Low-complexity Single-view and Multi-view Visual Compression for Visual Sensor Networks using Block-based Compressive Sensing

by

MANSOOR EBRAHIM

A thesis presented in total fulfilment of the requirements
for the degree of

Doctor of Philosophy (Computing)
at
Sunway University
Submission Date: 11th July 2016

List of Publications

The following publications were produced during the time of this project:

Peer Review International Conferences:

- **Mansoor Ebrahim**, and Wai Chong Chai; "Multi-phase Joint Reconstruction Framework for Multi-view Video Compression using Block-based Compressive Sensing". International Conference on Visual Communications and Image Processing (VCIP), 2015, December 2015, IEEE.
- **Mansoor Ebrahim,** and Wai Chong Chia; **"**Multi-view image compression for Visual Sensor Network based on Block Compressive Sensing and multi-phase joint decoding". International Conference on Computational Science and Technology (ICCST), 2014, August 2014, IEEE
- **Mansoor Ebrahim,** and Wai Chong Chia; **"**Secure Force: A Low-Complexity Cryptographic Algorithm for Wireless Sensor Network (WSN)", International Conference on Control System, Computing and Engineering ICCSCE, Nov 2013,IEEE

Book Chapter:

- **Mansoor Ebrahim**, and Wai Chong Chia; "Block Compressive Sensing (BCS) based Multi-Phase Reconstruction (MPR) Framework for Video". Lecture Notes in Electrical Engineering, 2015, December 2015, Springer.

Peer Review Journals:

- **Mansoor Ebrahim**, and Wai Chong Chia; "Multi-view Image Block Compressive Sensing with Multi-phase Joint Decoding for Visual Sensor Network". ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM), September 2015, (ISI indexed).
- **Mansoor Ebrahim,** and Wai Chong Chia; **"**A Comprehensive Review of Distributed Coding Algorithms for Visual Sensor Network (VSN)". International Journal of Communication Networks and Information Security (IJCNIS), vol. 6, no. 2, August 2014. (Scopus indexed).

Acknowledgement

Pursuing this PhD has been a truly life-changing experience for me and it would not have been possible to do without the support and guidance that I received from many people. First of all, I'd like to give my deepest gratitude to my great supervisor, Dr. Wai Chong Chia, who guided me throughout this thesis with his valuable suggestions and advices, most importantly for giving me the flexibility to work. I've learned a lot from his experience and vision. I could not have imagined a better advisor and mentor for my PhD. I also greatly appreciate his dedication in reading and correcting this manuscript.

I am thankful to the FST faculty, especially my PhD coordinator Dr. Yau Kok Lim for his unconditional guidance, help and support during my PhD. I also thank the FST staffs, in particular Ms. Winnie Teo and Ms. Lydia for guiding and supporting in the administrative matters. I thank all my colleagues in the research lab: Aqeel, Kingsley, Quang, Yasir, Yoke, Leslie and Yx-Boon for creating such a friendly working environment and for the many great moments spent together.

A special thanks to my family. Words cannot express how grateful I am to my father, my mother, my brothers and sisters for their prayers, support and love that have made me what I am and helped me through the most difficult periods of my life. The firm personality of my parents has affected me to be steadfast and never bend to difficulty.

Last but not least, I am greatly indebted to my devoted wife Madiha Mansoor and my son Rabie Ahmed. Their patience, endless love and support without any complaint have enabled me to complete this Ph.D. project. My wife took every responsibility and suffered all the bitterness to take care of my son and my family. I owe my every achievement to both of them.

Abstract

A Visual Sensor Network (VSN) is a wireless platform consists of a set of visual nodes, intermediate nodes, and a gateway. Visual nodes are the end devices responsible for capturing and sending the visual information to the intermediate nodes. They will then relay the information to a workstation via a gateway. The use of cameras in the visual nodes has brought with it a set of new challenges because all the nodes are powered by batteries. Hence, energy consumption is one of the main concerns in the field of VSN. One of the solutions to this is to reduce the amount of data transmission by using compression. In this thesis, a compression scheme that focuses on improving the reconstruction of measurements encoded using Compressive Sensing (CS) is proposed. As opposed to conventional compression scheme, the use of CS creates a simple-encoder complex-decoder paradigm that is more suitable for the VSN. On the one hand, the visual nodes, which serve as the encoders, are only required to quantize and transmit the measurements produced by CS. On the other hand, the server, which acts as a joint-decoder, will perform the complex task of exploiting the correlations and redundancies of information collected by different visual nodes. This reduces the amount of processing to be done on the encoder as well as the energy consumption. In the proposed scheme, certain visual nodes are configured to encode and transfer the information (I_{NR}) at subrates lower than others. Image registration and fusion are used to generate projected image (I_P) that closely resembles I_{NR}. This procedure is approximately 2-3 times shorter than the use of Motion Estimation and Compensation. The difference between I_P and I_{NR} at the measurements level is determined, and the difference is added to I_P to produce the final reconstructed output. The proposed scheme can outperform other compression schemes at lower subrates by up to 2dB-3dB when it was applied to images and up to 1.5dB-2.5dB when it was used on videos. In addition to this, the proposed scheme has also been implemented on Arduino and XBee to evaluate its effectiveness in real-world.

Keywords

Visual Sensor Network, Compressive Sensing, Joint-Decoding, Image Registration, Hardware, Encryption

Table of Contents

List of Figures

List of Tables

List of Abbreviations

ADPCM	Adaptive Differential Pulse Code Modulation
API	Application Programming Interface
BCS	Block Compressive Sensing
BP	Basis Pursuit
BPDN	Basis-Pursuit De-Noising
CS	Compressive Sensing
DC	Distributed Coding
DCT	Discrete Cosine Transform
DPCM	Differential Pulse Code Modulation
DSC	Distributed Source Coding
FB	Feature-Based
GD	Gradient Descent
GoP	Group of Pictures
GPSR	Gradient Projection for Sparse Reconstruction
IBR	Intensity-Based mage Registration
ISM	Image Similarity Metric
IST	Iterative Splitting and Thresholding
JMD	Joint Multi-phase Decoding
ME/MC	Motion Estimation/ Motion Compensation
MI	Mutual Information
MSD	Mean Square Difference
MSE	Mean Square Error
NESTA	Nesterov’s Algorithm
NP	Nondeterministic Polynomial
NSP	Null Space Property
OE	One Evolutionary
OMP	Orthogonal Matching Pursuits
PSNR	Peak-Signal to Noise-Ratio
R-D	Rate-Distortion

RIP	Restricted Isometry Property
SpaRSA	Sparse Reconstruction via Separable Approximation
SPI	Side Projection Information
SPL	Smooth Projection Land-weber
SQ	Scalar Quantization
SSIM	Structural Similarity Index Metric
SW	Slepian-wolf
TV	Total Variation
TV-AL3	Total Variation Augmented Lagrangian
VSN	Visual Sensor Network
WSN	Wireless Sensor Network
WZ	Wyner-ziv

Chapter 1
INTRODUCTION

The creation of the low-cost camera has caused Wireless Sensor Network (WSN) [1, 2] to emerge and form the Visual Sensor Network (VSN) [3, 4]. Typically, a VSN is constructed by using multiple visual sensors nodes (encoder) that are tiny devices with limited processing power. These visual nodes are capable of capturing the geometrically correlated visual data of a particular event at the same time from different viewpoints. The captured data is then transmitted to a server (decoder) for further processing.

VSN has been widely adopted in various multi-visual applications, such as 3D reconstructions, multi-camera imaging, armed tracing, and surveillance. One of the greatest features of VSN is their independent nature. If the sensor nodes are simply dropped in the field, they will automatically establish connection with every other node in the range to form a flexible ad-hoc network [5]. However, the use of cameras has increased the amount of information that has to be transmitted in the network significantly, bringing with it greater challenges, such as the limitation of battery, and computational resources (processing power and memory) on the visual sensor [6].

1.1. Motivation

When the VSN is capturing a specific event concurrently with different visual nodes, the acquired visual data are highly geometrically correlated [7,8]. However, such data cannot be directly transmitted as they contain a large volume of visual redundant information.

Consequently, the capturing and encoding of visual data increase the needs of computational resources (processing power and energy). In such battery-powered networks, energy consumption is the most critical issue that needs to be taken into consideration [9].One of the general solutions to this is to reduce the data transmission (compression). To achieve data reduction before transmission, development of an efficient compression and reconstruction scheme that exploits the intra-view and inter-view redundancies is required.

In the literature, many algorithms for compressing single and multi-view visual data have been proposed [10, 11, 12]. However, most of these conventional algorithms require significant amount of processing to be done on the encoder, which increases the energy consumption. Some algorithms have to reduce the visual data quality to work around the hardware constraints mentioned above [13]. Due to the high energy consumption and encoding complexity, the conventional approaches are not attractive for compressing the correlated images captured in a VSN.

Recently, distributed coding has appeared to be one of the more efficient mechanisms in resolving issues related to performing complex processing on small visual sensor. Distributed coding is capable of forming a simple-encoder complex-decoder paradigm, where devices serving as the encoder are only required to perform simple computation to compress the visual data. The server, which usually serves as the decoder, is now required to perform more complex computation in order to reconstruct the compressed visual data. In other words, the computation is shifted from the encoder side to the decoder side. This is the direct opposite of conventional approaches that performs most of the computation on the encoder. In situation where the visual data are correlated, the decoder could also exploit the correlation and make use of the information to improve quality of the reconstructed visual data. It is commonly known as joint-decoding. However, joint-decoding often fails to work when comes to real world operation. The main challenge now is on how to better exploit the correlation practically and make use of the available information.

1.2. Challenges

In distributed compression framework, images are captured independently and are compressed to resolve the issues related to complex processing and energy consumption on small visual sensor. In this case, the decoder has to estimate the correlation of the original scene from the multiple compressed views, in order to perform the joint reconstruction of multiple images as shown in Fig. 1.1.

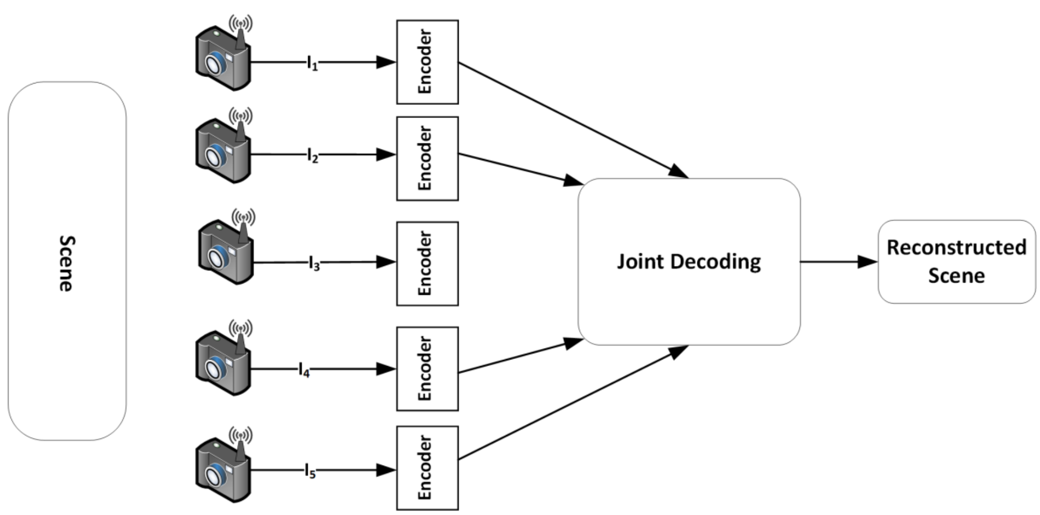


Figure 1.1: An overview of Distributed coding paradigm i.e. images are encoded independently by the respective cameras, but decoded jointly at a central decoder.

In the literature, different schemes are proposed to estimate the original correlation associated between images. However, these state-of-the-art techniques cannot efficiently handle (i.e. fail to capture the actual correlation from compressed views) the compressed images at low subrate, which results in low rate-distortion performance in distributed scenarios.

In this thesis, we need to consider the different correlational and representational challenges related to the processing of visual information collected by multiple sensors. It should be noted that in order to provide a better representation of a scene, we need to efficiently handle the correlation that exist between the images. In this regard, the research conducted in this thesis addresses the following research issues:

a) How to better exploit the correlation in captured images and videos and make use of the information to improve their quality at low subrate when distributed coding is used?
b) How distributed coding can be practically implemented for VSN?
c) How much energy could be conserved using distributed coding in real world operations?

1.3. Aim and Objectives

Motivated by the above discussed paradigm, the aim of this project is to develop a practical visual compression scheme for VSN that provides better coding performance and takes into consideration the energy consumption and security. Based on the nature of

VSN where visual nodes could be deployed unevenly in a field, the scheme should be able to work with minimum calibration. The key objectives are highlighted as follows:

a) Develop a low-complexity single-view and multi-view visual (image and video) compression scheme that gives better performance at low subrate.
b) Create a VSN platform that can be used to evaluate the proposed compression scheme in real world operation.
c) Minimize calibration that is needed for the scheme to operate.

1.4. Overview and Contributions

Generally, the contributions can be categorized into four categories.

a) Review of Distributed Coding

After comparing different distributed coding schemes that could be used to achieve the simple-encoder complex-decoder paradigm, we have decided to develop the proposed scheme based on Compressive Sensing. Among the many distributed coding schemes, Compressive Sensing (CS) is one of the latest schemes that has achieved popularity in recent years and is applied to various imaging applications [14], such as Magnetic Resonance Imaging (MRI) [15, 16] and seismic identification [17].

CS [18, 19] works on the principle of representing a signal (image/video) with only those sample measurements that are necessary for the recreation of the signal. In other words, the sampling rate is much lower than that of the Nyquist rate. One of the leading edge of this is the single-pixel camera [14] that directly reduces the sampling and number of data that will be streaming out. This reduces the amount of data that has to be processed and transmitted by the visual node, but increases the complexity of reconstructing the original signal. In other words, the main challenge now is to reconstruct the original signal from limited number of sample measurements.

b) Low-complexity Single-view and Multi-view Visual Compression

Because visual nodes could be deployed at places that are hard to reach, it is not feasible to reprogram the visual nodes in cases where the settings or configurations on the nodes have to be changed. In this case, we proposed a compression scheme for VSN based on Block-based Compressive Sensing (BCS). An overview of the proposed scheme is shown in Figure 1.2 and our contributions are highlighted with bolded boxes.

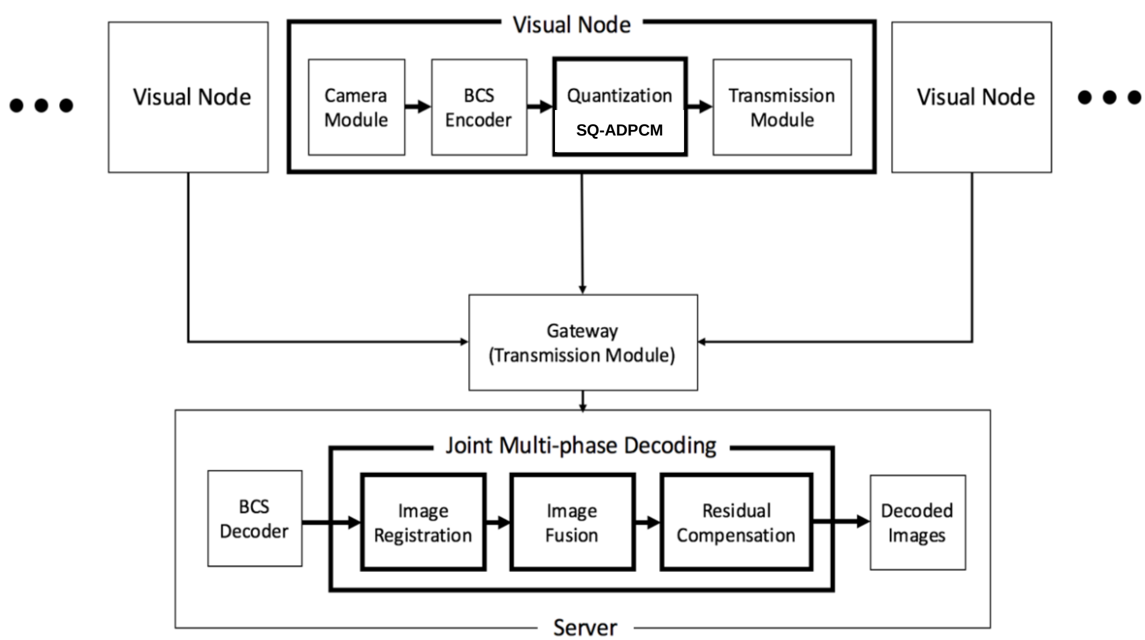


Figure 1.2: An overview of the proposed compression scheme

Consider the cases where there are a set of visual nodes, the images captured by each node are encoded using BCS. The output from BCS is known as measurements, which consists of a set of negative and positive numbers. A proposed Scalar Quantization-Adaptive Differential Pulse Code Modulation (SQ-ADPCM) quantization method is then applied to prepare the measurements for transmission. Each visual node transmits the encoded data to the gateway independently. The gateway relays the received data to the server for further processing.

At the server, the received data is first decoded by BCS to recover the images. The images then go through the proposed Joint Multi-phase Decoding (JMD), where the correlations among the images are exploited. The exploited information is used to improve the visual quality of the decoded images. The proposed JMD involves three main steps, namely (i) image registration, (ii) image fusion, and (iii) residual compensation. Depending on the deployment and configuration of the visual nodes, the entire compression scheme can handle the changes with minimum recalibration or reprogramming. Overall, the proposed scheme is able to cope with three setups, (i) multi-view image, (ii) single-view video, and (iii) multi-view video

Simulation results show that the image registration and fusion approach takes 40% - 50% less time to reconstruct the images and the image quality is on average 2dB - 3dB better when compared to other existing CS-based compression schemes at lower subrate. On the other hand, when the proposed scheme is applied on videos, the proposed scheme takes 50% less time to execute and the reconstructed frames are on average 1.5dB - 3dB better. In addition to this, we have also evaluated the effect of changing the block size in BCS.

c) A VSN Prototype Platform

There is a lack of visual node prototype that capable of taking RAW images (uncompressed imaged). We need RAW images to implement compressive sensing but most of the embedded camera only provides output in JPEG format. Moreover, the output is in the form of compressed bit stream. Unless the bit stream is decompressed on the spot to reconstruct the image, it is hard to directly apply other processing onto the compressed bit stream.

Although CMUCam4 is capable of capturing RAW image, the capturing process is performed in a row-by-row basis. This means that if the observed scene consists of moving object, then a row of image data might see unconnected to the next row of image data. Other prototype such as Mesh-Eye is bulk in size and Cyclops can only capture RAW image at low resolution (352x244). Therefore, it is necessary to develop a new visual node prototype that is capable of capturing RAW images.

We have developed an Arduino Due microcontroller board, XBee transmitter, and uCAM-II camera. In this case, images or videos captured by the camera are first compressed using BCS on the microcontroller. The measurements produced from the compression are then transmitted to the server via XBee transmitter. In order to reduce the memory consumption of BCS, we have also compared the effect of using smaller block size. We noticed that larger block size produces better result than smaller block size at lower subrates. However, larger block size consumes 5%-8% more energy than smaller block size, but the image quality is 0.5dB-1dB better.

1.5. Thesis Organization

The rest of the thesis is organized as follows. Chapter 2 first presents a survey of existing distributed coding schemes from the literature. An in-depth qualitative analysis of the existing distributed coding schemes is carried out and a summary of pros and cons, as well as the open issues in the existing distributed coding schemes is also included. Then, chapter 3 provides the theoretical basics of CS that explains how an image can be compressed into measurements. In addition to this, quantization used to convert measurements to data for transmission is also described. Chapter 4, 5, and 6 describes how the multi-view images, single-view video, and multi-view videos can be compressed by using the proposed compression scheme respectively. The simulation results of the proposed scheme are also presented and discussed in the respective chapter. Next, chapter 7 explains about the hardware implementation of the proposed compression scheme on the VSN platform that we created. This is followed by a low-complexity symmetric key encryption algorithm designed for VSN, denoted as Secure Force (SF), which is described in chapter 8. Finally, chapter 9 concludes the thesis and suggests future directions to the research related to this thesis.

