead us to measure the lifetime of the entire network.

**Table 4.2** - Packet loss model parameters

Parameters	Value
Transmission power ( $P_{out}$ )	-1 dBm
Power decay ( $PL(d_o)$ )	55 dB
Path loss exponent ( $c$ )	4.7
Shadowing standard deviation ( $\sigma$ )	3.2 dB
Inter-node distance ( $d$ )	5 m
Number of hops ( $n$ )	1-6
Noise floor ( $P_c$ )	-105 dBm
Data rate ( $R$ )	19.2 kbps
Noise bandwidth ( $B_N$ )	30 kHz

We present the relationship between node and network lifetime with the imposed processing and communication loads emphasized by various video codecs. For the source node, we assumed a TelosB mote integrated with a camera board and battery pack of  $4 \times 1.5$  V AA alkaline batteries [49], each having a capacity of 2700 mAh [50]. On the other hand, the store-and-forward relay node is an ordinary TelosB mote operating with a battery pack of 2

× 1.5 V AA alkaline batteries of the same capacity. The estimation model for the lifetime  $T_{i\_source}$  of an individual source node is based on the work presented in [149] is as follows:

$$T_{i\_source} = \frac{E_b}{(e_{ij} \times q_{ij}) + (e_k \times q_r)} \quad (4.12)$$

where  $E_b$  represents the offered energy of the battery pack (in joules),  $e_{ij}$  is the energy to transmit one information unit from source node  $i$  to some neighbour node  $j$  (relay or sink),  $q_{ij}$  is the rate at which information is generated from  $i$  to  $j$ ,  $e_k$  is the energy required to encode a single frame, and  $q_r$  is the rate at which video frames are captured at the source node. Similarly, the lifetime of a relay node is given by:

$$T_{i\_relay} = \frac{E_b}{(e_{ij} \times q_{ij}) + (e_{ik} \times q_{ik})} \quad (4.13)$$

where  $e_{ik}$  is the energy required by relay node  $i$  to receive one information unit from some neighbour node  $k$ , and  $q_{ik}$  denotes the rate at which information is received at the relay node.

The node (or battery) lifetime can also be estimated by using different duty cycles that represent the time proportion when certain current demand is enabled (such as when a node is in active or idle state) and can be evaluated as follows [150]:

$$T_{max} = \frac{E_{offered}}{I_{average}} = \frac{E_{offered}}{\sum_{x=1}^{X = \text{number of different current demands}} DC_x \times I_{x\_average}} \quad (4.14)$$

where  $E_{offered}$  is the offered capacity of the battery pack (in mAh),  $I_{average}$  denotes the average current drawn, and  $DC_x$  is the coefficient that represents time proportion where  $x$  current demand is enabled. Finally, the network (or system) lifetime is defined as the time at which the first node (source or relay node) is drained of energy, computed as follows:

$$T_{sys} = \min_{i \in n} (T_i) \quad (4.15)$$

where  $n$  is the set of nodes in the network, including the source and relay nodes. The parameters to compute the energy capacity of AA alkaline batteries are taken from the datasheet given in [140].

Another key factor upon which we evaluated the video codecs is the minimum processing and memory requirement at the source node which facilitates the selection of appropriate hardware platform for a given WMSN application. We computed the minimum required clock frequency for each of the video codec using the encoder complexity with frame processing rate of 15fps. On the other hand, memory requirements at the source node normally include storage of a compressed video frame and a temporal raw frame (YUV). The size of the compressed video frame is primarily dependent upon the type of video compression algorithm/codec while the size of the raw frame is dependent on the frame resolution and the colour depth [10, 151]. Following the work in [151], we assumed that the size of frame buffer at encoder site is allocated according to the GOP setting. For example, when GOP size is 4, frame buffer should be able to store at least four compressed frames.

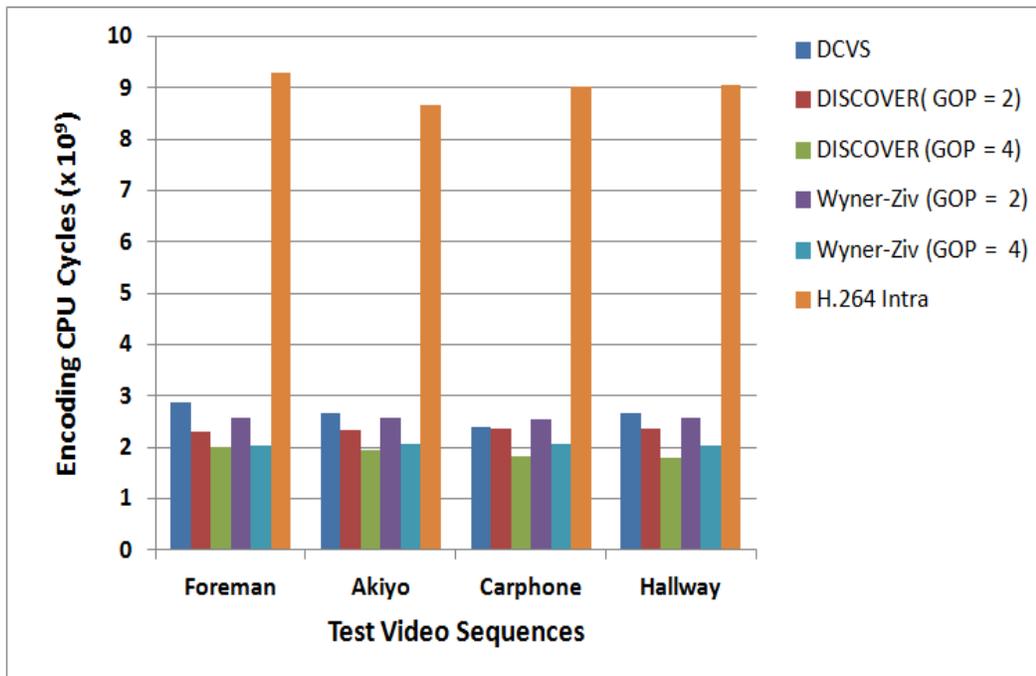
As mentioned, we assumed that the sink is a resourceful machine. Therefore, we are only concerned with its quality of reconstruction in terms of peak signal to noise ratio (PSNR). In addition, for a broader overview of the behaviour of the codecs, we also evaluated the decoding complexity in terms of the required number of CPU cycles. All results are presented for a sequence of 150 frames, unless stated otherwise.

## **4.4 Results and Discussion**

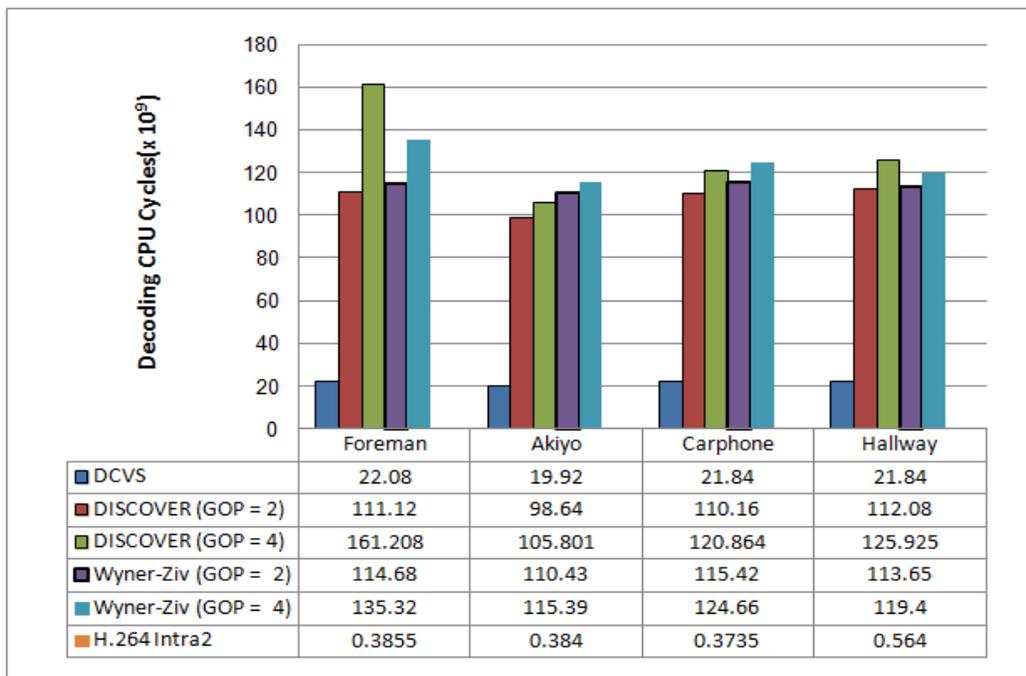
We compared the performance of the DISCOVER, Wyner-Ziv video codec, DVCS, and H.264 Intra in terms of their computation (encoding and decoding) complexity, energy (computation and communication) consumption, and PSNR. Two variants of the DISCOVER and Wyner-Ziv video codec were investigated, one with GOP size 2, and another with GOP size 4. The use of a short and longer GOP size is to test the behaviour of DISCOVER when temporal correlation is exploited more than once for every key frame. For DVCS, we used a GOP size of 3 as suggested in [138].

### **4.4.1 Encoding and Decoding Computational Complexity**

Fig. 4.3(a) presents the computation complexity of each codec in terms of the number of CPU cycles required to encode a sequence of 150 frames for each test video sequence. It shows that DISCOVER is computationally more efficient than the other three codecs. With GOP 4, its



(a)



(b)

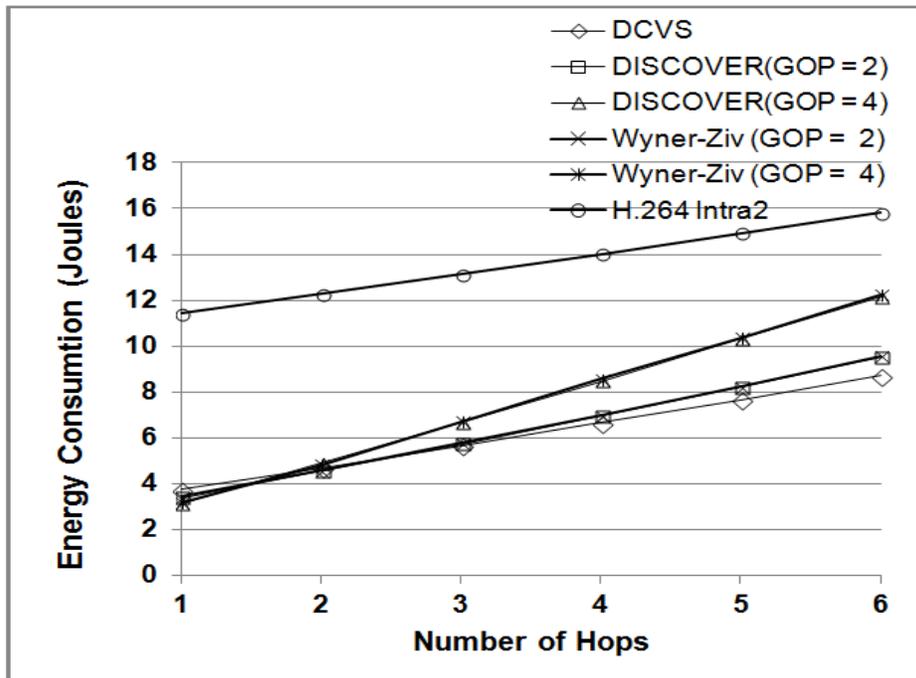
**Figure 4.3** - Codec computational complexity (a) at Encoding Site (b) at Decoding Site

complexity is even lower, which is expected, since fewer frames are required than GOP 2 to be encoded as key frames using conventional codec. On the other hand, Wyner-Ziv is slightly heavier than the state-of-the-art DISCOVER encoder for both GOP configurations due to less sophisticated channel coding and rate-estimation module as compared to its counterpart (DISCOVER). DVCS exhibits higher but still comparable complexity with respect to DISCOVER. Its complexity, however, also depends on its measurement rates used. While lower measurement rates can be used to further reduce encoding complexity of its key and non-key frames, this might come at the cost of a reduced quality of reconstruction. Finally H.264 Intra, being a conventional video codec, is computationally less efficient than all the other encoders.

Fig. 4.3(b) represents the decoding computational complexity of the respective codecs for each video sequence. It is clear that H.264 Intra, being a conventional video codec with simple decoder architecture shows the best decoding performance. In addition, DCVS decoder is significantly more efficient than both DISCOVER and Wyner-Ziv for each of the GOP configurations. This is because these codecs involves two major decoding components: Key frame decoding and WZ frame decoding, which involve complex operations such as motion estimation, frame interpolation/extrapolation, and side information generation. In addition, the higher the GOP size, the higher the complexity share of WZ frames decoding. Thus, GOP 4 configuration shows higher decoding complexity than one with GOP 2.

#### **4.4.2 Energy Consumption**

Fig. 4.4 shows the total energy consumption averaged over four video sequences in a multi-hop scenario. It is observed that for one-hop transmission, DISCOVER and Wyner-Ziv with GOP 4 exhibits a better energy performance than other codecs, which is not unexpected since they have the lower computation complexity, and computation energy (to be shown shortly) is found to be a dominant factor in the energy consumption.



**Figure 4.4** - Average total energy consumption over different number of hops

However, as the number of hops increases, their energy consumption is found to increase at a faster rate in contrast to the corresponding GOP 2 configuration and for DCVS as well. This is due to increased feedback channel requests and response overhead with a higher GOP size, which accumulates when transmitted over multiple hops.

**Table 4.3** - Computation and communication energy consumption (joules)

Hop(s)		1	2	3	4	5	6
DCVS	Computation	3.29					
	Communication	0.45	1.40	2.37	3.36	4.37	5.42
	Total	3.74	4.69	5.66	6.65	7.66	8.71
DISCOVER (GOP=2)	Computation	2.85					
	Communication	0.54	1.69	2.87	4.09	5.36	6.69
	Total	3.39	4.54	5.72	6.94	8.21	9.54
DISCOVER (GOP=4)	Computation	2.30					
	Communication	0.83	2.58	4.38	6.17	8.00	9.87
	Total	3.13	4.88	6.68	8.47	10.30	12.17

<b>Wyner-Ziv (GOP=2)</b>	<b>Computation</b>	2.89					
	<b>Communication</b>	0.54	1.7	2.91	4.1	5.37	6.69
	<b>Total</b>	3.43	4.59	5.8	6.99	8.26	9.58
<b>Wyner-Ziv (GOP=4)</b>	<b>Computation</b>	2.35					
	<b>Communication</b>	0.84	2.54	4.38	6.21	8.02	9.89
	<b>Total</b>	3.19	4.89	6.73	8.56	10.37	12.24
<b>H.264 Intra</b>	<b>Computation</b>	11.01					
	<b>Communication</b>	0.40	1.26	2.13	3.01	3.91	4.82
	<b>Total</b>	11.41	12.28	13.14	14.02	14.92	15.83

Table 4.3 shows the decomposition of the total energy consumption of each codec into computation energy and communication energy. From the results, we made the following observations. Firstly, it is observed that for all encoders with, the computation energy dominates the energy consumption, i.e. more energy is needed for video processing than for video transmission, an observation similarly made by Margi et al. [152]. This highlights the critical need for lower complexity encoders, as they would have a significant impact on the lifetime of energy-constrained WMSNs.

Secondly, it is evident that the total energy consumption of DISCOVER with GOP 2, Wyner-Ziv with GOP 2, and DCVS is comparable over multiple hops. This is because even though DCVS consumes higher computation energy, it is more efficient in communication (no feedback channel) and exhibits more energy effective behaviour from three hops onwards. While DISCOVER and Wyner-Ziv with GOP 4 have lower computational complexities, but they consumes the most communication energy as the number of hops increases. The reason is that, with higher GOP size, there are fewer key frames, which leads to lower compression ratio and more transmission bits. Another reason is its frequency of feedback channel usage, which is found to be higher than with GOP 2 because of more mismatches due to fewer occurrences of key frames.

In DISCOVER and most of the DVC-based codecs, the feedback channel has the responsibility to transmit parity bits to minimize the errors/mismatches present in the Wyner-Ziv frame to be decoded by changing the statistics provided by side information (an estimation of the frame

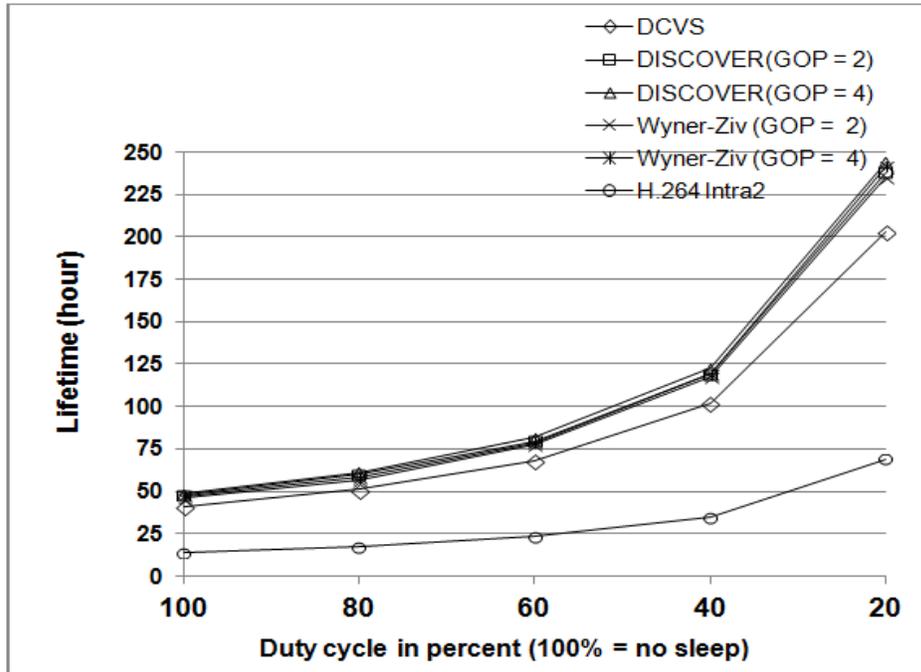
to be decoded) [91, 98]. This implies that the lower the mismatches/errors, the fewer the decoder requests and lower the feedback channel rate. For each request made by the decoder, the encoder sends parity bits to correct the errors/mismatches. DISCOVER and Wyner-Ziv with GOP 2 and GOP 4 exhibits on average 0.59, 1.05, 0.63, and 1.11 feedback requests per frame, respectively for each of the video sequences. For each request, the encoder will send 8-bits corresponding to 8-bitplanes to correct mismatches in the reconstructed frame of the decoder. With a higher feedback channel usage, the amount of energy expended for the transmission of overhead bits can be exacerbated in a multihop scenario due to transmission of feedback requests and responses between the source and sink.

The overall energy depletion of H.264 Intra is found to be higher than DISCOVER the other codecs, but at the same time its energy consumed for communication is the lowest among the codecs. This is due to its conventional codec design, which leads to better compression ratio and less transmission bits per frame, at the expense of higher encoding complexity.

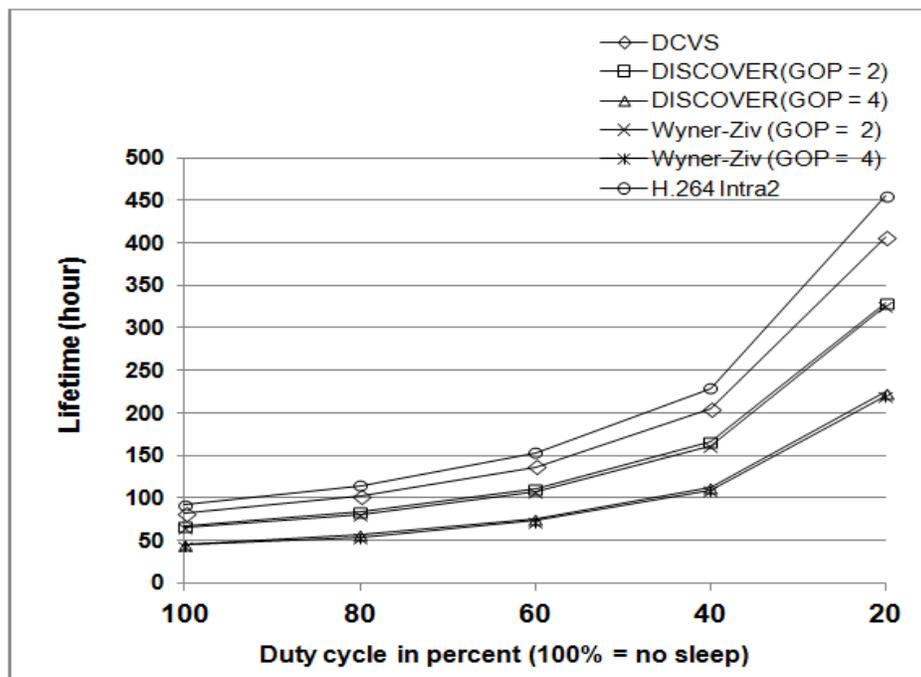
#### **4.4.3 Node and Network Lifetime**

In this section, we discuss the results of node and network lifetime estimation for a given WMSN using TelosB platform. To address different application scenarios ranging from continuous operation (100% duty cycle) to low-power (low duty cycle) mode operation where nodes alternate between active and dormant (sleep) states, we evaluated the lifetimes for five different duty cycles: 20%, 40%, 60%, 80%, and 100%, corresponding to DC coefficient values of 0.2, 0.4, 0.6, 0.8, and 1.0, respectively, in Eqn. (4.14).

From Fig. 4.5a, it is evident (and also reinforces our previous results) that a source node using H.264 Intra has the shortest lifetime due to its high encoding complexity that contributes significantly to the node's energy requirement. With 100% duty cycle, the lifetime of the source node with H.264 Intra is less than 14 hours, which increases to 69 hours when duty cycle is decreased to 20%. By contrast, the relay node using H.264 Intra has the longest lifetime as compared to its counterparts as shown in Fig. 4.5b. This is because, being a conventional video codec, H.264 has best compression ratio as opposed to DVC-based codecs. As a consequence, relay nodes with H.264 Intra have the lowest communication load, which is the major source of their energy consumption. As the duty

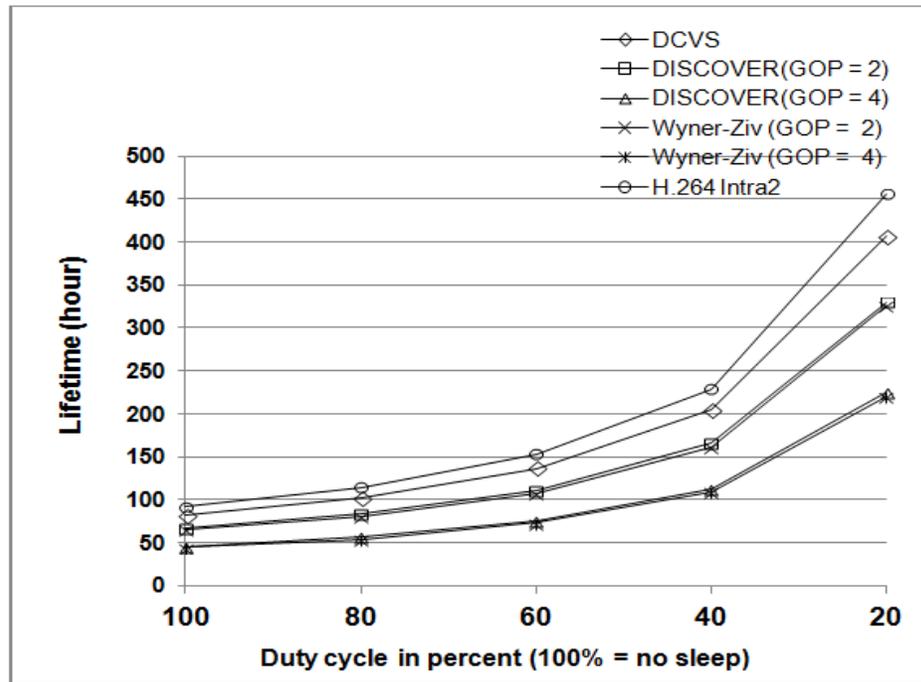


(a)



(b)

**Figure 4.5** - Node lifetime at (a) Source; (b) Relay; under different duty cycles and video codecs



**Figure 4.6** - Network lifetime under different duty cycles and video codecs

cycle decreases from 100% to 20%, the lifetime of the relay node with H.264 Intra increases from about 92 hours to 456 hours.

Using DISCOVER codec with GOP 4, the source node has a longer lifetime irrespective of the duty cycle. More specifically, the lifetime of the source node using DISCOVER with GOP 4 increases from 49 hours to 244 hours as its duty cycle decreases from 100% to 20%. On the other hand, Wyner-Ziv with similar GOP configuration falls slightly behind DISCOVER with lifetime of 47 hours to 237 hours for duty cycle variation from 100% to 20%.

Conversely, the shortest relay node lifetime of 43 hours, and 219 hours at 100%, and 20% duty cycle, respectively, is shown by Wyner-Ziv with GOP 4 configuration. This is due to its poorer compression ratio than its counterparts and thus a higher communication load.

In terms of the network lifetime, which considers the minimum of the source and relay node lifetime as shown in Fig. 4.6, it is observed that DISCOVER GOP 2 and H.264 Intra have the longest, and shortest predicted network lifetime, respectively.

#### 4.4.4 Quality of Reconstruction

Fig. 4.7 shows the quality of reconstruction from the encoded frames for each video sequence in terms of PSNR, which is the commonly used objective quality measure in literature. The PSNR is evaluated in decibels and is inversely proportional to the Mean Squared Error (MSE) of the pixel values between the original and compressed video frame:

$$MSE = \frac{1}{MN} \sum_{x=1}^M \sum_{y=1}^N \{ (f(x,y) - f'(x,y))^2 \} \quad (4.16)$$

$$PSNR = 10 \log \left( \frac{MAX_f^2}{\sqrt{MSE}} \right) \quad (4.17)$$

where  $M$  and  $N$  represent the height and width of the video frame, respectively,  $f(x,y)$  and  $f'(x,y)$  represents the original and compressed video frame at location  $(x,y)$ , respectively, and  $MAX_f$  is the maximum possible pixel value of the frame.

Generally, it is observed that the PSNR performance of DCVS and both variants of DVC i.e. DISCOVER and Wyner-Ziv is comparable across the four test video sequences, while H.264 Intra consistently achieved a higher PSNR than the other codecs. This is because H.264 Intra encodes every frame in the video like a key frame, which preserves more details for later reconstruction, and since each frame is self-contained that can be decoded independently, it is not subject to quality deterioration due to dependency between coded frames as in other codecs.

H.264 Intra also exploits spatial correlation efficiently which leads to higher PSNR values at the expense of greater computational complexity at the encoder, whereas DISCOVER and Wyner-Ziv exploits temporal correlation using simple, low complexity encoder at the expense of lower reconstruction quality. However, from the overall findings, it is evident that in multihop scenario, DCVS as compared to DISCOVER and Wyner-Ziv codec, still manages to achieve acceptable PSNR while maintaining low energy and decoding complexity at the same time. It is also observed that each of the considered codecs has achieved a PSNR greater than 30dB for all sequences. In wireless video transmission, a PSNR greater than 25dB is generally considered acceptable [153].