Chapter 2
LITERATURE REVIEW

Distributed coding schemes have appeared to be one of the more effectual, and competent visual data compression mechanism for modern visual processing and sensor based applications in the recent era. This chapter presents a survey of existing distributed coding schemes that are closely related to the problems addressed in this thesis. In addition, the technique that will be followed in our proposed system is also analyzed and compared. In section 2.1, an overview and classification of visual compression schemes for VSN is provided. Next, the existing distributed coding schemes are discussed in details in section 2.2 and 2.3, based on their advantages, disadvantages and open issues. An in-depth qualitative analysis of the stated distributed coding schemes is carried out in section 2.4 based on a set of evaluation criteria. This analysis can help to select the appropriate approach or technique that is suitable to resolve the computational issues in VSN. This is followed by section 2.5 that discusses about the current state-of-the-art visual reconstruction methods adopted by different Compressive Sensing (CS) schemes. As reconstruction methods play an important role in recovering the compressed visual data. Hence, it is important to look into these reconstruction methods. Not only that it would help to better understand the proposed schemes in later chapters, the proposed schemes are to be compared with these state-of-the-art reconstruction methods as well. Finally, in section 2.6 a summary of the stated distributed coding schemes is provided.

2.1 Visual Sensor Network (VSN)

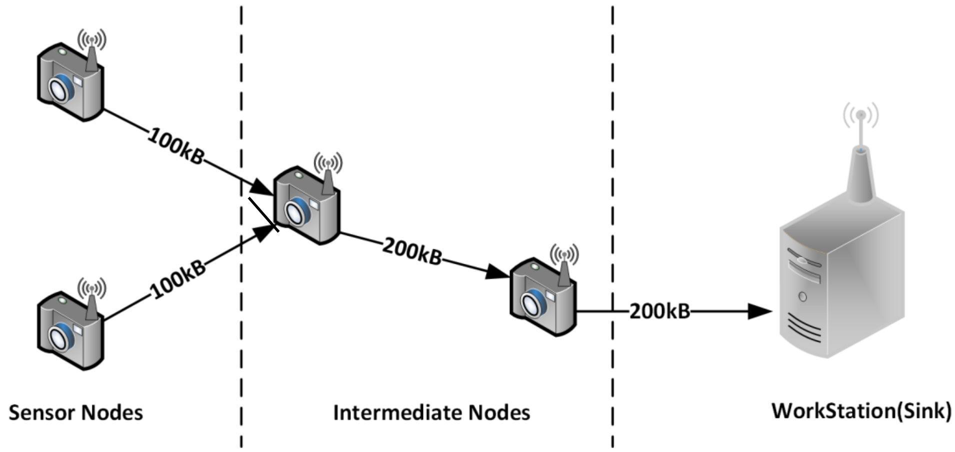
The Visual Sensor Network (VSN) is a type of wireless network that can provide multiple views of a specific event simultaneously from different visual sensors. Due to the resource constraints, direct transmission of large amount of captured visual data in such network is unfeasible. Many efforts have been made to resolve the aforementioned problem. Generally the problem can be addressed in two ways:

i. To take into account redundancy of nodes in a sensor network,

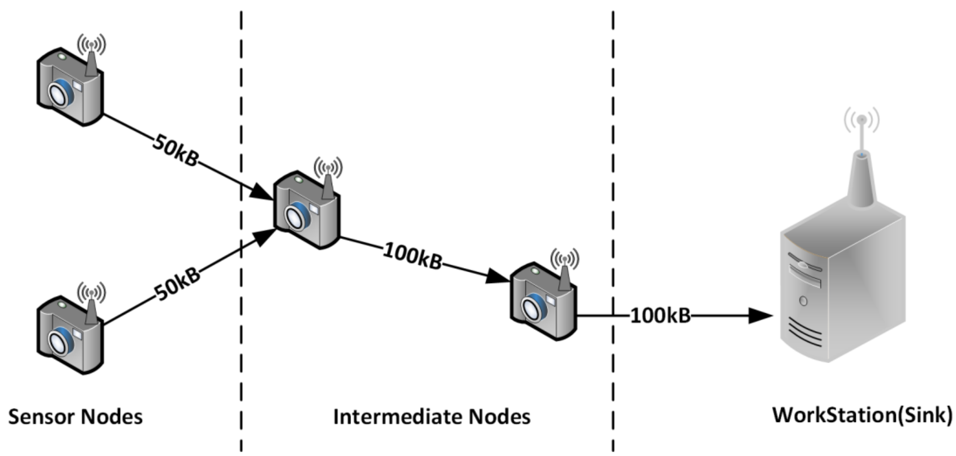
ii. To apply compression schemes to decrease the information transmission.

In the first case, only a subset of visual nodes will be selected and used by turns, avoiding the utilization of too much power by certain visual nodes and thus increase the network lifespan. However, this approach may not work well when there is no redundant node. When visual nodes have to frequently monitor and transmit large amount of visual data for a long time, compression of information is a very expedient solution. It is observed from [20] that the power consumption for data transmission is higher than that of data processing. The research work in [21] stated that the energy cost of transmitting 1kB information is equivalent to the executing of 3 million instructions when using a 100 Million Instructions per Second (MIPS) /W processor.

In VSN, visual data compression schemes are usually implemented before transmission takes place. This results in reducing power consumption and computational complexity. The effect is much more significant when the data have to be transmitted via a multi-hop network as illustrated in Figure 2.1a and 2.1b.



a. Un-compressed visual data transmission



b. Compressed visual data transmission

Figure 2.1: Visual Sensor Network with and without visual compression

The basic idea of compression is to reduce the redundant information that is insensitive to the human eye. Spatial, temporal, inter-view and intra-view redundancies are the few types of redundancies that are commonly found in single and multi-view images and videos. These redundancies are described in Table 2.1.

Table 2-1: Types of image redundancies found in single and multi-view images

Redundancy Type	Description
Spatial Redundancy	Refers to correlation in between adjacent pixels. Thus, the redundant information inside one frame can be removed.
Temporal Redundancy	Deals with reduce amount of bits that are desirable to characterize a specified image or its facts
Intra-view Redundancy	Reduces the multiple redundant information within a single image
Inter-view Redundancy	Used in the case when multiple images are coded and the images contain certain level of similarity.

The compression schemes aim to reduce the stated redundancies to decrease the required amount of data used to represent an image or video sequence [22].

2.1.1 Classification of Visual Data Compression Schemes for VSN

The compression schemes for visual data can be analyzed either as lossless or lossy. In general, the visual compression schemes can be categories into the following subsequent sets as displayed in Figure 2.2 [23]:

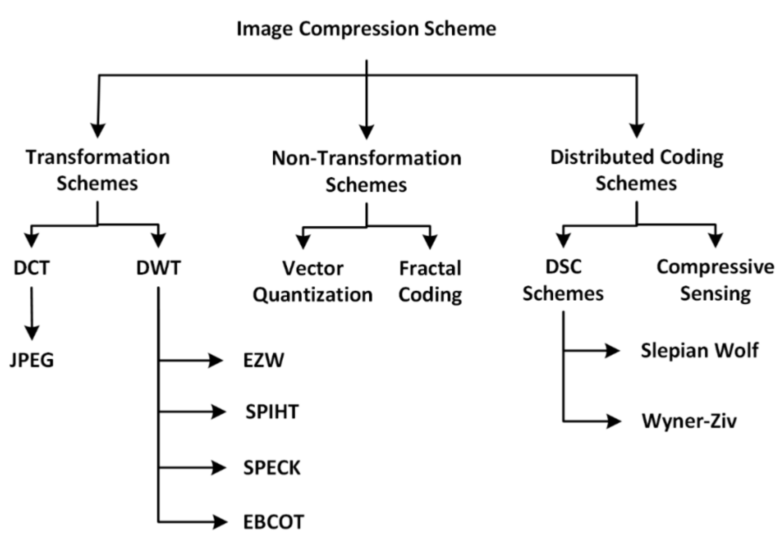


Figure 2.2: Classification of visual compression scheme for VSN

The transformation / non-transformation compression schemes are usually capable of removing the redundancies that appear within the visual data itself. The basic idea of these schemes is to locate and remove the redundancy in the spatial domain. This is to be done at the visual nodes before the transmission takes place. These schemes are not suitable for VSN as they perform all the complex processing at the visual node and require additional side information from the decoder. This increases the computational burden at the battery powered visual node and in result reduces the life span of the network. In cases, where multiple visual nodes are observing the same scene, the field-of-view of the visual nodes may overlap with each other. Hence, it is possible to further reduce the amount of information to be transmitted, by removing the overlapping regions (interview redundancy). In this context, the distributed coding schemes appeared to be a better solution for VSN as they are based on simple-encoder and complex-decoder paradigm. In other words, each visual node (encoder) independently compresses and transmits the data to a central server (decoder) for joint reconstruction i.e. shifting the computational burden from the visual node (encoder) to the server (decoder). The distributed coding schemes will be further discussed in later sections of this chapter. The transformation/ non-transformation compression schemes will not be discussed further. However, the detailed description and comparative analysis of the above-mentioned transformation/non-transformation schemes can be studied from the research papers [23-31].

2.1.2 Evaluation Criteria

In order to implement a suitable visual compression scheme for a particular application, it is essential to know its strength and limitation. Subsequently, the evaluation of existing distributed coding schemes based on certain criteria is necessary. The evaluation is carried out by in-depth qualitative analyses of the existing compression schemes. For qualitative analysis we select a scale of Lowest, Low, Medium, High, and Highest as presented in Table 2.2 to indicate the prospect of each scheme in terms of power consumption, memory utilization, complexity, execution time, and compression ratio.

Table 2-2: Qualitative analysis evaluation scale

Impact	Power Consumption, Memory Utilization, Complexity, Execution Time	Compression ratio
Lowest	• Independent encoding • Few non-zero coefficients < Nyquist rate.	• ≥ 1:1.25 compression ratio
Low	• Independent encoding + Side information • Non-zero coefficients ≤ Nyquist rate.	• > 1:1.6 < 1:1.25 compression ratio
Moderate	• Independent encoding + Side information • Non-zero coefficients ≥ Nyquist rate.	• ≥ 1:2.5 ≤ 1:1.6 compression ratio
High	• Joint encoding • Non-zero coefficients = Nyquist rate.	• ≥ 1: 5 ≤ 1:2.5 compression ratio
Highest	• Joint encoding + Side information • Non-zero coefficients > Nyquist rate.	• ≥ 1:20 ≤ 1:5 compression ratio

In the following sections, different distributed coding schemes are reviewed on the criteria defined in Table 2-2.

2.2 Distributed Source Coding (DSC)

Distributed Source Coding (DSC) schemes have been an area of research since the 1970s, but it gains popularity as a core compression mechanism after the current increase in the adoption of visual sensor applications by the modern world. In contrast to conventional source coding schemes, the DSC schemes are based on the phenomenon of individual compression of correlated sources (visual nodes) that does not interact with each other. In other words, the transmission of a set of independently compressed sensor outputs (no communication with each other) to a common point (base station) for joint decoding [32, 33]). DSC is a fundamental approach that exploits the spatial association of sensors resulting in reduced computational complexity at the visual node (encoder) whereas, increasing the complexity at the central base station (decoder) without performance degradation [34-36].

With the DSC model, one of the major questions is: is it possible to correctly recover the sequences at the decoder, by encoding them at a total rate smaller than the sum of individual entropies? Different DSC based schemes are introduced such as Low-Density Parity Check (LDPC) [37], DISCOVER [38], Power-efficient, Robust, hIgh compression Syndrome based Multimedia coding (PRISM)[39] to resolve the above discuss problem. Though, the theoretical foundation of all these distributed schemes is based on two theorems, the Slepian-Wolf (SW) and Wyner-Ziv (WZ). Both the SW and WZ theorems are discussed in the following subsections. However, for more details the reader can refer to the introductory articles on distributed coding [40, 41].

2.2.1 Slepian-Wolf Compression Scheme

i. Overview

In 1970's, Slepian and Wolf considered the problem related to distributed coding and revealed that a total rate R is sufficient to reconstruct the sources correctly. According to [42], same performance as of joint encoding can be achieved when multiple interrelated sources (in this case, two) are independently encoded and jointly decoded at the decoder as shown in Figure 2.3.

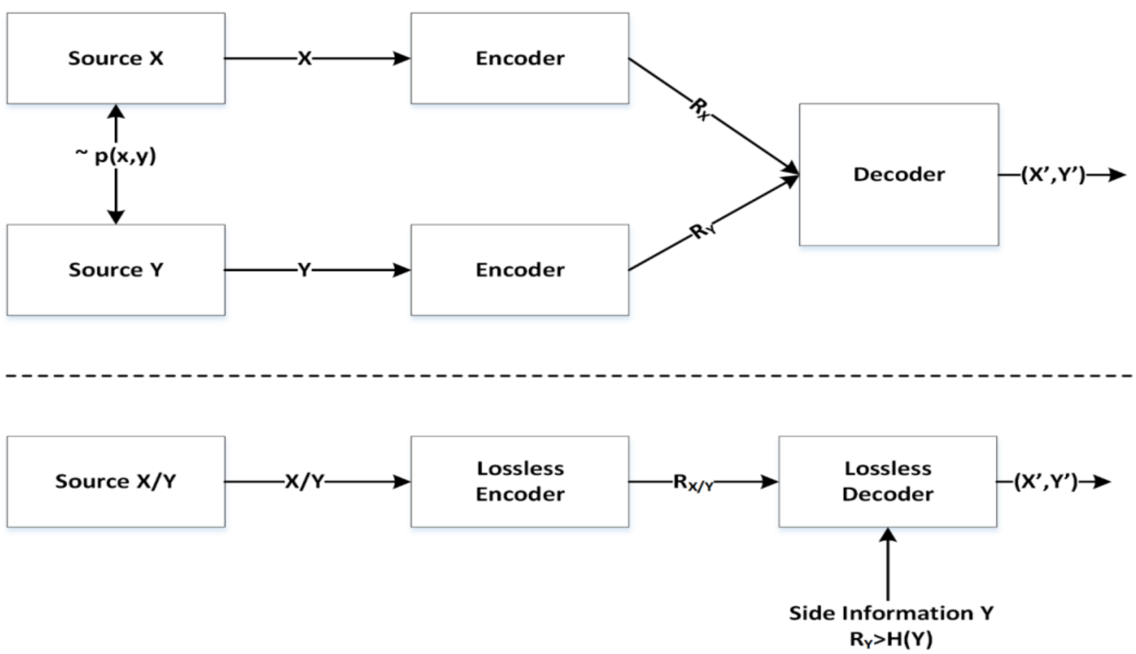


Figure 2.3: General Structure of SW algorithm along with lossless decoder

Such a mechanism is known as SW coding, which is the basis of DSC. As SW is a Lossless compression scheme, therefore, the output is independent of small errors and losses. More specifically, consider two interrelated sources X and Y that are independently encoded but the decoding process is performed jointly. Then, the attainable rate region (R_X; R_Y) for a lossless compression of the interrelated sources X and Y is given as in Eq. (2.1).

$$R_X \geq H(X|Y);R_Y \geq H(Y|X);$$

$$R_X + R_Y \geq H(X, Y) \qquad \textbf{(2.1)}$$

Where, H(X|Y) and H(Y|X) represents conditional entropy while, H(X,Y) represents joint entropy of (X,Y). Additionally, with joint decoding (but separate encoding) if the residual error probability is satisfied for encoding long sequences, the SW shows that much better compression rate can be obtained. In such instance, the SW theorem indicates that the residual error probability inclines to 0, and the minimum rate remains the same i.e. H(X,Y) for long sequences. Figure 2.4 illustrates the achievable rate region.

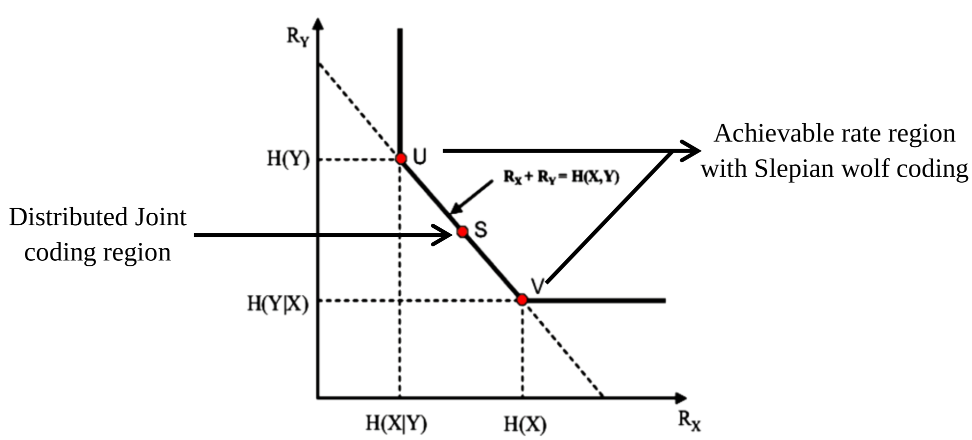


Figure 2.4: Slepian-Wolf (SW) achievable rate region for a set of interrelated sources X and Y

The SW region represents a particular case of the distributed source coding i.e. coding with side information. In such case, the source X is encoded based on the knowledge of Y that is available at the decoder but not accessible at the encoder side. A complete description of the algorithm with mathematical proofs can be found in [42].

ii. Related Work

The SW coding has practical significances for VSN where visual data produced by independent visual nodes are interrelated. The research papers published by different authors in [42-50] have contributed towards the understanding of the concept of SW coding and analyses of its implementation aspects with VSN.

In [43], a distributed linear code construction is proposed for arbitrarily correlated sources to attain any point on the SW achievable region. The research paper also focuses on how the incorporation of LDPC codes into the proposed scheme. In general, an efficient algorithm is provided that can be extended to any number of sources. The simulation result verifies that the scheme performs well for the entire SW rate region for arbitrarily correlated sources, but with small distributed linear codes and block sizes. Further performance improvements are anticipated, with more extensive code selection study and larger block sizes.

Further, in [44-47] problems regarding SW scheme are discussed with their possible solutions, such that [44] proposes a new asymmetric distributed algorithm that makes use of arithmetic codes for the distributed case. In particular, this scheme proposes a distributed binary arithmetic coder for SW coding with decoder side information, along with a soft joint decoder. The scheme provides satisfactory results over SW as well as when compared with turbo and LDPC codes the proposed scheme shows very competitive performance.

The research paper [45] identifies that the SW problem is related to dual channel coding and establish a linear code book duality in between the SW coding and channel coding. This duality leads the way towards the study of linear SW codes regarding trade-off among error rates. Additionally, the linear codebook-level duality is also created for general sources and channels.

The research paper [46] discusses a novel probability proof of the SW theorem with vanishing probability of error for two correlated Independent and Identically Distributed (i.i.d.) sources X and Y over finite alphabets. The encoding process determines that for attaining the standard SW rate region the random codes are linear over the real field and are hence called Real SW Codes (RSWCs). The research work shows that RSWCs can be used in a way that enables the receiver to decode by solving a set of integer programs.

Another constructive approach for the attainability standard SW rate region can be found in [47]. The work suggests an intuitive approach for symmetric and non-symmetric SW coding that can be used with systematic and non-systematic linear codes and can be extended for more than two sources. Specifically, the two correlated sources can be input and output respectively of a certain channel used to model their correlation.

Additionally, [48-50] work on the implementation and analysis of the SW for WSN. The authors have highlighted that SW coding is a promising scheme for WSN because it can eliminate the spatial redundancies. The papers mainly focus on the major problem in the execution of SW in different WSN that includes power constraints and their possible solutions.

The research paper in [48] addresses the rate allocation problem caused by SW coding when dealing with multiple interrelated sources. In addition to this, a novel water-filling algorithm is proposed to identify the optimal rate-point in order to provide a lossless recreation of the sources, while reducing the overall transmission power consumption of the VSN.

The research papers [49, 50] discuss the energy problems caused by SW coding in clustered based VSN and propose possible solutions. In [49] a SW coding based Energy-Efficient Clustering (SWEEC) algorithm is proposed. The proposed algorithm relies on a heuristic algorithm for solving the minimum set weight cover problem in graph theory. The simulation results show that the proposed SWEEC algorithm generates slightly more data (2-3%) than the Slepian-Wolf coding based Energy Minimization Clustering (SWEMC) algorithm, but can significantly improve the overall energy cost (on average 0.12x104) of the network when compared with SWEMC algorithm.

Similarly in [50] a suitable distributed optimal compression clustering protocol (DOC2) is proposed that reduces the volume of data in a clustered network. The proposed algorithm is based on an approximation algorithm for solving the minimum set weight cover problem in graph theory. The simulation results show that the proposed algorithm can considerably reduce the total amount of data in the network while the transmission cost within each cluster can be reduced up to 20-25% by performing the optimal intra-cluster rate allocation.

The performance outcomes of SW coding scheme that are drawn after studying and analyzing the above-discussed research literature based on earlier defined evaluation criteria are presented in Table 2.3.

Table 2-3: Performance analysis of DSC (SW) visual compression scheme

Parameters	Performance
Power Consumption	Moderate power consumption i.e. it completely removes redundancy produced by spatially interrelated data sets. However, in clustered network the power consumption increases due to increasing in the volume of data within each cluster.
Memory Utilization	Minimum memory utilization i.e. it consists of independent encoder phenomenon (correlated data are encoded separately and decoded jointly)
Complexity	Low complexity level at the encoder side i.e. it consists of individual low complexity encoders (shifting major computational load to the decoder).
Execution Time	Moderate execution time (encoding + decoding) as the processing load is shifted from the independent encoders to a joint decoder.
Compression Ratio & Lossyness	Moderate compression ratio. The overall image quality is moderate or high as SW is of lossless nature.

2.2.2 Wyner-Ziv Compression Scheme:

i. Overview

Wyner-Ziv (WZ) [51] scheme was proposed in 1976 and is based on the extended idea of SW theorem. The WZ makes use of lossy compression scheme with the Side Information (SI) feature at the decoder. It also assumes that the sources are mutually Gaussian. In general, consider X (Source data) and Y (side information) are the two statistically dependent sequences, where X (Source data) is encoded by the encoder without accessing Y (side information) as shown in Figure 2.5. The decoder recreates the source data $\hat{X}$ by making use of the side information Y with some adequate distortion D given as:

$$E\,[d\,(\hat{X}, X)] \leq D \qquad \textbf{(2.2)}$$

In such case, WZ shows that the transmission rate increases, when only the decoder exploits the statistical dependency between sources, compared to the case where the dependency is exploited at both the encoder and the decoder. Mathematically, the WZ theorem is given as in Eq. (2.3)

$$R^{WZ}(D) \geq R_{X|Y}(D)\ ,\ D > 0 \qquad \textbf{(2.3)}$$

Where, $R^{WZ}(D)$ and $R_{X|Y}(D)$ are rate-distortion functions with the average distortion D. $R^{WZ}(D)$ represents the minimum encoding rate for X when Y is available only at the decoder, and $R_{X|Y}(D)$ accounts for the minimum encoding rate for X, when Y is simultaneously available at both the encoder and the decoder. A complete description of the algorithm with mathematical proofs can be found in the research paper [51].

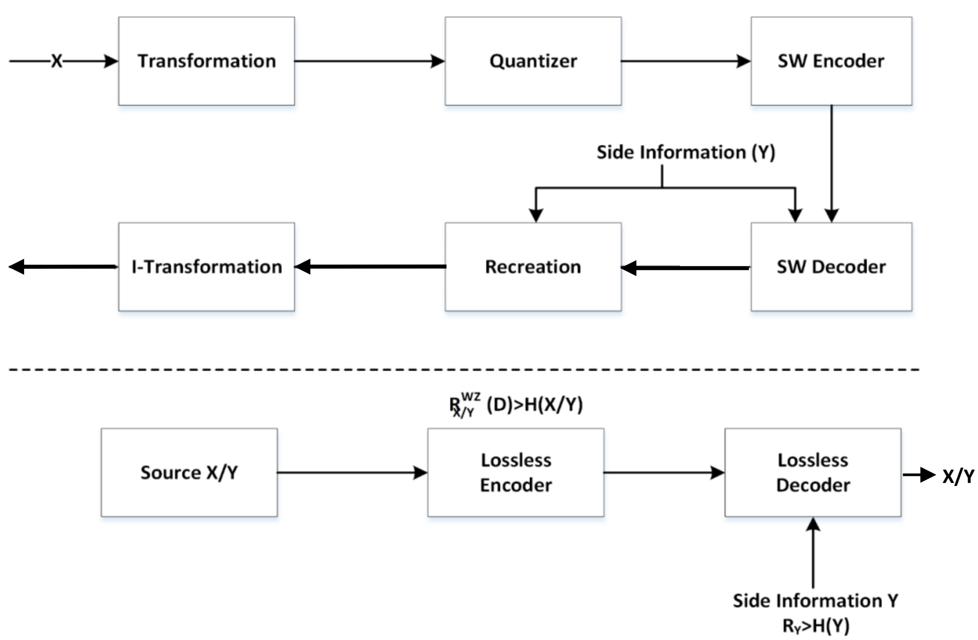


Figure 2.5: Overall block diagram of WZ encoder along with lossy decoder

ii. Related Work

The research work by different researchers in [51-58] leads the way towards the exploration of the WZ coding scheme. The research papers [52, 53] provide different approaches by extending the WZ scheme for multimedia networks.

The research work published in [52] is based on the extended version of WZ scheme implemented in a network scenario having multi-camera in which the base station also has a camera attached to it. The paper extends the work on the previous result by increasing the number of sources to more than two. Further, it derives an achievable rate distortion region and outer bound to the best rate distortion region (only possible where the sources are independent of the SI).

In [53], a modified version of WZ that focuses on using Motion Estimation (ME) parameters to improve the efficiency of video coding is proposed. In most of the video coding schemes, ME is performed at the decoder without the availability of the current frame. This results in imprecise ME that causes degradation of coding efficiency. This paper provides an analytical model for the estimation of potential gain by using Multi-resolution Motion Refinement (MMR) and assumes that the decoder has limited access to the current frame. The experimental results show that, at high rates, the coding performance of using MMR is lower than H.264 coding by 1.5dB. However, it outperforms WZ video coding using motion extrapolation by 0.9dB to 5dB. Simulations show a significant gain using real video data.

The research carried out in [54], aims to discuss, analyze and compare the two early WZ video coding solutions (Stanford, and PRISM [39]), notably from the functional aspects. The paper also analyzes some important developments of the Stanford WZ coding architecture. In contrast, the Stanford design works at the frame level and is describe by feedback channel based decoder side rate control. The feedback channel approach simplifies the rate control problem as the availability of side information is known to the decoder. However, such approach limits the extent to real-time applications and the delay related to feedback channeling need to be considered. Moreover, Stanford architecture is exposed to error corruption. While, the PRISM architecture, works at the block level and is described by encoder side rate control based on the availability of a reference frame based. The block-based coding in PRISM helps to accommodate coding adaptability to address the high motion statistics of video signals. Further, it is more resilience to error corruption due to its motion search approach performed at the decoder.

Further, the paper highlights many improvements that have been proposed in the initial Stanford WZ video codec. Some of the improvements include (i) removal of feedback channel that enhances its scope to real-time applications, however, such variant with encoder side rate control (without) feedback channel result in a loss up to 1.2 dB, when compared to decoder rate control solutions. (ii) The addition of selective Intra coding of blocks in the WZ frame that allows selecting a coding mode adapted to the available temporal correlation. (iii) Improving error resilience by adding redundant information that is encoded based on WZ video coding principles

In [55, 56] the use of WZ in VSN is conducted. The energy efficiency of different video coding schemes for predictive and distributed video encoding paradigms over the sensor platform were evaluated and presented in [55]. For predictive video coding, the compression-communication tradeoffs between two variant of predictive video coding i.e. inter and intra video coding are analyzed. The analysis performed by [55] shows that the inter-coded video utilizes much more energy (on average 763.68mJ/frame) than the intra-coded video (on average 60.03mJ/frame) due to the use of motion compensation/compensation block. For distributed video coding, two prominent techniques, namely the PRISM and WZ are analyzed. The results in [55] suggest that the WZ encoder consumes 40% less energy than the PRISM encoder. Moreover, the paper proposes variations to present video encoders to improve their energy efficiency.

In [56], the WZ problem, i.e. coding of the source data with SI existing only at the decoder in the form of a lossy scheme of the source in discussed. The paper explores both the theoretical and numerical designing aspects of WZ scheme based on multi-dimensional nested lattices. Further, a precise calculation from high-resolution assumption is also developed. The results from the above assumption can be used to analyze the performance and can assist as an applied director in selecting worthy lattices for WZ coding. However, the mentioned work has some open problems such as the upper bound expressions used must be improved; the maximization of theta series ' derivative due to the upper bound is another issue. Moreover, the need for more systematic approach to low-complexity code design is to be followed.

Further, in [57] another approach for multimedia coding is proposed to focus on developing a low complexity WZ video codec with intra-frame encoder and decoder. In this work, the WZ video coding is improved by using run length coding scheme for high-frequency coefficients and utilizing them at the decoder side for accurate motion estimation. Such scheme allows the implementation of low delay system with SI generated from the previously reconstructed frame. The experimental results (simulations) verify that proposed WZ codec for low motion videos shows 6dB-8dB PSNR improvement and a bit rate savings of 60 to 70% over traditional DCT-based intra-frame coding. Whereas, for high motion videos the improvement is around 1.5 dB and bit rate savings is about 15 to 20%.

In addition to this, in [58] a new WZ based multi-view video coding (MVC) is proposed, in which the complex processes (temporal and interview correlation) are shifted from the encoder to the decoder side. The core part of proposed approach is based on wavelet and SPIHT-based WZ video coding. Both the proposed wavelet and SPIHT-based WZ video coding scheme are much better (2dB-3dB) than the H.263+ intra coding. Moreover, the proposed SPIHT-based WZ video coding also outperforms the wavelet-domain WZ coding up to 1.2 dB. The results in [58] show that the proposed MVC system with joint temporal and inter-view prediction outperforms the H.263+ intra-coding up to 7 dB, and also outperform the WZ coding with only temporal prediction up to 1.5 dB.

The performance outcomes of WZ coding drawn after analyzing the above-discussed research papers based on earlier defined evaluation criteria are presented in Table 2.4.

Table 2-4: Performance analysis of DSC (WZ) visual compression scheme

Parameters	Performance
Power Consumption	Low power consumption as in WZ source data is individually encoded at the encoder and jointly decoded at the decoder. So it involves intra-frame coding and no predictive coding (motion approximation, recompense).
Memory Utilization	WZ is based on the principle of SW, so it also provides minimum memory utilization.
Complexity	Low complexity level as it consists of low complexity encoders i.e. efficient subdivision of the convolution between the encoder and decoder
Execution Time	Moderate execution time (encoding + decoding) as the processing load is shifted from the independent encoders to joint decoder.
Compression Ratio & Lossyness	High compression ratio. The overall image quality is moderate as WZ is of lossy nature.

2.3 Compressive Sensing (CS) Scheme

i. Overview

Compressive sensing (CS) [18, 19] is an emerging technique that has open new domain for effective transmission of correlated data. The conventional methodology towards the transformation of the signal is based on the Shannon sampling theorem (the so-called Nyquist rate). The CS works on the principle of representing a signal with a few non-zero coefficients, lesser than that of Nyquist rate. The CS scheme effectively reduces the computational requirements such as memory, processing power, and transmission bandwidth at the encoder by combining signal acquisition and dimensionality reduction into a single phase.

CS is effectual in two situations. First, when direct measurements of a high-resolution signal are hard to attain. Secondly, when encoding of one or more high-resolution signals is complicated. In literature, CS is a standard and is not specified for any particular signal other than underlying sparsity suppositions. However, the CS scheme has attained much attention for image and video coding, and its hardware implementation in the form of single pixel camera [14] has been created, with different schemes proposed for signal/image reconstruction based on such mechanism.

CS allows high prospect of signal recreation by using a minimum number of unsystematic estimations, provided that the signal/image is sparse. Unlike conventional compression scheme, in CS the visual node (encoder) only captures the signal measurements rather than the whole signal, this helps in reducing the computational complexity and bandwidth. Consider a real-valued signal x with length N from M measurements ($M \ll N$) is to be recovered, the signal must have some sparse representations in the transformation domain Ψ with random measurement matrix Φ as shown in Figure 2.6. Then, the set of measurements y is given as:

$$y = \Phi x \qquad \textbf{(2.4)}$$

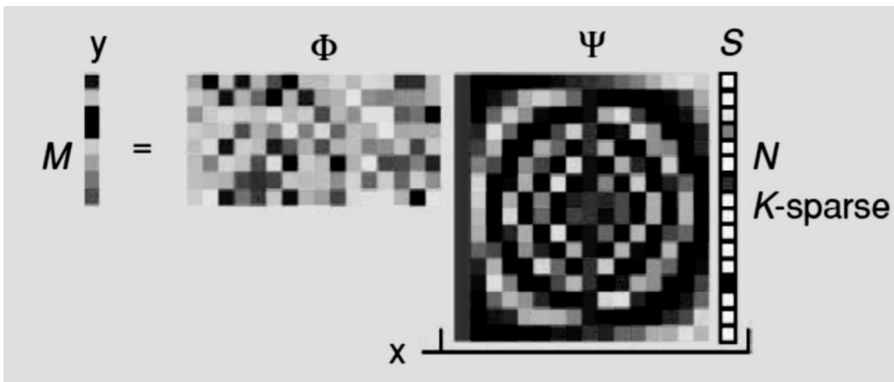


Figure 2.6: Compressive sensing acquisition process with a random measurement matrix Φ and transformation matrix Ψ. The K-spare vector of coefficients S [19]

It is also assumed that Φ is orthonormal i.e. $\Phi\,\Phi^T = I$. Where, I is the identity matrix. Nevertheless, to recover x from such small measurements is not directly possible, i.e. inverse projection of $x = \Phi^{-1} y$ is ill-posed [**95**]. But since x has some sparse representations in Ψ, x can be reconstructed from the sparse representations $\hat{x} = \Psi x$ by solving the ℓ_0 optimization problem that can be expressed as:

$$\hat{x} = argmin_{\hat{x}} \|\hat{x}\|_{lo}, \text{ s.t. } y = \Phi\,\Psi^{-1}\hat{x} \qquad \textbf{(2.5)}$$

The block diagram of CS scheme is presented in Figure 2.7. Detail description of CS scheme with mathematical proofs can be obtained from [18, 19].

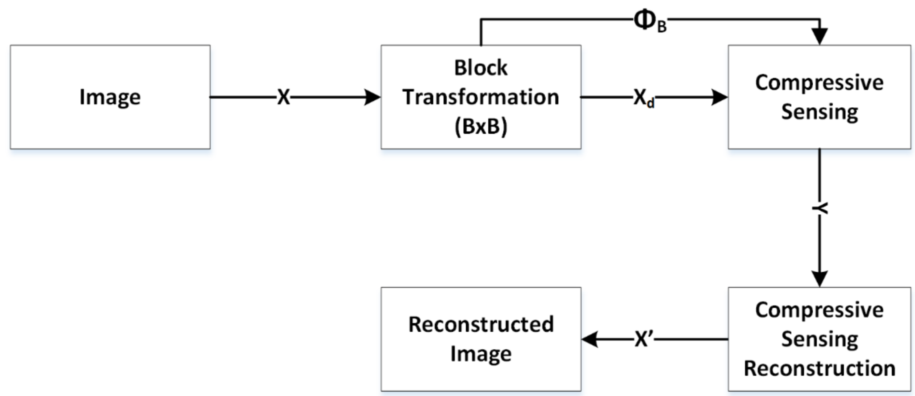


Figure 2.7: Block diagram of Visual compression and decompression by using compressive sensing scheme

CS is based on the central concept of signal representation using a set of linear, non-adaptive measurements i.e. a representation of signal/image by making use of the few non-zero coefficient (sparse expansion) present in the source. In such schemes, the reconstruction of the signal/image from a small set of measurements can be performed by using the nonlinear optimization. However, the CS reconstruction must satisfy two properties: Sparsity and incoherence i.e. the signal must be sparse in some domain, as well as the encoding matrix and the sparsity basis must satiate [68].

ii. Related Work

A comprehensive discussion on different aspects of the CS scheme can be found in [18, 19, 59-67]. The research work in [18] provides a detail introduction and analysis of the theoretical and mathematical aspects of CS, and it also discusses its potential applications in signal and image processing. Similarly, in [19] a survey of the theoretical features of compressive sampling is performed, discussing its fundamental principle based on Sparsity and incoherence i.e. signals/ images can be recovered from a small number of samples or measurements other than the conventional methods used. Whereas, in [59] CS is discussed as an alternative to Shannon/Nyquist sampling for sparse attainment or signal compression. The paper mainly focuses on the substantial performance gains that can be achieved by utilizing more realistic signal models other than simple sparsity and compressibility (inclusion of dependencies) among values and locations of the signal constants that are governing the CS writings.

Further, in [60] the mathematical characteristics of the CS for sparse signals and question related to training and optimal linear projections are discussed. Furthermore, different experimental results are performed to answer the related question. The outcomes show that the trained projection sets can provide much better results than optimal projections.

The applications of CS for WSN data gathering and energy efficiency have been studied in a few papers [61- 65]. The work done in [61] delivers current reviews of CS implementation on WSN. This paper shows that CS embraces encouraging enhancements to reduce the particular constraints of WSN such as power depletion, lifetime, time delay and cost. Also, it also analyses the effectiveness of implementing CS on WSN. The CS scheme combines the data collection and compression steps into a single step and does not require intermediate steps to attain the signals. Hence, transmitting the entire image, only a smaller amount of image is required to be transmitted or stored. This paper leads the way towards the improvements revealed by the application of CS in WSN in terms of power management, lifetime, and time delay.

In [62] study and performance (energy, latency) analysis based on the implementation of CS for data gathering in WSN is conducted. In advance, a data gathering problem in WSN is expressed, and different solutions are proposed, i.e. tree based and gossip based protocols scalable with energy and latency necessities. The experimental results show that both the protocols perform better for data gathering in WSN in terms of energy and latency. However, a tree-based protocol is vulnerable to the link lost.

On the other hand, [63] presents the first complete design for the application of CS to gather data for large scale WSN. The benefits that can be delivered by the proposed scheme includes the reduction in communication cost without increasing computational complexity, the maximum lifetime as well as it can handle unusual sensor outputs efficiently. Moreover, this novel scheme is tested practically, and the experimental results verify its competence and toughness. However, the scheme is not appropriate for small-scale sensor networks (due to limited signal sparsity).

The research paper [64] focuses on the temporal-spatial field measurement (data collection) issue in WSN that utilizes maximum energy and proposes a scheme based on CS that gathers data from WSN without using maximum energy. The proposed scheme was designed with the idea of performing repeated projections to maximize the data volume gain per energy costs. The scheme was tested both theoretically (simulations) and practically (real WSN), and the experimental results in [63] show that the proposed scheme provides a perfect approximation of the indefinite data for assuming energy cost.

The research paper [65-67] discusses the implementation aspects and the possible outcomes of CS scheme. [65], inspects the gains that can be achieved by the application of CS for data (image) collection in WSN, for these two different approaches were proposed (plan-CS and hybrid-CS) in the form of a particular data collection mechanism. The schemes were formulated and were helpful in solving flow-based optimization problems. However, the experimental results show that the first approach does not show any improvement, whereas, in the hybrid CS approach, a substantial improvement can be seen in the throughput. Further, the results were only tested for low-power systems only.

In this context [66] proposes a scheme for altering CS sample volume and update signal recreation in WSN resulting in a reduction in computational complexity, energy and processing time. The proposed scheme was tested theoretically (simulations) in numerous WSN conditions and the simulation results provided in this paper show that the proposed scheme can recreate the output signal by utilizing small sample volumes. Thus, provides better performance in terms of resource utilization and energy efficiency for WSN.

In [67], a survey based on the theory of CS as well as implement and analysis of the fundamental principle of CS (the signal/image can be recreated by making use of the limited volume of samples or measurements) in VSN are provided. Further, the results obtained by CS were compared with the results of JPEG Compression standard that indicates that CS is better in performance (power, memory, complexity, image quality) than the DCT based JPEG scheme.

The performance outcomes of CS coding scheme that are drawn after scrutinizing the above-discussed research papers based on earlier defined evaluation criteria are presented in Table 2.5.