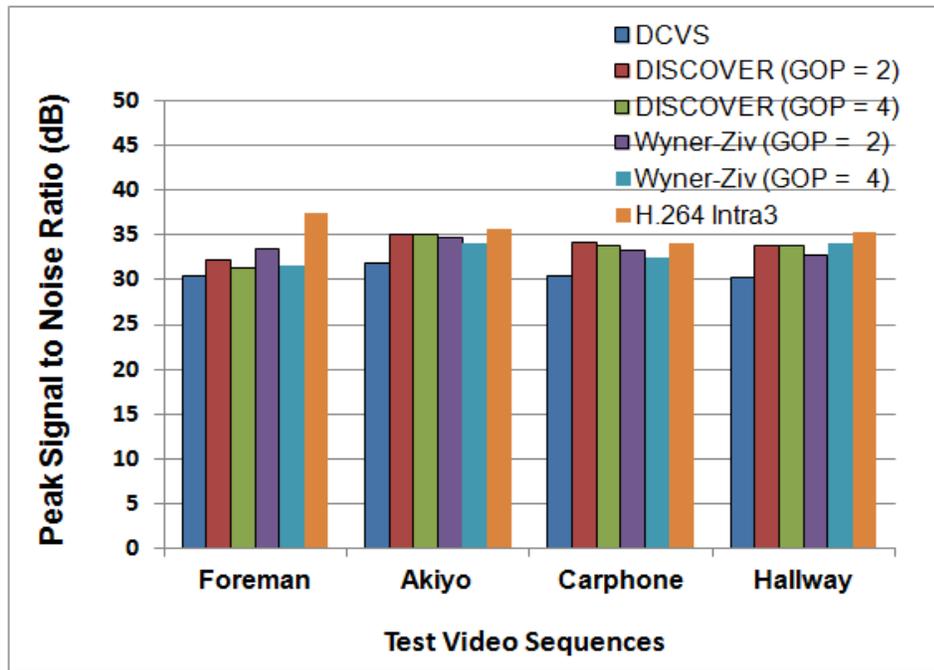


Figure 4.7- Quality of reconstruction

#### 4.5 Analysis of Codecs

Based on the findings above, we proceed to discuss the appropriate selection of video codecs given the characteristics of the WMSN application and identify possible areas of enhancement to these codecs. Of the encoders evaluated, DISCOVER and Wyner-Ziv with GOP 4 are the most energy efficient, and thus are suitable for applications with very strict energy budget at the encoding nodes. However, the application would be required to use a relatively powerful device as decoding node(s) since their decoder is 3–4 times more complex than its encoder [115]. Both DISCOVER and Wyner-Ziv codecs requires a feedback channel for the decoder to receive parity bits from encoder for error correction during frame reconstruction. Thus, their usage is restricted only to applications that support video encoding and decoding in real-time (active communication mode), e.g. real-time video streaming [19, 154] between the video source and sink.

Among the existing Wyner-Ziv implementations, the DISCOVER provides the best known rate-distortion performance due to its improved quality of generated side-information, online correlation noise estimation, and exploitation of the correlation model between source and side information for optimal reconstruction [91]. However, the performance in terms of reconstruction quality and compression ratio of DISCOVER or Wyner-Ziv codecs in

general, has not yet reached the level of predictive coding (H.264 Intra) as observed from the results in Section 4.4. Further improvements in achievable compression, optimisation on side-information generation and feedback channel rate are also required to reduce the communication overhead during frame reconstruction and to enhance the overall lifetime of the network. It is also worth mentioning that in contrast to the Wyner-Ziv codec [137], DISCOVER implementation is not available as an open source project, which restricts the researchers to employ it in their respective applications.

DCVS is based on an integration of DVC and CS; the latter is a relatively new coding paradigm. DCVS clearly performed better than H.264 Intra and is comparable to DISCOVER with GOP 2 in terms of encoding complexity, energy consumption and network lifetime. As far as decoding complexity is concerned, DCVS performs 3-4 times better than DISCOVER. From Fig. 4.3a, it can be seen that DCVS's encoder structure is actually simpler than DISCOVER, and thus has greater potential for reduced encoding complexity. Its efficiency may be improved by adapting the measurement rate for key and CS frames according to the decoding success rate established using given application criteria. An adaptive GOP scheme can also allow fewer key frames to be inserted between CS frames whenever few or no changes in background or foreground are detected in the video sequence. This reduces the overall measurements required and energy consumption at the encoder. Also, since DCVS does not require feedback channel, it is suitable for the applications that operate in passive communication mode.

## **4.6 Chapter Summary**

In this chapter, we have analysed the performance of four potential video codecs for WMSN: DISCOVER, Wyner-Ziv, DCVS, and H.264 Intra, in terms of their computation complexity at encoder, overall energy consumption due to computation and transmission of encoded frames from source to sink, decoding complexity, lifetime of node and network, processing and memory requirements and the quality of reconstruction. Using an energy consumption model based on TelosB mote specifications, and evaluating them over four different video sequences, DISCOVER and Wyner-Ziv codec with GOP 4 is found to be the most energy-efficient encoder for single-hop environment. However, due to their dependency on feedback channel for decoding operation and higher communication energy, its

usage may be limited and suitable to only real-time applications in single hop environment. On the other hand, the performance of DISCOVER and Wyner-Ziv codec with GOP 2 and DCVS encoder is comparable in terms of energy efficiency for multihop environment. Analysis of its energy consumption results revealed that DCVS consumes lower communication and higher computation energy than both variants of DISCOVER and Wyner-Ziv codec. The latter comes as a bit of surprise given its simpler encoder structure, and may be attributed to the measurement rates used for its key and CS frames which may be sub-optimal. However, DCVS can be used in both real-time and non-real time applications. H.264 Intra is found to have lower communication energy consumption than DISCOVER, Wyner-Ziv and DCVS due to its better compression ratio. However, it consumes much higher computation energy due to its greater encoding complexity, which significantly increases its overall energy consumption.

In terms of decoding complexity, the performance of H.264 Intra clearly stands out from other codecs, while DCVS is found to outperform DISCOVER and Wyner-Ziv. From our results, it is evident that none of the codecs met the requirements of having very low-complexity encoder and decoder at the same time. For the WMSN applications, where receivers are also resource-constrained devices such as PDAs or smart phones, and thus require simple decoders. In addition to simple encoders, transcoder-based solutions [82, 88, 90] that aim to achieve low complexity encoding (e.g. DISCOVER/Wyner-Ziv encoder) and low complexity decoding (e.g. H.264 Intra decoder) could be considered. We evaluated the lifetime of both source and relay nodes and the overall network for each of the video codecs. The relative order of the codecs in terms of the lifetime of the source node, which is considerably shorter than that of relay node, and thus dictates the network lifetime, expectedly follows the computational complexity results.

Finally, in terms of quality of reconstruction, it is found that DISCOVER, Wyner-Ziv codec and DCVS have comparable and reasonable performance with an achieved PSNR of at least 30 dB for all test sequences. However, as expected, it is still inferior to H.264 Intra. In addition to the performance analysis, we have also identified possible areas for further enhancement to the above codecs. As for future work, we would like to build on the knowledge gained about these video codecs to design robust and energy-efficient video watermarking techniques for WMSN.

Focusing on the significance of image authentication in WMSN environment, in the next chapter, we presented an enhanced DWT based energy-aware, semi-oblivious, adaptive image watermarking scheme (discussed in Section 3.3.2) for WMSNs. We evaluated various aspects of the enhanced version of the scheme including the distortion in cover image due to watermark redundancies, the number of embedding locations with respect to two channel adaptive parameters, and the impact of compression of the cover image on the correctness of extracted watermark.

# Chapter 5

## An Energy-Aware Adaptive DWT Based Image Watermarking

### 5.1 Introduction

In this chapter, we adapted a semi-oblivious energy-aware adaptive watermarking scheme for WMSNs, which considers key characteristics such as the embedding capacity, security, imperceptibility, computation and communication energy requirements. We evaluated the distortion in cover image due to watermark redundancies, the number of embedding locations with respect to two channel adaptive parameters, and the impact of compression of cover image on the correctness of extracted watermark. In addition, we investigated the robustness of the scheme against statistical analysis attacks. The results show that the proposed scheme has sufficient capacity to embed redundant watermarks in the cover image in an imperceptible manner with reasonably low distortion. The scheme is also considered relatively robust against collusion and middleman attacks.

The rest of the chapter is organised as follows: Section 5.2 explains the scope and motivation for the proposed watermarking scheme, Section 5.3 presents the design of the watermarking scheme. Section 5.4 describes the experimental setup. Section 5.5 presents and discusses the results. Finally, we conclude our work in Section 5.6.

## 5.2 Related Work

Over the past few years, WMSNs have been considered for various application domains ranging from sensitive military systems for enemy tracking and surveillance, to civilian and scientific systems for crime control, environmental and health monitoring. WMSN based applications employ wireless nodes equipped with camera sensors that monitor and track the changes within their field of vision and transmit the visual information either continuously or periodically to some central server or base-station. However, in contrast to simple wireless sensor networks (WSNs) which process scalar data such as temperature and pressure, camera sensors in WMSNs can generate huge amounts of data for transmission which may considerably reduce not only the lifetime but the performance of the network as well due to its limited bandwidth, memory, and energy capacities [45, 60, 155].

We propose a watermarking system blended with the essence of all three approaches (discussed in Section 2.3), namely watermarking, hashing, and cryptography. In the context of WMSN, robustness, imperceptibility, and energy efficiency are regarded as the key goals of a watermarking system [130, 156].

Transform domain watermarking techniques based on discrete cosine transform (DCT) [43, 157], discrete wavelet transform (DWT) [158-159], or discrete Fourier transform (DFT) [68], which embed the watermark into transform coefficients, are known to be more robust and resistant to compression algorithms since the watermark bit sequence is spread over the entire cover object rather than on the particular set of pixels. However, the cover object is subjected to a certain level of distortion due to watermark embedding at source coding site along with the error-prone wireless transmission channel which may affect the watermarking system's performance to a significant extent. In order to minimize the distortion in cover image object due to watermark embedding, the watermark should be embedded into a small number of locations as possible, but on the other side, watermark coding redundancies are needed to make the scheme robust against wireless channel errors which eventually presents a trade-off between watermarked image object distortion and robustness of the watermarking system. For example, a higher packet loss ratio may cause the watermark undetectable and result in failed authentication of the cover watermarked image object at receiver site. Thus, the watermarking scheme should be robust enough to survive not only the source coding compression, but also errors induced by the wireless channel.

On the other hand, watermarking schemes can be classified into oblivious, semi-oblivious, and non-oblivious with respect to their detection mechanism.

- In an oblivious watermarking system, the cover image (the original image) is not needed and only the watermarked image is transmitted to the receiver for watermark detection.
- In a semi-oblivious watermarking system, the cover image (the original image) is not needed at receiver. However, information such as shared keys in addition to the watermarked image may be required for successful watermark detection.
- A non-oblivious watermarking system requires the original image, watermarked image and optionally some additional information at the receiver for watermark detection.

In context of WMSNs, only oblivious and semi-oblivious detection mechanisms offer a viable watermarking solution, since most of the applications do not have access to the original image at the receiver site, which is required for the watermark extraction in non-oblivious watermarking schemes.

We proposed an enhanced DWT based energy-aware, semi-oblivious, adaptive image watermarking scheme that is more practical for WMSNs than its original non-oblivious version presented in [51]. We evaluated various aspects of the enhanced version of the scheme including the distortion in cover image due to watermark redundancies, the number of embedding locations with respect to two channel adaptive parameters, and the impact of compression of the cover image on the correctness of extracted watermark. Based on channel conditions, the scheme dynamically selects the watermark embedding locations to ensure watermark security and energy efficiency at the same time.

### **5.3 Energy-Aware Image DWT based Watermarking for WMSNs**

Watermarking schemes can be classified into two main branches based on their implementation domain, namely, spatial domain watermarking [160-161] and transform domain watermarking [43, 68, 158, 162]. Spatial domain watermarking schemes embed the watermark into the cover image object by directly modifying pixel values, e.g. replacing the least significant bits of cover image with watermark bits at certain locations. However, in contrast to transform domain watermarking (explained in Section 1), these schemes have less embedding capacity and usually are not very robust against wireless channel errors and

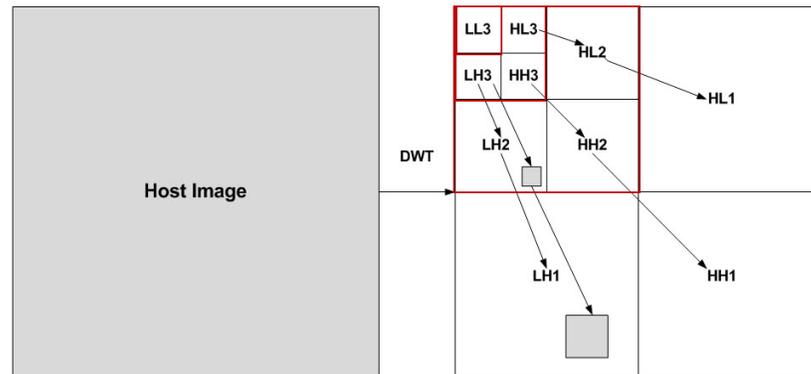
compression attacks [160]. On the other hand, among the three categories of transform domain, watermarking schemes based on DWT have shown reasonable robustness against various attacks than their counterparts [163-164].

This section describes an enhanced embedding algorithm based on the original version presented in [51] (also covered in our literature review), which uses qualified significant wavelet trees (QSWT) [165] for watermark embedding. The enhancement includes a proposed hashing algorithm used to generate a synchronized coefficient variation factor  $\sigma$ , which executes at both source and destination sites for watermark embedding and detection, respectively. A semi-oblivious watermark detection algorithm is then presented which is more practical than the original non-oblivious design [51]. Presence of the original image at the receiver site seems to be an unlikely scenario for WMSN applications. Therefore, we enhanced the embedding algorithm and devised a hashing based semi-oblivious extraction method that does not require the original image at the receiver site. However, a small amount of encrypted information is transmitted along with each watermarked image to enable the extraction of the watermark at the receiver's end. The given frame undergoes a 3-level discrete wavelet transform and embeds the watermark into selected coefficients (QSWTs) of LH3 sub-band. The two adaptive thresholds  $\alpha_1$  and  $\alpha_2$  are used to discover sufficient number of QSWTs to ensure proper watermark embedding and to allow watermark redundancies in case of poor channel conditions. These thresholds  $\alpha_1$  and  $\alpha_2$  decide the embedding locations in LH3, and LH2, sub-bands respectively. Additionally,  $\sigma$  is a factor by which the watermarked image varies from the original image while maintaining imperceptibility. The value of sigma remains constant for the entire sub-block/image but must be changed for every image to maintain the imperceptibility as well as security of the watermark.

- **DWT Transformation and QSWTs**

DWT transformation samples horizontal and vertical channels using band filters (high pass and low pass) and decompose image into four partitions of low, middle and high frequency bands namely LL, HL, LH, and HH respectively. In a multi-resolution DWT, the subbands at level 1 are the finest scale of DWT coefficients. For coarsed DWT transformation, LL1 sub-band is re-sampled and decomposed into LL2, HL2, LH2, and HH2 sub-bands

respectively as shown in Fig. 5.1. The partitions outlined with red colour are representing LL1, LL2 and LL3 sub-bands respectively.



**Figure 5.1** - Multi-resolution DWT transformation of an image

Conversely, the original image can be reconstructed by performing an inverse DWT transformation on the DWT coefficients. In the case of multi-resolution DWT, the inverse operation is performed multiple times to scale up to the original image resolution. In the scheme under discussion, the watermark is embedded into selected coefficients of LH3 sub-band and the selection approach is based on QSWTs. From Fig.5.1, it is evident that a parent-child relationship can be defined between coefficients of each band at different transformation levels. For example, coefficients at one coarser level (parent) are related to the coefficients at next finer level (children) until the finest level which has no children is traversed. Hence a complete wavelet tree or hierarchy can be seen. The QSWT can be formally defined as:

“ If a wavelet coefficient  $x_n(i, j) \in D$  at the coarser level is a parent of  $x_{n-1}(p, q)$  where  $D = \{HL_n, LH_n, HH_n\}$  (represents the sub-bands at level  $n$ ), and for a given set of thresholds  $\alpha_1$  and  $\alpha_2$ , it satisfies the conditions  $|x_n(i, j)| > \alpha_1, |x_{n-1}(p, q)| > \alpha_2$ , then  $|x_n(i, j)|$  and its children form a QSWT [165].”

### 5.3.1 Embedding Algorithm

**Input:** Image frame  $\chi$ , pixel variation factor  $\sigma$  for the entire sub-block/image, initial values for channel adaptive thresholds  $\alpha_1$  and  $\alpha_2$ . The steps for watermark embedding at source coding site are as follows: Let the frame resolution be  $n \times n$ , dimension of sub-blocks in LH3 and LH2 be  $\omega \times \omega$ , and  $2\omega \times 2\omega$ , respectively. Let the watermark frame resolution be  $\gamma \times \gamma$ , and the watermark sub-block size be  $\mathcal{Y} \times \mathcal{Y}$ .

#### Processing Steps:

1. For a given frame  $\chi$ , perform 3-level DWT on  $\chi$  and obtain LH2 and LH3 sub-bands from 2-level, and 3-level DWT, respectively, where LH3 is  $\ell \times \ell$ .
2. Transform LH3 into  $k$  non-overlapping sub-blocks of dimension  $\omega \times \omega$ , where  $\omega \in \{1, \dots, \ell\}$  given that the total number of sub-blocks should not exceed the total number of pixels in the watermark image/logo.
3. Transform LH2 into non-overlapping sub-blocks of dimension  $2\omega \times 2\omega$ .
4. Transform the watermark image/logo  $W$  into  $k$  sub-blocks of dimension  $\mathcal{Y} \times \mathcal{Y}$ . The total number of sub-blocks in watermark image should be equal to the number of sub-blocks in LH3.
5. Store the mean of all sub-blocks of LH3 in an array  $T_1$  such that  $T_1(i)$  represents the mean of  $i^{\text{th}}$  sub-block.
6. Store the mean of all sub-blocks of LH2 in an array  $T_2$  such that  $T_2(j)$  represents the mean of  $j^{\text{th}}$  sub-block.
7. Find the QSWT for each sub-block in LH3:

*for*  $m = 1$  to  $k$

*for*  $i = 1$  to  $\omega$

*for*  $j = 1$  to  $\omega$

*if*  $LH3(i, j, m) > T_1(m) + \alpha_1$

*then* {

*if*  $LH2(2i - 1, 2j - 1, m) > T_2(m) + \alpha_2$

*and if*  $LH2(2i - 1, 2j, m) > T_2(m) + \alpha_2$

and if  $LH2(2i, 2j - 1, m) > T_2(m) + \alpha_2$

and if  $LH2(2i, 2j, m) > T_2(m) + \alpha_2$

then {

a. Select  $LH3(i, j)$  as one of the QSWT of block  $m$ . The  $QSWT_{EWE}$  data structure holds the (location, value) pair of the selected  $LH3(i, j, m)$  before watermark embedding.

b. Embed watermark into the QSWTs of each sub-block using the following formula:

$$LH3\_watermarked(i, j, m) = LH3(i, j, m) + LH3(i, j, m) * \sigma * logo\_sublk(x, y, m)$$

where  $x$  and  $y$  are used to index pixel values in logo sub-block  $m$  and  $\sigma$  is a factor by which the watermarked image is allowed to vary from the original image depending on the sensitivity of the application while maintaining watermark imperceptibility. We can either employ block based or image based configuration for  $\sigma$  value. In the case of block-based setting, the hashing algorithm (Section 5.3.2) executes for each block and generates the  $\sigma$  value used for embedding the watermark in that particular block only. However, in image based setting, the hashing algorithm executes only once for the entire image and the corresponding  $\sigma$  value is used to embed watermark throughout the image.

}

}

where thresholds  $\alpha_1$  and  $\alpha_2$  are adaptive parameters whose values are subjected to channel conditions.

8. Transform *LH3\_watermarked* into  $\ell \times \ell$  dimension.

**Output:** 3<sup>rd</sup> level, watermarked LH subband (*LH3\_watermarked*) and *QSWT\_BWE* data structure that holds location, value pair of the selected QSWTs before watermark embedding. *QSWT\_BWE* plays a significant role for upholding the robustness of the watermarking scheme. Since it is being transmitted along with the watermarked image, if the attacker gets hold of this data structure, the watermark could have been revealed with ease. Therefore, we proposed to encrypt *QSWT\_BWE* before transmitting it to the base-station using elliptic curve cryptographic (ECC) algorithm [166]. ECC is a public key cryptographic algorithm widely employed in WSNs because it offers the same level of security with much lower computational complexity than its counterparts such as RSA and AES [41, 44, 167]. On the other hand, shared key cryptography may have even lower computational and storage costs, but it is a less secure solution since it may take only one compromised node to jeopardize the security of the entire network. Moreover, for almost all the images we have tested, the size of *QSWT\_BWE* is considerably small. Hence, the overall impact of employing ECC to encrypt the *QSWT\_BWE* on energy consumption of sensor nodes is still tolerable.

### 5.3.2 Hashing Algorithm

For the watermark security, we proposed a hashing algorithm that generates the same value of  $\sigma$  at both source and destination, and is evaluated based on LH3 sub-blocks of LH3 and the watermark logo. It is required for the correct embedding and extraction of the watermark to the original and from the watermarked image respectively. As mentioned in Section 5.3.1, it is a factor by which the watermarked image is allowed to vary from the original image depending on the sensitivity of the application while maintaining watermark imperceptibility.

Although hashing incurs additional complexity in terms of computation at both ends of the application, the proposed steps provide a viable watermarking solution for WMSNs with little extra energy consumption. The algorithm works as follows:

1. Compute the mean pixel value of entire LH3/sub-block and store it in Hash\_Input<sub>img</sub>/ Hash\_Input<sub>img</sub>(i). where Hash\_Input<sub>img</sub> and Hash\_Input<sub>img</sub>(i) represent the mean of entire LH3 and the mean of i<sub>th</sub> sub-block of LH3 respectively.
2. Compute the mean pixel value of entire watermark/Sub-block and store it in Hash\_Input<sub>wat</sub>/ Hash\_Input<sub>wat</sub>(i). where Hash\_Input<sub>wat</sub> and Hash\_Input<sub>wat</sub>(i) represents the mean of entire watermark image and the mean of i<sub>th</sub> sub-block of watermark image respectively. Similar to embedding algorithm, the number of sub-blocks in LH3 and watermark image must be equal.
3. Take the product of Hash\_Input<sub>img</sub>/ Hash\_Input<sub>img</sub>(i) and Hash\_Input<sub>wat</sub>/ Hash\_Input<sub>wat</sub>(i) and store it in Hash\_Product/Hash\_Product(i).
4. Divide Hash\_Product/Hash\_Product(i) with 255 to scale it down to gray level as follows:

$$\text{Hash\_Product} = \text{Hash\_Product}/255$$

5. Take mod with 0.002 to generate values less than 0.002,

$$\sigma = \text{Hash\_Product} \bmod 0.002$$

if Hash\_product is zero or  $\sigma$  is less than 0.001, return  $\sigma = 0.001$

where the range of values for  $\sigma$  varies from 0.001 to 0.002. Through experimentation on series of images varying from the famous *Lena* to *Coffee Cup* (taken from CITRIC camera board [168]), we evaluated the range of  $\sigma$  that suits most of the images with different colour, brightness, indoor, outdoor effects and detailing. Within the specified range, the watermark remains imperceptible and recognizable with acceptable Normalised correlation (NC) at receiver's end.

However, for an entirely uniform image with any of the colour code from 0-255, a watermark cannot be embedded, since Step 7 in embedding algorithm is unable to identify any of the QSWTs.

### 5.3.3 Detection Algorithm

The watermark detection algorithm in [51] is originally based on multi-resolution wavelet transform watermarking [165] which employs a non-oblivious detection mechanism for

watermark extraction. As stated earlier, it may not be realistic for WMSN applications to assume the presence of the original image at the receiver site. Therefore, a semi-oblivious detection mechanism is proposed which only requires the watermarked image and some additional encrypted information (QSWT\_BWE data structure) at the receiver. The mechanism is designed such that even if an eavesdropper manages to get hold of both pieces of information, it will still be difficult to extract the watermark due to the encrypted QSWT\_BWE, the diversity in pixel varying factor  $\sigma$ , and its combined effect with watermark on source image. Upon receiving the watermarked image and the QSWT\_BWE data structure, the receiver decrypts the QSWT\_BWE using its own private key and the ECC decryption algorithm. The resulting array is then fed into the watermark detection algorithm. The following steps are used by the receiver site to detect the watermark and authenticate a given image object.

**Input:** Denote a received frame  $\chi'$ , pixel variation factor  $\sigma$  for the entire sub-block/image, an original watermark  $W$ , and a decrypted QSWT\_BWE data structure. Let the frame resolution be  $n \times n$ , dimension of sub-blocks in  $LH3'$  and  $LH2'$  be  $\omega \times \omega$ , and  $2\omega \times 2\omega$ , respectively. Let the watermark resolution be  $\gamma \times \gamma$ , and the watermark sub-block size be  $\mathcal{Y} \times \mathcal{Y}$ .

**Processing Steps:**

1. For a given frame  $\chi'$ , perform 3-level DWT on  $\chi'$  and obtain  $LH2'$  and  $LH3'$  sub-bands from 2-level, and 3-level DWT respectively, where  $LH3'$  is  $\ell \times \ell$ .
2. Transform  $LH3'$  into  $k$  non-overlapping sub-blocks of dimension  $\omega \times \omega$  where  $\omega \in \{1, \dots, \ell\}$ , given that the total number of sub-blocks should not exceed the total number of pixels in the watermark image/logo.
3. Transform  $LH2'$  into non-overlapping sub-blocks of dimension  $2\omega \times 2\omega$ .
4. Create a data structure  $LH3\_original$  of dimension equivalent to  $LH3'$ , initialize it with  $LH3'$  values, and transform it into  $k$  non-overlapping sub-blocks of dimension  $\omega \times \omega$ .
  - a. Select the locations specified in QSWT\_BWE for each sub-block and assign  $LH3\_original$  the corresponding coefficient value at the identified location.

- For each coefficient  $(i, j)$  in sub-block  $m$  from sub-band  $LH3'$ , extract watermark value for sub-block  $m$  at location  $(x, y)$

*for*  $m = 1$  to  $k$

*for*  $i = 1$  to  $\omega$

*for*  $j = 1$  to  $\omega$

*if*  $LH3\_original(i, j, m)$  in  $QSWT\_BWE$

*then* {

$$W'(x, y, m) = \frac{[LH3'(i, j, m) - LH3\_original(i, j, m)]}{[LH3\_original(i, j, m) * \sigma]}$$

}

- Repeat the process in step 4 until watermark is completely extracted.
- Transform the watermark image  $W'$  back into  $\gamma \times \gamma$  dimension.

**Output:** Extracted watermark  $W'$ .

## 5.4 Experimental Setup

We evaluated the above-mentioned hashing, watermark embedding, and detection algorithms using three test images as shown in Fig. 5.2. Lena and Pepper images are selected due to their being widely used in the image processing field, while the Coffee Cup image is taken from our CITRIC board that comes with a 1.3MP camera [168].



(a) Lena



(b) Pepper



(c) Coffee Cup

**Figure 5.2** - Test images

We investigated various aspects of the scheme as follows:

- Normalised correlation (NC) between the original and the extracted watermarks while utilizing the maximum embedding capacity of the cover image.
- Peak signal to noise ratio (PSNR) of cover image at source coding site against the number of watermarks embedded in it.
- Total energy consumption for transmitting an image object from the source coding site to the base station (or central server) against available data rates under different channel conditions.
- Total number of embedding locations (QSWT trees) against thresholds  $\alpha_1$  and  $\alpha_2$ .
- Normalised correlation (NC) between the original and the extracted watermarks at different compression ratios of the watermarked image.

We performed our experiments in Matlab on grayscale cover and watermark images having image resolution of  $512 \times 512$ , and  $16 \times 16$ , pixels respectively. The T-MAC protocol parameters for TinyOS [169] and TelosB [11] specifications were used to evaluate the transmission energy based on the energy model presented in [170] (Table 5.1). Throughout the simulations, we have used dimensions of LH3 and watermark sub-blocks as  $8 \times 8$ , and  $4 \times 4$ , respectively.

Eqns. (5.1) and (5.2) describe the probability and the energy consumption of RTS packet transmission failure as a result of channel errors, e.g. packet loss, collision etc.

$$p_{RTS}^{Fail} = 1 - (1 - BER)^{L_{RTS} \cdot p \cdot (1 - p)^N} \quad (5.1)$$

where  $p$  is the source node probability of sending a packet,  $L_{RTS}$  represents the RTS packet size, and the total number of direct neighbours is denoted by  $N$ .

$$E_{RTS}^{Fail} = (P_{RTS}^{TX} \cdot T_{RTS}) + (P^{RX} \cdot T_{CTS\_Timeout}) \quad (5.2)$$

where  $P_{RTS}^{TX}$ ,  $T_{RTS}$  represents the transmission power, and transmission time, respectively, needed to transmit an RTS packet.  $P^{RX}$  is the received power of the circuit, and the timeout interval for receiving CTS packets is denoted by  $T_{CTS\_Timeout}$ .