Table 2-5: Performance analysis of CS visual compression scheme

Parameters	Performance
Power Consumption	Lowest power consumption as it reduces the total amount of data to be processed such that it recreates the signal by using only fewer sampling values lesser than that of Nyquist rate.
Memory Utilization	Minimum memory utilization as it acquires and compresses the data at the same time. Therefore, there is no need to store the data before compression.
Complexity	Low complexity level as it consists of simple and low complexity encoder and decoder.
Execution Time	Moderate execution time (encoding + decoding) as it consists of a low complexity encoder and decoder. But, initially, it requires the processing of the whole image information from all the sources connected.
Compression Ratio & Lossyness	The CS scheme provides a High compression ratio. The overall image quality is Low i.e. it makes use of a minimum number of sample measurements to encode the image data to the reconstruction of images from such small measurements is difficult.

2.4 Comparative Analysis of Distributed Coding Schemes

In this section, the above discussed distributed coding schemes are analyzed on the earlier defined evaluation criteria. The comparative analysis outcomes presented in Table 2.6 shows that the DSC (SW) scheme provides low complexity encoders, low execution time, and low memory utilization. The power consumption of SW scheme is moderate as it removes the redundancy of data produced by spatially interrelated sources that result in low energy consumption. Furthermore, the use of simple-encoder and complex-decode paradigm reduces the complexity of all independent encoders. As SW scheme, is a lossless scheme so it provides good quality image outputs with moderate compression ratios. Additionally, the overall execution time is moderate as the decoding process requires more processing as compared to the encoding process.

Table 2-6: Comparative analysis of existing visual compression schemes

	Compression Schemes		
Parameters	**SW**	**WZ**	**CS**
Power Efficiency	Moderate	Low	Lowest
Memory Utilization	Low	Low	Lowest
Compression Ratio	Moderate	High	High
Complexity	Low	Low	Low
Execution Time	Moderate	Moderate	Moderate

However, the real-world implementation of such scheme is very complicated, because each visual node is required to have the correlation structure of the entire network. Moreover, in few cases, the power efficiency of SW scheme might be unstable, and the energy consumption increases by 15% when it is applied to clustered network, network with multiple interrelated sources, and when the dual channel is used for data transmission. This is due to the increase in the amount of data that has to be processed within each cluster.

The power consumption of WZ is low such that it consists of an individual encoder and joint decoder (low complexity) involving intra-frame coding. The scheme is also robust against channel coding and noise errors, as it does not make use of the predictive looping scheme and delivers independently scalable codec as with the case in SW scheme.

However, due to high compression ratio and lossy nature of WZ scheme the output image quality is not that high because it does not provide efficient handling of the compressed image data throughout the whole time (encoding and decoding). Furthermore, WZ scheme might be exposed to some rate losses (system loss, source coding loss).

CS provides the lowest power consumptions such that it focuses on the fact that original signal can be recreated by capturing only the necessary samples of the original signal rather than signal as a whole. Furthermore, the sampling rate of CS scheme is much smaller than that of Nyquist rate. In addition to this, the CS scheme has low memory requirements as it does not require prior information regarding the data deals. Further, the complexity level of CS scheme is low because it consists of low complexity independent encoder and joint decoder. The compression ratio in CS scheme is high due to the reason that it makes use of a minimum number of measurements to compress the signal.

However, in certain cases CS initially requires the accomplishment of the whole signal information from all the sources connected so, in such cases the compression ratio is moderate. With this high compression ratio, the restoration of a signal is very difficult and might result in low signal quality. Furthermore, CS is only applicable for sparse data and its practical implementation in the actual world is very complicated, time consuming and expensive. The reason behind is that the CS scheme does not require prior information regarding the data (image/video). Thus, the prediction regarding the sparse sources in a particular transformation domain is a challenge. Also, it initially requires complete source (Image / Signal) information (information regarding all the sources that the visual nodes are sending) which requires extra cost and time.

To summarize the above discussion, CS scheme has an edge over the SW scheme and WZ scheme. The CS scheme provides lowest power consumption (it does not require prior information regarding the signal as the case with SW and WZ) and minimum memory utilization (recreates the signal with less information that results in less memory utilization). Further, it is less complex (based on independent encoding and joint decoder) and requires shorter execution time. Although CS has been envisioned as a useful technique to improve the performance of VSN, it is still not very clear how exactly it will be applied and how big the improvements will be when comes to real-world operation. This is due to the reconstruction process of CS that is difficult and hard to implement. This often causes the restored signal to be of low quality. After decided that CS is more suitable for VSN, several reconstruction schemes commonly adopted in CS are reviewed in the next section.

2.5 CS based Visual Reconstruction

The use of CS in the compression of multi-view visual data (image and video) has gained substantial attention in the recent years. The basic idea is to sample each multi-view visual data independently using CS and then a joint reconstruction scheme is applied to exploit the correlation within the multi-view visual data. However, only the state-of-the-art schemes providing better results than others are discussed. Later, the experimental results of the proposed scheme presented will be compared with those acquired from these state-of-the-art schemes.

2.5.1 Dictionary based Coding

A joint reconstruction scheme for multi-view visual compression is proposed by [69] that make use of the geometrical correlation model rather than simply parsing the image difference. The proposed scheme samples the multiple images independently with random projections and then reconstruct them jointly at the decoder with redundancies represented as local geometric transformations. The proposed scheme solves the ℓ_1 norm-ℓ_2 norm optimization problem and reduces the Mean Square Error (MSE) distortion of the reconstructed image. However, the images must be sparsed and correlated over a structured dictionary.

A similar approach was adopted by Li et al. [70], in which independent encoding and joint decoding of multi-view images is performed. The joint reconstruction is expressed as an unconstrained optimization problem. Additionally, an iterative algorithm for solving the optimization problem is also presented. The proposed scheme independently imposes sparsity in each image captured by the visual node, as well as in the view-to-view difference images in the neighboring visual nodes.

2.5.2 3D Transformation coding

The principle of joint reconstruction scheme has also been featured in the work done in [71, 72]. The research work employs a non-collaborative geometric manifold lifting framework for joint reconstruction. The proposed framework in [71] uses a set of cameras to observe overlapping of a larger scene to form multiple images from a distance (far-field imaging). The scheme is more focused towards the far-field problem. Such approach is relatively simpler to implement than the full multi-view setup because far-field imaging will not have to deal with problems such as parallax. Moreover, the scheme assumes that the views must be along a low-dimensional manifold to exhibit view-to-view correlation.

In [72], the author extends the work done in [71] and also considers the case of monitoring a scene from a closer distance (near-field imaging). The work suggests that near-field imaging has to be handled by employing a plenoptic function (hypothetical 5D function used to describe the intensity of light observed from any point in space) with the manifold framework. Moreover, the proposed work also left with a few open questions: (i) accuracy of Isometric Map embedding depends on the relative size of the multiple images with a scene, (ii) realistic embedding of reduced multiple images require significant amount of cameras positioning, (iii) require an extensive amount of computation for large images.

In [78], a CS-based video reconstruction scheme that recovers each frame within a video sequence independently using 2D Discrete Wavelet Transform (2D-DWT) is proposed. However, the temporal correlations between consecutive frames were not taken into consideration. An alternative approach is to exploit the temporal correlations by makes use of 3D-DWT and reconstructs a group of frames all at once [78]. But the increase in dimensionality also leads to the increase in memory requirement and computational burden.

[79], applied coded aperture mask designs to each frame and attempted to solve multiple frames altogether to exploit the correlations between consecutive frames.

In [80], 3D transformation is used in combination with Motion Estimation (ME) at the decoder. Each frame is encoded independently with random CS measurements. A multi-scale CS based video reconstruction scheme is presented for reconstruction. It uses an iteration mechanism between Motion Estimation (ME) and sparsity-based reconstruction of the frames themselves. The scheme rests on the LIMAT method (use Motion Compensation (MC) to improve sparsity in the 3D-DWT) for standard video compression.

2.5.3 Residual Based Coding

In [73] a residual based reconstruction scheme named as Modified-CS-Residual is presented. The proposed scheme deals with the reconstruction problem related to sparse signals from a minimum number of linear projections, assuming that some side information is known. The side information is generated by using Least Mean Squares or Kalman filtered based prediction methods. However, the side information can be exposed to errors. The principle of the proposed scheme is to solve a convex relaxation related with data constraints and sparsity outside the side information.

Another reconstruction scheme based on the above-mentioned principle is presented in [74,75]. The scheme is known as k-t FOCUSS. The scheme relies on Disparity Compensation (DC) and Disparity Estimation (DE) based prediction and residual encoding, leading to the optimal sample allocation between the prediction and residual encoding steps. k-t FOCUSS assumes that there exists multiple key frames, i.e. side information, and then CS reconstruction is performed by taking residuals between each non-key frame and a bidirectional (DC/DE) prediction for each of the key-frames.

Another joint reconstruction scheme based on side information and residual reconstruction is introduced in [76]. The scheme incorporates the use of DC/DE based prediction to establish the side information and it is, known as DC-BCS-SPL. The residual between the side information and the original view is calculated to aid the reconstruction of the final view.

Also, in [77] a joint reconstruction algorithm based on DC/DE is proposed. This proposed scheme makes use of proximal gradient method to solve the optimization problem.

In [81], an ME/MC based scheme is presented. The scheme incorporated MC/ME into the reconstruction process of BCS-SPL for video and referred as MC-BCS-SPL. Initially, block-based random CS measurements are applied frame by frame for the video sequence. Then, the decoder incorporates the reconstruction from an ME/MC-based residual; the proposed MC-BCS-SPL scheme alternatively reconstructs frames of the video sequence and their corresponding motion fields, using one to improve the quality of the other in an iterative fashion.

Also, in [82] a joint reconstruction algorithm based on MC/ME and fusion is proposed. The proposed scheme first down-sample the views and then makes use MC/ME and fusion approach to generate a view prediction, which aid in the reconstruction of final view.

A joint reconstruction scheme based on view prediction and residual reconstruction is introduced in [83]. The scheme incorporates the use of MC/ME into the reconstruction to establish the side information. The side information then aids to the reconstruction of final improved view.

The above discussed residual coding schemes are exposed to few issues such as inaccurate prediction and computational complexity. For example, the scheme by [73] make use of Least Mean Squares or Kalman filter based prediction methods that are more suitable for video applications, due to the assumption that sparsity pattern evolves gradually from frame to frame. For the schemes [74-77, 81-83], that employs DC/MC and DE/ME prediction methods, accurate predictions are hard to achieve with basic transformation (translation / affine) model. This is because images captured from different view angles may exhibit some deformations, and this issue has to be resolved by using more complex transformation model, leading to the additional computational burden.

A summary of the various CS based single and multi-view visual (image and video) reconstruction schemes that we have reviewed are shown in Table 2.7.

Table 2-7: Summary of various CS based single and multi-view visual reconstruction schemes

Approach	Work	Scheme	Description	Issues
Dictionary Based Coding	**69**	**Geometrical Correlation**	Independent sampling of images with random projections and joint reconstruction with redundancies represented as local geometric transformations the decoder.	Images must be sparse and correlated over a structured dictionary. The learning and usage of dictionaries is computationally intensive. In particular, searching the sparse representation of a signal in these dictionaries requires solving an optimization problem that leads to very long computational times, especially in 3D imaging.
	70		Similar to [69].However, the joint reconstruction is expressed as unconstrained optimization problem that contains two regularization terms that are used to capture the correlation. An iterative algorithm is proposed to solve the optimization problem.	
3D Transformation Coding	**71**	**Non-Collaborative Geometric Manifold Lifting**	Uses a set of cameras to observe a scene from a distance (far-field imaging). Make use of geometric manifold lifting framework for joint reconstruction. Relatively easier to implement in case of far-field imaging as will not have to deal with problems such as parallax.	Mainly focuses on the basic far-field problem of a single large scene. Require large amount of camera positioning. In case of near field problem need to handle the parallax and occlusion issues along with large camera positioning that leads to additional computational complexity.
	72			
	78	**Wavelet Transform**	CS-based video reconstruction scheme that independently recovers each frame within a video sequence using 2D Discrete Wavelet Transform (2D-DWT). Also make use of 3D-DWT for the reconstruction of multiple frames.	Does not consider temporal correlations between consecutive frames. Increase in dimensionality also increases the memory requirement and computational burden
	79	**Coded Aperture**	CS-based video reconstruction scheme that attempts to solve multiple frames altogether to exploit the correlations between consecutive frames.	Projections recorded based on coded apertures have weaker theoretical guarantees and such approach is hard to implement practically.
	80	**3D-Transformation & Motion Estimation (ME)**	Each frame is encoded independently with random CS measurements. It uses an iteration mechanism between Motion Estimation (ME) and sparsity-based reconstruction of the frames themselves.	3D transformation requires additional memory and increases computationally complex due to increase in dimensionality. Accurate estimation is hard to achieve with ME using basic estimation algorithms as images/frames captured from different view angles may exhibit some deformations.

<table>
<tr><td rowspan="7">Residual Based Coding</td><td>73</td><td>Kalman-Filter (KF) based Prediction</td><td>Deals with the reconstruction problem related to sparse signals from a minimum number of linear projections, assuming that some side information is known.

Solve the convex relaxation related to data constraints and sparsity outside the side information</td><td>Suitable for video applications, due to the assumption that sparsity pattern evolves gradually from frame to frame.

Prediction fails if the knowledge of the state dynamics and measurement models is imprecise or due to inaccurate initialization of the filter.

The problem can be overcome by using a tree structured KF algorithm but it will lead to computational complexity.</td></tr>
<tr><td>76</td><td rowspan="2">Disparity Estimation & Compensation (DE/DC) based Prediction</td><td>Incorporates disparity prediction methods into the reconstruction process of BCS-SPL to establish the side information.</td><td rowspan="6">Introduces discontinuities at the block borders (blocking artifacts).

Accurate predictions are hard to achieve with basic estimation and compensation algorithms as images/frames captured from different view angles may exhibit some deformations.

May result in producing false edges and ringing effects.

Requires more complex estimation and compensation algorithms that results in additional computational burden.

Significant improvements at higher subrate as compared to lower subrate as at lower subrates the scheme does not estimate and compensate the motion due to smaller number of measurement that leads to low quality initial reconstructions</td></tr>
<tr><td>77</td><td>Makes use of the proximal-gradient method to solve the optimization problem and DE and DC for side information generation.</td></tr>
<tr><td>74, 75</td><td rowspan="4">Motion Estimation & Compensation (ME/MC) based Predication</td><td>Uses motion based prediction and residual encoding to optimal the sample allocation between the prediction and residual encoding steps.</td></tr>
<tr><td>81</td><td>The scheme incorporated MC/ME into the reconstruction process of BCS-SPL for video. The scheme alternatively reconstructs frames of the video sequence, using one to improve the quality of the other in an iterative fashion</td></tr>
<tr><td>82</td><td>The scheme makes use down sampling, ME/MC and fusion approach to generate a view prediction, which aid in the reconstruction of final view for multi-view video.</td></tr>
<tr><td>83</td><td>The scheme incorporated MC/ME into the reconstruction process of BCS for multi-view video.</td></tr>
</table>

2.6 Conclusive Remarks

Devices in VSN are mostly powered by batteries and in such network the power consumption is one of the most critical issues that need to be taken into consideration. The most suitable solution to such issues is to compress the captured visual data before transmission takes place as research in [20] shows that the power consumption for data transmission is higher than that of data processing. In cases, where multiple visual nodes are observing the same scene, the field-of-view of the visual nodes may overlap with each other. Hence, it is possible to further reduce the amount of information to be transmitted, by also removing inter-view redundancy. This is usually achieved by using the distributed coding. Among the many distributed coding schemes, the efficient sampling mechanism of CS can help to resolve the energy consumption issue in VSN. The implementation of CS for VSN reduces the total amount of data to be processed such that it recreates the signal by using only fewer sampling values than that of Nyquist rate. This results in an extended lifetime of the visual node. However, there are many open issues related to the reconstruction quality and practical implementation of CS. The current researches of CS are more concentrated towards hypothetical characteristics with simulated results, rather than on the understanding the potential issues in the practical implementation of CS.

Chapter 3
COMPRESSIVE SENSING

This chapter presents the theoretical background of Compressive Sensing (CS) that would help to better understand the proposed schemes in later chapters. In section 3.1, the core fundamentals concepts of CS that includes sparsity, incoherence, signal sensing and signal reconstruction is discussed. Next, various CS based compression is presented in section 3.2. This is followed by the differences between CS-based compression and conventional compression. Section 3.4 describes quantization that is the core element in transforming measurements to data for transmission. The chapter is concluded in section 3.5.

3.1. Theoretical Basics

CS states that a signal that is sparse in some transform domain could be entirely reconstructed with a number of samples lower than the requirement stated in Shannon-Nyquist theorem. CS relies on two important concepts, known as sparsity (signal of interest) and incoherence (sensing modality).

3.1.1. Sparsity

Sparsity is an important parameter for sampling and reconstruction of a signal. CS takes advantage of the fact that numerous sorts of real-world and manmade signals (e.g. images or videos) are sparse in certain transformation domain. This means that after the transformation, a signal could be left with only a small number of significant coefficients.

These coefficients can be ignored to achieve compression. Some known transformation basis includes wavelets, localized sinusoids, curve-lets, and wave field propagation [68].

Let $x \in R^N$ be a signal with N elements that can be expanded into an orthonormal basis Ψ **= [** $\Psi_{\mathbf{1}}, \Psi_{\mathbf{2}} \ldots \Psi_{\mathbf{N}}$]. The transformation basis of *x* is then represented as:

$$x\,(t) = \sum_{i=1}^{N} S_i \Psi_{\mathbf{i}} \qquad \textbf{(3.1)}$$

Where, S is an N × 1 vector of coefficients of x. Then, the general form of Eq. (3.1) can be written as:

$$x = \Psi S \qquad \textbf{(3.2)}$$

The signal x is said to be *K*-sparse if there exists *K* non-zero elements.

Most natural images are characterized based on large smooth regions or textured regions with sharp edges. A signal with such characteristics is said to be sparse when represented using a transformation basis for instance, wavelet. The wavelet basis splits the signal into its lower and higher frequency components. The coarse scale approximations of the signal are provided by the lowest frequency components, whereas the details and edges are found in the higher frequency components. The signal represented by the wavelet basis usually contains very small coefficients making them sparse in nature. In other words, a worthy approximation of the signal can be attained by setting the small coefficients to zero, or thresholding the coefficients, to obtain a *K*-sparse representation.

3.1.2. Incoherence

The maximum correlation measured between any two elements of two different matrices (might be represented by different basis or domains) is referred as coherence [68].

Consider a pair of orthonormal basis (Φ, Ψ) of R^N where Φ is used for sensing and Ψ is used for representing the signal. Then the coherence μ between the two bases is defined as:

$$\mu(\Phi, \Psi) = (\sqrt{N}) (\max |\Phi_k, \Psi_j|), \quad 1 \le j, k \le N \tag{3.3}$$

$$1 \le \mu (\Phi, \Psi) \le \sqrt{N} \tag{3.4}$$

In CS, the key concern is the incoherence of sensing domain Φ of the signal and the orthonormal basis Ψ. Generally, low incoherence between Φ and Ψ leads to fewer samples been required for the recovery of the signal. An example of low coherence pair includes spikes (for Fourier domain) and noise-lets (for wavelet domain) that are incoherent in any dimension.

3.1.3. CS Signal Acquisition/Sensing

The signal acquisition process of CS is different from the conventional sensing process. The conventional process operates by collecting large amount of information and then discards the unnecessary information using compression. CS operates by collecting only the necessary information related to the object of interest by taking certain random projection that is enough for the reconstruction of a signal.

Consider a signal x with length N to be recovered from M measurements ($M \ll N$) that is sparse in some transformation domain Ψ with random measurement matrix Φ. The set of measurements y is given as:

$$y = \Phi x \qquad \textbf{(3.5)}$$

Where, $x \in R_N$, is the input signal; $y \in R_M$ is the measurement vector. It is assumed that the random sensing matrix Φ is orthonormal i.e. $\Phi \Phi^T = A$. Where, A is the identity matrix.

However, as describe earlier a signal has to be sparse in some transformation domain Ψ. The reconstruction of a signal *x* lies within the set of sparse significant transformation coefficients $x = \Psi S$, and can be expressed as:

$$y = \Phi \Psi S \quad \text{s.t. } y = \Theta S \qquad \textbf{(3.6)}$$

Where, $\Theta = \Phi \Psi$ is a single MxN pseudo random measurements and S is the sparse vector. The underdetermined representation of S is given by measurement vector y. Moreover, the process of measurement is non-adaptive because Φ matrix does not depend on signal x and is fixed. The random measurements can only apply to *K*-sparse signal if Φ conforms to the following given relationship

$$M = K\text{: } \log (N/K) \qquad \textbf{(3.7)}$$

Where, K represents signal sparsity, M and N are pseudo random measurements.

Generally, in CS we focus more towards discrete signals $x \in R_N$, rather than the continuous time space signals. This is because discrete signals are theoretically simpler, and discrete CS theory is far more developed.

3.1.4. Reconstruction Algorithms

The recovery of the encoded measurements is the main challenge of using CS. As the number of unknowns is much larger than the number of observations, recovery of $x \in R_N$ from its corresponding $y \in R^M$ i.e. inverse projection of $\hat{x} = \Phi^{-1}$ y is ill-posed [95]. Since the signal to be compressed by CS should be sparse in nature, the reconstruction can be carried out by solving a convex optimization problem using sparsity in transformed domain with either ℓ-norm or image gradient with Total Variation (TV) norm.

The reconstruction of a signal x lies within the set of sparse significant transformation coefficients $x = \Psi$ S and can be obtained by solving different ℓ-norm optimization problem**.** The primary ℓ_0 optimization problem function can be expressed as:

$$\hat{x} = argmin_S \|S\|_{lo}, \qquad \text{s.t. } y = \Phi \, \Psi \, S = \Theta S \qquad \textbf{(3.8)}$$

However, solving the ℓ_0 constrained optimization problem is computationally infeasible due to its combinational and non-differentiable (presence of the absolute value function) property i.e. Nondeterministic Polynomial (NP) completeness [84].

Hence, several alternative optimizations schemes such as convex relaxation [85], greedy-iterative [86], gradient-descent [87], and iterative-thresholding [88] have been proposed to solve Eq. (3.11). However, most of the proposed schemes are exposed to certain issues, such that as the size of the natural image increases, so does the size of the sampling matrix, resulting in higher computational and memory consumption. A brief overview of the algorithms used in the proposed scheme is described in the following subsections.

i. Convex Relaxation

Algorithms based on convex relaxation approach achieve reconstruction by solving the convex optimization problem through linear programming [85]. Such algorithms require a small number of measurements for exact reconstruction but are computationally more complex due to multidimensional signals such as images and video.

The most prominent of convex relaxation algorithms is basis pursuit (BP) [89] which applies a convex relaxation to the ℓ_0 problem resulting in a ℓ_1 optimization,

$$\hat{x} = argmin_S \|S\|_{l1}, \qquad \text{s.t. } y = \Phi\, \Psi S \tag{3.9}$$

Where Ψ is the sparsity transform**.** Moreover, consider a case in which the CS measurements obtained are exposed to some noise. In this context Eq. (3.11) becomes

$$y = \Theta\, S + n \tag{3.10}$$

So, the equality constraint in the ℓ_1 formulation of Eq. (3.13) can be reduced as given in Eq. (3.14).

$$argmin_S \|S\|_{l1,} \text{ s.t.} | y- \Phi \Psi S |_{l2} \leq e \tag{3.11}$$

Where, y is a noisy CS measurement with noise vector e such that e>0. This is a favoured CS measurement reconstruction formulation. Such constrained optimization is closely related to the unconstrained optimization problem using Lagrangian multiplier ℓ_1 and ℓ_2 [88] i.e. it further reduces the problem by expanding the equality constraint from ℓ_1 to ℓ_2 penalty as stated below

$$S = argmin_S \|S\|_{l1} + \lambda \ | y- \Phi \Psi S |_{l2} \tag{3.12}$$

This formulation is known as Basis-Pursuit De-Noising (BPDN) [89], where, Φ is referring to the observed measurement matrix, Ψ is the sparse matrix in the transform domain and λ is the scaling factor that balances the ℓ_1-driven sparsity against the ℓ_2-based measure of distortion. Such property of BPDN makes it appropriate for signal and image processing applications.

Also, block-based image and video can use a Total Variation (TV) minimization. The TV minimization finds the smoothest solution within the potential space by making use of piece-wise smooth characteristics of natural signals rather than finding the sparse solution within the transformation domainΨ. The basic TV minimization function is given as in [90, 91].

$$\mathrm{TV(S)} = \sum_{i,j} |S_{i+1,j} - S_{i,j}| + |S_{i+1,j} - S_{i,j}| \quad \textbf{(3.13)}$$

$$S = argmin_S \| y - \Theta S \|_{l2} + \lambda \, \mathrm{TV(S)} \quad \textbf{(3.14)}$$

A variety of approaches has been developed to solve the optimization problem in Eq. (3.12), (3.14), and (3.15) [92]. The linear programming and second order cone approaches prove to be quite effective in solving BP (3.12) and BPDN (3.15) optimizations, with great accuracy. However, as the size of the natural image increases due to multidimensionality, such convex programming methods are exposed to higher computational and memory consumption issues. As an alternative, more efficient algorithms [86-88] that require fewer iterations or less computational resources are preferred such as iterative-thresholding [88].

ii. Iterative thresholding

This algorithm is faster than the convex optimization. In this case the recovery of exact measurements is based on hard or soft thresholding [88], [93] given that the signal is sparse and is initiated from noisy measurements. The thresholding function rests on a number of iterations and problem structure.

The Iterative-thresholding algorithms recover S by consecutive projection and thresholding. The reconstruction begins with some initial approximation i.e. S(0) and is further advanced in an iterative manner, as in Eq. (3.18, 3.19) [94]:

$$\check{\hat{\boldsymbol{x}}}^{(i)} = \hat{x}^{(i)} + 1/\gamma \, \Psi \, \Phi^{T}(y - \Psi^{-1} \, \Phi \, \hat{x}^{(i)}) \quad \textbf{(3.15)}$$

$$\hat{x}^{(i+1)} = \begin{cases} \breve{\tilde{\boldsymbol{x}}}_0^{(i)}, \left|\breve{\tilde{\boldsymbol{x}}}^{(i)}\right| \geq \tau(i)_{else} \end{cases} \quad \textbf{(3.16)}$$

Where, $\breve{\tilde{\boldsymbol{x}}}^{(i)}$represents the consecutive projections**,** τ(i) is a threshold at ith iteration and $\boldsymbol{\gamma}$ is the eigenvalue of $\Phi.\Phi^T$. It is clear that such procedure is a specific instance of a Projected Land-weber (PL) algorithm [95]. The consecutive projection and thresholding scheme speed up the reconstruction process to a certain extent and conserves a high level of precision [96].

The most prominent iterative thresholding algorithm includes Expander Matching Pursuits [97], Sparse Matching Pursuit [98] and Sequential Sparse Matching Pursuits [99] that attains near-linear recovery time while using O (s.log (n/s)) measurements only. Moreover, recently, proposed Belief Propagation algorithm also falls in this category [100].

3.2. CS based Compression Schemes

General, the CS based compression schemes can be categorised into full coding and block coding. The former acquires the CS measurements of the visual data by sampling it with appropriate sensing matrix Φ. However, in most cases Φ is not directly applied to the visual data, rather a sparse transformation is applied initially. The Φ is then applied to transform coefficients to attain the CS measurements.

In contrast, the latter acquire the CS measurements by first dividing the visual data into the small independent block. Each block is then individually sampled by the same sensing matrix Φ. Such approach helps to reduce the computational complexity and memory requirements at the encoder and is appropriate for low power applications such as VSN.

3.2.1. Block based Compressive Sensing (BCS)

In [95] a block coding based CS scheme is proposed. The scheme denoted as Block-based Compressive Sensing (BCS) attempts to process an image in a block-by-block basis. An image is first divided into small **BxB** independent block. Each block is then individually sampled using the same measurement matrix **Φ** with a constrained (block-diagonal) structure as shown in Eq. (3.20).

$$\Phi = \begin{bmatrix} \Phi_B & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \Phi_B \end{bmatrix} \tag{3.17}$$

The benefits of using BCS include:

(i) the implementation and storage of the measurement operator are simple;

(ii) block-based measurement is more expedient for practical applications as the sampled image data need not be encoder as a whole rather in a block by block fashion until the measurement of entire image is done;

(iii) the individual processing of each block of image data results in easy initial solution with significantly fast reconstruction process [95].

The two different variants that can be used to reconstruct measurements encoded using BCS, known as Smooth Projection Land-weber (SPL) and Total Variation (TV) minimization, are discussed as follows.

i. Smooth Projection Land-weber (SPL)

This reconstruction incorporates Wiener smoothing step with an iterative threshold recovery. Such approach removes the blocking artifacts occur due to block-based sampling. The reconstruction process is described in pseudo code 3.1[95, 96].

PSEUDO CODE- 3.1. SPL Based Reconstruction

Function $x^{(i+1)} = \mathrm{SPL}(x^{(i)}, y, \Phi_{\mathrm{block}}, \Psi, \lambda)$

$\hat{x}^{(i)} = \mathrm{Wiener}(x^{(i)})$

for each block j

$\check{\hat{x}}_{(j)}^{(i)} = \hat{x}_{(j)}^{(i)} + \Phi_{\mathrm{block}}^{T}(y - \Phi_{\mathrm{block}}\hat{x}_{(j)}^{(i)})$

$\check{\check{x}}^{(i)} = \hat{\check{x}}^{(i)}\,\Psi$

$\check{x}^{(i)} = \mathrm{Threshold}(\check{\check{x}}^{(i)}, \lambda)$

$\bar{x}^{(i)} = \check{x}^{(i)}\Psi^{-1}$

for each block j

$x_{(j)}^{(i+1)} = \bar{x}^{(i)} + \Phi_{\mathrm{block}}^{T}(y - \Phi_{\mathrm{block}}\bar{x}_{(j)}^{(i)})$

In the pseudo code the reconstruction variant of SPL is incorporated with Wiener filtering to search for a CS reconstruction, simultaneously achieving sparsity and smoothness. Such approach helps to reduce the blocking artifacts. The Wiener(.) function filters a 2D image x degraded by constant power additive noise. The function make use of a pixel-wise adaptive Wiener filtering method on the statistics estimated from a local neighborhood of 3×3. Φ is assumed to be a random orthonormal matrix. Later, a hard threshold (.) function is used for thresholding by SPL (•). To set a proper τ for hard thresholding, a universal threshold method is adopted [103].

$$\tau^i = \lambda \, \sigma^i \sqrt{(2 \text{Log } K)}$$

Where, λ = constant control factor to manage convergence, and K = number of the transform coefficients. σ^i = estimated using a robust median estimator.

ii. Total Variation Minimization (TV)

In [90, 91], another reconstruction based on Total Variation (TV) minimization is presented. The TV minimization finds the smoothest solution within the potential space by making use of piece-wise smooth characteristics of natural signals rather than finding the sparse solution within the transformation domain Ψ. The basic TV minimization function is given as in Eq. (3.16, 3.17).

However, the TV minimization based CS reconstruction problem in Eq. (3.17) is exposed to additional computational burden, i.e. the non-differentiable (presence of the absolute value function) and non-linear properties of TV minimization are hard to access and elucidate computationally than ℓ_1 minimization, restricting its use for CS reconstruction. In [101], a scheme named as TV-AL3 is proposed to solve Eq. (3.17). The scheme is based on the combination of the conventional augmented Lagrangian method with variable splitting and alternating direction method. TV-AL3 generates same high quality reconstructed image as that of standard TV but reduces the computational burden by applying splitting and alternating approaches. The purpose of splitting is to distinct the non-differentiable terms from the differentiable terms, i.e. separating the differentiation function from the TV so as to facilitate low-complexity sub-problems in an alternating minimization way [102], resulting in decreased computational burden. Moreover, the augmented Lagrangian method of TV-AL3 differs from the standard Lagrangian method by adding a square penalty term, whereas from the quadratic penalty method by the presence of the linear term involving the multiplier λ [102].

3.3. Difference between Compressive Sensing v/s Conventional Compression Scheme

The idea of using Compressive Sensing (CS) coding scheme is to create a simple encoder complex decoder paradigm for low-power applications (VSN). As CS represents a signal with a few non-zero coefficients (below Nyquist rate), such lower sampling rate implies less energy required for visual data processing. On the contrary, the conventional (image/video) coding schemes (MPEG2, MPEG4, H.264) are designed based on the complex encoder simple decoder paradigm. In order to exploit and remove the redundancies of captured data at the encoder, the encoding process is typically 5 to 10 times more complex than the decoder [57]. Such complex encoder requires a larger amount of energy for visual data processing.

Additionally, it is not valid to compare CS with conventional coding schemes. CS is based on simple encoder with all the exploitation and removal of redundancies for better reconstruction are done at the decoder side. While, conventional video coding schemes performs all the major operations at the encoder.

CS can be used for compression as long as the dimensionality reduction that it provides is coupled with quantization and some form of entropy coding (i.e. to produce a bit stream from the CS measurements). Currently, it is observed that CS compression performance is not good as modern image compression schemes (like JPEG-2000 or H.264) when considered from the perspective of rate-distortion performance (i.e., PSNR vs. bpp).

3.4. Quantization

Quantization is one of the essential elements of the encoding process, and it should be designed in accordance with the concerned signal. Also, it should reduce the amount of distortion in the reconstructed signal. However, in practical, sub-optimal fixed quantizers are usually used.

For most of the conventional compression schemes, the design of the quantizer is based on human visual sensitivity and linear transformation coefficient for a variety of images. The conventional image compression schemes apply certain transformation on the image to produce transform coefficients. Thus, when these coefficients are quantized a substantial number of quantization coefficients will be zero and need not to be encoded. After quantization, entropy coding is used that encodes the data into bits for transmission. Conventional image compression standards make use of pre-defined quantization matrices. For instance, the low-frequency Discrete Cosine Transform (DCT) coefficients are better quantized with JPEG and MPEG rather than the high-frequency DCT coefficients. This is because human visual system is less sensitive to error in the higher frequency domain than the lower frequency. For different H.264 contours, a uniform quantization scheme is used in which equality is given to all the coefficients.

For CS based schemes, the process is entirely different from the conventional compression processes as it generates a set of linear measurements by using a sensing matrix rather than applying the transformation on the image. Moreover, as discussed earlier the numbers of measurements obtained from CS are much smaller than the original signal. In order to quantize such small measurements, a direct solution is only to apply Scalar Quantization (SQ) to each of the CS measurements obtained. However, from analysis it is observed that such quantization solution is highly inefficient in terms of rate-distortion performance as compared to traditional coding schemes as discussed in [104]. Additionally, when the standard and uniform quantization matrixes are applied to each of the CS measurements obtained for image/video, the performance trend is similar as of scalar quantization (i.e. inefficient R-D performance). In this regard, various researchers have focused their work towards the improvement of rate-distortion performance of quantized CS measurements which is also an open research problem. Most of the efforts are based on quantizer optimization for instance [105], the reconstruction process [106, 107]), or both optimization and reconstruction [108, 109]). Additional, few of the work is based on combining the simple uniform SQ with differential modulation of the CS measurements [110]. The advantage of such quantization scheme is that the CS encoder and decoder operations need not to be modified accordingly and results in better reconstruction quality. Although, the encoder might acquire some additional complexity but that will not be significant.

3.5. Conclusive Remarks

In this chapter, core fundamentals of CS that includes sparsity, incoherence, signal sensing and signal reconstructions are discussed. These concepts will be used in the research work presented in this thesis. The CS theory states that certain signals (image/video) can be recovered from far fewer samples or measurements (less than Nyquist rate) than traditional methods used. However, the reconstruction of signal from such small measurements is the main challenge of using CS.

Chapter 4
MULTI-VIEW IMAGE COMPRESSION AND RECONSTRUCTION

In the case of VSN, images captured by different visual nodes are often correlated. Hence, a joint-decoding approach that exploits the correlation among them, and make use of the extracted information to produce image of better quality can be applied.

The joint reconstruction schemes discussed in section 2.5 of chapter 2 are exposed to few main issues. Firstly, the inter-view redundancy was not fully exploited due to parallax and occlusion problems in real-world application. Secondly, the use of disparity or motion estimation and compensation in the reconstruction is not suitable for real-time application due to the rather complex and slow processing.

In this chapter, we proposed a multi-view compression scheme for VSN based on Block-based Compressive Sensing (BCS) and Joint Multi-phase Decoding (JMD). First, images captured by different visual nodes are encoded using BCS. One of the visual nodes is configured to serve as the reference node, whereas the others as non-reference nodes. In this case, images captured by the non-reference nodes are encoded at a lower subrate when compared with the images from the reference nodes. The core idea is to improve the reconstruction of images captured by the non-reference nodes, by using information in the image captured by the reference node. This is achieved by exploiting the high correlation between them at the joint-decoder. The encoded measurements are then transmitted independently to the server that serves at the joint-decoder.

At the joint-decoder, the proposed JMD is applied on the received images. The proposed JMD produces and uses Side Projection Information (SPI) to aid the reconstruction of the final image. One reason of using BCS is that it managed to provide an initial reconstruction of an image in shorter period of time [95]. The initial reconstruction helps in the generation of the SPI, which is the core component of the proposed scheme. Besides using the initial reconstruction, residual reconstruction and prediction method are added to produce a SPI that could better represent the visual data to be decoded. Simulation results show that the proposed scheme works well for both near-field and far-field images, and could also handle parallax and occlusion issues. This is achieved by aligning and fusing the images captured from different view angles. Furthermore, the proposed JMD relies on simplified operations that are less complex when compared to the other reconstruction schemes.

The rest of this chapter is organized as follows. Section 4.1 provides an overview of using the proposed scheme to compress and reconstruct multi-view images. Then section 4.2 and 4.3 explain about the proposed JMD that is used to reconstruct the compressed images and to exploit the correlation among them. All the experimental results are presented in section 4.4, and the chapter is concluded in section 4.5.

4.1. Overview of the Proposed Multi-View Image Compression and Reconstruction

Before explaining the proposed scheme, it is important to first define and summarize the notations that are used in the chapter. The notations are summarized in Table 4.1.

Table 4-1: The notations used in the chapter

Notation	Description
$\mathbf{I_R}$	Reference Image
$\mathbf{I_{NR}}$	Non-Reference Image (Image to be Improved)
$\mathbf{Y_R}$	Reference Image Measurements
$\mathbf{Y_{NR}}$	Non-Reference Image Measurements
$\mathbf{I'_R}$	Reconstructed Reference Image (Decoded Reference Image)
$\mathbf{I'_{NR}}$	Intermediate Reconstructed Non-Reference Image
$\mathbf{I'_T}$	Image Transformation based on $\mathbf{I'_R}$ and $\mathbf{I'_{NR}}$
$\mathbf{I'_{RT}}$	Transformed $\mathbf{I'_R}$
$\mathbf{I''_R}$	$\mathbf{I'_R}$ registration w.r.t. $\mathbf{I'_{NR}}$
$\mathbf{I_P}$	Projected Image
$\mathbf{Y_P}$	Projected Image Measurements
$\mathbf{Y_r}$	Residual Measurements
$\mathbf{I_r}$	Residual Image
$\mathbf{I''_{NR}}$	Final Reconstructed Image

To better explain the proposed scheme, we consider a Visual Sensor Network (VSN) that consists of S_n number of visual nodes (encoder) observing the same scene from different positions shown in Figure 4.1.

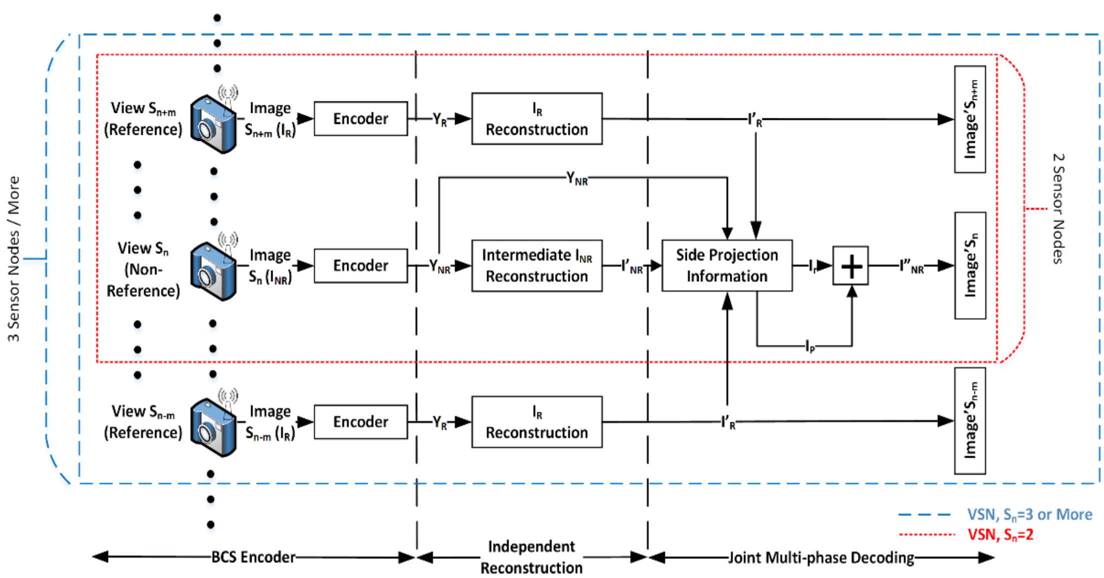


Figure 4.1: Using the proposed scheme for multi-view image compression.