Similarly, the probability and energy consumed due to transmission failure of CTS, DATA and ACK packet failures are expressed in Eqns. (5.3) – (5.4), (5.5) – (5.6), and (5.7) – (5.8), respectively. Note that the transmission energy due to DATA and ACK packet failure

is exactly the same. The reason is that if either DATA or ACK packet fails, the transmission party experiences the same behaviour.

$$p_{CTS}^{Fail} = (1 - p_{RTS}^{Fail}) \cdot (1 - (1 - BER)^{L_{CTS}}) \quad (5.3)$$

$$E_{CTS}^{Fail} = (P_{RTS}^{TX}) + (P^{RX} \cdot T_{CTS_{timeout}}) \quad (5.4)$$

$$p_{DATA}^{Fail} = (1 - p_{RTS}^{Fail}) \cdot (1 - p_{CTS}^{Fail}) \cdot (1 - (1 - BER)^{L_{DATA}}) \quad (5.5)$$

$$E_{DATA}^{Fail} = (P_{RTS}^{TX} \cdot T_{RTS}) + (P^{RX} \cdot T_{CTS}) + (P_{DATA}^{TX} \cdot T_{DATA}) + (P^{RX} \cdot T_{ACK_{timeout}}) \quad (5.6)$$

$$p_{ACK}^{Fail} = (1 - p_{RTS}^{Fail}) \cdot (1 - p_{CTS}^{Fail}) \cdot (1 - p_{DATA}^{Fail}) \cdot (1 - (1 - BER)^{L_{ACK}}) \quad (5.7)$$

$$E_{ACK}^{Fail} = (P_{RTS}^{TX} \cdot T_{RTS}) + (P^{RX} \cdot T_{CTS}) + (P_{DATA}^{TX} \cdot T_{DATA}) + (P^{RX} \cdot T_{ACK_{timeout}}) \quad (5.8)$$

From Eqns. (5.1) – (5.8), the probability and the energy consumption of successfully delivering a packet at sending site is given by:

$$p^{SUCC} = (1 - p_{RTS}^{Fail}) \cdot (1 - p_{CTS}^{Fail}) \cdot (1 - p_{DATA}^{Fail}) \cdot (1 - p_{ACK}^{Fail}) \quad (5.9)$$

$$E^{SUCC} = (P_{RTS}^{TX} \cdot T_{RTS}) + (P^{RX} \cdot T_{CTS}) + (P_{DATA}^{TX} \cdot T_{DATA}) + (P^{RX} \cdot T_{ACK}) \quad (5.10)$$

Finally,  $\bar{E}_{TX}$  represents the average transmission energy consumption of an upper layer protocol data unit (PDU) and is given by:

$$\bar{E}_{TX} = (E^{SUCC} \cdot p^{SUCC}) + (E_{RTS}^{Fail} \cdot p_{RTS}^{Fail}) + (E_{CTS}^{Fail} \cdot p_{CTS}^{Fail}) + (E_{DATA}^{Fail} \cdot p_{DATA}^{Fail}) + (E_{ACK}^{Fail} \cdot p_{ACK}^{Fail}) \quad (5.11)$$

**Table 5.1:** Parameters for evaluating transmission energy consumption

Parameters	Value
Data packet size (header + payload)	47 (11 + 36) bytes
RTS packet size	13 bytes
CTS packet size	15 bytes
ACK packet size	13 bytes

$P^{RX}$	35 mW
$P_{RTS}^{TX}, P_{CTS}^{TX}, P_{ACK}^{TX}, P_{DATA}^{TX}$	31 mW
Voltage	1.8V

Now, the energy consumed at the receiving site due to RTS, CTS, DATA, and ACK packets failure is given by:

$$E_{RTS\_R}^{Fail} = ( P^{RX} \cdot T_{RTS\_timeout} ) \quad (5.12)$$

$$E_{CTS\_R}^{Fail} = ( P^{RX} \cdot T_{RTS} ) + ( P_{CTS}^{TX} \cdot T_{CTS} ) + ( P^{RX} \cdot T_{DATA\_timeout} ) \quad (5.13)$$

$$E_{DATA\_R}^{Fail} = ( P^{RX} \cdot T_{RTS} ) + ( P_{CTS}^{TX} \cdot T_{CTS} ) + \left( P^{RX} \cdot \frac{L_{DATA}}{R_{DATA}} \right) \quad (5.14)$$

$$E_{ACK\_R}^{Fail} = ( P^{RX} \cdot T_{RTS} ) + ( P_{CTS}^{TX} \cdot T_{CTS} ) + \left( P^{RX} \cdot \frac{L_{DATA}}{R_{DATA}} \right) + ( P_{ACK}^{TX} \cdot T_{ACK} ) \quad (5.15)$$

Therefore, the overall energy consumption at the receiving end can be expressed as follows:

$$\bar{E}_{RX} = ( E_{DATA\_R}^{SUCC} + p^{SUCC} ) + ( E_{RTS\_R}^{Fail} + p_{RTS}^{Fail} ) + ( E_{CTS\_R}^{Fail} + p_{CTS}^{Fail} ) + ( E_{DATA\_R}^{Fail} + p_{DATA}^{Fail} ) + ( E_{ACK\_R}^{Fail} + p_{ACK}^{Fail} ) \quad (5.16)$$

The value of  $E_{DATA\_R}^{SUCC}$  will be identical to that of  $E_{ACK\_R}^{Fail}$  since after the transmission of ACK, the behaviour of receiving site will not change irrespective of whether the ACK is received successfully or not. Therefore, using Eqns. (5.11) and (5.16), the average total energy in transmitting and receiving a packet is given by:

$$\bar{E}_{Total} = \bar{E}_{TX} + \bar{E}_{RX} \quad (5.17)$$

It is evident from Eqns.(5.1) – (5.17) that for each type of packet, the packet transmission time which is a function of the packet length and data rate, is an important contributing factor to the total energy consumption. We will further examine the relationship between the total energy consumption and the available data rate in Section 5.5.4.

## 5.5 Results and Discussion

In this section, the proposed image watermarking scheme is evaluated and analysed based on standard measures such as distortion caused by watermarking embedding against capacity of embedding, thresholds relationship with QSWTs, energy consumption against data rate, robustness against compression and statistical analysis attacks.

### 5.5.1 Distortion in Cover Image Object and Capacity of Embedding

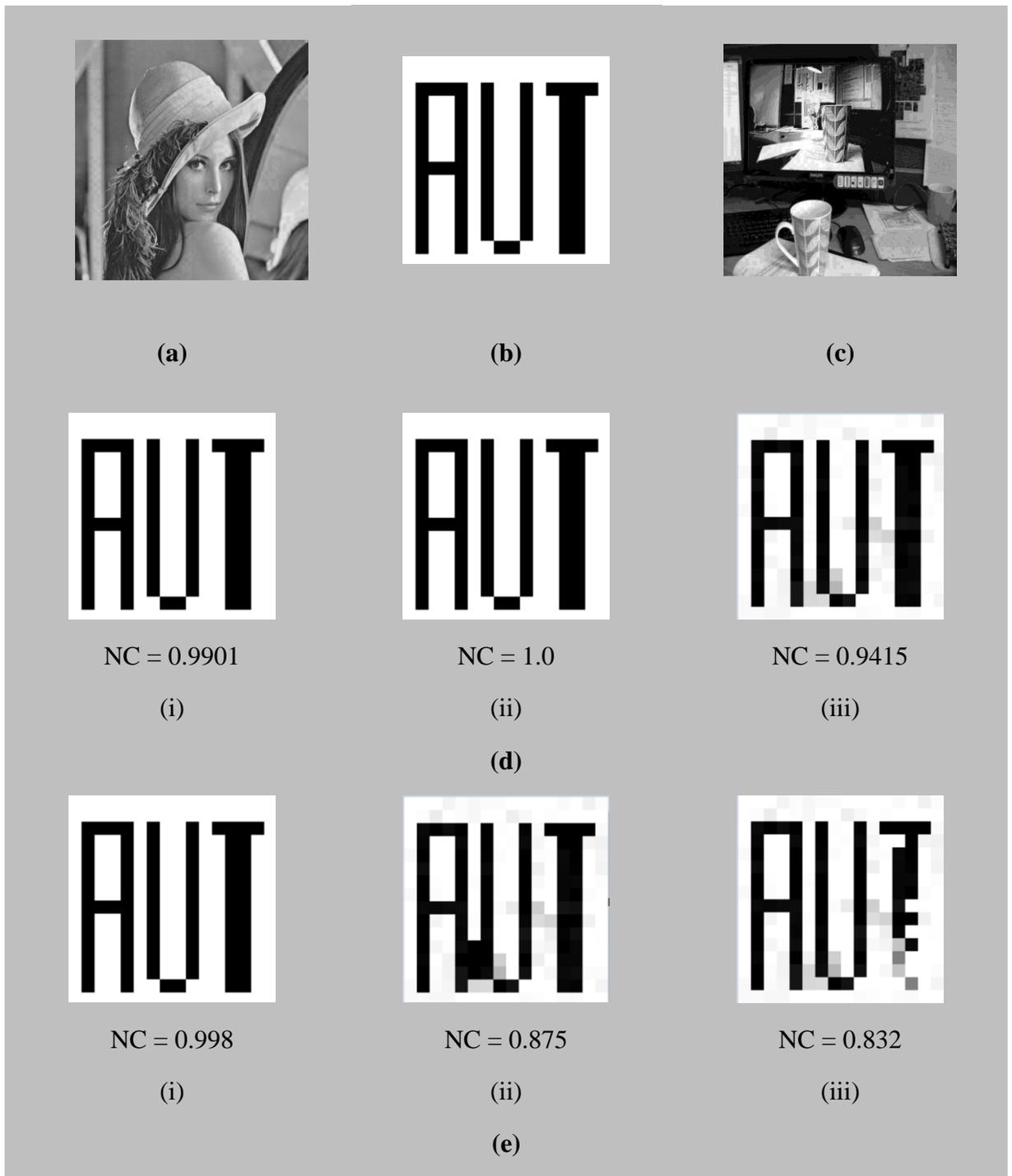
In this section, we describe the amount of distortion (in terms of PSNR) that watermark embedding may have caused to the cover image by gradually increasing the utilization of the embedding algorithm's capacity up to its maximum limit. The capacity of a watermarking algorithm is defined as the maximum allowable limit (in terms of bits) of watermark sequence that can be embedded into the cover object [159]. We performed our experiment on Lena, Pepper, and Coffee Cup images using threshold values of 25, and 50, for  $\alpha_1$ , and  $\alpha_2$ , respectively. These threshold values select the maximum QSWTs from the cover image and therefore the maximum number of embedding locations. However, for each subsequent embedding, the algorithm only proceeds if the remaining number of QSWTs are sufficient to hold one complete watermark. Results have shown that the scheme upholds the watermark imperceptibility even after maximum redundant embeddings as shown in Fig. 5.3(a) and 5.3(c).

The quality of the watermark upon extraction from the noisy watermarked image is computed using Normalised correlation (NC), which is widely used to measure the similarity between two images i.e. the original and extracted watermark, and is given by:

$$NC = \frac{\sum_{i=1}^w \sum_{j=1}^m \omega_{(i,j)} \omega'_{(i,j)}}{\sum_{i=1}^w \sum_{j=1}^m [\omega_{(i,j)}]^2} \quad (5.18)$$

where  $\omega (i, j)$  and  $\omega' (i, j)$  represents the original, and extracted watermarks, respectively.

Fig. 5.3(d) and 5.3(e) show the extracted watermarks with their NC values from Lena and Coffee Cup images respectively.

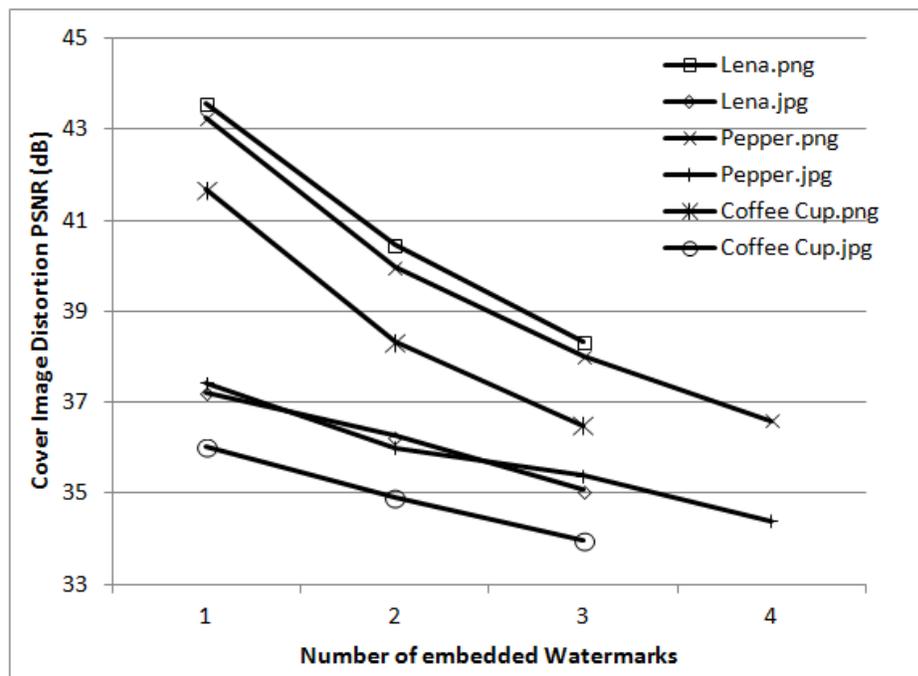


**Figure 5.3** - (a) Watermarked Lena image (512 × 512); (b) Watermark composed of letters "AUT" (16 × 16); (c) Watermarked Coffee Cup image (512 × 512); (d)

Recovered watermarks ( $16 \times 16$ ) from watermarked Lena; (e) Recovered watermarks ( $16 \times 16$ ) from watermarked Coffee Cup

One complete watermark image/logo embedding takes at least 256 QSWTs. However, Lena and Coffee Cup have 941 and 1002 QSWTs that support the embedding of the entire watermark three times only. A detailed explanation about the embedding locations (QSWTs) will be given shortly in Section 5.5.2. It can also be seen that the NC of extracted watermarks from the Lena is relatively higher than those extracted from the Coffee Cup image. This could be due to the fact that the Coffee Cup image is relatively more cluttered compared to the Lena Image.

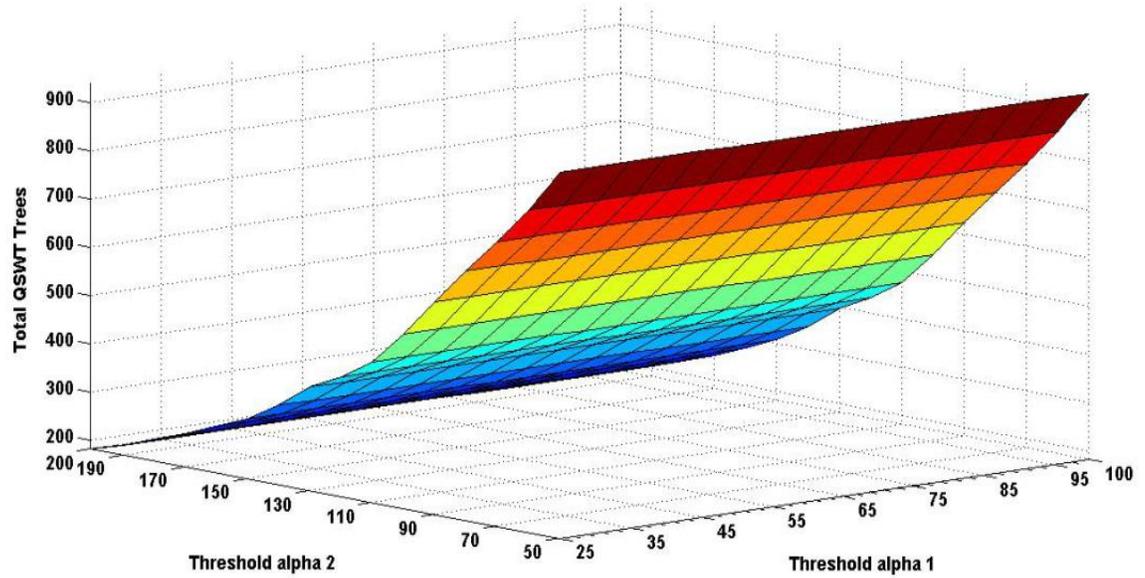
The discontinuity in PSNR values after embedding the 3<sup>rd</sup> and 4<sup>th</sup> watermark for a given set of thresholds, implies that the images no longer have sufficient capacity to embed another replica of the watermark. Although some of the QSWTs are still available, their number is less than that required for one complete embedding (Fig. 5.4).



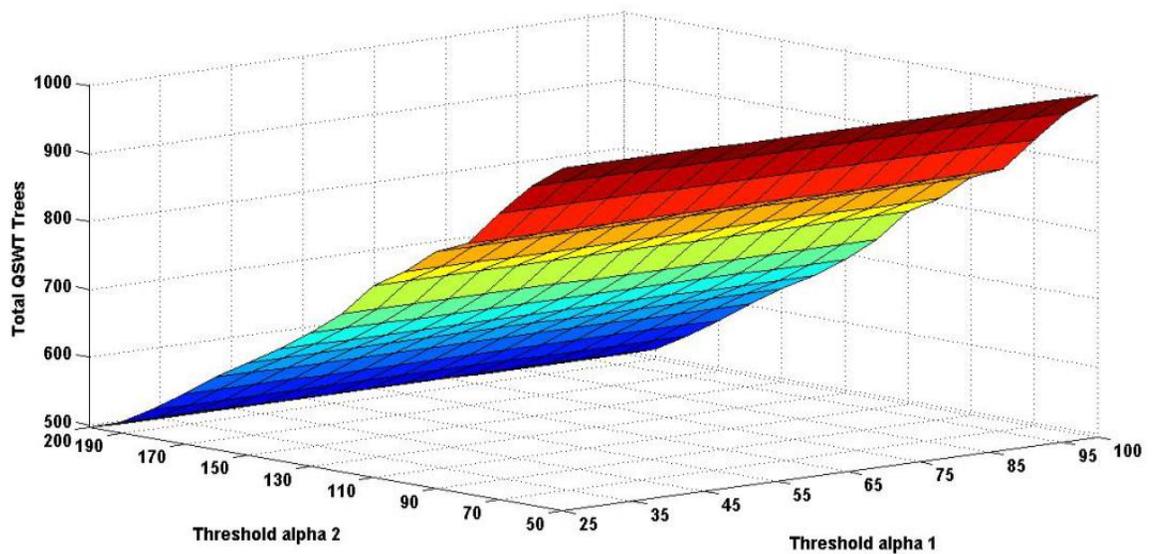
**Figure 5.4** - Distortion of cover image (PSNR) due to watermark embedding

The scheme only embeds replicas of the watermark when the remaining number of QSWTs is equal or greater than that essential for one complete embedding. Even after utilizing maximum embedding capacity, the scheme is still found capable of hiding the redundant watermarks in an invisible manner.

The effect of lossy and lossless image compression formats, i.e. JPG, and PNG, respectively, on the cover image distortion for a given number of embedded watermarks, is also examined. As expected, the lossless image compression (PNG) resulted in higher PSNR with the same number of embeddings than its lossy counterpart (JPG) as shown in Fig. 5.4.



(a)



(b)

**Figure 5.5** - Threshold  $\alpha_1$  vs.  $\alpha_2$  vs. QSWT trees: (a) Lena; (b) Coffee Cup

### 5.5.2 Thresholds and QSWTs

We examined the relationship between the thresholds  $\alpha_1$  and  $\alpha_2$  and the selected QSWTs for various images. As discussed in Section 5.3, watermark embedding is a function of these thresholds which are adaptive to channel conditions. Similar to other wireless networks, WMSNs are likely to suffer from channel errors, which may cause a valid image to be rejected due to failure in the watermark authentication process. The receiver may request the retransmission of an image if the watermark authentication process fails, which inevitably increases the overall network communication cost and delay.

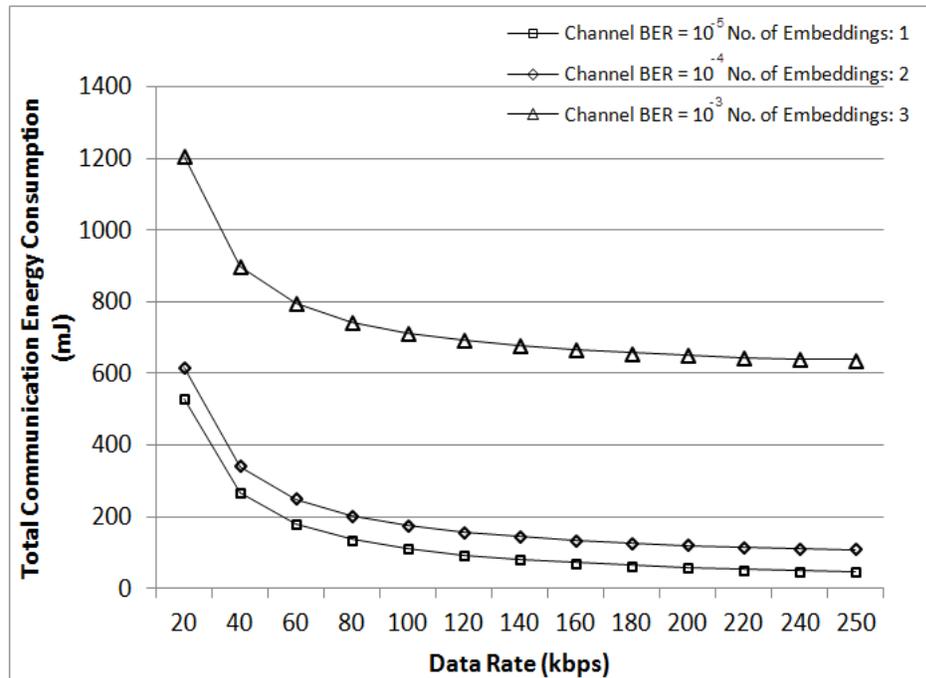
However, the scheme under discussion adaptively selects the number of watermarks to be embedded in the cover image object based on the threshold pair  $\alpha_1$  and  $\alpha_2$ . In the case of higher channel bit error rate (BER), the scheme uses lower threshold values which eventually select a larger number of QSWTs that can accommodate more watermarking bits, and therefore allow watermark redundancies. As described earlier, optimal watermark embedding is directly related to the optimal threshold values  $\alpha_1$  and  $\alpha_2$  used, which ultimately affects the watermark redundancies, total energy consumption and watermark authentication performance. The problem can be described as an Optimisation problem which can be solved using techniques such as genetic algorithms. In [130], a solution based on a genetic algorithm was proposed which takes communication energy consumption based on Eqn. (5.17) as input along with other parameters such as minimum QSWT required for one watermark embedding, packet length, channel factor, distortion reduction factor of watermarked and non-watermarked packets, to compute the optimal threshold pair  $\alpha_1$  and  $\alpha_2$ . This eventually generates the optimal number of QSWTs for single/redundant watermark embeddings. We will discuss energy consumption behavior in Section 5.5.3.

In our experiments, we used  $16 \times 16$  gray scale watermark image/logo which requires almost 256 QSWTs for one complete embedding. The maximum number of QSWTs for Lena, Pepper and Coffee Cup images using threshold value pair ( $\alpha_1 = 25$ ,  $\alpha_2 = 50$ ) are 941, 1125, and 1002 respectively. The total number of QSWTs under varying threshold  $\alpha_1$  and  $\alpha_2$  values is shown in Fig. 5.5. Due to similarity in data patterns, for brevity, only the case for Lena and Coffee Cup images are shown. As discussed previously, the chosen threshold values should be optimal relative to channel conditions, and should be able to accommodate at least one complete watermark. On the other hand, redundant watermark embedding will increase

the complexity of embedding algorithm at source-coding site, and its subsequent impact on energy efficiency will be discussed in the Section 5.5.3.

### 5.5.3 Energy Consumption and Data Rate

Fig. 5.6 shows the total communication energy consumption (based on Eqn. 5.17) of transmitting and receiving sites for the whole image, the encrypted QSWT\_BWE data structure relative to the available data rates under different channel BERs. A unique optimal pair of thresholds has been selected for a given channel BER, which in turn defines the number of watermarks to be embedded. As shown in Fig. 5.6, as the BER increases from  $10^{-5}$  to  $10^{-3}$ , the number of required watermarks defined by the selected optimal threshold pair increases from 1 to 3. Generally, the energy consumption is found to be higher at lower data rates due to a longer transmission time which eventually consumes more transmit and circuit power. As the data rate increases, its effect on packet transmission time becomes diminutive. Therefore, only a slight variation in overall energy consumption has been noticed after channel data rate exceeds 120kbps as shown in Fig.5.6.



**Figure 5.6** - Total communication energy consumption versus data rate

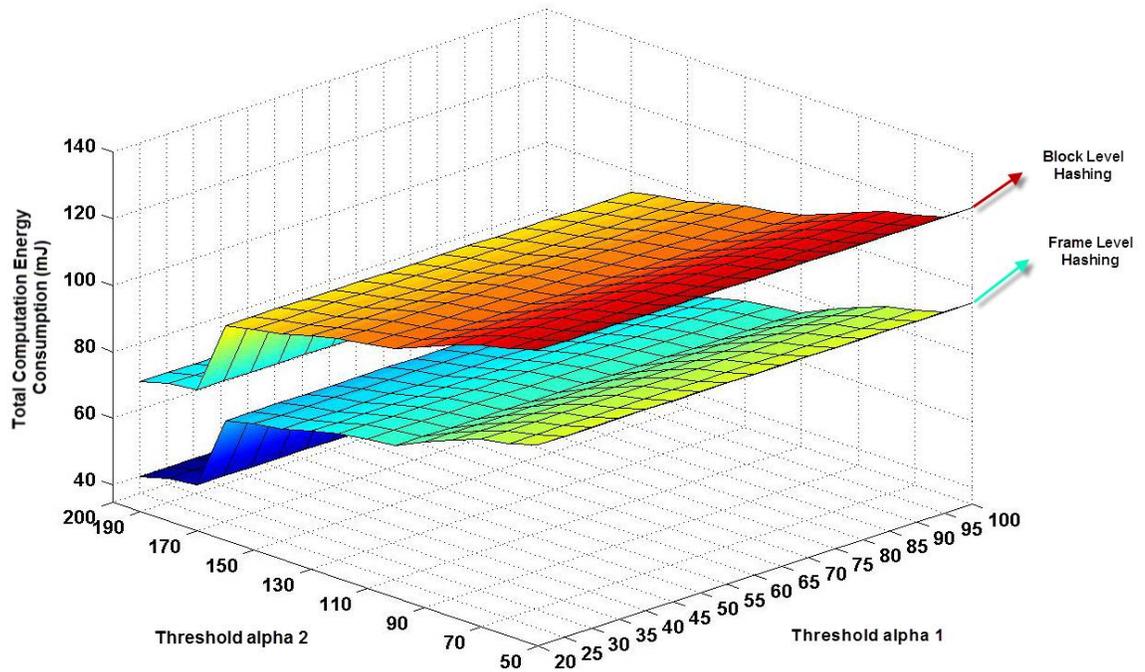
On the other hand, an increase in channel BER, i.e. the rate at which errors occur in a transmission medium, leads to more packet retransmissions, which increase the total energy

consumption. Since the test images yield almost the same statistical pattern being encoded using JPEG library[112] with quality factor 75 , we plotted only the results for the Lena image in Fig. 5.6, Fig.5.7, and Table.5.2.