In this case, n represents the non-reference node and n-m, …, n+m are its neighboring left and right reference nodes, where m can be extended until the overlap between the field-of-view of the visual nodes diminishes. The proposed scheme is flexible in such a way that one may choose to have m numbers of reference node with n numbers of non-reference node. In the following subsections, we will discuss about how the proposed scheme will operate in the case of single reference node and multiple reference nodes.

4.2. Joint Reconstruction with Single Reference Node

We first consider a joint reconstruction scenario that only consists of one reference node. All the other non-reference nodes will rely on using the information from the only reference node to reconstruct the images. To simplify the explanation, we assume that there are only two visual nodes, where one is configured to serve as the reference node, while another as the non-reference node, as highlighted by the short red dotted lines in Figure 4.1.

In this case, each node first captures the measurements of an image using BCS. The encoded measurements are then transmitted to the server, where they are independently reconstructed using BCS-SPL or BCS-TV-AL3. Later, the proposed JMD is applied to the encoded measurements and decoded image to produce the SPI that will be used to aid the reconstruction of the final image.

4.2.1 Encoding using BCS

The images captured by the reference node (I_R) and non-reference node (I_{NR}) are encoded independently using BCS, at a subrate of M_R and M_{NR} respectively, with $M_R \geq M_{NR}$. To simplify the processing, the images captured (I_R / I_{NR}) by the sensor nodes are first divided into small blocks of size $X \times X$, and each block is sampled with respect to the sensing matrix Φ_x. This produces the measurements (Y_R / Y_{NR}) as defined in Eq. (4.1). Y_R and Y_{NR} are then transmitted to the server (decoder) independently.

$$Y_x = \Phi_x I_x, \quad \text{where } x = R \text{ or } NR \tag{4.1}$$

4.2.2 Independent Reconstruction

At the server, I'_R and I'_{NR} are independently reconstructed by using Y_R and Y_{NR} respectively while Φ_R and Φ_{NR} are their corresponding sensing matrix. The reconstruction of I'_R and I'_{NR} can be achieved by solving either Eq. (3.15) using BCS-SPL [96] or Eq. (3.17) using BCS-TV-AL3 [101], which are two different variants of CS reconstruction.

4.2.3 Joint Multi-phase Decoding (JMD)

After we obtained $\mathbf{I'_R}$ and $\mathbf{I'_{NR}}$, the proposed JMD is applied to them. The goal is to obtain an improved version of $\mathbf{I'_{NR}}$ (known as $\mathbf{I''_{NR}}$). The scheme involves generating the SPI using $\mathbf{I'_R}$, $\mathbf{I'_{NR}}$ and $\mathbf{Y_{NR}}$ as shown in Figure 4.2 (highlighted short red dotted lines). Generally, the process can be categorized into two main phases that are described in the following subsections.

i. Side Projection Information (SPI)

The approach developed to exploit the correlations between the multi-view images is divided into two phases as explain follows.

Phase 1 - Prediction: This phase consists of two key steps: (i) registration and (ii) fusion. First, image registration is used to project I'_R onto I'_{NR} as in Figure 4.3

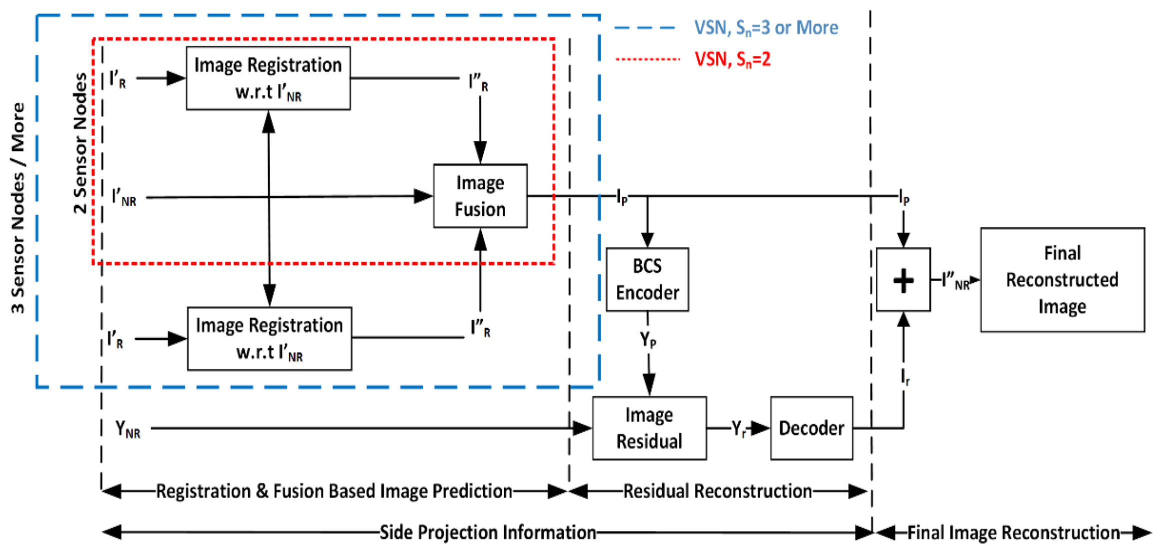


Figure 4.2: Block diagram of side projection information generator

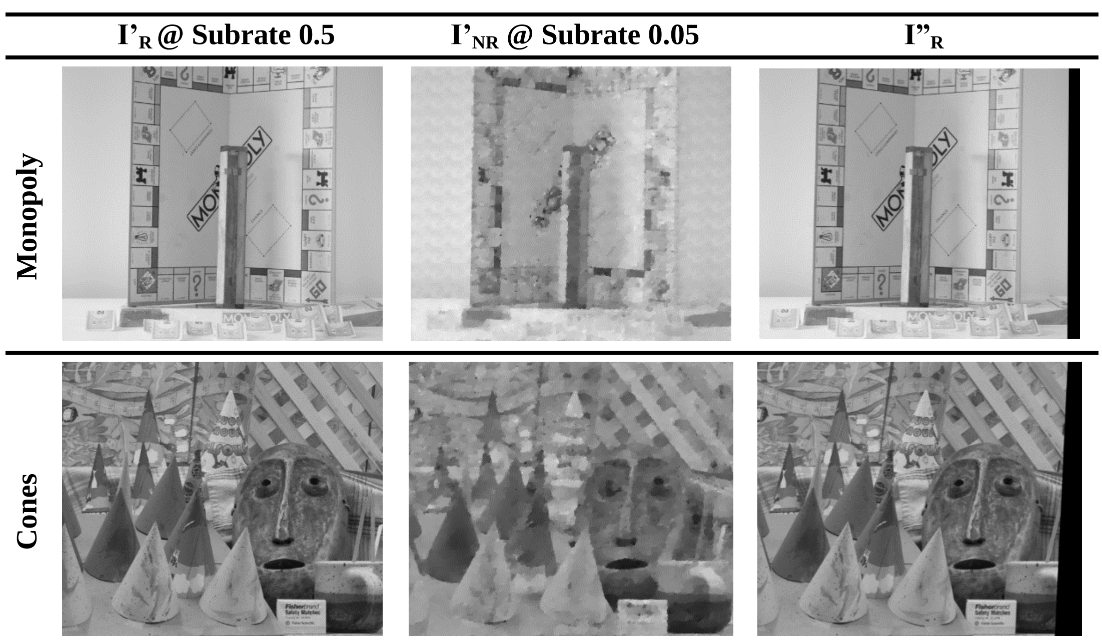


Figure 4.3: Registration process in which I'_R is projected onto I'_{NR} to produce I''_R

Registration: The registration process aligns $\mathbf{I'_R}$ onto $\mathbf{I'_{NR}}$ to exploit the correlation between them, and the output of this is referred to as $\mathbf{I''_R}$. In this case, Intensity-based image Registration (IBR) is adopted, as it requires less amount of pre-processing and able to achieve better alignment than that of Feature-Based (FB) methods [111] as shown in Figure 4.4.

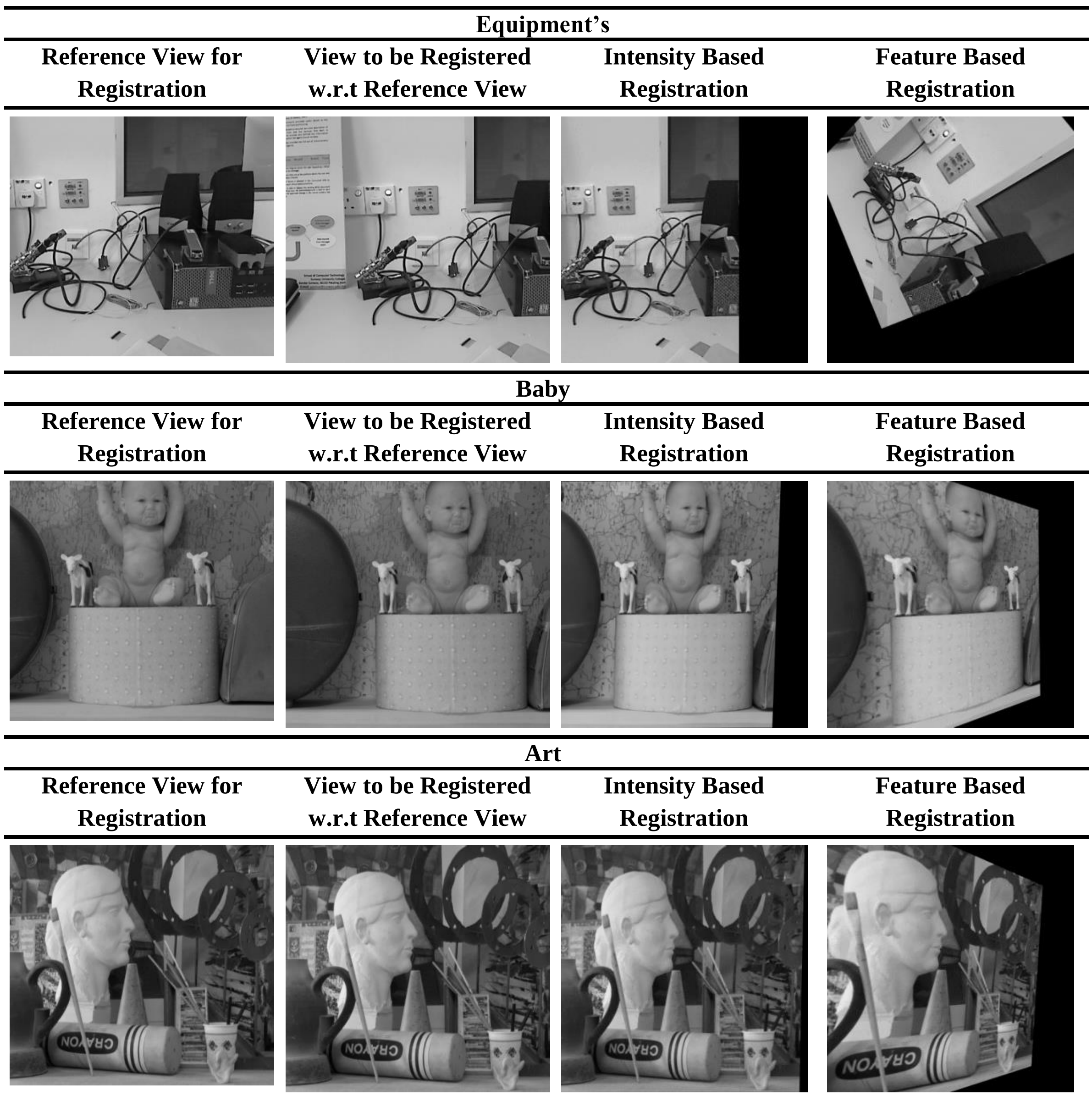


Figure 4.4: Intensity based registration vs Feature based registration

The IBR shown in Figure 4.5 is an iterative process that can be divided into three parts: (i) pre-processing, (ii) image transformation, (iii) evaluation and optimization.

The registration process begins with the pre-processing of I'_R and I'_{NR}. In this case, a phase correlation is used to find the gross alignment between the two images to estimate an initial transformation matrix (I-tform).

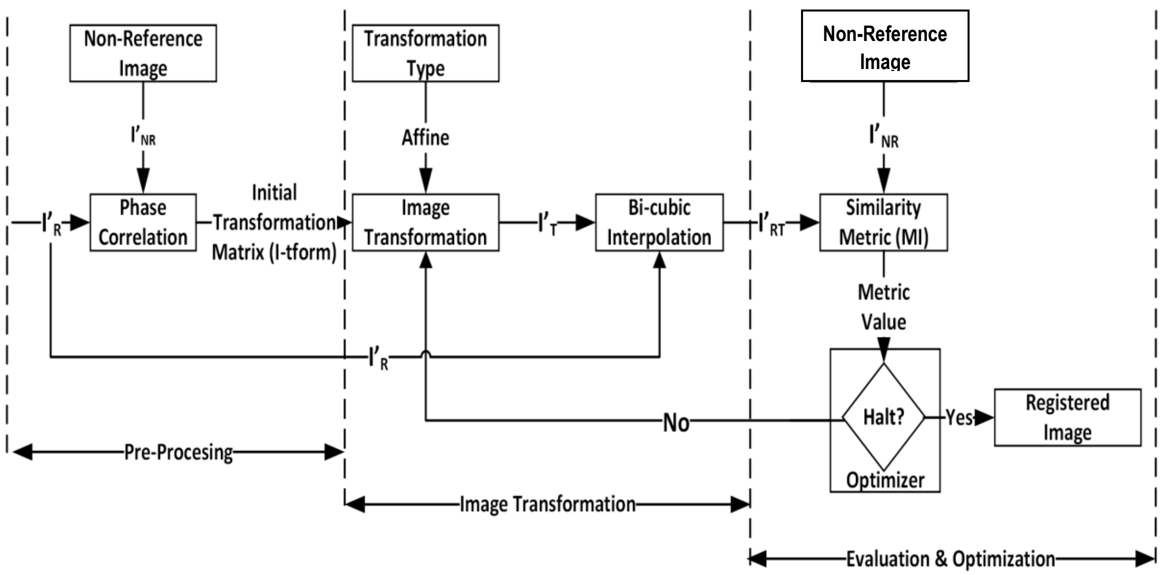


Figure 4.5: The complete process of intensity based image registration

After pre-processing, a particular geometric transformation (translation, affine, etc.) has to be used to align I'_R w.r.t. I'_{NR}. In this case, we have chosen affine transformation for our proposed JMD. The affine transform is based on translation (in *x* and *y*), rotation, scaling (in *x* and *y*) and skew geometric transformations. Such properties of affine transform help not only to preserve co-linearity and incidence but also preserve the parallelism unlike projective transformation as shown in Figure 4.6.

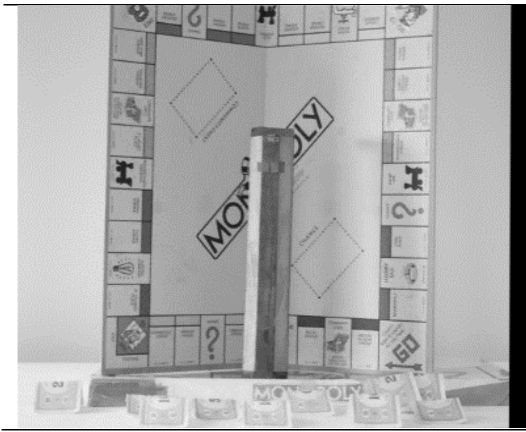
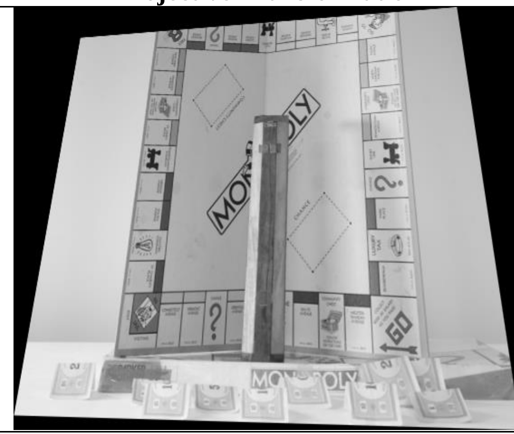
Affine Transformation	Projective Transformation
	

Figure 4.6: Comparison of affine and projective transformations when applied to Monopoly image

In this way, those correlated multi-view images can be better aligned and benefit the fusion process of exploiting the interview redundancies in a more accurate way. Also, the transformation applied can be modified based on top-left (x_0, y_0) or centre pixel (x_c, y_c) coordinates with a different definition of the translation parameters.

The earlier estimated initial transformation matrix (I-tform) and affine transform then produce a transformed image **I'$_{\mathbf{T}}$**. This **I$_{\mathbf{T}}$** is to be applied on the **I'$_{\mathbf{R}}$** with bi-cubic interpolation to give transformed I'$_R$ that is called **I'$_{\mathbf{RT}}$**.

Afterward, evaluation and optimization based on an image similarity metric and the optimizer are performed respectively. The image similarity metric is used to evaluate the accuracy of the registration. It is defined mainly based on the widely used Mutual Information (MI) or Mean Square Difference (MSD) iterative algorithms [111, 112]. However, the proposed JMD makes use of MI [113] that depends on different information theoretical techniques based on joint probability distribution. Such technique samples the pixel values from the two images to assure that the similar set of pixel values are mapped among them. The Image Similarity Metric (ISM) returns a metric value by comparing the I'_{RT} to the I'_{NR}. The ISM metric value M can be expressed as follow:

$$M(I'_{NR}, I'_{RT}) = \sum_{y \in L_{I'_{RT}}} \sum_{x \in L_{I'_{NR}}} P_{I'_{RT}\,I'_{NR}}(y,x;a) \log_2 \left(\frac{P_{I'_{RT}\,I'_{NR}}(y,x;a)}{P_{I'_{NR}}(x) P_{I'_{RT}}(y;a)} \right) \quad \textbf{(4.2)}$$

Where $L_{I'_{NR}}$ and $L_{I'_{RT}}$ represents the discrete sets of intensities related with images I'_{NR} and I'_{R}, respectively, $P(y,x:a)$ is the joint probability distribution function of random variables x, y, a is the transformation parameters and $P_{I'_{NR}}$, $P_{I'_{RT}}$, and $P_{I'_{NR}I'_{RT}}$ are the marginal and joint probability distributions [113], respectively.

Finally, the optimizer states the methodology to maximize the achieved similarity metric M to produce a final registered image I''_{R}. When the specified number of iterations is completed or when a point of diminishing is reached, the process terminates. Otherwise, the optimizer adjusts the transformation matrix to begin the next iteration. The optimization parameters of transformation N are determined based on the maximization of mutual information expressed as follow:

$$N = argmax_N \left\| M(I'_{NR}, I'_{RT}) \right\| \quad \textbf{(4.3)}$$

$$N = argmax_N \left\| M(I'_{NR}, I'_{RT}) = \sum_{y \in L_{I'_{RT}}} \sum_{x \in L_{I'_{NR}}} P_{I'_{RT}\,I'_{NR}}(y,x;a) \log_2 \left(\frac{P_{I'_{RT}\,I'_{NR}}(y,x;a)}{P_{I'_{NR}}(x) P_{I'_{RT}}(y;a)} \right) \right\| \quad \textbf{(4.4)}$$

The above optimization problem can solve by making use of either Gradient Descent (GD) or one evolutionary (OE) algorithms to optimize the image similarity metric. The proposed JMD employs the GD algorithm [114] to adjust the transformation parameters so that the optimization follows the gradient of the ISM in the direction of the maxima.

The complete process of IBR as implemented in the proposed scheme is summarized in pseudo-code 4.1. The defined set of parameters above was found best when compared to other sets of parameters for ten different datasets that have been used.

PSEUDO-CODE 4.1. Registration of Images

Input $I'_{NR} = \{I'_n\}$, $I'_R = \{I'_{n\pm1}\}$, Transform= affine, Interpolation=Bi-cubic
OutputI''_R
Function Registration **(**I'_{NR}, I'_R**)**
for all a = {Number of Views -1} do

STEP I: Pre-processing of images to find the angle offset among them using phase correlation

STEP II: Transformation of I using I-tform, geometric transform and interpolation.

STEP III: Evaluation and optimization of I_{RT} w.r.t I'_n using mutual information similarity metric and gradient optimizer.

Optimization Condition:

If

Optimization =N= max-iterations ‖ point of diminish

Halt (process complete)

Else

Go to Step II and repeat the steps till process completes.

end for

Fusion: In this step, **I"$_{R}$** is fused with **I'$_{NR}$** using wavelet transform to produce the projected image **I$_{P}$**. The reason for doing this is to preserve the quality and detail information of the image [115, 116]. The entire process is best described as an example illustrated in Figure 4.7.

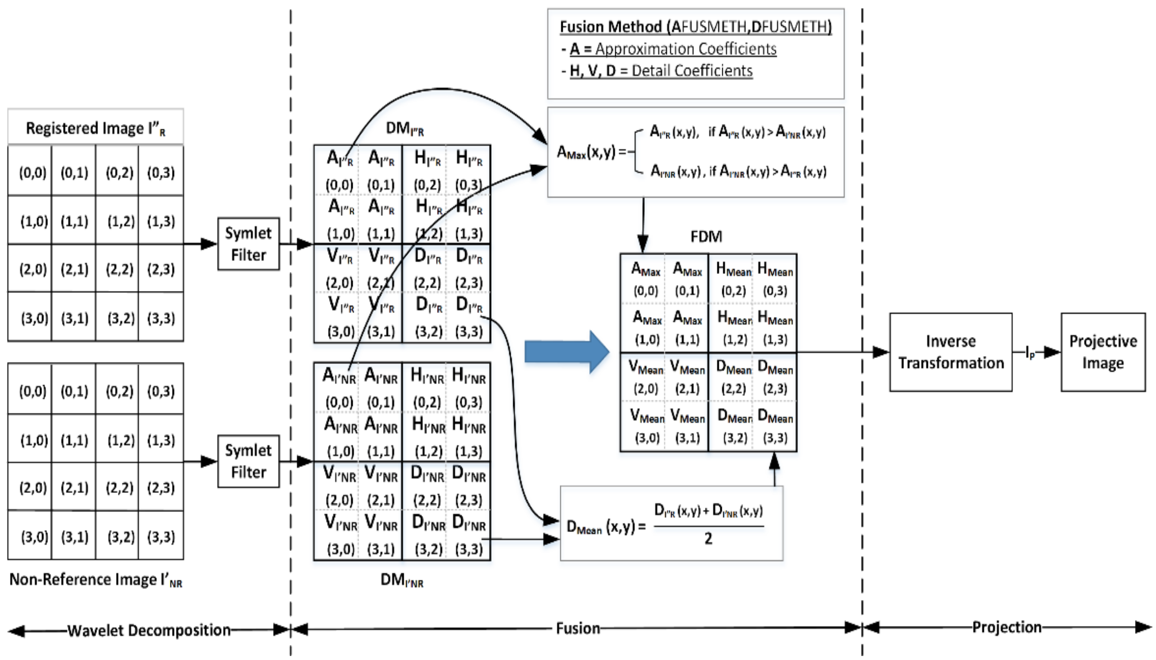


Figure 4.7: The complete process of image fusion based on wavelets with approximation & detail coefficients

First, a Symlet 4-tap filter is applied to I"$_{R}$ and I'$_{NR}$ to decompose the images into two decomposition maps, DM$_{I"R}$ and DM$_{I'NR}$ respectively. The Symlet filter is used rather than dB4 because it provides better results at same decomposition levels as shown in Table 4.2 for different image dataset. Each map contains coefficients that can be categorized as the approximation (A) and detail (D).

Table 4-2: Fusion of various multi-view image using different wavelet filters

			Baby			
Subrates	**0.05**	**0.1**	**0.15**	**0.2**	**0.25**	**0.3**
dB4	28.33	30.71	32.56	33.86	35.26	36.47
Symlet4	**28.98**	**31.32**	**33.15**	**34.41**	**35.81**	**36.89**
			Monopoly			
Subrate	**0.05**	**0.1**	**0.15**	**0.2**	**0.25**	**0.3**
dB4	25.12	27.42	29.36	31.38	33.00	34.57
Symlet4	**25.97**	**28.14**	**30.01**	**32.07**	**33.67**	**35.17**
			Middlebury			
Subrate	**0.05**	**0.1**	**0.15**	**0.2**	**0.25**	**0.3**
dB4	28.13	30.58	32.35	33.57	34.66	35.72
Symlet4	**28.99**	**31.24**	**32.95**	**34.10**	**35.11**	**36.23**

In the example shown in Figure 4.7, only one level of decomposition is adopted. However, in our implementation five level of decomposition is used. Next, the two decomposition maps are fused together using point-to-point operations. Assuming that (x, y) is used to represent the coordinate of each coefficient. Each approximation coefficient $A_{I'R}(x, y)$ from $DM_{I'R}$ is compared to the approximation coefficient $A_{I'NR}(x, y)$ from $DM_{I'NR}$. The coefficient with larger magnitude is then selected and stored in the fused decomposition map, FDM(x, y). The detail coefficients (H / V / D) are handled in a slightly different way. Instead of taking the coefficient with larger magnitude, the average value of detail coefficients located at coordinate (x, y) from $DM_{I'R}$ and $DM_{I'NR}$ is calculated and stored in the FDM(x, y). After fusion, the inverse transformation is applied to the fused decomposition map to generate the projective image I_P. The defined set of parameters above was found best when compared to other sets of parameters for ten different datasets that have been used.

Overall, the fusion process is described in pseudo-code 4.2.

PSEUDO-CODE 4.2. Fusion of Images

Input $I'_{NR} = \{I'_n\}$, $I''_R = \{I''_{n\pm1}\}$, Transformation = sym4

Output I_P

Function Fusion **(**I'_{NR}, I''_R**)**

Initializing $I_P = I'_n$

for all b = {a} do , where **a** = {Number of Views -1}

STEP I: Five level decomposition of image I'_n and $I''_{n\pm1}$ into Decomposition Maps (DM) using wavelet transformation. Each DM includes one Approximation (A) coefficient and three Detail (D) coefficients.

$$\mathbf{DM}\ (\mathbf{I_P}) = [\mathbf{A}_{\mathbf{I}\boldsymbol{P}}\ (\boldsymbol{x}, \boldsymbol{y}), \mathbf{D}_{\mathbf{I}\boldsymbol{P}}\ (\boldsymbol{x}, \boldsymbol{y})]$$

$$\mathbf{DM}(\boldsymbol{I}_{\boldsymbol{n}\pm\mathbf{1}}^{\boldsymbol{b}}) = [\mathbf{A}_{\mathbf{I}''\boldsymbol{n}\pm\mathbf{1}}^{\boldsymbol{b}}(\boldsymbol{x}, \boldsymbol{y}), \mathbf{D}_{\mathbf{I}''\boldsymbol{n}\pm\mathbf{1}}^{\boldsymbol{b}}(\boldsymbol{x}, \boldsymbol{y})]$$

STEP II: Fusion of the decomposition maps based on approximation (A) and detail (D) coefficients.

$$\mathbf{FDM}\ (\mathbf{I_P}, \boldsymbol{I}_{\boldsymbol{n}\pm\mathbf{1}}^{\boldsymbol{b}}) = \mathbf{Fusion}\ [\mathbf{DM}(\boldsymbol{I}_{\boldsymbol{P}}), \mathbf{DM}(\boldsymbol{I}_{\boldsymbol{n}\pm\mathbf{1}}^{\boldsymbol{b}})]$$

$$\textbf{s.t. Fusion}\ \{[\mathbf{A}_{\mathbf{I}\boldsymbol{P}}\ (\boldsymbol{x}, \boldsymbol{y}), \mathbf{D}_{\mathbf{I}\boldsymbol{P}}\ (\boldsymbol{x}, \boldsymbol{y})]\ , [\mathbf{A}_{\mathbf{I}''\boldsymbol{n}\pm\mathbf{1}}^{\boldsymbol{b}}(\boldsymbol{x}, \boldsymbol{y}), \mathbf{D}_{\mathbf{I}''\boldsymbol{n}\pm\mathbf{1}}^{\boldsymbol{b}}(\boldsymbol{x}, \boldsymbol{y})]\}$$

- For fusing approximation (A) coefficients of the DM(I_P) **and** DM($I_{n\pm1}^{b}$)

$$\mathbf{FDM_A(x,y)} = \begin{cases} \textbf{If}\quad \mathbf{A}_{\mathbf{I}''\boldsymbol{n}\pm\mathbf{1}}^{\boldsymbol{b}}(\boldsymbol{x}, \boldsymbol{y}) > \mathbf{A}_{\mathbf{I}\boldsymbol{P}}\ (\boldsymbol{x}, \boldsymbol{y}) \\ \textbf{then}\ \mathbf{A}_{\mathbf{I}''\boldsymbol{n}\pm\mathbf{1}}^{\boldsymbol{b}}(\boldsymbol{x}, \boldsymbol{y}) \\ \textbf{Else}\ \mathbf{A}_{\mathbf{I}\boldsymbol{P}}\ (\boldsymbol{x}, \boldsymbol{y}) \end{cases}$$

- For fusing Detail (D) coefficients of the DM(I_P) **and** DM($I_{n\pm1}^{b}$)

$$\mathbf{FDM_D(x,y)} = [\mathbf{D}_{\mathbf{I}\boldsymbol{P}}^{\mathbf{1}}(\boldsymbol{x}, \boldsymbol{y}) + \mathbf{D}_{\mathbf{I}''\boldsymbol{n}\pm\mathbf{1}}^{\boldsymbol{b}}(\boldsymbol{x}, \boldsymbol{y})]\ /\ \mathbf{2}$$

- Same steps will be repeated for each decomposition level.

STEP III: Transformation of the fused decomposition map into image I_P.

$$\mathbf{I_P} = \mathbf{Transform}^{-1}\ [\mathbf{FDM}\ (\mathbf{I_P}, I_{n\pm1}^{b})]$$

end for

The registration and fusion method creates a projective image I_P that contains improved pixel information than that of I'_{NR} as represented in Figure 4.8.

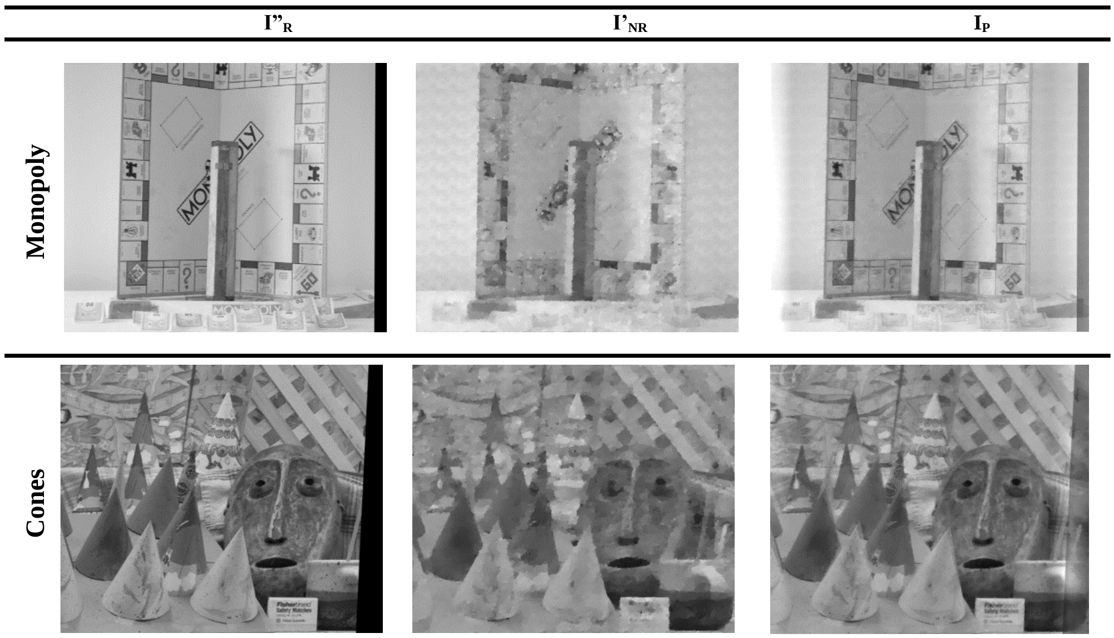


Figure 4.8: Fusion of I''_R with I'_{NR} to produce I_P

- **Phase 2 - Residual Image:** After we have the projected image $\mathbf{I_P}$ ready, it is encoded with BCS at the joint decoder to produce measurement $\mathbf{Y_P}$. Then, the difference between $\mathbf{Y_P}$ and $\mathbf{Y_{NR}}$ is determined as expressed in Eq. (4.5) and the output is known as the residual measurement $\mathbf{Y_r}$.

$$Y_r = Y_{NR} - Y_P \tag{4.5}$$

Based on the observation shown in [117], it is better to generate the residual measurements first and then decode it to get the residual image rather than generating the residual image from the two decoded images. The reason for doing this is to ensure maximum correlation with minimum prediction errors as compared to the original one. To obtain the residual image I_r, the residual measurements are then decoded by solving either Eq. (3.15) or (3.17) depending on the variant used. The reconstruction of the residual image yields better results when similar blocks exist in both images. In the datasets that we used, occlusions take place due to depth discontinuity i.e. overlapping of objects in the images. The residual of such occluded blocks exhibits features different from the other blocks of the image and result in higher correlation [117].

ii. Final Image Reconstruction

In order to produce the final reconstructed image I''_{NR}, the side projection information generated i.e. residual image I_r and the projected image I_P are added together. It is a normal point-to-point addition that is expressed in Eq. (4.6).

$$I''_{NR} = I_r + I_P \tag{4.6}$$

By doing so, uniformity in terms of image measurements (Y) is achieved i.e. the measurements computed for the final reconstructed image is hypothetically equal to the measurements Y_{NR}. The final reconstructed image is shown in Figure 4.9.

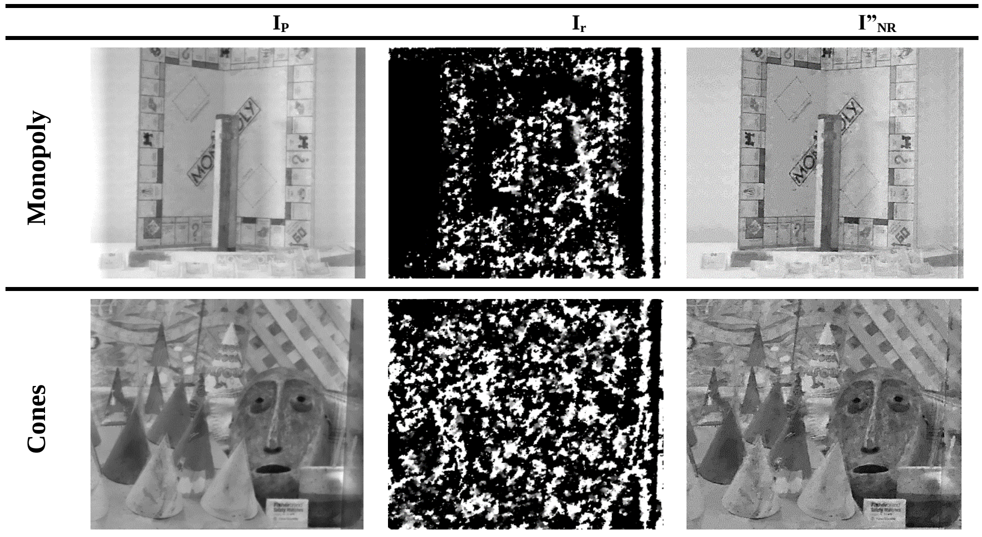


Figure 4.9: Reconstruction of final image I''_{NR} by adding I_P and I_r

By comparing the highlighted regions (boxes with dotted outlines) in Figure 4.10 (b), (c), it can be noticed that the image reconstructed by using the proposed JMD (JMD-TV-AL3) scheme looks much sharper when compared to that of using conventional independent BCS-TV-AL3.

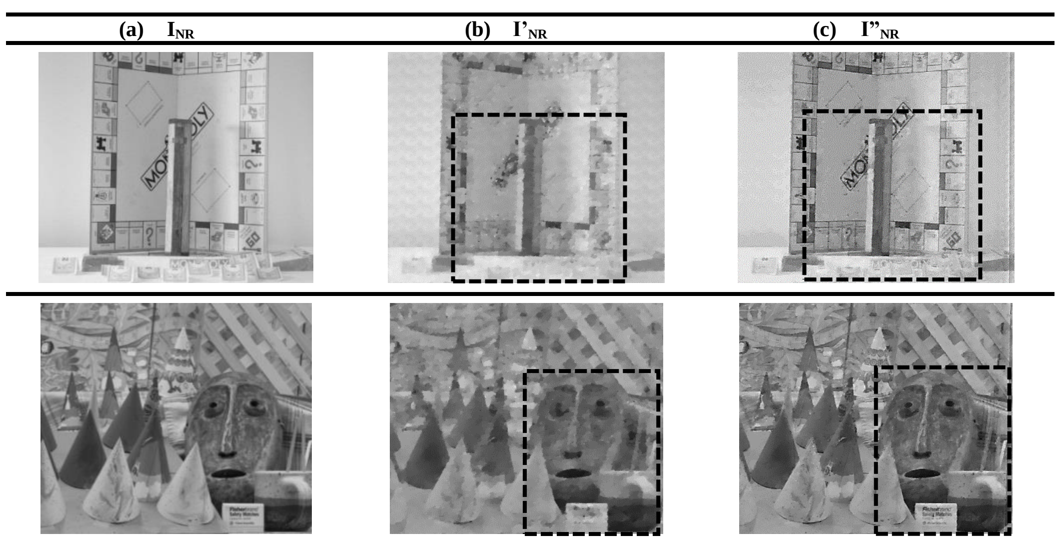


Figure 4.10: Samples of independent & final reconstruction of non-reference image at subrate of 0.05

4.3. Joint Reconstruction with Multiple Reference Nodes

In this section, the proposed JMD scheme is extended to consider cases with more than two reference nodes as shown in Figure 4.1 (highlighted with long blue dotted lines).The main process remains the same as defined in Section 4.2. The only fundamental difference is in the registration and fusion methods. Instead of dealing with only two images, they are now required using more than two images to generate the projective image I_P, as illustrated in Figure 4.2 (highlighted long blue dotted line). The complete process of this is also described in pseudo-code 4.3.

PSEUDO-CODE 4.3. Complete JMD Image Reconstruction

Input $I'_{NR} = \{I'_n\}$, $I'_R = \{I'_{n\pm1}\}$, $Y_{NR} = \{Y_n\}$, $\{\Phi_n\}$
Output I''_n
for all n $\in \{1, 2, 3\ldots$ Number of Views$\}$ do

STEP I: Registration of images $I'_{n\pm1}$ w.r.t I'_n using pseudo-code 4.1

STEP II: Fusion of registered images $I''_{n\pm1}$ with I'_n using pseudo-code 4.2

STEP III: Encoding of projected image $\mathbf{I_P}$ with BCS to produce measurement $\mathbf{Y_P}$.

STEP IV: Determine the difference between $\mathbf{Y_P}$ and $\mathbf{Y_n}$ to acquire a residual measurement $\mathbf{Y_r}$.

STEP V: Decode the residual measurements $\mathbf{I_r}$ by solving Eq. (3.15) or Eq. (3.17) based on the BCS variant used.