The computational energy consumption due to the process of encrypting QSWT\_BWE and implementation of the hashing algorithm at block level relative to frame/image level against the threshold pair generated for different BERs is shown in Fig. 5.7. There is also a fixed cost of 154.1mJ associated with the initial setup operations for ECC such as key exchange and handshake protocol at the source nodes [44]. However, this setup cost has not been included in our energy consumption results as the setup operation is not required for every single image. The resulting  $\sigma$  value from the hashing algorithm is used to randomize the effect of watermark embedding from block level to frame level such that it is non-trivial for the attacker to reveal the watermark even if it captures the watermarked image and the corresponding QSWT\_BWE data structure.

Block level hashing is more robust than the frame level hashing but the added security is at the expense of additional complexity and hence of the energy. The choice between the two candidates depends upon the sensitivity of application. For example, if the application is deployed in a hostile environment, one is likely to be more concerned with the authenticity and integrity of data transmission than the additional energy due to block level computation which is still less significant as compared to transmission energy (Table 5.2). With frame/image level hashing, a different  $\sigma$  value is generated for different frame/image that similarly randomizes the effect of embedding the watermark to the cover image and makes it difficult to extract the watermark even if multiple frames/images are captured by the attacker.

The details of each component (computation and communication) that constitutes the total energy consumption at the source coding site to encode, embed redundant watermarks into the cover image, encrypt QSWT\_BWE, and transmit it along with the watermarked image to the base station (or central server) is given in Table 5.2. The computational energy consumption is evaluated by finding the total CPU cycles consumed by the process and multiplying it by the consumed energy of 1.215 nJ per CPU cycle for TelosB mote [18].



**Figure 5.7** - Threshold  $\alpha_1$  vs.  $\alpha_2$  vs. computational energy consumption

Each BER can be supported by a different non-overlapping range of threshold pair values that corresponds to the least number of redundant watermark embeddings. However, the genetic algorithm in [130] selects only one threshold pair in a given range that yields the optimal energy consumption (shaded rows in Table 5.2).

#### 5.5.4 Effect of Compression on Extracted Watermark

We analyze the robustness of the scheme by encoding the image using different quality factors (widely known as Q factor). The Independent JPEG group provides the JPEG library [112] with Q factor ranges from 1 to 100. A value of 1 tends to give a highly compressed but worst quality image, while a value of 100 tends to generate the largest file size but best quality image. The optimal Q factor varies from image to image based on its content; however, the JPEG image library is using a default Q factor value of 75. A single watermark is embedded into the cover image which is encoded at different Q factor values (Table 5.3).

**Table 5.2** - Total energy consumption (computation + communication) corresponding to different BERs at frame- and block-Level hashing

		Frame Level Hashing										Block Level Hashing									
$\alpha 1$	$\alpha 2$	QSWT Trees	Computation			Communication			Total Energy(ml)	Hashing	Computation			Communication			Total Energy(ml)				
			Hashing	Embedding	Encryption	QSWT	Image	Encryption			Embedding	QSWT	Image								
<b>BER <math>10^{-3}</math></b>	25	50	941	0.452	61.24	33.68	150.7864	1064.929	1311.23	28.928	61.24	33.68	150.7864	1064.929	1339.71						
	30	60	850	0.452	57.35	32.02	150.7864	1064.929	1305.55	28.928	57.35	32.02	150.7864	1064.929	1334.03						
	35	70	772	0.452	55.04	31.90	150.7864	1064.929	1303.12	28.928	55.04	31.90	150.7864	1064.929	1331.60						
<b>BER <math>10^{-4}</math></b>	40	80	699	0.452	53.97	31.22	55.02689	565.2762	705.96	28.928	53.97	31.22	55.02689	565.2762	734.43						
	45	90	624	0.452	52.36	29.78	55.02689	565.2762	702.91	28.928	52.36	29.78	55.02689	565.2762	731.38						
	50	100	545	0.452	50.39	28.33	55.02689	565.2762	699.49	28.928	50.39	28.33	55.02689	565.2762	727.96						
<b>BER <math>10^{-5}</math></b>	55	110	474	0.452	48.23	27.15	26.99309	508.3699	611.21	28.928	48.23	27.15	26.99309	508.3699	639.68						
	60	120	431	0.452	47.33	26.49	26.99309	508.3699	609.65	28.928	47.33	26.49	26.99309	508.3699	638.12						
	65	130	397	0.452	46.54	25.97	26.99309	508.3699	608.34	28.928	46.54	25.97	26.99309	508.3699	636.81						
	70	140	347	0.452	45.14	25.24	26.99309	508.3699	606.20	28.928	45.14	25.24	26.99309	508.3699	634.68						
	75	150	397	0.452	44.03	24.13	26.99309	508.3699	603.98	28.928	44.03	24.13	26.99309	508.3699	632.46						
	80	160	278	0.452	43.89	22.89	26.99309	508.3699	602.60	28.928	43.89	22.89	26.99309	508.3699	631.08						
	85	170	258	0.452	43.41	21.55	26.99309	508.3699	600.78	28.928	43.41	21.55	26.99309	508.3699	629.26						
	90	180	227	0.452	42.76	0	0	508.3699	551.58	28.928	42.76	0	0	508.3699	580.06						
	95	190	293	0.452	41.78	0	0	508.3699	550.60	28.928	41.78	0	0	508.3699	579.08						
	100	200	182	0.452	39.54	0	0	508.3699	548.36	28.928	39.54	0	0	508.3699	576.84						

**Table 5.3 - Watermarked and extracted watermark images under different Q factor values.**

Watermarked Image	Q Factor		
	75	65	55
	 NC=0.982	 NC= 0.89	 NC= 0.834
	 NC=0.98	 NC= 0.895	 NC= 0.81
	 NC= 0.974	 NC= 0.846	 NC= 0.783
Average = 0.978			Average = 0.877
Average = 0.978			Average = 0.809

It is evident from Table 5.3 that the scheme exhibits reasonable resistance against compression attacks. Selecting a lower Q factor reduces the image size and therefore the transmission energy. Often sensor board based camera devices are programmed to vary the Q factor of the captured image to increase the node and hence the network lifetime [159].

NC degrades gracefully with increasing compression ratio. Even at a Q factor value of 55, the detection algorithm is still able to extract the watermark with an average NC of 0.809 (Table. 5.3). The detection algorithm rejects the image frame if NC falls below 0.80.

### 5.5.5 Statistical Analysis Attacks

- The scheme to certain extent is safe against *collusion attacks*. Collusion attacks perform statistical analysis on a large number of intercepted watermarked frames to discover the potential watermark embedding locations. However, the given scheme does not embed watermarks at fixed predetermined locations, but rather identifies the embedding locations dynamically, based on thresholds and the image/frame itself. With the same threshold values applied to a different image, the embedding locations would be different. With the same image applied with different threshold values, the embedding locations would again be different. Thus, statistical analysis performed by collusion attacks may not be as useful as for schemes which use fixed predetermined embedding locations for all images/frames.

On the other hand, if an attacker captures both the image and the QSWT\_BWE data structure, it would require strong statistical analysis to figure out the embedding locations. It should be noticed that the data structure does not hold direct information about the QSWT locations but instead its child's location at LH1/LH2 level, and the lower the wavelet level, the harder will be the analysis.

- The scheme also appears to be resistant against *middleman attacks*. Middleman attacks are those in which the attacker injects counterfeit frames into the transmission channel to fool the system. For example, the attacker intercepts an authentic frame and jams the receiver using high-powered and highly directional antenna. Due to jamming, the receiver is unable to receive the authentic frame transmitted by the trusted transmitter. Thereafter, based on statistical analysis of a large number of previously intercepted frames, the attacker copies the bits from the locations assumed to contain the watermark in an intercepted frame and embeds them to the same locations of the counterfeit frame. At the same time, the attacker sends a forged acknowledgement to the trusted transmitter. As a result, this transmitter assumes that its frame transmission is successful. The attacker then transmits the watermarked counterfeit frame to the receiver, which upon reception, extracts the watermark sequence and authenticates the forged frame.

However, for the proposed watermarking scheme, the watermark embedding locations are based on the image/frame itself, and vary with each frame and with different thresholds. Therefore, it is particularly hard to identify the watermark embedding locations even with strong statistical analysis. Even if the embedding locations have been identified by the attacker, and a counterfeited frame injected with the original QSWT\_BWE data structure, the detection algorithm will still generate an incorrect  $\sigma$  value (at frame or block level). This will lead to an incorrect watermark extraction, and the image will be rejected.

## 5.6 Chapter Summary

In this chapter, we proposed a semi-oblivious image watermarking scheme for WMSN which enhances the original non-oblivious version presented in [51]. It uses a low-complexity public key cryptography to encrypt some essential information to be transmitted with the watermarked image to the base-station. It is found that the proposed scheme is robust against compression and has significant capacity to embed watermark redundantly into cover image while maintaining the imperceptibility requirement. The scheme is also found to be reasonably energy efficient under different channel BERs. Lastly, our theoretical analysis shows that the scheme can survive statistical analysis attacks such as collision and middleman attacks.

Following the image watermarking, our next chapter pays attention to the security of video data in a given WMSN environment. In Chapter 6, we discussed a novel, low-complexity, blind, and an imperceptible video watermarking scheme based on DVC for WMSNs, as a primary contribution of this thesis.

# Chapter 6

## A Novel DVC Based Video Watermarking

### 6.1 Introduction

Distributed video coding (DVC) is an emerging video coding paradigm designed to allow low-complexity compression by resource-constrained camera sensor nodes in wireless camera sensor networks (WMSN). Several DVC architectures have already been proposed in literature (as discussed in Chapter 3, Section 3.2) but there are hardly any security mechanisms designed to validate the integrity and authenticity of the decoded video.

In this chapter, a novel, low complexity, blind, and imperceptible watermarking scheme for WMSNs based on Wyner-Ziv (WZ) video coding is presented. The rest of the chapter is organised as follows: Section 6.2 briefly outlines the related work in the field; Section 6.3 presents the proposed watermarking scheme and Section 6.4 describes the experimental setup. Section 6.5 discusses the results obtained and finally Section 6.6 summarises the chapter.

## 6.2 Related Work

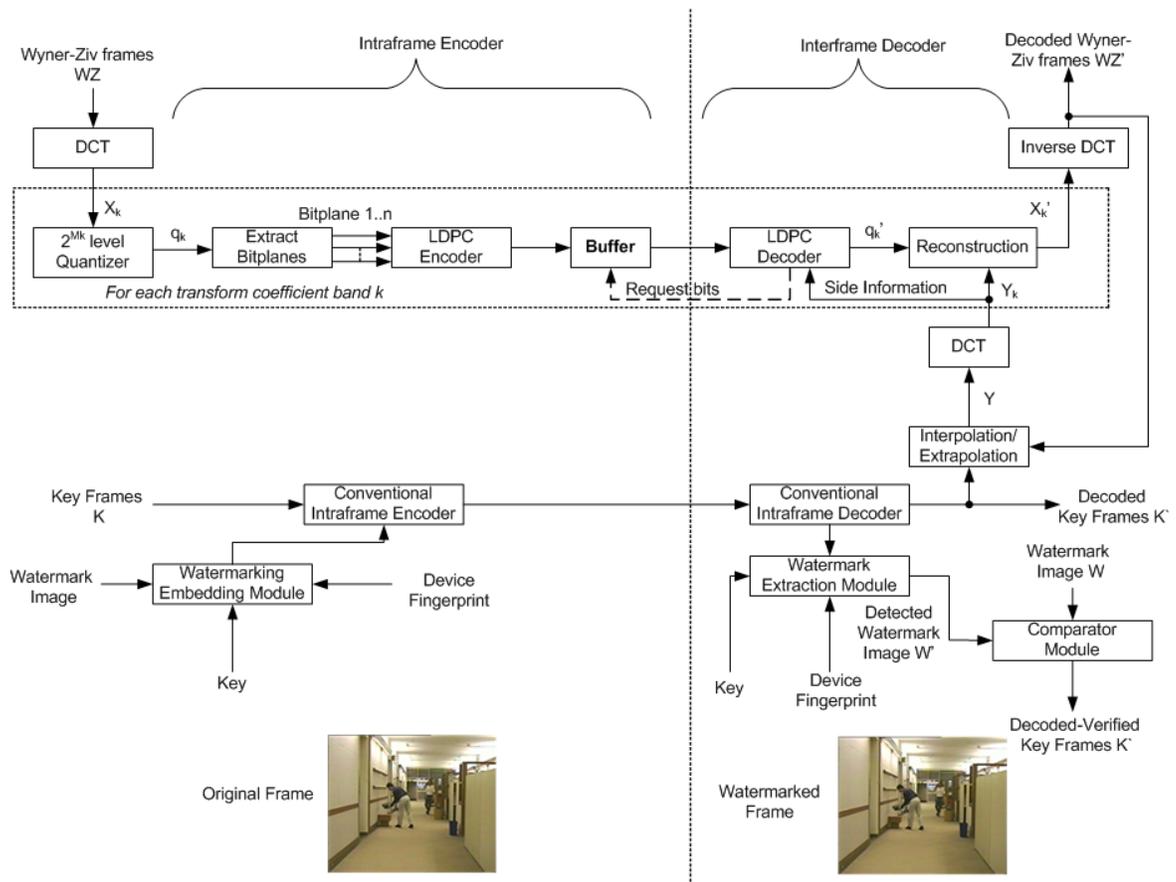
To the best of my knowledge, there exists only a single work by Ning et al., [73] (discussed in Chapter 3, Section 3.2.1, and thus will be used as the benchmarking scheme in this chapter) on a DVC based video watermarking scheme where the watermark is embedded into least significant bits (LSB) of selected discrete cosine transform (DCT) coefficients of each key frame. The scheme uses Arnold transformation [130] and Harris corner detector [171] to scramble the watermark image, and identify the interest points in each key frame for watermark embedding, respectively. Harris corner detector processes each key frame to evaluate the number of interest points which by itself is a complex operation that may cause additional delay in real-time applications. Moreover, the capacity of embedding is dependent on the type of frame itself, while embedding in DCT coefficient increases the bitrate requirement and may cause a drift error in WZ frame reconstruction.

## 6.3 Proposed Scheme

The WZ coding architecture encodes video sequence using a GOP configuration. For a given GOP size  $n$ , every first frame in a GOP is classified as a key frame and compressed via conventional intra-frame encoding, while the remaining  $n - 1$  non-key frames are compressed via WZ encoding. At the decoder site, key frames are reconstructed using conventional intra-frame decoder, whereas decoding of the WZ frames requires side-information generated from both previously decoded key and WZ frames. The WZ video coding architecture along with our proposed watermarking module is presented in Fig.6.1.

### 6.3.1 Key Frame Watermarking Inside Reconstruction Loop of H.264/AVC - Intra Mode

Due to the resource constraints and codec related dependencies, we propose to watermark only the key frames in every GOP configuration. Key frame is the most significant frame, since it provides the basis (side-information) for the rest of non-key (WZ) frames in a given GOP.



**Figure 6.1** - Proposed watermarking framework based on Wyner-Ziv video coding

Operations such as frame interpolation/extrapolation are performed on a decoded key frame along with previously reconstructed WZ frames to produce the side-information for the currently decoding WZ frame. Upon receiving the side-information, block-based DCT is performed and the resulting transform coefficients are aligned to form the coefficient bands, which represent an estimate of each decoded bitplane of the received WZ frame.

The Low Density Parity Check (LDPC) encoder-decoder is being utilized as a Slepian-Wolf codec. Each bit-plane vector is decoded when side-information and residual statistics are available. However, if the decoder cannot decode a bit-plane, it requests additional parity bits from the encoder via feedback channel, and the process continues until a certain acceptable level of bit error rate (BER) performance is achieved. Such an arrangement ensures that the WZ encoder will transmit only a small amount of parity bits to the decoder for reconstruction of quantized bitstream. However, the decoder continues to generate feedback requests until the quantized bitstream is reconstructed with the desired quality parameter. Feedback channel is also used to transmit rate control information to the

encoder, which is justified since the encoder does not need to perform any computation associated with the selection of encoding bit rate. Similar to inter-frames, WZ-frames are highly compressed and have lower payload due to fewer residual statistics. Therefore, embedding the watermark in these frames may adversely affect the coding efficiency as well as the complexity of the watermark embedding module. Hence, for WZ coding, key frame watermarking is a viable and energy efficient solution for uplink application architectures.

The watermarking module is embedded inside the reconstruction loop of Intra prediction module to keep the bitrate intact while streaming over heterogeneous network environments. Data hiding techniques inside the reconstruction loop for H.264/AVC are proposed by Zafar et.al [172]. The watermark embedding module is implemented inside the reconstruction loop next to transformation and quantization operations. Several existing video watermarking techniques embed the watermark in quantized transform coefficients before entropy coding and outside the reconstruction loop using two different approaches as follows:

- The first approach uses VLC domain embedding that requires compressed bitstream to be entropy decoded to embed the watermark.
- In contrast, the second approach embeds the watermark in DCT domain that requires compressed bitstream to be entropy decoded and inversely quantized.

Both of the approaches encounter the following shortfalls:

- Since the watermark embedding is performed outside the reconstruction loop, the reconstruction on the encoder is therefore performed using non-watermarked quantized transform coefficients. Meanwhile, the decoder is performing the reconstruction using watermarked quantized transform coefficients which eventually leads to a mismatch (drift error) between encoder and decoder reconstruction which further escalates due to the prediction process and causes significant distortion in reconstructed signal at decoder site.
- Additionally, the rate-distortion bit allocation operation executes inside the encoder's quantization module, and the mismatch between encoder and decoder reconstructed coefficients has a negative impact on quality versus bitrate trade off.

However, these issues can be resolved if watermark embedding is performed inside the reconstruction loop as in Fig. 6.2. In this case, both encoder and decoder process watermarked quantized transform coefficients in their prediction module which helps in restraining the bitrate meanwhile maintaining the reconstruction quality.

### 6.3.2 Key Frame Watermarking and H.264/AVC - Intra Mode

WZ encoder implementation in [137] employed H.264/AVC Intra mode for key frame encoding. H.264/AVC has three alternative macroblock (MB) settings for Intra prediction mode, namely, *Intra 4×4*, *Intra 16×16* and *I\_PCM*. We utilized *Intra 16×16* mode for our watermarking scheme which uses Hadamard transform to encode discrete cosine (DC) coefficients. Each *16×16* macroblock is decomposed into *4×4* sub-blocks and processed one by one. In this mode, macroblock is predicted from top and left neighbouring pixels and subsequently executes in four modes namely, horizontal, vertical, DC and plane modes. At decoder, a macroblock is reconstructed using the predicted block generated from the previously decoded neighbouring macroblocks identified by the Intra-prediction mode of the current macroblock. Later on, the decoded residual block values are added to the predicted macroblock, resulting in a finally reconstructed macroblock ( $mblk_f$ ), which is given by:

$$mblk_f = mblk_p + mblk_r \quad (6.1)$$

where  $mblk_p$  and  $mblk_r$  represent predicted and residual macroblocks respectively.

We proposed to embed the watermark information only into selective *4×4* sub-blocks of luminance macroblocks by suitably modifying the number of non-zero quantized AC coefficients (NZQAC) with the aim of restraining the bitrate escalation due to watermark embedding. Based on Eqn. (6.1), the difference between watermarked and non-watermarked reconstructed macroblock is denoted by  $\Delta mblk_f$ , which is similar to the difference between the predicted watermarked and non-watermarked macroblocks ( $\Delta mblk_p$ ) and is given by:

$$\Delta mblk_F = \Delta mblk_p \quad (6.2)$$

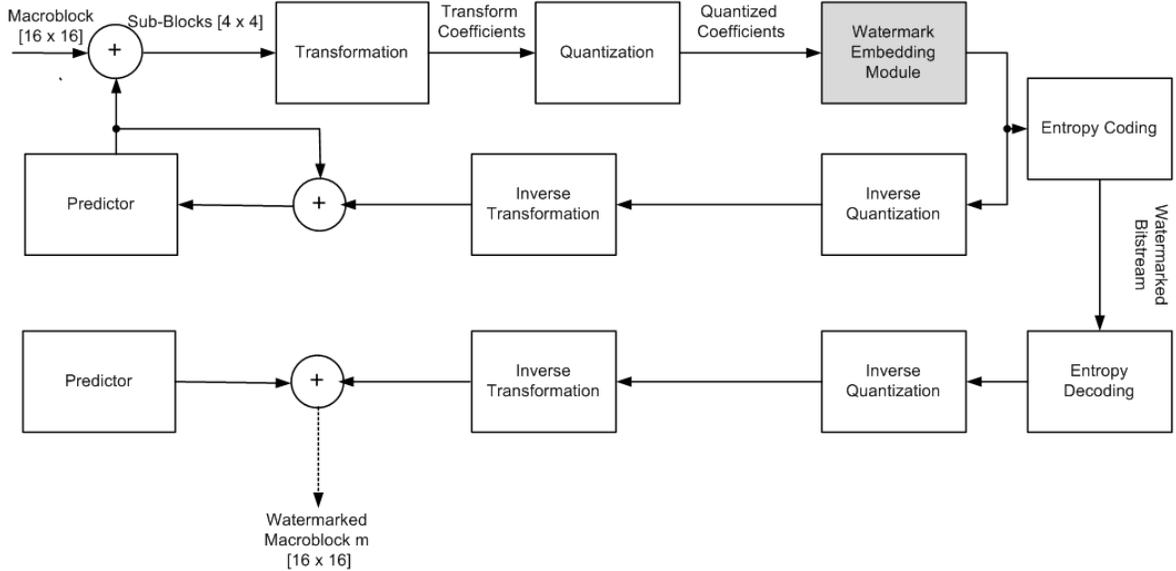
where,

$$\Delta mblk_f = mblk_f^w - mblk_f \quad (6.3)$$

$$\Delta mblk_p = mblk_p^w - mblk_p \quad (6.4)$$

and the  $MB_F^w$  and  $MB_P^w$  denotes the watermarked reconstructed and predicted macroblocks respectively.

Most of the existing video watermarking schemes work on modifying the DCT coefficients for watermark embedding, which not only adversely elevates the bit rate requirement but also degrades the video reconstruction quality. In contrast, working with AC coefficients for watermark embedding may cause only a slight distortion to the signal and minimal impact to the required bitrate. Key frame watermarking framework based on H.264/AVC Intra codec is shown in Fig. 6.2.



**Figure 6.2** - H.264/AVC Intra based key frame watermarking framework

The key frame is the most significant frame, since it provides the basis (side-information) for the rest of non-key (WZ) frames in a given GOP. Operations such as frame interpolation/extrapolation are performed on decoded key frames along with previously reconstructed WZ frames  $X^{N-1}$  to produce side-information  $Y^N$  for the currently decoding WZ frame  $Z_N$ . Upon receiving the side-information, block-based DCT is performed and resulting transform coefficients are aligned to form the coefficient bands, which represent an estimate of each decoded bitplane of the received WZ frame. The distortion function

between  $x^N$  and  $z^N$  with encoder and decoder mappings as  $f(X^N)$ , and  $g(f(X^N), Y^N)$ , respectively, is given by:

$$D(x^N, z^N) = \frac{1}{N} \sum_{n=1}^N D(x_n, z_n) \quad (6.5)$$

where  $x \in X, z \in Z$ , and  $Y^N$  are the watermarked side-information block of length  $N$ . Using mappings  $f$  and  $g$  with  $N$  indices, the average distortion is given by:

$$\Delta = E \left\{ \frac{1}{N} \sum_{n=1}^N D(X_n, Z_n) \right\} \quad (6.6)$$

The optimal rate-distortion pair  $(R, d)$  can be obtained if, for each WZ macroblock:

$$\frac{1}{N} \log(Y^N) \leq R \pm \dot{w}_R, \text{ and } \Delta \leq d \pm \dot{w}_d \quad (6.7)$$

where  $\dot{w}_R$  and  $\dot{w}_d$  denotes the relative change in  $(R, d)$  pair due to watermarked side-information block. Furthermore, the WZ rate-distortion function is given by:

$$R_{WZ}(d) \triangleq \min_{(R,d) \in R_{WZ}} R \quad (6.8)$$

where  $R_{WZ}$  represents the set of all achievable  $(R, d)$  pairs.

Our proposed embedding mechanism considers minimizing the rate of change of feedback requests for WZ macroblock reconstruction, which is a function of  $(R, d)$  pair and the number of embedded watermark bits  $c_k$  in the corresponding key frame macroblock embedding mechanism:

$$\min \Delta \partial_{WZ}(Z_N) \cong \partial_{WZ(R,d)}^{Z_N} \pm \partial_{WZ(\dot{w}_R, \dot{w}_d, c_k)}^{Z_N} \quad (6.9)$$

Odyssey CPU availability model [140] and TelosB per cycle energy count value of  $1.215nJ$  [18] are used to compute the total computational energy consumption using a smoothing filter given as follows:

$$P_{t+1} = \alpha P_t + (1-\alpha)(n_t - p_w) \quad (6.10)$$

To compute  $P$ , Odyssey counts the number of processes at time  $t$  and estimates the number of processes at a future time  $t + 1$  using Eqn.(6.10). Variables  $P, n_t, p_w, \alpha$  represent number of

predicted processes, periodically sampled average CPU load, load consumed by watermarking process, and  $e^{-\frac{t_{pw}}{T}}$  respectively, where  $t_{pw}$  and  $T$  denote the sampling period for the watermarking process and the prediction horizon which considers history data for long-term prediction respectively.

### 6.3.3 Watermark Embedding at Encoder

Details of the watermarking embedding algorithm are as follows:

#### Step 1- Calculating Macroblock Mean Matrix from Camera Fingerprint

Every digital camera has its own distinctive characteristics; even identical units produced by the same manufacturer under the same conditions have their own unique features. This is due to the infinitesimal physical irregularities such as variation among pixels in relation to their sensitivity to light, which is also referred to as Photo Response Non-Uniformity (PRNU) [173]. Every image/frame captured from a particular camera exhibits a distinctive PRNU pattern which represents its unique fingerprint. We assume that such a fingerprint can be captured and placed within the video sensor's internal memory at the time of network installation. The results presented in this paper are based on the fingerprint we computed from our CITRIC smart camera [166][174][174][174][173][173][173][163][173][173].

---

**Input:** *Camera\_Fingerprint* [ ] of dimension (176 × 144)

**Output:** *Macroblock\_Mean* [ ] of dimension (1 × 99)

---

- 1: Decompose *Camera\_Fingerprint*[ ] into 99 macroblocks of dimension (16 × 16).
  - 2: Calculate the mean of each macroblock and store the result in *Macroblock\_Mean* [ ].
- 

#### Step 2 - Evaluating Reference Point from *Macroblock\_Mean*[ ]

Determine the *Reference\_Point* in *Macroblock\_Mean*[ ] which decomposes the matrix into approximately two equal halves based on the  $limit_{upper}$  and  $limit_{lower}$  variables.

---

**Input:** *Macroblock\_Mean*[ ]

**Output:** *Reference\_Point*

---

- 1:  $limit_{upper} = \max (Macroblock\_Mean[ ])$
  - 2:  $limit_{lower} = \min (Macroblock\_Mean[ ])$
  - 3:  $\{limit_{lower}, \dots, i - 1\} < Reference\_Point \leq \{i + 1, \dots, limit_{upper}\}$
-

### Step 3 - Watermark Scrambling

Each pair (sensor node, base-station) is assumed to share a 99-bit secret key. The watermark that needs to be embedded into the key frames is also a 99-bit sequence which can be derived from a secret binary message or a binary logo.

---

**Input:** *Macroblock\_Mean*[ ], *Key* [ ], *Watermark* [ ] all of dimension  $(1 \times 99)$ , and a *Reference\_Point* variable.

**Output:** *Scrambled\_Watermark* [ ] of dimension  $(1 \times 99)$

---

```
1: for each i, where  $i \in \{0,1,2 \dots,98\}$  {
2:     if (Macroblock_Mean[i] > Reference_Point)
3:         Scrambled_Watermark [i] = Key[i]  $\otimes$  Watermark [i]
4:     else
5:         Scrambled_Watermark [i] = Watermark [i]
6: }
```

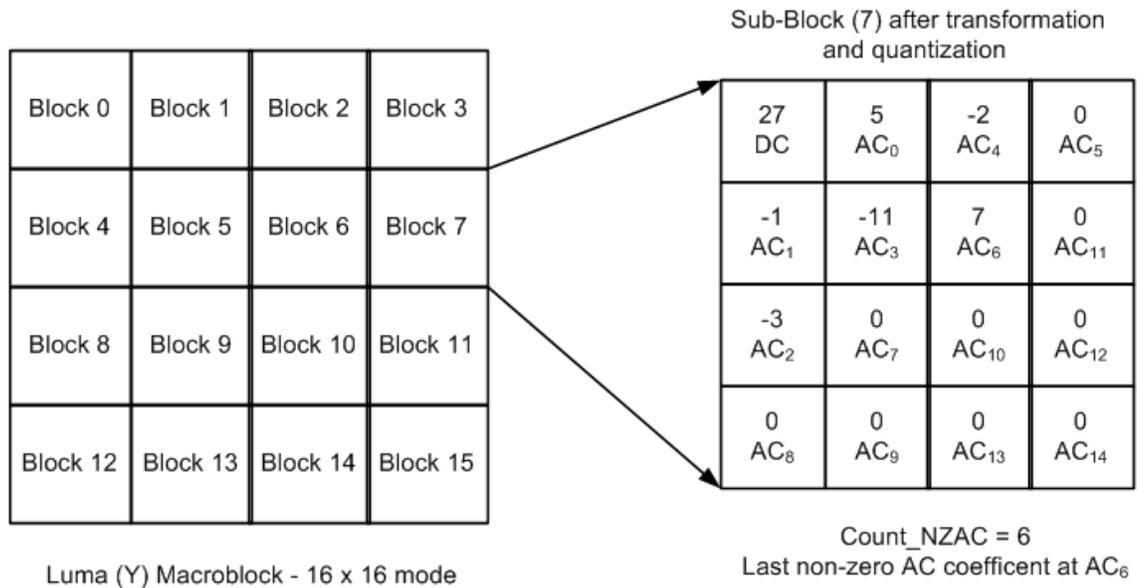
---

### Step 4 - Watermark Embedding

The proposed watermarking scheme embeds the watermark in luminance component of Intra frame after DCT transformation and quantization within the reconstruction loop, which employs  $(16 \times 16)$  macroblock as a basic processing unit. Each  $(16 \times 16)$  macroblock is then further decomposed into  $(4 \times 4)$  sub-blocks. Following the integer DC transformation and quantization, the number of non-zero AC coefficients *Count\_NZAC*[*i*][*j*] in the selected sub-block *j* of macroblock *i* is counted. Thereafter, the last non-zero AC coefficient is marked as *AC<sub>l</sub>*, where *l* is the order of this AC coefficient by Zig-Zag scan (Fig. 6.3). Given the 99-bit *Scrambled\_Watermark*[ ] and 99 macroblocks per frame, each macroblock *i* is embedded with the corresponding *Scrambled\_Watermark*[*ith*] bit, and is able to embed redundant copies of the given watermark bit ranging from  $\{1, \dots, 16\}$ .

The Boolean data structure *Embed\_Watermark*[ ] is utilized to choose a sub-block for embedding in each macroblock and to cause the disparity in the embedding pattern in each key frame. Watermark bit will be embedded in a particular sub-block if and only if the corresponding sub-block entry in *Embed\_Watermark*[ ] is set to 1. If the entire *Embed\_Watermark*[ ] is set to 1, it implies that the same watermark bit will be embedded in all 16 sub-blocks. At least one bit in *Embed\_Watermark*[ ] should be set to 1 in order to

embed one complete watermark in a given key/Intra frame. We employed a simple hashing algorithm at the encoding and decoding nodes to generate a similar *Embed\_Watermark*[ ] array for each frame using seed parameters such as frame number and the number of redundant watermarks to be embedded.



**Figure 6.3** - Luminance component of  $16 \times 16$  macroblock and expanded  $4 \times 4$  sub-block

---

**Input:** *Key\_Frame* [ ], *Scrambled\_Watermark* [ ], and *Embed\_Watermark* [ ] of dimension  $(176 \times 144)$ ,  $(1 \times 99)$ , and  $(1 \times 16)$  respectively.

**Output:** *Watermarked\_Key\_Frame* [ ] of dimension  $(176 \times 144)$

---

```

1: for each macroblock i {
2:     for each subblock j in macroblock i {
3:         if Embed_Watermark [j] is 1 {
4:             Count total number of nonzero AC coefficients and store the
5:             result in Count_NZAC [i][j]
6:             if (Count_NZAC[i][j] is odd) AND (Scrambled_Watermark[i]
7:             is 0)
8:                 Change last nonzero AC coefficient to 1
9:             else
10:                if (Count_NZAC[i][j] is odd) AND (Scrambled_Watermark[i]
11:                is 1)
12:                    Do not change last nonzero AC coefficient
13:                else
14:                    if (Count_NZAC[i][j] is even) AND (Scrambled_Watermark[i]
15:                    is 0)
16:                        Do not change last nonzero AC coefficient

```

---

```

17:         else
18:         if (Count_NZAC[i][j] is even) AND (Scrambled_Watermark[i]
19:         is 1)
20:             Change last nonzero AC coefficient to 1
21:         else
22:         if (Count_NZAC[i][j] is zero) AND (Scrambled_Watermark[i]
23:         is 1)
24:             Change last AC coefficient of the subblock to 1
25:         else
26:         if (Count_NZAC[i][j] is zero) AND (Scrambled_Watermark [i]
27:         is 0)
28:             Do not change last nonzero AC coefficient
29:         }
30:     }
31: }
32: }

```

---

### Step 5 – Hashing for Sub-block Selection

**Input:** Number of redundant watermarks to be embedded *redundant\_wat*, the current frame number *frame\_no*, and the maximum number of embedding corresponding to given macroblock configuration *max\_embed*.

**Output:** A binary array *Embed\_Watermark[ ]* of dimension  $(1 \times 16)$

```

1: Initialize an array Embed_Watermark[ ] of dimension  $(1 \times 16)$  with zero.
2: if ( redundant_wat equals max_embed )
3:     Set entire Embed_Watermark[ ] to 1
4: else
5:     for each i where  $i \in \{ 1..redundant\_wat \}$  {
6:          $index = (i * frame\_no + \sigma) \bmod max\_embed$  ,
7:         where  $\sigma$  is a constant
8:         if (Embed_Watermark [index] is 1)
9:             Look out for first index in Embed_Watermark with Zero
10:            value and set it to 1
11:         else
12:             Set Embed_Watermark[index] to 1
13:         }
14:     }

```

---

### 6.3.4 Watermark Extraction at Decoder

The watermark detection algorithm is similar to the embedding algorithm in that it also performs the steps such as calculating the fingerprint macroblock mean, reference point value for macroblock mean matrix, watermark scrambling, and hashing for *Extract\_Watermark[ ]*. The main difference from the embedding algorithm is that rather

than embedding the watermark and modifying non-zero AC coefficients, the detection algorithm will count the total number of non-zero AC coefficients in a particular sub-block. If it is odd, the extracted watermark bit is 1, or 0 otherwise. The watermark extraction function from a sub-block  $i$  can be expressed as:

$$W'_i = \begin{cases} 1 & \text{if } Count\_NZAC'_i \% 2 \simeq 1 \\ 0 & \text{if } Count\_NZAC'_i \% 2 \simeq 0 \end{cases} \quad (6.11)$$

The watermark bit is also extracted only from those sub-blocks with corresponding value set to 1 in *Extract\_Watermark*[ ] array. The procedure to populate *Extract\_Watermark*[ ] is the same as *Embed\_Watermark*[ ].

The wireless communication is prone to channel errors which inevitably affect various bitstream components such as bit errors in headers and quantized transform coefficients. The quantized transform coefficients will be incorrectly decoded due to an erroneous header. Furthermore, the erroneous transform coefficients lead to incorrect decoding of luminance and chrominance components of the given macroblock which eventually cause the watermark bit/s in that block to be extracted incorrectly. Even though, the bit errors in bitstream header have more severe consequences on the decoding process, the probability of their occurrence is relatively lower than the errors in quantized transform coefficients due to the fact that the major portion of the bitstream is comprised of these coefficients [175]. Also, the bitstream can be maliciously altered to manipulate the integrity of the video content. Our proposed watermarking scheme provides the error/tampering detection capability even at macroblock level with a considerably smaller increase in computational energy at source nodes.

The scheme has a provision to associate Bit error rate (BER) and error/tampering detection capability with the redundancy of an embedded watermark in encoded video, depending upon the given application requirements. For example, the hashing algorithm can be modified to find the optimal watermark redundancy in relation with the channel BER, which expectedly affects the total energy consumption. The problem can be described as an Optimisation problem that can be solved using techniques such as genetic algorithms. In [130], a solution for adaptive watermark embedding based on genetic algorithm was proposed which takes communication energy consumption as input along with other parameters such as packet length, channel factor distortion reduction factor of watermarked packets, to compute the optimal number of embeddings. On the other hand, the proposed

scheme also provides the error/temper detection capability by exploiting the maximum embedding capacity (16 redundant 99-bits watermarks) at the expense of additional computation energy required for embedding redundant watermark bits in each Sub-block of a given macroblock. At decoder, the watermark detection algorithm extracts 16 redundantly embedded watermark bits (one bit from each  $4 \times 4$  sub-block ) from each  $16 \times 16$  macroblock  $mblk_i$ . The distortion in decoded watermarked macroblock at the reciever site can be expressed in Eqn. (6.8), which is closely related to BER requirement that channel exhibits during the transmission of watermarked video and the redundantly embedded watermarks in  $mblk_i$ .

$$Q_{di} = \{ f_{redundant\_wat}(mblk_i), BER \}, \quad \text{where } i = \{ 0, 1, \dots, 98 \} \quad (6.12)$$

$$WAS(mblk_i) = \left\{ \begin{array}{l} 1, \quad \left( 1 - BER^{[wr'_i/wr_i]} \right) \geq 0.98 \\ 0, \quad \text{Otherwise} \end{array} \right\} \quad (6.13)$$

*WAS* in Eqn.(6.13) represents the Watermark Authentication Success for macroblock  $mblk_i$  as a function of channel BER requirement,  $wr'_i$ , and  $wr_i$ . Where  $wr'_i$  and  $wr_i$  denotes the number of redundant watermark bits extracted correctly and the total number of redundant bits embedded in a given macroblock respectively.

## 6.4 Experimental Setup

The proposed watermarking scheme was implemented on the DVC codec from CMLAB [137] and experiments were performed on 150 frames of two video sequences (*CoffeeCup* and *Hillview*) captured from our CITRIC camera platform [8]. The *CoffeeCup* represents an indoor office scenario with objects having sharp corners and edges and some foreground motion, while the *Hillview* represents an outdoor scenario with regions of uniform texture, background motion and low-complexity content. Performance of the proposed scheme in terms of rate-distortion, feedback requests, and computation energy consumption, is compared against that of Ning's scheme [3], which is the closest to ours as mentioned in Section 6.1. Moreover, Ning's scheme originally used WZ codec that exploits the turbo coder for channel coding. However, for consistency in performance analysis, we implemented Ning's scheme on same WZ codec used for the proposed scheme which employs LDPCA channel coder rather than turbo coder.

A 99-bit binary image watermark was embedded into both video sequences. The raw video (YUV) of QCIF resolution was encoded using GOP sizes of 2 and 4 frames at a frame rate of 15 fps. The DVC codec employed JM9.5 reference software to encode key frames in GOP configuration. Key frames were encoded using Intra main profile, with GOP dependent I-frame period, while WZ encoder enabled Intra-mode, skip-block, de-blocking filter under various quantization levels. Each  $16 \times 16$  macroblock in a key frame can carry watermark bits ranging from 1-16, which means the maximum embedding capacity using QCIF will be 1584 bits ( $99 \times 16$ ) per key frame.

The encoder settings used for the key and WZ frames are shown in Table 6.1.

**Table 6.1 - WZ encoder configuration parameters**

H.264 Intra Encoder Configuration for Key Frames		DVC Encoder Configuration for WZ Frames	
Parameter	Value	Parameter	Value
Profile/Level IDC	77 main	Intra Mode	1 (Enable key frame sequence)
Frame Rate	15fps	Frame Rate	15fps
QP	28	QIndex	4
Period of I Frame	GOP Dependent	IntraQP	28
Rate Control	0 (Default)	Quantization parameter of Intra Deblocking filter	1 (Enable)
RD Optimisation	0 (Low Complexity Mode)	Skip Block	1 (Enable)
		GOP Size	2 or 4

In order to maintain consistency in performance analysis of the proposed scheme, the analytical models specified in Chapter 4, Section 4.2 used to compute the encoding and decoding complexity, computational and communication energy, network and node lifetime due to video encoding and watermark embedding. We also evaluated average normalised correlation (NC) between the original watermark and extracted watermarks from two video sequences watermarked at their maximum embedding capacity. In addition, the impact of watermark embedding on average encoding bitrate requirement and PSNR of tested video sequences is presented. All the presented results are using a sequence of 150 frames, unless stated otherwise.

## 6.5 Results and Discussions

### 6.5.1 Capacity and Imperceptibility

Each  $16 \times 16$  macroblock in a key frame is able to carry watermark bits ranging from 1-16, which means the maximum embedding capacity using QCIF resolution will be 1584 (99 macroblocks  $\times$  16 sub-blocks/macroblock) bits per key frame. Instead of embedding a single watermark of 1584 bits, however we have chosen to embed multiple smaller watermarks (collectively comprised of 1584 bits) since radio transmission is more prone to channel errors which may cause a valid frame to be rejected due to a failure in the watermark extraction process.

Fig. 6.4 shows the average normalised correlation (NC) between the original watermark and extracted watermarks from two video sequences. A video frame from each of the video sequences is also shown in Fig. 6.4 watermarked at its maximum embedding capacity while maintaining visual imperceptibility. We gradually increased the watermark redundancy from 2 to 16 and extracted the watermark(s) from the decoded video. The average NC between the original and the extracted watermark in each of the cases is at least 0.97 which seems to outperform the Ning's scheme [73] that came up with least average NC of 0.84. In contrast to Ning's scheme that embeds the watermark into selective interest points around corners and edges within the key frame; our proposed scheme has uniform and consistent embedding capacity irrespective of the content of the frame itself. Furthermore, the watermark cannot be embedded if the entire frame content is comprised of smooth and uniformly textured regions that have no corners or edges inside it. However, the selected video sequences have been chosen carefully, such that the Ning's scheme exhibits similar embedding capacity in key frames as in proposed scheme.

Video Sequences / No. of Embeddings		2	4	8	16
		Average Normalised Correlation (NC)			
Hillview		Proposed Scheme			
		1.0	0.97	0.98	0.97
		Ning's Scheme			
		0.97	0.95	0.91	0.84
CoffeeCup		Proposed Scheme			
		1.0	0.97	0.98	0.98
		Ning's Scheme			
		0.98	0.96	0.93	0.86

**Figure 6.4** - Normalised correlation (NC) and watermarked video frame under maximum embedding capacity

Another primary aspect that needs to be considered is impact of the watermark embedding on video quality. Several video processing applications measure the quality of a video with reference to HVS which is a relatively weak modelling aspect. Structural similarity index (SSIM), an intelligent metric to measure subjective video quality, estimates the visual impact of shifts inside frame's luminance components, changes in contrast along with any other errors collectively termed as structural changes. SSIM is based on the assumption that HVS is specifically adapted for extracting structural information from the video frame and therefore could provide a good approximation of perceived video quality, measuring the structural similarity between the original and the reference frame. For original and watermarked frame signals  $x$  and  $y$ , the SSIM is defined as:

$$SSIM(x, y) = [l(x, y)]^\alpha [c(x, y)]^\beta [s(x, y)]^\gamma , \quad (6.14)$$

where  $\alpha > 0, \beta > 0,$  and  $\gamma > 0$  measure the relative significance of all three index terms representing luminance, contrast and structural components, and are given by:

$$l(x, y) = \frac{2\mu_x\mu_y + C_1}{\mu_x^2 + \mu_y^2 + C_1} , \quad (6.15)$$

$$c(x,y) = \frac{2\sigma_x\sigma_y + C_2}{\sigma_x^2 + \sigma_y^2 + C_2} \quad (6.16)$$

$$s(x,y) = \frac{\sigma_{xy} + C_3}{\sigma_x\sigma_y + C_3} \quad (6.17)$$

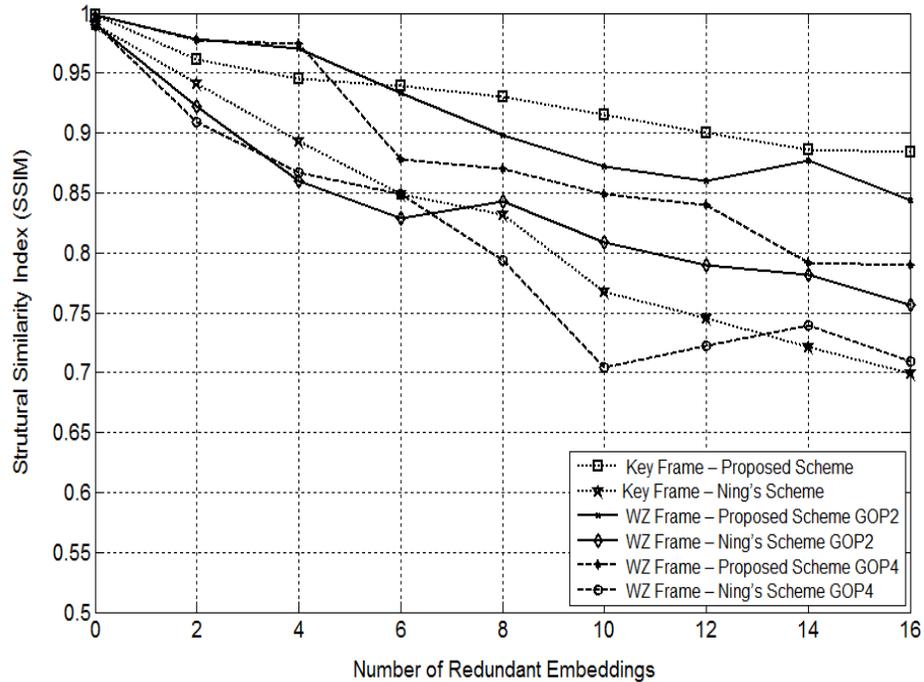
where  $\mu_x$ ,  $\mu_y$ ,  $\sigma_x$ , and  $\sigma_y$  denote the mean and standard deviations of the original and the watermarked frame respectively.  $\sigma_{xy}$  is the co-variance between two frames, and constants  $C_1$ ,  $C_2$ , and  $C_3$  are used to deal with situations where denominators are close to zero.

$$C_1 = (K_1L)^2, \quad (6.18)$$

$$C_2 = (K_2L)^2, \quad (6.19)$$

$$C_3 = C_2/2, \quad (6.20)$$

where  $L$  represents the colour levels,  $K_1 = 0.01$ , and  $K_2 = 0.03$ . The SSIM score in Fig.6.5 uses a scale of 0 – 1, comparing a non-watermarked video against a watermarked one for key as well as WZ frame. Perceptually closer video sequences leads to higher SSIM value.



**Figure 6.5** – SSIM under different redundancy levels - Hillview

**Table 6.2** - Rate-Distortion performance - Hillview

(a) GOP 2

QIndex	No Embedding		99-bit Embedding				1584-bit Embedding			
			Proposed Scheme		<i>Ning's Scheme</i>		Proposed Scheme		<i>Ning's Scheme</i>	
	Bitrate (kbps)	PSNR (dB)	Bitrate (kbps)	PSNR (dB)	Bitrate (kbps)	PSNR (dB)	Bitrate (kbps)	PSNR (dB)	Bitrate (kbps)	PSNR (dB)
Q1	74.54	28.61	75.29	28.40	76.03	27.55	80.33	27.63	82.11	25.93
Q2	93.23	29.71	94.16	29.50	95.09	28.61	98.47	28.70	102.70	26.94
Q3	99.78	31.11	100.78	30.89	101.76	29.96	103.53	30.05	109.92	28.20
Q4	141.56	32.59	138.73	32.36	144.39	31.38	144.28	31.48	155.94	29.55
Q5	165.43	32.85	166.08	32.61	168.74	31.30	169.59	31.73	182.24	29.76
Q6	220.98	32.92	221.19	32.69	225.40	31.37	225.54	31.80	243.43	29.83
Q7	263.61	33.63	262.25	33.39	268.88	32.04	270.24	32.48	290.39	30.47
Q8	414.92	35.16	415.07	34.91	423.22	33.50	419.36	33.96	457.08	31.86

(b) GOP 4

QIndex	No Embedding		99-bit Embedding				1584-bit Embedding			
			Proposed Scheme		<i>Ning's Scheme</i>		Proposed Scheme		<i>Ning's Scheme</i>	
	Bitrate (kbps)	PSNR (dB)	Bitrate (kbps)	PSNR (dB)	Bitrate (kbps)	PSNR (dB)	Bitrate (kbps)	PSNR (dB)	Bitrate (kbps)	PSNR (dB)
Q1	50.54	27.95	51.55	27.73	52.58	27.42	52.88	26.62	57.65	25.54
Q2	67.44	28.13	67.79	27.91	69.16	27.60	70.16	26.80	78.89	25.71
Q3	75.03	28.87	75.53	28.65	76.06	28.34	77.04	27.52	85.06	26.41
Q4	115.62	30.11	115.93	29.89	116.89	29.58	118.25	28.72	124.49	27.59
Q5	133.78	30.24	133.46	30.02	134.18	29.71	136.12	28.85	146.47	27.71
Q6	181.45	31.55	182.08	31.33	183.78	31.02	185.72	30.12	201.02	28.96
Q7	240.89	32.16	240.17	31.94	242.62	31.63	244.54	30.71	257.18	29.54
Q8	397.01	34.63	397.95	34.41	399.05	34.10	405.32	33.11	424.76	31.88

It is evident from Fig.6.5 that the proposed scheme overall maintains a better perceptual quality than Ning's scheme for key frames and WZ frames under both GOP configurations.

### 6.5.2 Rate Distortion Performance

The rate-distortion performance by Ning et al. [73] and the proposed scheme under various capacity utilization levels using GOP 2 and GOP 4 configuration is shown in Table 6.2. The result shows that the proposed scheme caused an acceptable change in encoding bitrate and PSNR of the tested video sequence in contrast to [73]. This is due to the fact that the

embedding algorithm adaptively truncates the most insignificant NZQAC coefficient from the  $4 \times 4$  Sub-block (only if required), which consequently minimizes its effect on both evaluated parameters. Moreover, the watermarking of the key frame is performed within the reconstruction loop of H.264 Intra [172]. Therefore, the distortion introduced by the watermark bits can be estimated by successive macroblock predictions, which eventually reduce the drift error. In contrast, Ning's scheme [73] works on embedding the entire watermark in LSBs of *DCT coefficients* of the selected interest regions, which elevates the bitrate as well as distortion as the capacity gradually increases.

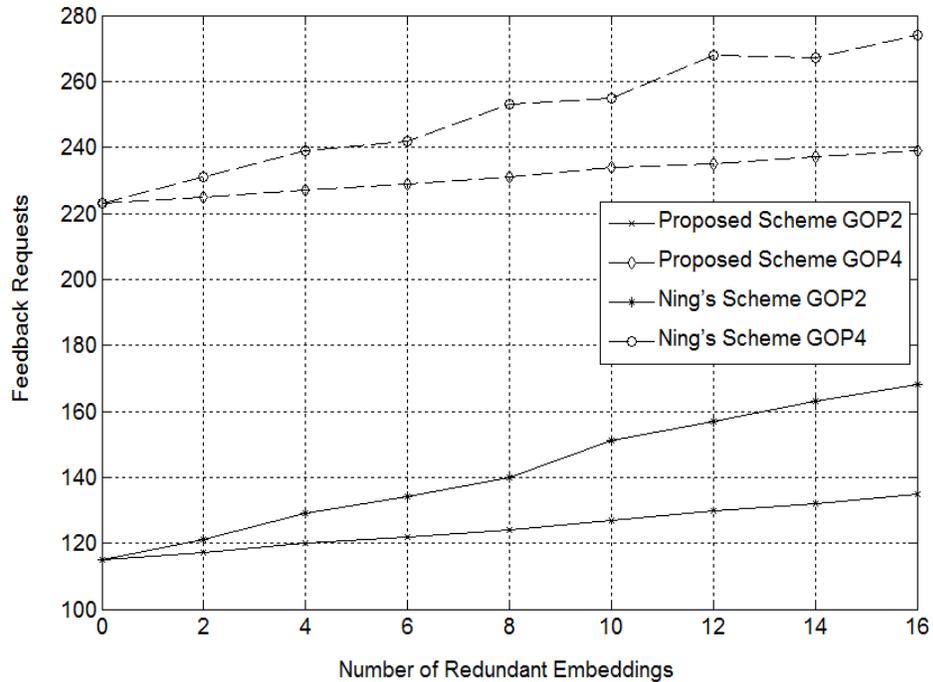
### **6.5.3 Impact of Watermarking on Feedback Channel**

In this section, we examine the impact of watermark embedding on Wyner-Ziv's feedback channel. The LDPCA decoder receives successive chunks of parity bits from the encoder following the requests made through the feedback channel. The decoder continues to generate feedback requests until the quantized bitstream has been reconstructed with the desired quality parameter. It is important to measure the impact of embedding on WZ-frame reconstruction, since the side-information for the WZ-frame to be decoded is constructed based on the key frame, and our scheme only embeds the watermark into the key frame. Therefore, the number of feedback requests generated from the decoder with or without watermark embedding helps us to measure the effect of key-frame watermarking on WZ-frame reconstruction.

Since key frame forms the basis of WZ reconstruction, changes to DCT coefficients also lead to an increase in feedback requests sent due to mismatch between estimated and currently decoding WZ frame as shown in Fig. 6.6. This phenomenon presents a trade-off between robustness of the watermarking and rate-distortion performance, as well as the delay introduced due to excessive feedback requests in WZ frame reconstruction [73].

### **6.5.4 Encoding and Decoding Complexity**

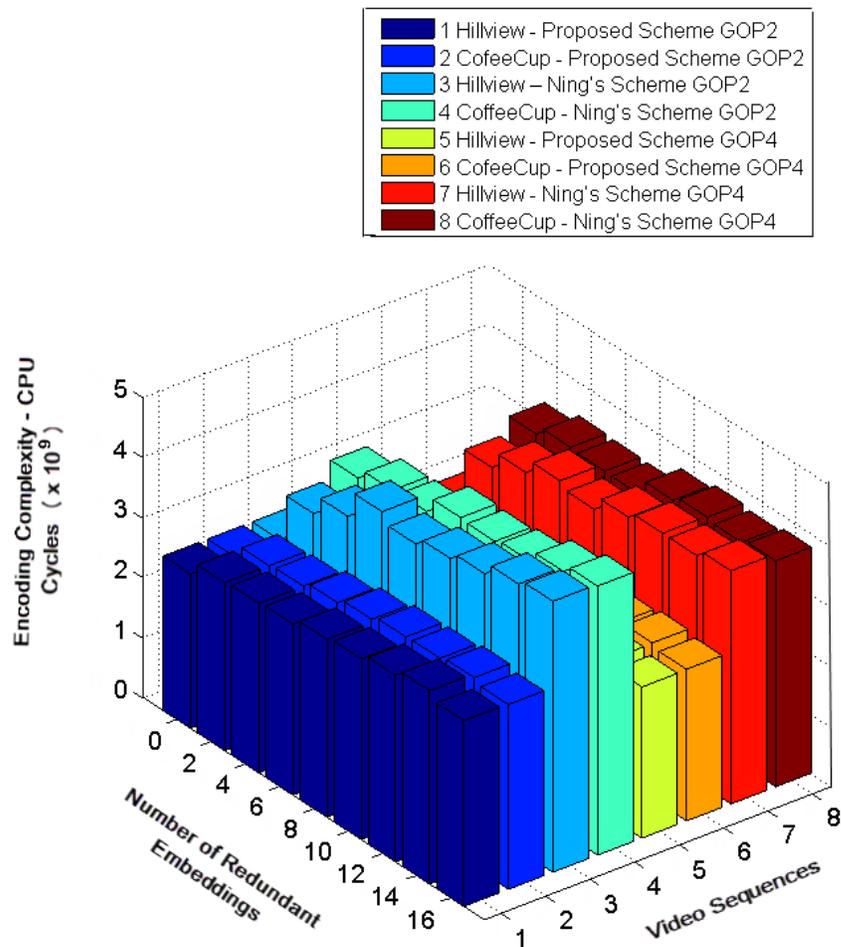
The computational load on resource-limited sensor nodes is considered as a primary design issue not only for the video coding schemes but also for designing security mechanisms in a WMSN environment. The encoder complexity includes key frame and WZ coding as two primary components.



**Figure 6.6** - Feedback requests under different redundancy levels - Hillview

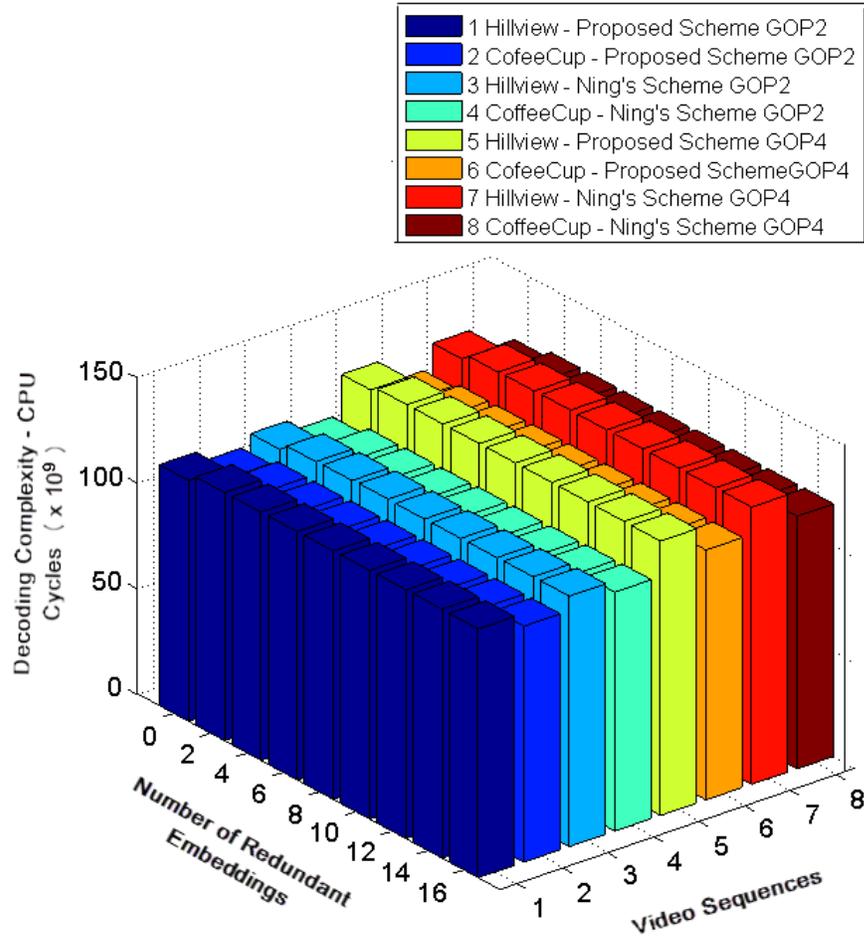
The higher the GOP, the less the key frames, and therefore lowers the complexity share of key frames. We evaluated the encoding complexity of WZ codec for the test video sequences with and without watermark embedding under GOP 2 and GOP 4 configurations for both schemes. In Fig. 6.7, the encoding complexity associated with zero redundant watermarks represents encoding-only computations, while that for sixteen redundant watermarks represent encoding computations along with embedding under maximum capacity utilization. It can be observed that video sequences with GOP 2 configuration have a higher complexity than GOP 4 due to the alternate key frame encoding. It can be seen that the encoding complexity elevates with increasing redundancy in watermark embedding as expected. However, there is a significant performance gap observed in the proposed scheme relative to Ning's work. Under maximum capacity utilization (1584 bits), the added complexity share due to watermarking is approximately 17% and 33% for GOP size 2, and 22% and 43% for GOP size 4, by the proposed and Ning's scheme respectively. This is due to the fact that Ning's scheme processes each key frame individually, performs Arnold's transformation of watermark, look for the suitable and sufficient number of corners and edges for watermark embedding, which consumes a substantial amount of CPU cycles that varies depending on the content of frame itself. For the proposed scheme, a slight drop in the encoding complexity has been witnessed for all the video sequences for both GOP

configurations with maximum embedding. This could be due to the fact that no iterations for the hashing algorithm need to be executed in order to determine the subblocks for watermarking bit embedding.



**Figure 6.7** - Encoding complexity versus embedding capacity under GOP 2 and 4 configurations

As with encoder complexity, the major components that conclude the decoder complexity are key and WZ frame decoding. In contrast to encoding complexity, the impact of watermark detection on the decoding algorithm with varying watermarking payload for both GOP configurations is relatively insignificant, as shown in Fig. 6.8. The computations for embedding and detection algorithm are more or less the same but this behaviour is due to the heavy decoder which makes the additional computations for the detection algorithm relatively insignificant with reference to the video decoding. However, the proposed scheme still outperforms the Ning's decoding (including watermark detection) complexity.



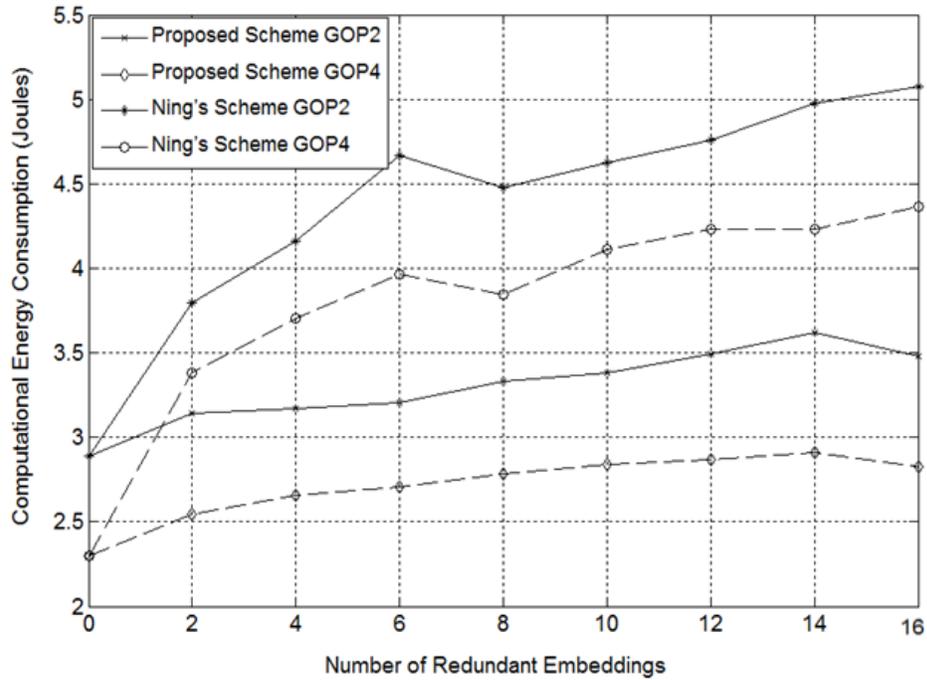
**Figure 6.8** - Decoding complexity versus embedding capacity under GOP 2 and 4 configurations

### 6.5.5 Energy Consumption

Energy-efficiency is one of the primary concerns for the battery-operated camera sensor nodes that affect not only the lifetime of the node itself but also the lifetime of the entire network. Therefore, it is considered a key design issue for the development of security mechanisms in the WMSN environment that exhibits a trade off between security, computation and communication energy consumption. In this section, in order to appreciate the practical effectiveness of the proposed scheme, the energy cost of computation (encoding, embedding), and communication has been presented.

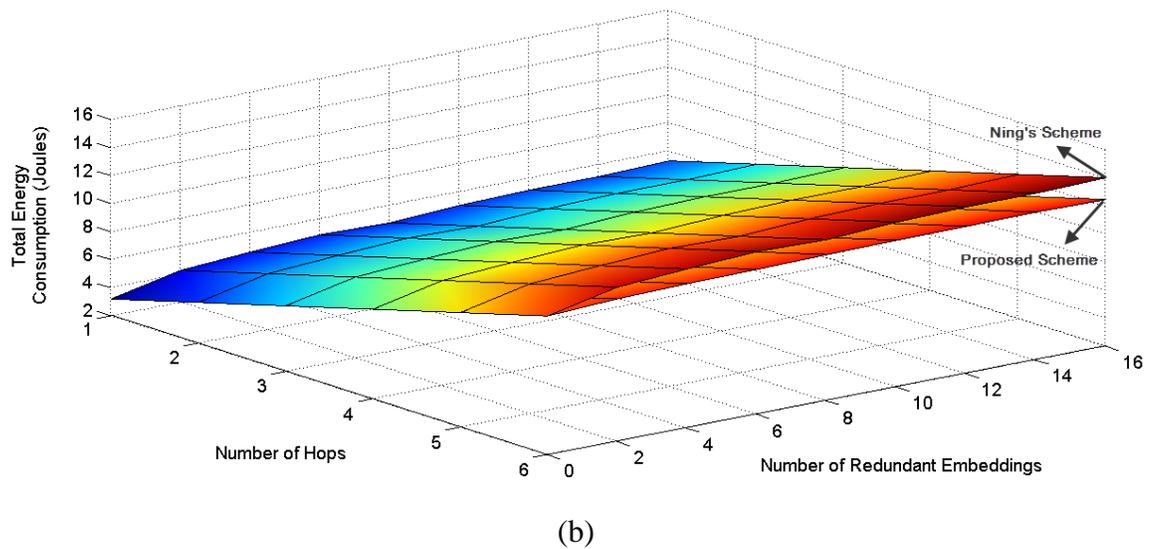
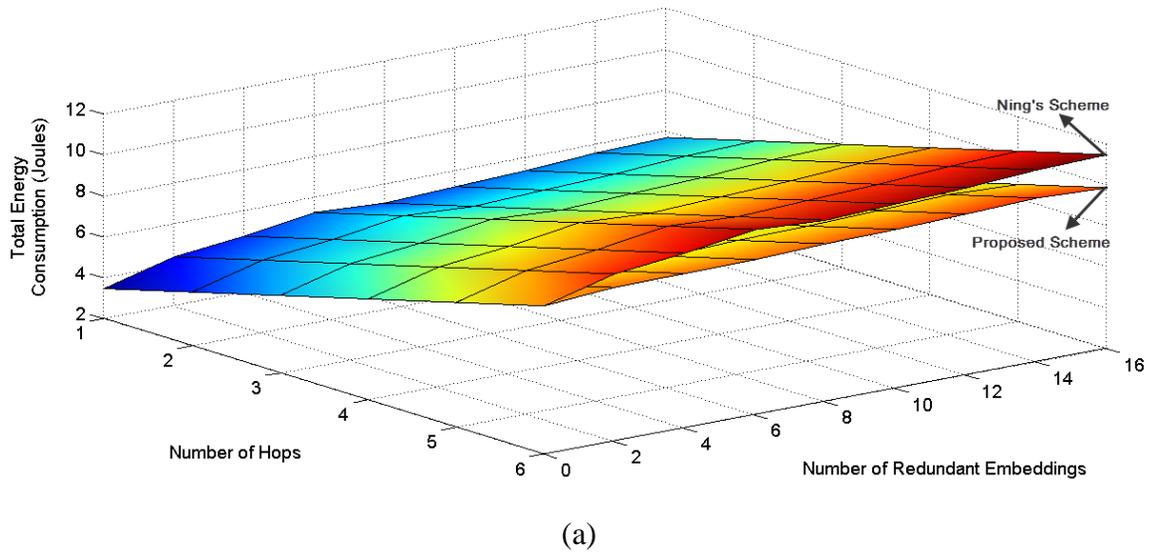
The computational energy consumption under various redundancy levels is shown in Fig. 6.9. It can be seen that the proposed scheme with a simple embedding algorithm exhibits a more energy-efficient behaviour than [73] for both GOP configurations. This can be attributed to

the embedding mechanism being dependent on subblocks selected by the hashing algorithm rather than the contents of video frame itself. On the other hand, in [73], every key frame undergoes Harris corner and edge detection, and from the resulting interest regions, a few are selected to embed the entire watermark following the criterion value and embedding requirement.



**Figure 6.9** - Computational energy consumption (encoding + embedding) under different redundancy levels

Fig. 6.10a and 6.10b show the total energy consumption for the Hillview sequence in a multi-hop WMSN scenario that includes energy consumed due to encoding, watermark embedding, and communication of watermarked bitstream from source to sink over multiple hops using GOP size 2 and 4 respectively. For maintaining the consistency, we employed the same computation, communication energy and channel models used in Chapter 4. The energy consumption results from the Hillview sequence are only provided to avoid overlapping results. The breakdown of the factors contributing to total energy consumption in Fig. 6.10 is shown in Table-6.3.



**Figure 6.10-** Overall energy consumption (computation + communication) under different redundancy levels using: (a) GOP 2 (b) GOP 4 – Hillview

The general behaviour of both schemes is observed as follows: For single or double hop of transmission; GOP size 4 exhibits a better energy performance than GOP size 2 for both of the watermarking schemes. This is due to the lower computational energy which is found to be a dominant factor in the overall energy consumption for single or double hop of transmission. However, as the number of hops increases, the energy consumption due to communication dominates over the computational share resulting from the increased feedback channel requests and response overhead with a higher GOP size, which accumulates when transmitted over multiple hops. Although the energy consumed due to watermark embedding in GOP 4 is relatively lower than the counterpart GOP 2 due to

lesser number of key frames, the corresponding impact on total energy consumption is not significant because the communication energy leads in the overall energy consumption share.

Table 6.3a and 6.3b depict the energy breakdown under GOP size 2, while 6.3c and 6.3d under GOP size 4, for the proposed and Ning's scheme respectively. Energy consumption due to encoding and embedding remains almost the same over multi-hops while communication varies. Energy consumption due to embedding changes by varying the redundancy of watermarks inside key frames. However, it is evident from Fig. 6.10 and Table 6.3 that the proposed scheme outperforms the Ning's watermarking in almost every aspect from computation to communication. We have already discussed the computational energy behaviour in the start of the Section 6.5.5. However, regarding communication, it can be seen from Table 6.3 that even with same number of encoded frames, the communication energy consumption for the proposed scheme and Ning's scheme is different with the same number of embedded watermarks. This is because Ning's scheme works on embedding the watermark in LSBs of *DCT coefficients* of the selected interest regions, which increases the number of bits to encode a symbol (DCT coefficient) and gradually reduce the compression ratio which worsens with several redundant embeddings.

**Table 6.3 a** – Overall energy consumption for Hillview sequence using proposed scheme under GOP 2 configuration

Hillview - GOP 2 Proposed Scheme																		
Hops/ Watermarks	1			2			3			4			5			6		
	ENC	EMB	COMM															
<b>0</b>	2.89	0	0.548	2.89	0	1.703	2.89	0	2.858	2.89	0	4.012	2.89	0	5.167	2.89	0	6.322
<b>2</b>	2.89	0.255	0.548	2.89	0.255	1.703	2.89	0.255	2.855	2.89	0.255	4.013	2.89	0.255	5.168	2.89	0.255	6.323
<b>4</b>	2.89	0.282	0.549	2.89	0.282	1.703	2.89	0.282	2.858	2.89	0.282	4.013	2.89	0.282	5.168	2.89	0.282	6.323
<b>6</b>	2.89	0.314	0.548	2.89	0.314	1.703	2.89	0.314	2.858	2.89	0.314	4.013	2.89	0.314	5.168	2.89	0.314	6.323
<b>8</b>	2.89	0.443	0.548	2.89	0.443	1.703	2.89	0.443	2.858	2.89	0.443	4.013	2.89	0.443	5.168	2.89	0.443	6.323
<b>10</b>	2.89	0.494	0.548	2.89	0.494	1.703	2.89	0.494	2.858	2.89	0.494	4.013	2.89	0.494	5.168	2.89	0.494	6.324
<b>12</b>	2.89	0.602	0.548	2.89	0.602	1.703	2.89	0.602	2.858	2.89	0.602	4.014	2.89	0.602	5.169	2.89	0.602	6.324
<b>14</b>	2.89	0.732	0.549	2.89	0.732	1.704	2.89	0.732	2.859	2.89	0.732	4.014	2.89	0.732	5.170	2.89	0.732	6.325
<b>16</b>	2.89	0.586	0.550	2.89	0.586	1.703	2.89	0.586	2.859	2.89	0.586	4.014	2.89	0.586	5.169	2.89	0.586	6.325

\*ENC – Encoding

\*EMB - Embedding

\*COMM - Communication

**Table 6.3 b** – Overall energy consumption for Hillview sequence using Ning’s scheme under GOP 2 configuration

Hops/ Watermarks		Hillview - GOP 2 Ning’s Scheme																	
		1			2			3			4			5			6		
		ENC	EMB	COMM	ENC	EMB	COMM	ENC	EMB	COMM	ENC	EMB	COMM	ENC	EMB	COMM	ENC	EMB	COMM
<b>0</b>	2.89	0	0.548	2.89	0	1.703	2.89	0	2.858	2.89	0	4.012	2.89	0	5.167	2.89	0	6.322	
<b>2</b>	2.89	0.906	0.548	2.89	0.906	1.704	2.89	0.906	2.859	2.89	0.906	4.014	2.89	0.906	5.171	2.89	0.906	6.325	
<b>4</b>	2.89	1.269	0.550	2.89	1.269	1.706	2.89	1.269	2.861	2.89	1.269	4.015	2.89	1.269	5.172	2.89	1.269	6.327	
<b>6</b>	2.89	1.775	0.552	2.89	1.775	1.707	2.89	1.775	2.862	2.89	1.775	4.017	2.89	1.775	5.175	2.89	1.775	6.327	
<b>8</b>	2.89	1.584	0.557	2.89	1.584	1.711	2.89	1.584	2.865	2.89	1.584	4.019	2.89	1.584	5.178	2.89	1.584	6.329	
<b>10</b>	2.89	1.733	0.558	2.89	1.733	1.713	2.89	1.733	2.867	2.89	1.733	4.023	2.89	1.733	5.181	2.89	1.733	6.332	
<b>12</b>	2.89	1.866	0.560	2.89	1.866	1.713	2.89	1.866	2.871	2.89	1.866	4.024	2.89	1.866	5.183	2.89	1.866	6.335	
<b>14</b>	2.89	2.088	0.561	2.89	2.088	1.715	2.89	2.088	2.872	2.89	2.088	4.027	2.89	2.088	5.188	2.89	2.088	6.337	
<b>16</b>	2.89	2.183	0.562	2.89	2.183	1.716	2.89	2.183	2.874	2.89	2.183	4.030	2.89	2.183	5.190	2.89	2.183	6.340	

\*ENC - Encoding

\*ENB - Embedding

\*COMM - Communication

**Table 6.3 c – Overall energy consumption for Hillview sequence using proposed scheme under GOP 4 configuration**

Hillview - GOP 4 Proposed Scheme																		
Hops/ Watermarks	1			2			3			4			5			6		
	ENC	EMB	COMM															
<b>0</b>	2.30	0	0.829	2.30	0	2.578	2.30	0	4.321	2.30	0	6.067	2.30	0	7.813	2.30	0	9.560
<b>2</b>	2.30	0.247	0.829	2.30	0.247	2.578	2.30	0.247	4.322	2.30	0.247	6.068	2.30	0.247	7.814	2.30	0.247	9.560
<b>4</b>	2.30	0.357	0.829	2.30	0.357	2.578	2.30	0.357	4.321	2.30	0.357	6.067	2.30	0.357	7.814	2.30	0.357	9.560
<b>6</b>	2.30	0.405	0.829	2.30	0.405	2.576	2.30	0.405	4.322	2.30	0.405	6.068	2.30	0.405	7.815	2.30	0.405	9.561
<b>8</b>	2.30	0.487	0.829	2.30	0.487	2.576	2.30	0.487	4.322	2.30	0.487	6.068	2.30	0.487	7.815	2.30	0.487	9.561
<b>10</b>	2.30	0.542	0.829	2.31	0.542	2.575	2.30	0.542	4.322	2.30	0.542	6.068	2.30	0.542	7.815	2.30	0.542	9.561
<b>12</b>	2.30	0.568	0.829	2.31	0.568	2.575	2.30	0.568	4.322	2.30	0.568	6.068	2.30	0.568	7.815	2.30	0.568	9.561
<b>14</b>	2.30	0.608	0.829	2.31	0.608	2.575	2.30	0.608	4.322	2.30	0.608	6.068	2.30	0.608	7.815	2.30	0.608	9.562
<b>16</b>	2.30	0.528	0.829	2.32	0.528	2.578	2.30	0.528	4.323	2.30	0.528	6.069	2.30	0.528	7.816	2.30	0.528	9.562

\*ENC - Encoding

\*ENB - Embedding

\*COMM - Communication

**Table 6.3 d** – Overall energy consumption for Hillview sequence using Ning’s scheme under GOP 4 configuration

Hillview - GOP 4 Ning’s Scheme																		
Hops/ Watermarks	1			2			3			4			5			6		
	ENC	EMB	COMM															
<b>0</b>	2.30	0	0.829	2.30	0	2.578	2.30	0	4.321	2.30	0	6.067	2.30	0	7.813	2.30	0	9.560
<b>2</b>	2.30	1.079	0.829	2.30	1.079	2.579	2.30	1.079	4.322	2.30	1.079	6.068	2.30	1.079	7.814	2.30	1.079	9.563
<b>4</b>	2.30	1.402	0.833	2.30	1.402	2.582	2.30	1.402	4.322	2.30	1.402	6.068	2.30	1.402	7.814	2.30	1.402	9.565
<b>6</b>	2.30	1.665	0.836	2.30	1.665	2.584	2.30	1.665	4.325	2.30	1.665	6.070	2.30	1.665	7.815	2.30	1.665	9.569
<b>8</b>	2.30	1.547	0.839	2.30	1.547	2.584	2.30	1.547	4.327	2.30	1.547	6.073	2.30	1.547	7.820	2.30	1.547	9.571
<b>10</b>	2.30	1.814	0.840	2.30	1.814	2.586	2.30	1.814	4.330	2.30	1.814	6.074	2.30	1.814	7.822	2.30	1.814	9.575
<b>12</b>	2.30	1.933	0.842	2.30	1.933	2.588	2.30	1.933	4.333	2.30	1.933	6.077	2.30	1.933	7.824	2.30	1.933	9.579
<b>14</b>	2.30	1.932	0.845	2.30	1.932	2.589	2.30	1.932	4.335	2.30	1.932	6.082	2.30	1.932	7.828	2.30	1.932	9.582
<b>16</b>	2.30	2.063	0.847	2.30	2.063	2.593	2.30	2.063	4.339	2.30	2.063	6.089	2.30	2.063	7.832	2.30	2.063	9.582

\*ENC - Encoding

\*EMB - Embedding

\*COMM - Communication

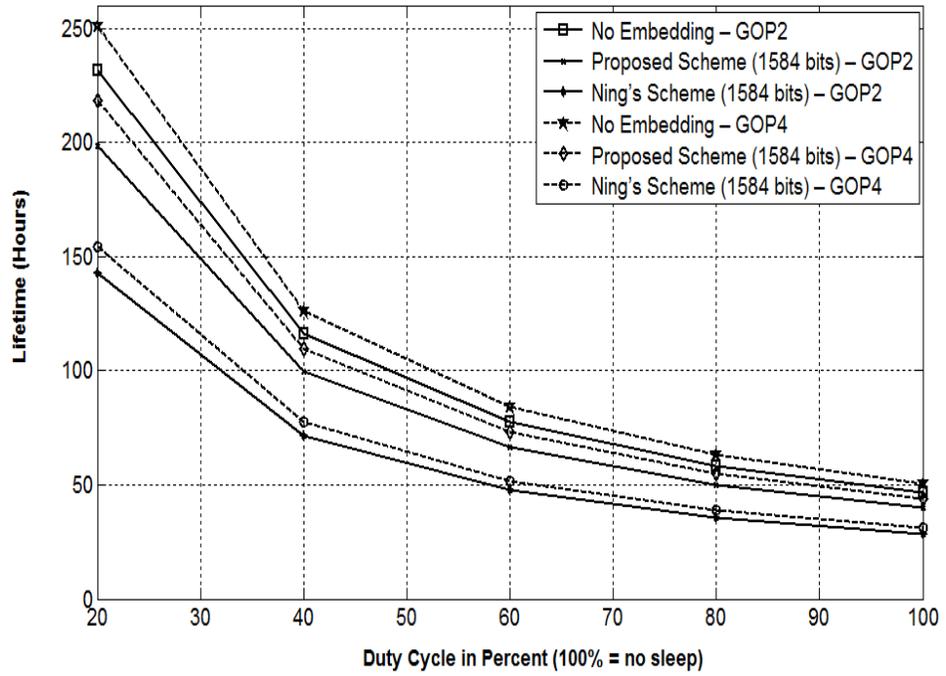
### 6.5.6 Node and Networks Lifetime

We have evaluated the node and network lifetime for a given WMSN using a TelosB platform with and without incorporating a watermarking scheme. Different application scenarios have been addressed ranging from continuous operation (100% duty cycle) to low-power (low duty cycle) operation modes where nodes switch between active and dormant (sleep) states. The lifetimes for five different duty cycles, 20%, 40%, 60%, 80%, and 100%, have been computed as shown in Fig. 6.11.

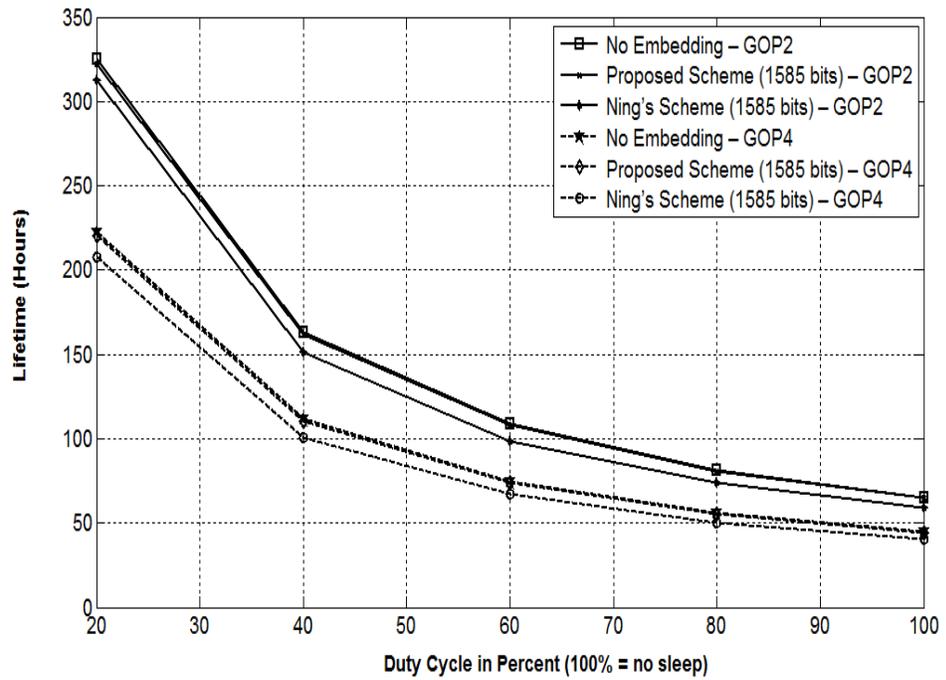
From Fig. 6.11a, it is clear that the source node has the shortest lifetime due to its high encoding and embedding complexity that contributes significantly to the node's energy requirement. These results have been computed based on the encoding and communication of bitstream with no watermark and sixteen redundantly embedded watermarks. For Ning's scheme, with maximum capacity utilization, under 100% duty cycle, the lifetime of the source node for GOP 2 and GOP 4 is approx. 28.5% and 31.01% hours respectively, which increases to 142.5 and 154.39 hours when the duty cycle is decreased to 20%. On the other hand, with similar embedding, under 100% duty cycle, the lifetime of the source node for the proposed scheme is approx. 39.8 and 43.8 hours, which increases to 198.31 and 218.25 hours when duty cycle is decreased to 20% for GOP size 2 and 4 respectively.

There is almost un-noticeable impact of watermarking on the lifetime of relay nodes for the proposed scheme, since the embedding mechanism tries to restrain the encoding bitrate as much as possible and does not significantly affect the number of encoded bits (Fig. 6.11b). For the proposed scheme, as the duty cycle decreases from 100% to 20%, the lifetime of the relay node for GOP 2 and GOP 4 increases from approximately 64.77 to 321.97 hours and 44.30 to 220.61 hours respectively. While for Ning's scheme, a duty cycle decrease from 100% to 20% accumulates the relay lifetime for GOP 2 and GOP 4 from approx. 59.23 to 312.07 hours and 40.40 to 207.71 hours respectively.

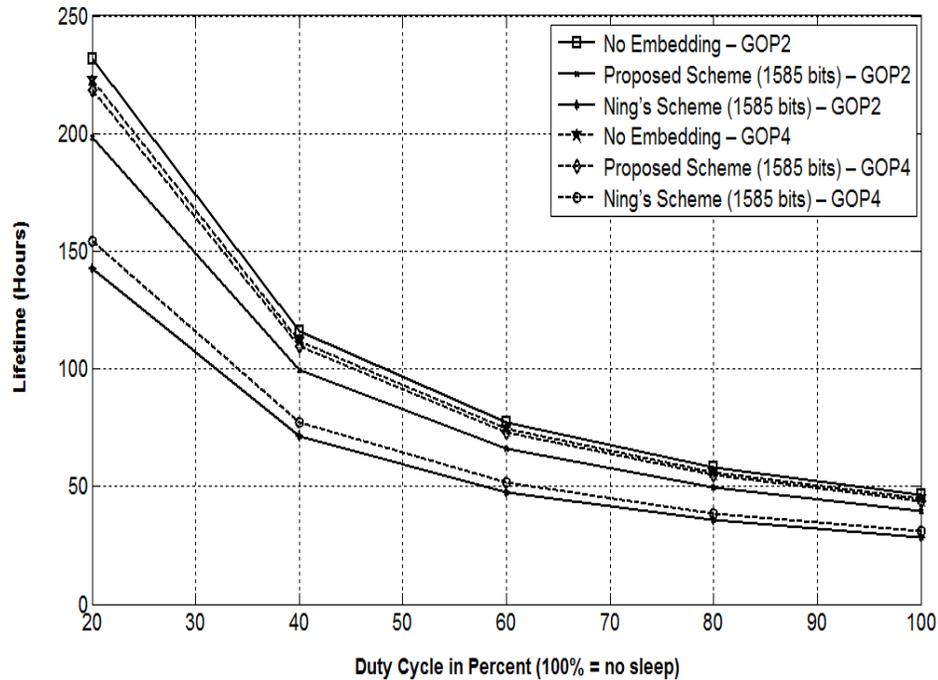
The network lifetime, which considers the shorter of the source and relay node lifetimes is shown in Fig. 6.11c.



(a)



(b)



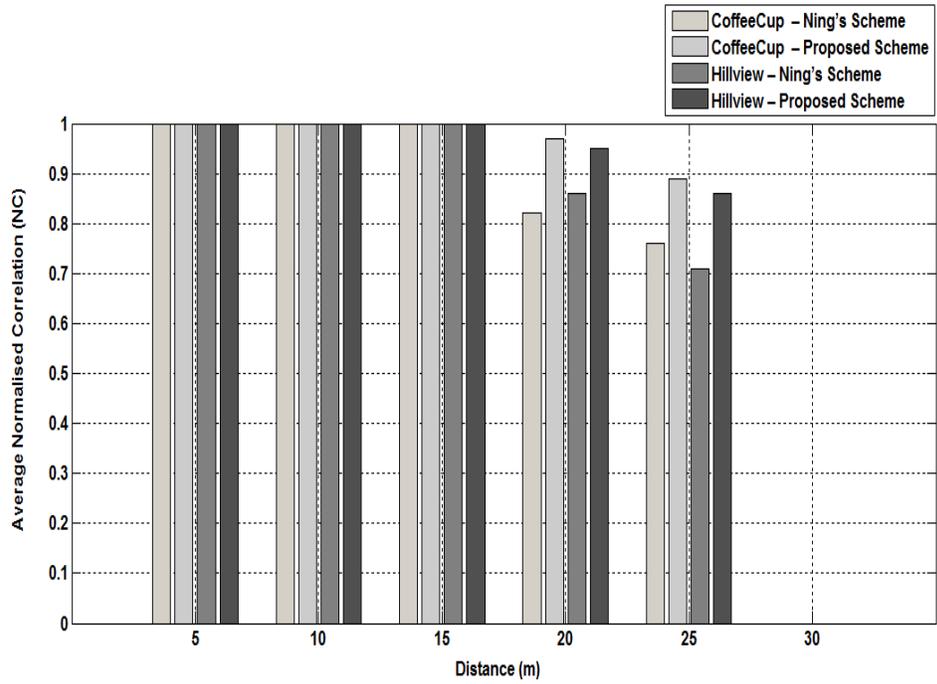
(c)

**Figure 6.11-** Lifetime at (a) Source node; (b) Relay node; (c) Network under different duty cycles

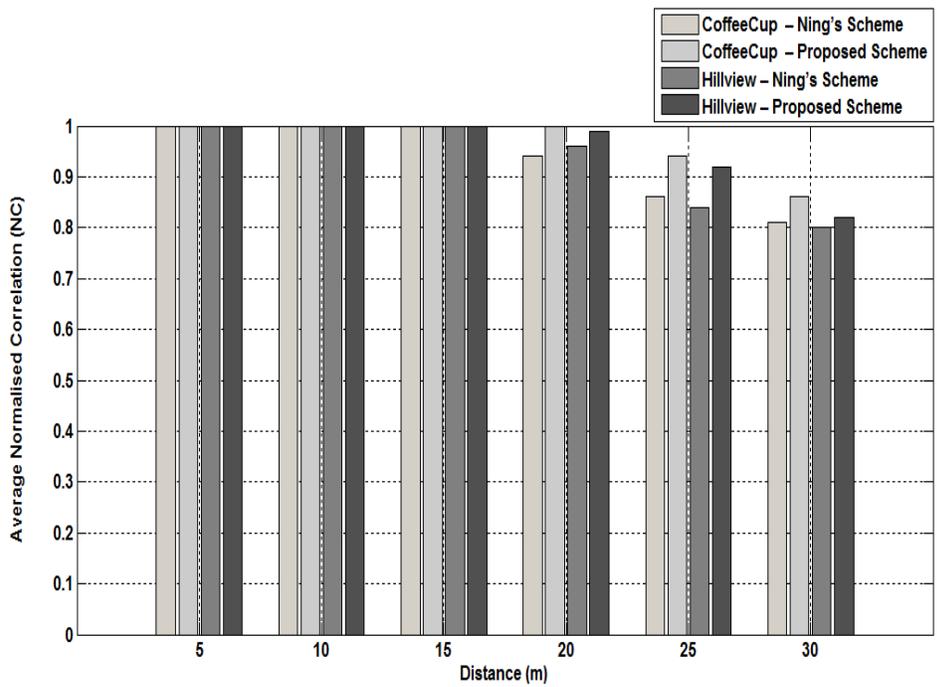
### 6.5.7 Robustness

Data transmission over radio links is a challenging task since the random errors may occur frequently and heavily deteriorates the performance of underlying compression as well as the watermarking scheme. To analyse the impact of these channel errors in WMSN environment, we have also conducted a series of real experiments on the compressed bitstreams of CoffeeCup and Hillview video sequences, watermarked using the proposed and Ning's scheme respectively.

The compressed watermarked bitstreams are transmitted using TelosB motes over single and multihop scenarios with varying distance between source, relay and base-station nodes as shown in Fig 6.12 and Fig. 6.13 respectively. Base-station executes the watermark extraction algorithm followed by the decoding process on the received watermarked bitstreams. We evaluated the average normalised correlation between the extracted and the original watermark to quantify the watermark existence in the received watermarked bitstream.



(a)



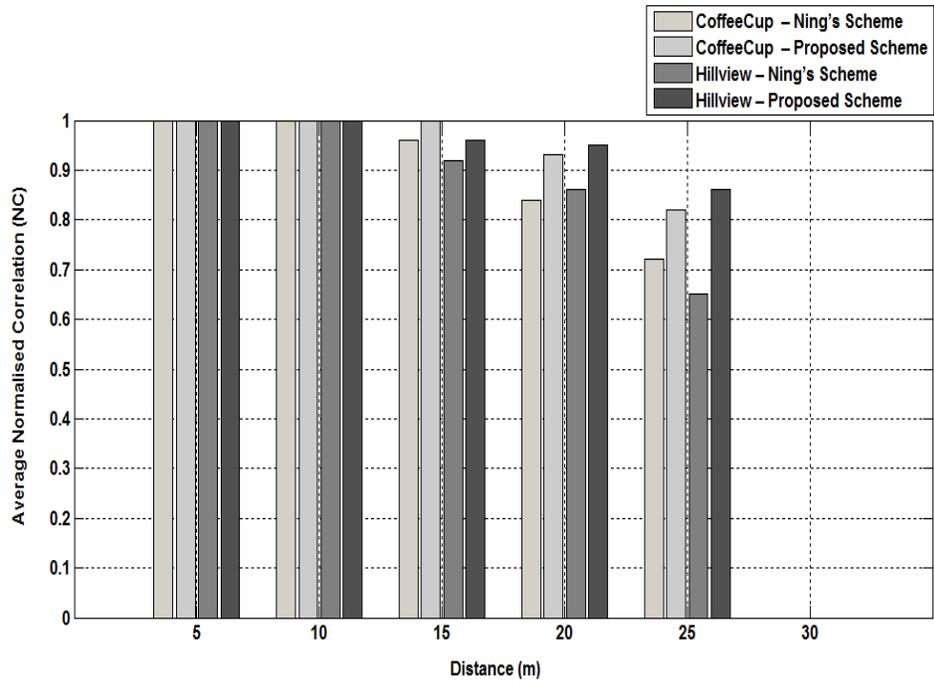
(b)

**Figure 6.12-** Average normalised correlation using single hop transmission in (a) Indoor scenario (b) Outdoor scenario

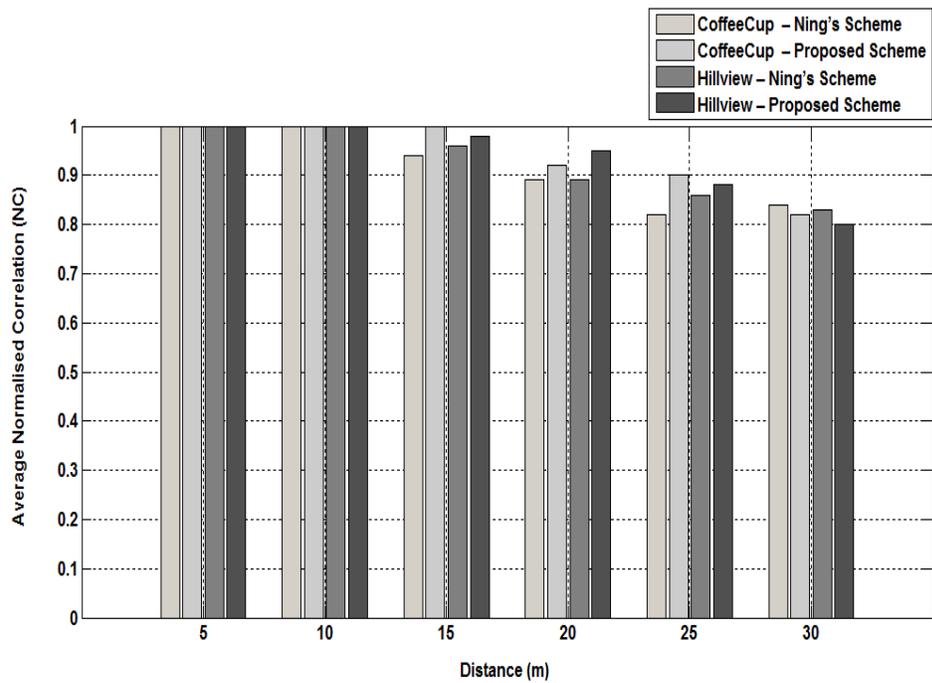
Fig 6.12(a) and 6.12(b) illustrate the NC with varying distance for indoor, and outdoor environment, respectively, at data rate of 38400 bps (maximum rate of TelosB mote). We

gradually increased the distance from 5 to 30 meters and extracted the watermark(s) from the decoded video. Eight redundant watermarks are embedded in each key frame using the proposed scheme and Ning's scheme. For both indoor and outdoor environments, the average NC between the original and the extracted watermark in each of the cases is at least 0.82, which seems to outperform the Ning's scheme [73] that has an average NC of 0.71. However, it excludes the indoor environment case where the distance between the source and the base-station is 30 meters; almost the entire bitstream packets are dropped which eventually fails the watermark extraction process as shown in Fig.6.12(a). It can also be seen clearly that for outdoor environment, the overall average NC for both schemes is relatively better than the indoor environment (Fig.6.12(b)).

On the other hand, similar tests using same settings have been performed with one intermediate relay node which is used to store and forward the watermarked bitstream packets (Fig.6.13(a) and 6.13(b)) to and from source and base-station. It is evident from the results that the proposed scheme clearly outperforms the Ning's scheme. This could be due to the fact that Ning's scheme embeds the watermark into selective interest points around corners and edges within the key frame. Consequently, if those packets that contain watermark information are lost (or corrupted) during bitstream transmission, it will lead to lower NC values or watermark authentication failure in worse-case scenario. In contrast to Ning's scheme, our proposed scheme has uniform and consistent embedding strategy, which is robust such that packet losses to watermarked bitstream only affect a portion of the watermark instead of ruining the entire watermark. Furthermore, the impact of packet losses can be minimised further by introducing the watermark redundancy.



(a)



(b)

**Figure 6.13-** Average normalised correlation using two hop transmission in (a) Indoor scenario (b) Outdoor scenario

## **6.6 Chapter Summary**

In this chapter, we have proposed a novel, low complexity, blind and imperceptible video watermarking scheme for DVC based WZ video coding architecture that embeds the watermark information into each key frame in a given GOP setting. We have analyzed the embedding capacity, imperceptibility, robustness, rate-distortion performance, energy requirements of the scheme, impact on the feedback channel, and lifetime of node and entire network. The results show that the proposed scheme offers a low-complexity and energy efficient watermarking solution for WMSNs and outperforms the existing work in the literature.

This thesis is an effort to bring together energy efficient and robust watermarking mechanisms for WMSN especially those based on DVC. The next Chapter concludes our research and will discuss the potential future research direction in the field.

# Chapter 7

## Conclusion and Future Work

### 7.1 Introduction

This thesis aimed at design and development of watermarking system for multimedia content (image/video) in the context of WMSN. In particular, it focuses on DVC paradigm for the purpose of low-complexity watermarking embedding due to the energy constraints at source-coding sites. The first three chapters explain the introduction, motivation, background, literature review and analysis. In chapter four, we investigated DVC and DCVS based codecs against the conventional H.264 Intra codec. Finding from our experimental results helps in the selection of an appropriate DVC video codec for a given WMSN application. Chapter five proposed an enhanced semi-oblivious energy-aware adaptive watermarking scheme for WMSNs, which considered key characteristics such as the embedding capacity, security, imperceptibility, computation, and communication energy requirements. Chapter 6 presented a novel, energy-efficient, low-complexity, blind, and imperceptible video watermarking scheme based on transform domain Wyner-Ziv (WZ) coding, which builds on the principles of DVC. We investigated the performance of the proposed scheme on a fully functional WZ codec using real video sequences captured from embedded video camera sensors.

The rest of the chapter is organised as follows: Section 7.2 summarise our contributions and Section 7.3, suggest avenues for future study.

## 7.2 Conclusions

The contribution of this thesis is as follows:

- We evaluated and analyzed the performance of video codecs based on emerging video coding paradigms such as distributed video coding (DVC) and distributed compressive video sensing (DCVS) for multihop WMSNs. The aim is to provide insights into the computational (encoding/decoding) complexity, energy consumption, node and network lifetime, processing and memory requirements, and the quality of reconstruction of these video codecs. Based on the findings, this work provides some guidelines for the selection of appropriate video codecs for a given WMSN application.
- We have proposed a semi-oblivious, energy-aware adaptive watermarking scheme for wireless multimedia sensor networks, according to key characteristics such as the embedding capacity, security, imperceptibility, computation and communication energy requirements. We evaluated the distortion in a cover image due to watermark redundancies, the number of embedding locations with respect to two channel adaptive parameters, and the impact of compression of a cover image on the correctness of extracted watermark. In addition, we investigated the robustness of the scheme against statistical analysis attacks. The results have shown that the proposed scheme has sufficient capacity to embed redundant watermarks in the cover image in an imperceptible manner with reasonably low distortion. The scheme is also considered relatively robust against collusion and middleman attacks.
- We have also proposed a novel, blind and imperceptible video watermarking scheme based on DVC for WMSNs. With the advent of distributed source coding theory, several distributed video coding (DVC) architectures have been proposed in the existing literature, which feature a simple encoding and complex decoding framework. Such coding frameworks are well-suited to the wireless multimedia sensor network (WMSN) environment as they support low-complexity video encoding at resource-constrained video sensing nodes, while shifting the computational intensive tasks of motion compensation

and estimation to the typically more resourceful video decoding (sink) node. To date, few watermarking schemes based on DVC have been proposed, and none of them can be directly applied to WMSNs due to their relatively complex embedding operation, high encoding bitrate requirement, and inconsistent embedding capacity. Therefore, this work aims to bridge this gap in research by proposing a novel approach to energy-efficient watermarking based on DVC architecture for WMSNs. The key features of this work are as follows:

- A novel, energy-efficient, low-complexity, blind and imperceptible video watermarking scheme based on WZ coding [5], which builds on the principles of DVC.
- Practical implementation of the proposed scheme on a fully functional DVC codec and its evaluation using real video sequences captured from embedded video sensors.
- Derived analytical models from which the capacity and imperceptibility; computational and communication energy consumption; feedback requests; rate-distortion performance, encoding and decoding complexity; node and network lifetime of WMSN in reference to the proposed scheme are estimated.

The experimental results show that the proposed scheme offers a low-complexity and energy-efficient watermarking solution for WMSN environments.

### **7.3 Future Work**

Experimental results at this stage have shown good potential for the proposed scheme. There are many avenues for pursuing further work in this area. Some of these are as follows:

- High efficiency video coding (HEVC) [176] is a new video compression format that succeeds H.264/M-PEG/AVC. HEVC is said to double the compression efficiency compared to existing compression formats (H.264/M-PEG/AVC) while maintaining the same video quality. As a part of future work, we will be looking into adapting the evolving HEVC coding scheme for key frame encoding and watermark embedding. Given higher compression efficiency, HEVC will help to further reduce the communication energy requirement for the codec as well as watermarking scheme.

- The Wyner-Ziv codec we have used is appropriate only for real-time applications with active communication mode. However, for passive communication of multimedia data in a WMSN environment, watermarking schemes still need to be designed and investigated. DVC or DCVS based codecs that do not require a feedback channel would be appropriate for applications that follow passive data transmission.
- Developing effective ways to optimises error/mismatch propagation in WZ-frames when the watermark is embedded in the key frames that are used to form side-information for WZ frames.
- Watermarking schemes also need to be explored with reference to DCVS coding architectures for both active and passive (feedback-less) data transmission modes.
- Low-complexity privacy protection mechanisms (as discussed in Section 2.3) based on watermarking for WMSN also require considerable attention and a subject for further research offering plenty of room for innovation.

# References

- [1] B. Harjito and H. Song, "Wireless Multimedia Sensor Networks Applications and Security Challenges," in *Broadband, Wireless Computing, Communication and Applications (BWCCA), International Conference on*, pp. 842-846,2010.
- [2] T. Melodia and I. Akyildiz, "Research Challenges for Wireless Multimedia Sensor Networks," in *Distributed Video Sensor Networks*, B. Bhanu, *et al.*, Eds., ed: Springer London, pp. 233-246, 2011.
- [3] Z. Yun, *et al.*, "Securing wireless sensor networks: a survey," *Communications Surveys & Tutorials, IEEE*, vol. 10, pp. 6-28, 2008.
- [4] M. Guerrero-Zapata, *et al.*, "The future of security in Wireless Multimedia Sensor Networks," *Telecommunication Systems*, vol. 45, pp. 77-91, 2010.
- [5] A. Aaron, *et al.*, "Transform-domain Wyner-Ziv codec for video," *Visual Communications and Image Processing 2004, January 2004, San Jose, Calif, USA, Proceedings of SPIE*, vol. 5308, pp. 520 - 528, 2004.
- [6] D. Kundur, *et al.*, "Security and Privacy for Distributed Multimedia Sensor Networks," *Proceedings of the IEEE*, vol. 96, pp. 112-130, 2008.
- [7] B. Harjito, *et al.*, "Secure communication in wireless multimedia sensor networks using watermarking," in *Digital Ecosystems and Technologies (DEST), 2010 4th IEEE International Conference on*, pp. 640-645,2010.

- [8] I. Lee, *et al.*, "Wireless Multimedia Sensor Networks Guide to Wireless Sensor Networks," ed: Springer London, pp. 561-582, 2009.
- [9] M. AlNuaimi, *et al.*, "A survey of Wireless Multimedia Sensor Networks challenges and solutions," in *Innovations in Information Technology (IIT), 2011 International Conference on*, pp. 191-196, 2011.
- [10] M. Javier, *et al.* Multimedia Data Processing and Delivery in Wireless Sensor Networks, *Wireless Sensor Networks: Application-Centric Design. Geoff V Merrett and Yen Kheng Tan (Ed.)*, 2010.
- [11] MEMSIC TelosB Mote Specifications. Available: <http://www.memsic.com/>
- [12] MEMSIC MICAz Mote Specifications. Available: <http://www.memsic.com/>
- [13] I. F. Akyildiz, *et al.*, "A survey on wireless multimedia sensor networks," *Computer Networks*, vol. 51, pp. 921-960, 2007.
- [14] S. Misra, *et al.*, "A survey of multimedia streaming in wireless sensor networks," *Communications Surveys & Tutorials, IEEE*, vol. 10, pp. 18-39, 2008.
- [15] H. Kalva, "The H.264 Video Coding Standard," *Multimedia, IEEE*, vol. 13, pp. 86-90, 2006.
- [16] P. Salembier and T. Sikora, *Introduction to MPEG-7: Multimedia Content Description Interface*: John Wiley & Sons, Inc., 2002.
- [17] F. Dufaux, *et al.*, "Distributed Video Coding: Trends and Perspectives," *EURASIP Journal on Image and Video Processing*, vol. 2009, 2009.
- [18] J. J. Ahmad, *et al.*, "Energy efficient video compression for wireless sensor networks," in *Information Sciences and Systems, 2009. CISS 2009. 43rd Annual Conference on*, pp. 629-634, 2009.
- [19] F. Pereira, *et al.*, "Distributed video coding: Selecting the most promising application scenarios," *Image Commun.*, vol. 23, pp. 339-352, 2009.
- [20] H. Schwarz, *et al.*, "Overview of the Scalable Video Coding Extension of the H.264/AVC Standard," *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 17, pp. 1103-1120, 2007.
- [21] D. Slepian and J. Wolf, "Noiseless coding of correlated information sources," *Information Theory, IEEE Transactions on*, vol. 19, pp. 471-480, 1973.

- [22] A. Wyner and J. Ziv, "The rate-distortion function for source coding with side information at the decoder," *Information Theory, IEEE Transactions on*, vol. 22, pp. 1-10, 1976.
- [23] A. Miguel, "Low Delay Distributed Video Coding," Masters in Electrical and Computer Engineering, Technical University of Lisbon, 2009.
- [24] W. Yin, *et al.*, "Bregman Iterative Algorithms for  $\ell_1$ -Minimization with Applications to Compressed Sensing," *SIAM J. Img. Sci.*, vol. 1, pp. 143-168, 2008.
- [25] W. Ju and L. Jonathan, "Video Authentication against Correlation Analysis Attack in Wireless Network," in *Multimedia, 2008. ISM 2008. Tenth IEEE International Symposium on*, 2008, pp. 396-403.
- [26] T. Goldstein and S. Osher, "The Split Bregman Method for L1-Regularized Problems," *SIAM Journal on Imaging Sciences*, vol. 2, pp. 323-343, 2009.
- [27] S. Chessa, *et al.*, "Mobile Application Security for Video Streaming Authentication and Data Integrity Combining Digital Signature and Watermarking Techniques," in *Vehicular Technology Conference, 2007. VTC2007-Spring. IEEE 65th*, 2007, pp. 634-638.
- [28] I. Cox, *et al.*, "Watermarking Is Not Cryptography  
Digital Watermarking." vol. 4283, Y. Shi and B. Jeon, Eds., ed: Springer Berlin / Heidelberg, 2006, pp. 1-15.
- [29] S. Katzenbeisser, "On the Integration of Watermarks and Cryptography  
Digital Watermarking." vol. 2939, T. Kalker, *et al.*, Eds., ed: Springer Berlin / Heidelberg, 2004, pp. 267-268.
- [30] A.-R. Sadeghi, "The Marriage of Cryptography and Watermarking — Beneficial and Challenging for Secure Watermarking and Detection  
Digital Watermarking." vol. 5041, Y. Shi, *et al.*, Eds., ed: Springer Berlin / Heidelberg, 2008, pp. 2-18.
- [31] M. M. Haque, *et al.*, "An Efficient PKC-Based Security Architecture for Wireless Sensor Networks," in *Military Communications Conference, 2007. MILCOM 2007. IEEE*, 2007, pp. 1-7.

- [32] Y. Pingping, *et al.*, "Copyright Protection for Digital Image in Wireless Sensor Network," in *Wireless Communications, Networking and Mobile Computing, 2009. WiCom '09. 5th International Conference on*, pp. 1-4, 2009.
- [33] X. Jin and J. Kim, "A Secure Image Watermarking Using Visual Cryptography," in *Computer Science and its Applications*. vol. 203, S.-S. Yeo, *et al.*, Eds., ed: Springer Netherlands, pp. 179-187, 2012.
- [34] G.-C. Tai and L.-W. Chang, "Visual Cryptography for Digital Watermarking in Still Images," in *Advances in Multimedia Information Processing - PCM 2004*. vol. 3332, K. Aizawa, *et al.*, Eds., ed: Springer Berlin Heidelberg, pp. 50-57, 2005.
- [35] A. Piva, "Cryptography and Data Hiding for Media Security," in *Multimedia Services in Intelligent Environments*. vol. 120, G. Tsihrintzis and L. Jain, Eds., ed: Springer Berlin Heidelberg, pp. 227-255, 2008.
- [36] W.-C. Yang, *et al.*, "Applying Public-Key Watermarking Techniques in Forensic Imaging to Preserve the Authenticity of the Evidence," in *Intelligence and Security Informatics*. vol. 5075, C. Yang, *et al.*, Eds., ed: Springer Berlin Heidelberg, pp. 278-287, 2008.
- [37] V. Kitanovski, *et al.*, "Combined hashing/watermarking method for image authentication," *International Journal of Signal Processing*, vol. 3, pp. 223-229, 2007.
- [38] J. Cannons and P. Moulin, "Design and statistical analysis of a hash-aided image watermarking system," *Image Processing, IEEE Transactions on*, vol. 13, pp. 1393-1408, 2004.
- [39] F. Liu, *et al.*, "Wave-atoms-based multipurpose scheme via perceptual image hashing and watermarking," *Applied Optics*, vol. 51, pp. 6561-6570, 2012.
- [40] L. Weng, *et al.*, "Robust image content authentication using perceptual hashing and watermarking," in *Proceedings of the 13th Pacific-Rim conference on Advances in Multimedia Information Processing*, pp. 315-326, 2012.
- [41] G. Gaubatz, *et al.*, "State of the art in ultra-low power public key cryptography for wireless sensor networks," in *Pervasive Computing and Communications Workshops, PerCom 2005 Workshops. Third IEEE International Conference on*, pp. 146-150, 2005.

- [42] S. Rahman, *et al.*, "Chaos-cryptography based privacy preservation technique for video surveillance," *Multimedia Systems*, pp. 1-11, 2010.
- [43] H. Rahmani, *et al.*, "A new lossless watermarking scheme based on DCT coefficients," in *Digital Content, Multimedia Technology and its Applications (IDC), 2010 6th International Conference on*, pp. 28-33, 2010.
- [44] A. S. Wander, *et al.*, "Energy analysis of public-key cryptography for wireless sensor networks," in *Pervasive Computing and Communications, 2005. PerCom 2005. Third IEEE International Conference on*, pp. 324-328, 2005.
- [45] I. T. Almalkawi, *et al.*, "Wireless Multimedia Sensor Networks: Current Trends and Future Directions," *Sensors*, vol. 10, pp. 6662-6717, 2010.
- [46] S. c. S. Cheung, *et al.*, "Managing privacy data in pervasive camera networks," in *Image Processing, 2008. ICIP 2008. 15th IEEE International Conference on*, pp. 1676-1679, 2008.
- [47] M. Saini, *et al.*, "Anonymous surveillance," in *Multimedia and Expo (ICME), IEEE International Conference on*, pp. 1-6, 2011.
- [48] J. Wang and G. L. Smith, "A cross layer authentication design for secure video transportation in wireless sensor network," *Int. J. Secur. Netw.*, vol. 5, pp. 63-76, 2010.
- [49] J. Shen and X. Zheng, "Security for Video Surveillance with Privacy," in *Internet Technology and Applications, 2010 International Conference on*, pp. 1-4, 2010.
- [50] P. Fakhari, *et al.*, "Protecting patient privacy from unauthorized release of medical images using a bio-inspired wavelet-based watermarking approach," *Digital Signal Processing*, vol. 21, pp. 433-446, 2011.
- [51] H. Wang, *et al.*, "Energy-Aware Adaptive Watermarking for Real-Time Image Delivery in Wireless Sensor Networks," in *Communications, 2008. ICC '08. IEEE International Conference on*, pp. 1479-1483, 2008.
- [52] H. Jiang, *et al.*, "A Solution of Video Semi-fragile Watermarking of Authentication Based on Binary Characteristic Strings," in *Multimedia Information Networking and Security, 2009. MINES '09. International Conference on*, pp. 167-170, 2009.
- [53] V. Kitanovski, *et al.*, "Semi-Fragile Watermarking Scheme for Authentication of MPEG-1/2 Coded Videos," in *Systems, Signals and Image Processing, 2007 and 6th*

- EURASIP Conference focused on Speech and Image Processing, Multimedia Communications and Services. 14th International Workshop on*, pp. 225-228, 2007.
- [54] C. Rey and J.-L. Dugelay, "A survey of watermarking algorithms for image authentication," *EURASIP J. Appl. Signal Process.*, vol. 2002, pp. 613-621, 2002.
- [55] C. Tsong-Yi, *et al.*, "H.264 Video Authentication Based on Semi-fragile Watermarking," in *Intelligent Information Hiding and Multimedia Signal Processing, 2008. IHHMSP '08 International Conference on*, pp. 659-662, 2008.
- [56] Chris Harris and M. Stephens., "A Combined Corner and Edge Detector. ," in *Proceeding of the fourth Alvey Vision Conference*, pp. 147-152, 1988.
- [57] J. J and K. B, "Public-Key Cryptography Standards (PKCS) #1: RSA Cryptography Specifications Version 2.1," 2003.
- [58] I. Kamel and H. Juma, "A Lightweight Data Integrity Scheme for Sensor Networks," *Sensors*, vol. 11, pp. 4118-4136, 2011.
- [59] H. Hui-Yu, *et al.*, "A video watermarking algorithm based on pseudo 3D DCT," in *Computational Intelligence for Image Processing, 2009. CIIP '09. IEEE Symposium on*, pp. 76-81, 2009.
- [60] L. A. Grieco, *et al.*, "Secure Wireless Multimedia Sensor Networks: A Survey," in *Mobile Ubiquitous Computing, Systems, Services and Technologies, 2009. UBICOMM '09. Third International Conference on*, pp. 194-201, 2009.
- [61] F. Hartung and M. Kutter, "Multimedia watermarking techniques," *Proceedings of the IEEE*, vol. 87, pp. 1079-1107, 1999.
- [62] D. Xu, *et al.*, "Video Watermarking Based on Spatio-temporal JND Profile Digital Watermarking." vol. 5450, H.-J. Kim, *et al.*, Eds., ed: Springer Berlin / Heidelberg, pp. 327-341, 2009.
- [63] R. Lancini, *et al.*, "A robust video watermarking technique in the spatial domain," in *Video/Image Processing and Multimedia Communications 4th EURASIP-IEEE Region 8 International Symposium on VIPromCom*, pp. 251-256., 2002.
- [64] H. Ling, *et al.*, "Robust video watermarking based on affine invariant regions in the compressed domain," *Signal Processing*, vol. 91, pp. 1863-1875, 2011.

- [65] Y. Cheng-Han, *et al.*, "An adaptive video watermarking technique based on DCT domain," in *Computer and Information Technology, 2008. CIT 2008. 8th IEEE International Conference on*, pp. 589-594, 2008.
- [66] K. Ahmed, *et al.*, "Novel DWT video watermarking schema," *MG&V*, vol. 18, pp. 363-380, 2009.
- [67] W. Chun-Xing, "A Blind Video Watermarking Scheme Based on DWT," pp. 434-437, 2009.
- [68] Y. Liu and J. Zhao, "A new video watermarking algorithm based on 1D DFT and Radon transform," *Signal Processing*, vol. 90, pp. 626-639, 2010.
- [69] P. Campisi and A. Neri, "Perceptual Video Watermarking in the 3D-DWT Domain Using a Multiplicative Approach Digital Watermarking." vol. 3710, M. Barni, *et al.*, Eds., ed: Springer Berlin / Heidelberg, pp. 432-443, 2005.
- [70] S. P. Mohanty, *et al.*, "VLSI architectures of perceptual based video watermarking for real-time copyright protection," in *Quality of Electronic Design, 2009. ISQED 2009. Quality Electronic Design*, pp. 527-534, 2009.
- [71] L. Zhi and C. Xiaowei, "The imperceptible video watermarking based on the model of entropy," in *Audio, Language and Image Processing, 2008. ICALIP 2008. International Conference on*, pp. 480-484, 2008.
- [72] J. Li, "A Novel Scheme of Robust and Blind Video Watermarking," in *Information Technology and Applications, 2009. IFITA '09. International Forum on*, pp. 430-434, 2009.
- [73] Z. Ning, *et al.*, "A Novel Watermarking Method For Wyner-Ziv Video Coding," in *Intelligent Information Hiding and Multimedia Signal Processing, 2008. IIHMSp '08 International Conference on*, pp. 857-860, 2008
- [74] F. Gui and W. Guo-Zheng, "Motion vector and mode selection based fragile video watermarking algorithm," in *Anti-Counterfeiting, Security and Identification (ASID), 2011 IEEE International Conference on*, pp. 73-76, 2011.
- [75] K. Tien-Ying, *et al.*, "Fragile Video Watermarking Technique by Motion Field Embedding with Rate-Distortion Minimization," in *Intelligent Information Hiding and Multimedia Signal Processing, 2008. IIHMSp '08 International Conference on*, pp. 853-856, 2008.

- [76] F. Dufaux, *et al.*, "Distributed Video Coding: Trends and Perspectives," *EURASIP Journal on Image and Video Processing*, vol. 2009, p. 13, 2009.
- [77] B. Girod, *et al.*, "Distributed Video Coding," *Proceedings of the IEEE*, vol. 93, pp. 71-83, 2005.
- [78] R. Puri, *et al.*, "Distributed video coding in wireless sensor networks," *Signal Processing Magazine, IEEE*, vol. 23, pp. 94-106, 2006.
- [79] A. Aaron, *et al.*, "Wyner-Ziv video coding with hash-based motion compensation at the receiver," in *Image Processing, 2004. ICIP '04. 2004 International Conference on*, pp. 3097-3100 Vol. 5, 2004.
- [80] A. Aaron, *et al.*, *Transform-domain Wyner-Ziv codec for video* vol. 5308: SPIE, 2004.
- [81] A. Aaron, *et al.*, "Wyner-Ziv coding of motion video," in *Signals, Systems and Computers, 2002. Conference Record of the Thirty-Sixth Asilomar Conference on*, pp. 240-244 vol.1, 2002.
- [82] Y. Aiguo, *et al.*, "A fast video transcoder from Wyner-Ziv to AVS," presented at the Proceedings of the Advances in multimedia information processing, and 11th Pacific Rim conference on Multimedia: Part II, Shanghai, China, 2010.
- [83] Y. Aiguo, *et al.*, "A Fast Video Transcoder from Wyner-Ziv to AVS," in *Advances in Multimedia Information Processing - PCM 2010*. vol. 6298, G. Qiu, *et al.*, Eds., ed: Springer Berlin / Heidelberg, pp. 328-339,2011.
- [84] A. Anne, "Wyner-Ziv Coding for Video: Applications to Compression and Error Resilience," pp. 93-93, 2003.
- [85] J. Ascenso and F. Pereira, "Adaptive Hash-Based Side Information Exploitation for Efficient Wyner-Ziv Video Coding," in *Image Processing, 2007. ICIP 2007. IEEE International Conference on*,pp. III - 29-III - 32, 2007.
- [86] C. Brites, *et al.*, "Studying Temporal Correlation Noise Modeling for Pixel Based Wyner-Ziv Video Coding," in *Image Processing, 2006 IEEE International Conference on*, pp. 273-276, 2006.
- [87] D. Kubasov, *et al.*, "Optimal Reconstruction in Wyner-Ziv Video Coding with Multiple Side Information," in *Multimedia Signal Processing, 2007. MMSP 2007. IEEE 9th Workshop on*, pp. 183-186, 2007.

- [88] J. L. Martinez, *et al.*, "Wyner-Ziv to H.264 video transcoder," in *Image Processing (ICIP), 2009 16th IEEE International Conference on*, pp. 2941-2944, 2009.
- [89] J. Pedro, "Studying Error Resilience Performance for a Feedback Channel based Transform Domain Wyner-Ziv Video Codec," presented at the Picture Coding Symposium Lisbon, Portugal, 2007.
- [90] E. Peixoto, *et al.*, "A Wyner-Ziv Video Transcoder," *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 20, pp. 189-200, 2010.
- [91] F. Pereira, *et al.*, "Wyner-Ziv video coding: A review of the early architectures and further developments," in *Multimedia and Expo, 2008 IEEE International Conference on*, pp. 625-628, 2008.
- [92] X. Qian and X. Zixiang, "Layered Wyner&ndash;Ziv Video Coding," *Image Processing, IEEE Transactions on*, vol. 15, pp. 3791-3803, 2006.
- [93] M. Tagliasacchi, *et al.*, "Intra Mode Decision Based on Spatio-Temporal Cues in Pixel Domain Wyner-ZIV Video Coding," in *Acoustics, Speech and Signal Processing, 2006. ICASSP 2006 Proceedings. 2006 IEEE International Conference on*, pp. II-II, 2006.
- [94] E. J. Candes and M. B. Wakin, "An Introduction To Compressive Sampling," *Signal Processing Magazine, IEEE*, vol. 25, pp. 21-30, 2008.
- [95] M. F. Duarte, *et al.*, "Single-Pixel Imaging via Compressive Sampling," *Signal Processing Magazine, IEEE*, vol. 25, pp. 83-91, 2008.
- [96] T. T. Do, *et al.*, "Distributed Compressed Video Sensing," in *Information Sciences and Systems, 2009. CISS 2009. 43rd Annual Conference on*, pp. 1-2, 2009.
- [97] C. Zhang and J. Leng, "Distributed video coding based on compressive sensing," in *Multimedia Technology (ICMT), 2011 International Conference on*, 2011, pp. 3046-3049, 2011.
- [98] X. Artigas, *et al.*, "The DISCOVER codec: Architecture, Techniques and Evaluation," *Picture Coding Symposium*, vol. 17, pp. 1103–1120, 2007.
- [99] Z. Xue, *et al.*, "Distributed video coding in wireless multimedia sensor network for multimedia broadcasting," *WTOC*, vol. 7, pp. 418-427, 2008.

- [100] R. Puri, "PRISM : a new robust video coding architecture based on distributed compression principles," *Proc. of Allerton Conf. Commun., Ctrl, Computing, Urbana-Champaign, IL, Oct. 2002*.
- [101] A. Majumdar and K. Ramchandran, "PRISM: an error-resilient video coding paradigm for wireless networks," in *Broadband Networks, 2004. BroadNets 2004. Proceedings. First International Conference on*, pp. 478-485, 2004.
- [102] October 2012). BCH Code. Available: [http://en.wikipedia.org/wiki/BCH\\_code](http://en.wikipedia.org/wiki/BCH_code)
- [103] R. Puri and R. Kannan, "PRISM: a "reversed" multimedia coding paradigm," in *Image Processing, 2003. ICIP 2003. Proceedings. 2003 International Conference on*, 2003, pp. I-617-20 vol.1, 2003.
- [104] G. D. Forney, Jr., "Coset codes. I. Introduction and geometrical classification," *Information Theory, IEEE Transactions on*, vol. 34, pp. 1123-1151, 1988.
- [105] A. Avudainayagam, *et al.*, "Hyper-Trellis Decoding of Pixel-Domain Wyner&#x2013;Ziv Video Coding," *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 18, pp. 557-568, 2008.
- [106] D. Baoguo and S. Hong, "Encoder Rate Control for Pixel-Domain Distributed Video Coding without Feedback Channel," in *Multimedia and Ubiquitous Engineering, 2009. MUE '09. Third International Conference on*, pp. 9-13, 2009.
- [107] M. Morbee, *et al.*, "Rate Allocation Algorithm for Pixel-Domain Distributed Video Coding Without Feedback Channel," in *Acoustics, Speech and Signal Processing, 2007. ICASSP 2007. IEEE International Conference on*, pp. I-521-I-524, 2007.
- [108] D. Kubasov, *et al.*, "A Hybrid Encoder/Decoder Rate Control for Wyner-Ziv Video Coding with a Feedback Channel," in *Multimedia Signal Processing, 2007. MMSP 2007. IEEE 9th Workshop on*, pp. 251-254, 2007.
- [109] B. Macchiavello, *et al.*, "Side-Information Generation for Temporally and Spatially Scalable Wyner-Ziv Codecs," *EURASIP Journal on Image and Video Processing*, vol. 2009, p. 171257, 2009.
- [110] S. Yun-Chung, *et al.*, "Progressive Side Information Refinement with Non-local Means Based Denoising Process for Wyner-Ziv Video Coding," in *Data Compression Conference (DCC)*, pp. 219-226, 2012.

- [111] B. Macchiavello, *et al.*, "Motion-Based Side-Information Generation for a Scalable Wyner-Ziv Video Coder," in *Image Processing, 2007. ICIP 2007. IEEE International Conference on*, pp. VI - 413-VI - 416, 2007.
- [112] J. Wen, *et al.*, "EXIT Chart-Based Side Information Refinement for Wyner-Ziv Video Coding," in *Data Compression Conference (DCC)*, , pp. 209-218, 2012.
- [113] B. Macchiavello, *et al.*, "Iterative Side-Information Generation in a Mixed Resolution Wyner-Ziv Framework," *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 19, pp. 1409-1423, 2009.
- [114] L. Wei, *et al.*, "Estimating side-information for Wyner-Ziv video coding using resolution-progressive decoding and extensive motion exploration," in *Acoustics, Speech and Signal Processing, 2009. ICASSP 2009. IEEE International Conference on*, pp. 721-724, 2009.
- [115] *DISCOVER Video Codec*. Available: [www.discoverdvc.org](http://www.discoverdvc.org)
- [116] D. Varodayan, *et al.*, "Rate-adaptive codes for distributed source coding," *Signal Process.*, vol. 86, pp. 3123-3130, 2006.
- [117] J. Ascenso, *et al.*, "Content Adaptive Wyner-ZIV Video Coding Driven by Motion Activity," in *Image Processing, 2006 IEEE International Conference on*, pp. 605-608, 2006.
- [118] (October 2012). Turbo Codes. Available: [http://en.wikipedia.org/wiki/Turbo\\_code](http://en.wikipedia.org/wiki/Turbo_code)
- [119] (October 2012). LDPC Code. Available: [http://en.wikipedia.org/wiki/LDPC\\_code](http://en.wikipedia.org/wiki/LDPC_code)
- [120] M. A. T. Figueiredo, *et al.*, "Gradient Projection for Sparse Reconstruction: Application to Compressed Sensing and Other Inverse Problems," *Selected Topics in Signal Processing, IEEE Journal of*, vol. 1, pp. 586-597, 2007.
- [121] J. A. Tropp and A. C. Gilbert, "Signal Recovery From Random Measurements Via Orthogonal Matching Pursuit," *Information Theory, IEEE Transactions on*, vol. 53, pp. 4655-4666, 2007.
- [122] L.-W. Kang and C.-S. Lu, "Distributed compressive video sensing," presented at the IEEE International Conference on Acoustics, Speech and SP;Taiwan, 2009.
- [123] G. L, *et al.*, "Fast compressive imaging using scrambled hadamard ensemble," in *Proc. EUSIPCO*, 2008.

- [124] A. Beck and M. Teboulle, "A Fast Iterative Shrinkage-Thresholding Algorithm for Linear Inverse Problems," *SIAM J. Img. Sci.*, vol. 2, pp. 183-202, 2009.
- [125] Chen H W, *et al.*, "Dynamic measurement rate allocation for distributed compressive video sensing," , p. 774401, 2010.
- [126] J. Ostermann, *et al.*, "Video coding with H.264/AVC: tools, performance, and complexity," *Circuits and Systems Magazine, IEEE*, vol. 4, pp. 7-28, 2004.
- [127] J. L. Wong, *et al.*, "Security in Sensor Networks: Watermarking Techniques Wireless Sensor Networks," C. S. Raghavendra, *et al.*, Eds., ed: Springer US,, pp. 305-323, 2004.
- [128] B. Harjito., Watermarking Technique based on Linear Feed Back Shift Register (LFSR),. *Seminar Nasional Konferda ke-9 Himpunan Matematika Wilayah Jateng dan DIY di FMIPA UNS, 2003.*
- [129] M. Li, *et al.*, *An Introduction to Kolmogorov Complexity and Its Applications*, 3rd ed.: Springer Theoretical Computer Science, 2008.
- [130] H. Wang, "Communication-resource-aware adaptive watermarking for multimedia authentication in wireless multimedia sensor networks," *The Journal of Supercomputing*, pp. 1-15, 2010.
- [131] H. Ming-Shing, *et al.*, "Hiding digital watermarks using multiresolution wavelet transform," *Industrial Electronics, IEEE Transactions on*, vol. 48, pp. 875-882, 2001.
- [132] H. M. Al-Otum and N. A. Samara, "A robust blind color image watermarking based on wavelet-tree bit host difference selection," *Signal Processing*, vol. 90, pp. 2498-2512, 2010.
- [133] M. He., " Adaptive Image Digital Watermarking Algorithm Based on Best Scramble.," Master's thesis: , 2007.
- [134] M. Wu, *et al.*, "Adaptive Dictionary Learning for Distributed Compressive Video Sensing," *JDCTA: International Journal of Digital Content Technology and its Applications*, vol. Vol.6, p. 9, 2012.
- [135] P. Luo, *et al.*, "Motion estimation using a compressive sensing architecture for H. 264/AVC," in *Multimedia Technology (ICMT), 2011 International Conference on*, pp. 3133-3135, 2011.

- [136] Y. Zhou, *et al.*, "The research for tamper forensics on MPEG-2 video based on compressed sensing," in *ICMLC*, pp. 1080-1084, 2012.
- [137] (2013). *CMLAB: DSP, Graphic and Network Systems*. Available: [http://www.cmlab.csie.ntu.edu.tw/new\\_cml\\_website/index.php](http://www.cmlab.csie.ntu.edu.tw/new_cml_website/index.php)
- [138] K. Li-Wei and L. Chun-Shein, "Distributed compressive video sensing," *IEEE International Conference on Acoustics, Speech and SP;Taiwan*, April 2009.
- [139] *H.264/AVC Software Coordination*. Available: <http://iphome.hhi.de/suehring/tml/>
- [140] S. Gurun, *et al.*, "NWSLite: A general-purpose, nonparametric prediction utility for embedded systems," *ACM Trans. Embed. Comput. Syst.*, vol. 7, pp. 1-36, 2008.
- [141] R. Jurdak, *et al.*, "Adaptive Radio Modes in Sensor Networks: How Deep to Sleep?," in *Sensor, Mesh and Ad Hoc Communications and Networks, 2008. SECON '08. 5th Annual IEEE Communications Society Conference on*, pp. 386-394, 2008.
- [142] K. Schwieger and G. Fettweis., *Multi-Hop Transmission: Benefits and Deficits*, 2004.
- [143] M. Shiwen, *et al.*, "Multipath video transport over ad hoc networks," *Wireless Communications, IEEE*, vol. 12, pp. 42-49, 2005.
- [144] M. Z̄, *et al.*, "An analysis of unreliability and asymmetry in low-power wireless links," *ACM Trans. Sen. Netw.*, vol. 3, p. 7, 2007.
- [145] P. S. Boluk, *et al.*, "Robust Image Transmission Over Wireless Sensor Networks," *Mob. Netw. Appl.*, vol. 16, pp. 149-170, 2011.
- [146] T. S. Rappaport, *Wireless communications: principles and practice*: Prentice Hall PTR, 2002.
- [147] N. Reijers, *et al.*, "Link layer measurements in sensor networks," in *Mobile Ad-hoc and Sensor Systems, 2004 IEEE International Conference on*, pp. 224-234, 2004.
- [148] "IEEE standard for part 15.4: Wireless MAC and PHY specifications for low rate WPAN," ed. New York: IEEE Std 802.15.4.
- [149] H. Karl and A. Willig, *Protocols and Architectures for Wireless Sensor Networks*: John Wiley & Sons, 2007.
- [150] F. Kerasiotis, *et al.*, "Battery Lifetime Prediction Model for a WSN Platform," in *Sensor Technologies and Applications (SENSORCOMM), 2010 Fourth International Conference on*, pp. 525-530, 2010.

- [151] K. V. Kumar and P. G. K. Mohan.. Distributed Video Coding (DVC): Challenges in Implementation and Practical Usage, 2011.
- [152] C. B. Margi, *et al.*, "Characterizing energy consumption in a visual sensor network testbed," in *Testbeds and Research Infrastructures for the Development of Networks and Communities, 2006. TRIDENTCOM 2006. 2nd International Conference on*, pp. 8 pp.-339, 2006.
- [153] N. Thomos, *et al.*, "Optimized transmission of JPEG2000 streams over wireless channels," *Image Processing, IEEE Transactions on*, vol. 15, pp. 54-67, 2006.
- [154] W. Ju, *et al.*, "Supporting Video Data in Wireless Sensor Networks," in *Multimedia, 2007. ISM 2007. Ninth IEEE International Symposium on*, pp. 310-317, 2007,.
- [155] N. Imran, *et al.*, "A comparative analysis of video codecs for multihop wireless video sensor networks," *Multimedia Systems*, pp. 1-17, 2011.
- [156] I. Kamel, *et al.*, "Distortion-Free Watermarking Scheme for Wireless Sensor Networks," in *Intelligent Networking and Collaborative Systems, 2009. INCOS '09. International Conference on*, pp. 135-140, 2009.
- [157] W. Burger and M. J. Burge, "The Discrete Cosine Transform DCT," in *Principles of Digital Image Processing*, ed: Springer London, pp. 1-8, 2009.
- [158] W.-T. Huang, *et al.*, "A robust watermarking technique for copyright protection using discrete wavelet transform," *W. Trans. on Comp.*, vol. 9, pp. 485-495, 2010.
- [159] R. Dubolia, *et al.*, "Digital Image Watermarking by Using Discrete Wavelet Transform and Discrete Cosine Transform and Comparison Based on PSNR," in *Communication Systems and Network Technologies (CSNT), 2011 International Conference on*, pp. 593-596, 2011.
- [160] R. K. Megalingam, *et al.*, "Performance Comparison of Novel, Robust Spatial Domain Digital Image Watermarking with the Conventional Frequency Domain Watermarking Techniques," in *Signal Acquisition and Processing, 2010. ICSAP '10. International Conference on*, pp. 349-353, 2010.
- [161] B. Abhishek, *et al.*, "FPGA Based Design of Robust Spatial Domain Image Watermarking Algorithm Power Electronics and Instrumentation Engineering." vol. 102, V. V. Das, *et al.*, Eds., ed: Springer Berlin Heidelberg, pp. 91-95. 2011.

- [162] M. N. Sakib, *et al.*, "A Basic Digital Watermarking Algorithm in Discrete Cosine Transformation Domain," in *Intelligent Systems, Modelling and Simulation (ISMS), 2011 Second International Conference on*, pp. 419-421, 2011.
- [163] H. Inoue, *et al.*, "A digital watermark based on the wavelet transform and its robustness on image compression," in *Image Processing, 1998. ICIP 98. Proceedings. 1998 International Conference on*, pp. 391-395 vol.2. , 1998
- [164] Y.-S. Kim, *et al.*, "Wavelet based watermarking method for digital images using the human visual system," in *Circuits and Systems, 1999. ISCAS '99. Proceedings of the 1999 IEEE International Symposium on*, pp. 80-83 vol.4. , 1999.
- [165] M.-S. Hsieh, *et al.*, "Hiding digital watermarks using multiresolution wavelet transform," *Industrial Electronics, IEEE Transactions on*, vol. 48, pp. 875-882, 2001.
- [166] L. An and N. Peng, "TinyECC: A Configurable Library for Elliptic Curve Cryptography in Wireless Sensor Networks," in *Information Processing in Sensor Networks, 2008. IPSN '08. International Conference on*, pp. 245-256, 2008.
- [167] L. Batina, *et al.*, "Low-Cost Elliptic Curve Cryptography for Wireless Sensor Networks," in *Security and Privacy in Ad-Hoc and Sensor Networks*. vol. 4357, L. Buttyán, *et al.*, Eds., ed: Springer Berlin Heidelberg, pp. 6-17, 2006.
- [168] A. Y. Yang. 2012, *CITRIC Smart Camera Platform and Applications*. Available: <http://www.eecs.berkeley.edu/~yang/software/CITRIC/index.html>
- [169] 2012, *TinyOS*. Available: <http://www.tinyos.net>
- [170] W. Wang, *et al.*, "Energy Efficient Multirate Interaction in Distributed Source Coding and Wireless Sensor Network," in *Wireless Communications and Networking Conference, 2007.WCNC 2007. IEEE*, pp. 4091-4095, 2007.
- [171] C. Harris and M. Stephens, "A combined corner and edge detector," in *In Proceedings of Fourth Alvey Vision Conference*, pp. 147-151, 1988.
- [172] Z. Shahid, *et al.*, "Considering the reconstruction loop for data hiding of intra-and inter-frames of H. 264/AVC," *Signal, Image and Video Processing*, vol. 7, pp. 75-93, 2013.
- [173] M. Goljan, *et al.*, "Sensor noise camera identification: Countering counter-forensics," in *IS&T/SPIE Electronic Imaging*, 2010, pp. 75410S-75410S-12.

- [174] A. Y. Yang. 2013. *CITRIC Smart Camera Platform and Applications*. Available: <http://www.eecs.berkeley.edu/~yang/software/CITRIC/>
- [175] M.-G. Ko, *et al.*, "Error detection scheme based on fragile watermarking for H. 264/AVC," in *Acoustics, Speech and Signal Processing (ICASSP), 2011 IEEE International Conference on*, pp. 1061-1064, 2011.
- [176] G. J. Sullivan, *et al.*, "Overview of the high efficiency video coding (HEVC) standard," *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 22, pp. 1649-1668, 2012.