STEP VI: Final reconstructed image $\mathbf{I''_n}$ is generated by adding the residual image $\mathbf{I_r}$ and the projected image $\mathbf{I_P}$ by using a normal point-to-point addition

end for,
return

Assuming that there are two reference nodes and one non-reference node, the two reconstructed I'_{R1} and I'_{R2} are first registered w.r.t I'_{NR} to produce I''_{R1} and I''_{R2} respectively as defined in Pseudo Code 4.1. Next, I''_{R1}, I''_{R2} and I'_{NR} are fused together following the procedure presented in Pseudo Code 4.2 to produce the projection image I_P. Finally, I_P is used to obtain the residual image that leads to the reconstruction of final image I''_{NR}. The block diagram for the joint reconstruction of entire multi-view images sequence based on proposed JMD is presented in Figure 4.11. Also, the results for the joint reconstruction of the entire Monopoly and Cones image sequence are shown in Figure 4.11.

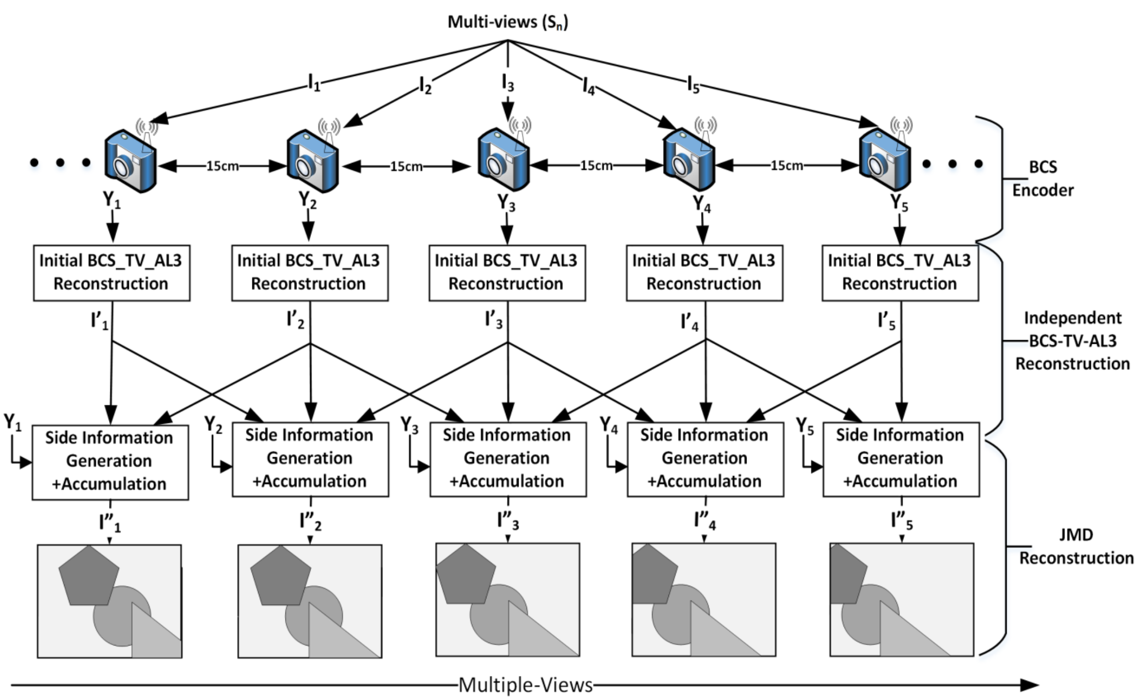


Figure 4.11: Different views of the same scene, where one view is selected as the non-reference view, whereas other are the reference views

4.4. Experimental Results

In the following, the evaluation of the proposed scheme with single and multiple reference nodes are presented. A set of test images (both near-field and far-field), is applied to evaluate its performance. Each dataset consists of 7 views taken from equidistant points along a line. The images are about 1300 × 1100 pixels (cropped to the overlapping field of view), with about 150 different integer disparities present.

For the work reported in this chapter we use ten datasets as shown in Figure 4.12, obtained from [118,119,120]. The selected dataset consists of images with various characteristics such as high and low percentage of un-textured surface, variations and disparity ranges to evaluate the performance of proposed system in different conditions. As input images a single image with multiple views taken with the same exposure and lighting is used. To make the images amenable by the proposed system, we down sample the original images size 512 × 512 pixels. The resulting images are still more challenging than standard stereo benchmarks such as the Aloe, Middlebury Teddy and Cones images, due to their larger disparity range and higher percentage of un-textured surface.

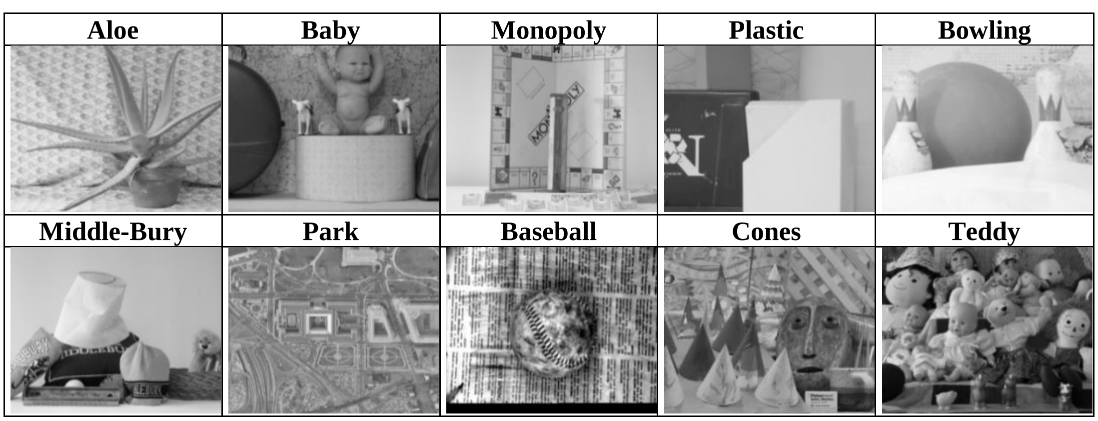


Figure 4.12: Several standard grayscale test image datasets of size 512x512

The evaluation also involves using the three different variants as summarized in Table 4.3. The purpose is to find out which variant works best with the registration and fusion process in the proposed JMD. The variants (SPL-DCT, SPL-DDWT, TV-AL3) were implemented by using their available source codes [121, 122] respectively. Furthermore, we also investigated the effect of using smaller block size.

Table 4-3: Proposed JMD with different BCS variants

Abbreviation	Combination
JMD-DCT	Proposed JMD + SPL-DCT [103]
JMD-DDWT	Proposed JMD + SPL-DDWT [103]
JMD-TV	Proposed JMD + TV-AL3 [102]

The evaluation is carried out by measuring the Rate-Distortion (R-D) in terms of Peak-Signal to Noise-Ratio (PSNR (dB)) at different sampling rate (subrate). Moreover, due to the random nature of the measurement matrix $\mathbf{\Phi}$, the quality of the reconstructed image might vary. Hence, all PSNR values presented represent an average of **5** independent trials. All the non-reference images are encoded at lower subrates (0.05, 0.1, 0.15, 0.2, 0.25, 0.3) with the reference image encoded at a fixed subrate of 0.5.

4.4.1 Impact of Block Size on the Reconstruction of Images

The evaluation is carried on various test images of size 512x512. Four different block size are tested i.e. 8x8, 16x16, 32x32, and 64x64. In this experimental setup, we focused the simulation till 64x64 block size, because higher block size will produce more measurements and hence will take more time and energy to transmit. Further, larger measurement block size provides better reconstruction quality at the expense of complexity. The selection of block size is a tradeoff between reconstruction quality and computational complexity. In this regard, it is not feasible for a battery-powered device to always encode and send the captured images at higher block size. In addition, most of the research works focused on low powered application have adopted block size of 32x32 as it provides better image quality with less computational complexity.

Table 4.4 shows the effect of using different block size on various images. From the results, it is noticeable that the reconstruction quality of using larger block size is better than using smaller block size.

Table 4-4: PSNR (dB) results for the impact of different block size (8x8, 16x16, 32x32, 64x64) on the CS reconstruction quality for various image sets.

			Aloe			
Subrate	**0.05**	**0.1**	**0.15**	**0.2**	**0.25**	**0.3**
TV 64x64	**24.81**	**27.28**	**27.94**	**28.56**	**29.34**	**29.88**
TV 32x32	24.31	26.65	27.35	27.98	28.89	29.35
TV 16x16	23.87	25.54	26.79	27.58	28.45	28.96
TV 8x8	21.33	22.87	26.18	27.05	28.09	28.43
			Baby			
Subrate	**0.05**	**0.1**	**0.15**	**0.2**	**0.25**	**0.3**
TV 64x64	**28.98**	**31.38**	**33.15**	**34.41**	**35.83**	**36.89**
TV 32x32	28.43	30.71	32.56	33.86	35.26	36.47
TV 16x16	27.93	29.92	32.09	33.26	34.74	35.88
TV 8x8	25.58	27.33	31.03	32.58	34.42	34.89
			Monopoly			
Subrate	**0.05**	**0.1**	**0.15**	**0.2**	**0.25**	**0.3**
TV 64x64	**25.40**	**28.34**	**30.59**	**32.64**	**36.57**	**36.57**
TV 32x32	25.20	27.42	29.36	31.38	33.00	34.57
TV 16x16	24.86	26.94	28.64	30.08	31.81	33.11
TV 8x8	21.45	23.38	27.49	29.60	31.48	32.24
			Plastic			
Subrate	**0.05**	**0.1**	**0.15**	**0.2**	**0.25**	**0.3**
TV 64x64	**30.01**	**35.17**	**38.27**	**41.05**	**43.25**	**45.20**
TV 32x32	29.76	34.87	37.70	39.47	41.59	43.60
TV 16x16	29.41	33.19	36.28	38.66	40.39	41.33
TV 8x8	27.31	31.00	35.88	37.49	38.54	39.44
			Bowling			
Subrate	**0.05**	**0.1**	**0.15**	**0.2**	**0.25**	**0.3**
TV 64x64	**30.98**	**34.02**	**36.21**	**37.98**	**39.78**	**41.68**
TV 32x32	30.56	33.43	35.29	36.84	38.23	39.33
TV 16x16	30.21	32.68	34.07	35.86	37.42	38.67
TV 8x8	27.88	30.45	33.89	35.29	36.76	37.70

***Note*:** The bold values relates to the maximum PSNR (dB) reached for a given subrate and image

Specifically, at low subrates the reconstruction quality of smaller block size, i.e. 8x8 is poor. However, the difference between using larger and smaller block size decreases as the subrate increases. Additionally, it should also be noted that larger measurement block sizes further benefits the reconstruction of low-variation images, as can be seen from the "Plastic" image set which shows a performance difference of $\sim$4 dBs when 64×64 blocks are used instead of 16×16 blocks.

Overall, larger measurement block size provides better reconstruction quality at the expense of complexity. For instance, the reconstruction speed of using the smaller block is 35% faster than the larger block size. The selection of block size is a tradeoff between reconstruction quality and computational complexity.

4.4.2 Joint Reconstruction with Single Reference Node

In this section, the Rate-Distortion (R-D) performance of the proposed scheme with one reference node is evaluated. The various setups are distinct as follow and are discussed in later subsections.

i. Proposed JMD with Different Variants

In this subsection, the effect of using different variants (SPL-DCT, SPL-DDWT, or TV-AL3) with proposed JMD is discussed. The purpose is to find out which variant is more suitable for our proposed JMD.

The results presented in Table 4.5 show that TV-AL3 outperforms other variants. By referring to the cases where JMD were not used, the PSNR achieve by using TV-AL3 is higher than that of SPL-DCT and SPL-DDWT. In other words, the quality of the initial reconstruction is better. This helps to improve the accuracy in image registration, and lead to better results when JMD is applied.

Table 4-5: R-D performance (dB) achieved by using the conventional BCS-SPL (DCT, DWT), BCS-TV-AL3, and the proposed scheme to encode different near-field and far-field images

			Aloe			
Subrate	**0.05**	**0.1**	**0.15**	**0.2**	**0.25**	**0.3**
SPL-DCT [103]	24.09	25.43	26.48	27.47	28.39	29.17
JMD-DCT	25.28	26.57	27.57	28.48	29.37	30.08
SPL-DDWT [103]	24.45	25.98	26.98	27.95	28.36	29.16
JMD-DDWT	25.75	27.15	28.11	28.99	29.36	30.11
TV-AL3 [102]	25.78	27.09	27.85	28.7	29.69	30.85
JMD-TV	**27.26**	**28.44**	**29.09**	**29.91**	**30.87**	**31.93**
			Baby			
Subrate	**0.05**	**0.1**	**0.15**	**0.2**	**0.25**	**0.3**
SPL-DCT [103]	27.52	30.62	32.12	33.33	34.46	35.39
JMD-DCT	29.87	32.48	33.85	34.75	35.79	36.69
SPL-DDWT [103]	27.95	30.87	32.65	33.87	34.91	35.94
JMD-DDWT	30.55	32.89	34.57	35.77	36.76	37.66
TV-AL3 [102]	29.09	31.63	33.36	34.71	36	37.17
JMD-TV	**32.34**	**34.38**	**35.79**	**36.85**	**37.93**	**38.95**
			Monopoly			
Subrate	**0.05**	**0.1**	**0.15**	**0.2**	**0.25**	**0.3**
SPL-DCT [103]	24.00	25.95	27.49	28.74	30.55	31.55
JMD-DCT	26.86	28.65	30.13	31.25	33.05	34.04
SPL-DDWT [103]	24.27	26.21	28.33	29.65	31.18	32.49
JMD-DDWT	27.20	29.09	31.10	32.3	33.81	35.09
TV-AL3 [102]	25.57	27.92	30.20	32.19	33.99	35.45
JMD-TV	**28.86**	**30.86**	**33.02**	**34.92**	**36.68**	**38.08**
			Plastic			
Subrate	**0.05**	**0.1**	**0.15**	**0.2**	**0.25**	**0.3**
SPL-DCT [103]	28.69	31.99	34.17	36.01	37.49	39.18
JMD-DCT	31.10	34.38	36.49	38.27	39.59	40.51
SPL-DDWT [103]	28.79	31.72	33.9	36.08	37.55	39.78
JMD-DDWT	29.46	32.1	34.03	35.93	37.28	39.46
TV-AL3 [102]	38.68	43.05	44.33	45.49	46.69	47.93
JMD-TV	**41.19**	**44.77**	**46.00**	**47.04**	**48.00**	**49.05**
			Bowling			
Subrate	**0.05**	**0.1**	**0.15**	**0.2**	**0.25**	**0.3**
SPL-DCT [103]	28.04	32.39	33.98	35.45	36.76	37.78
JMD-DCT	29.55	33.87	35.42	36.85	38.12	39.14
SPL-DDWT [103]	29.54	32.36	33.97	36.00	36.57	38.61
JMD-DDWT	30.18	32.94	34.34	36.2	36.69	38.67
TV-AL3 [102]	34.98	37.37	39.39	41.15	42.67	44.31
JMD-TV	**37.31**	**39.48**	**41.39**	**42.87**	**44.30**	**45.64**
			Middle-Bury			
Subrate	**0.05**	**0.1**	**0.15**	**0.2**	**0.25**	**0.3**
SPL-DCT [103]	24.13	26.27	27.59	28.92	29.92	31.03
JMD-DCT	26.85	29.03	30.15	31.22	32.19	33.12
SPL-DDWT [103]	24.87	26.85	28.55	29.96	30.93	32.08
JMD-DDWT	27.61	29.64	31.17	32.33	33.24	34.21
TV-AL3 [102]	**26.28**	**28.59**	**30.39**	**31.81**	**33.00**	**34.25**
JMD-TV	**29.14**	**31.50**	**33.13**	**34.25**	**35.36**	**36.40**

Park						
Subrate	**0.05**	**0.1**	**0.15**	**0.2**	**0.25**	**0.3**
SPL-DCT [103]	23.33	24.69	25.62	26.43	27.12	27.85
JMD-DCT	24.31	25.51	26.32	27.04	27.66	28.35
SPL-DDWT [103]	23.65	24.84	25.77	26.46	27.14	27.82
JMD-DDWT	24.50	25.54	26.37	26.96	27.74	28.17
TV-AL3 [102]	25.87	27.05	27.95	28.75	29.45	29.97
JMD-TV	**26.96**	**27.91**	**28.74**	**29.4**	**30.1**	**30.56**
Baseball						
Subrate	**0.05**	**0.1**	**0.15**	**0.2**	**0.25**	**0.3**
SPL-DCT [103]	17.12	19.12	20.77	21.98	23.14	24.44
JMD-DCT	18.98	21.01	22.61	23.71	24.81	25.90
SPL-DDWT [103]	17.15	18.70	19.98	21.29	22.37	23.73
JMD-DDWT	19.11	20.75	22.06	23.17	24.13	25.27
TV-AL3 [102]	18.05	19.85	21.64	23.18	24.83	25.95
JMD-TV	**19.61**	**21.39**	**22.98**	**24.39**	**25.81**	**26.91**
Cones						
Subrate	**0.05**	**0.1**	**0.15**	**0.2**	**0.25**	**0.3**
SPL-DCT [103]	23.87	25.67	26.88	27.95	28.77	29.64
JMD-DCT	26.41	27.65	28.67	29.60	30.34	31.04
SPL-DDWT [103]	24.40	26.14	27.20	28.22	29.03	29.83
JMD-DDWT	26.38	27.99	28.92	29.72	30.40	31.06
TV-AL3 [102]	25.53	27.64	29.02	30.21	31.25	32.21
JMD-TV	**27.67**	**29.34**	**30.54**	**31.54**	**32.51**	**33.33**
Teddy						
Subrate	**0.05**	**0.1**	**0.15**	**0.2**	**0.25**	**0.3**
SPL-DCT [103]	24.52	26.71	28.08	29.23	30.25	31.26
JMD-DCT	26.75	28.52	29.78	30.77	31.72	32.61
SPL-DDWT [103]	24.84	27.15	28.33	29.62	30.68	31.55
JMD-DDWT	26.87	29.00	30.06	31.06	31.98	32.72
TV-AL3 [102]	26.38	28.74	30.20	31.38	32.37	33.49
JMD-TV	**28.47**	**30.34**	**31.56**	**32.69**	**33.60**	**34.61**

***Note*:** The bold values relates to the maximum PSNR (dB) reached for a given subrate and image

ii. Subrates

In this subsection, the effect of I_R subrate on the reconstruction of I_{NR} using the proposed scheme is evaluated. From the previous evaluation, we know that JMD-TV performs better than JMD-DCT and JMD-DDWT. Hence, we are using JMD-TV in this experiment. The different setups are defined in Table 4.6.

Table 4-6: Different subrate setups

Setup	Subrate
1	$\mathbf{M_R = M_{NR}}$
2	$\mathbf{M_R= 0.3, M_{NR}=0.05,0.1,0.15,0.2,0.25,0.3}$
3	$\mathbf{M_R= 0.5, M_{NR}=0.05,0.1,0.15,0.2,0.25,0.3}$

In the first setup, both I_R and I_{NR} are transmitted at the same subrate i.e. $M_R=M_{NR}$. In the second setup, I_R is transmitted at a fixed subrate of $M_R=0.3$ and I_{NR} is transmitted at different subrates that range from $M_{NR}=0.05$ to 0.3. M_R is increased to 0.5 in the third setup, with the rest settings remain the same as the second setup. The gain in the first setup ($\mathbf{M_R=M_{NR}}$) is lower than that of the second and third setup. When $\mathbf{I_R}$ and $\mathbf{I_{NR}}$ are transmitted at the same rate, the reconstructed $\mathbf{I_R}$ does not contain information that could significantly help the reconstruction of $\mathbf{I_{NR}}$.

Table 4-7: R-D Performance (dB) comparison of the proposed JMD and conventional BCS-TV-AL3 with different subrate setups defined in Table 4.6 for various multi-view test images

			Aloe			
Subrate	**0.05**	**0.1**	**0.15**	**0.2**	**0.25**	**0.3**
BCS-TV-AL3 [102]	26.01	29.49	28.15	28.93	29.89	30.95
JMD-TV ($\mathbf{M_R=M_{NR}}$)	25.65	26.81	27.66	28.38	29.15	29.94
JMD-TV ($\mathbf{M_R= 0.3}$)	26.42	27.65	28.23	29.07	29.99	31.08
JMD-TV ($\mathbf{M_R= 0.5}$)	**27.26**	**28.44**	**29.09**	**29.91**	**30.87**	**31.93**
			Baby			
Subrate	**0.05**	**0.1**	**0.15**	**0.2**	**0.25**	**0.3**
BCS-TV-AL3 [102]	29.09	31.63	33.36	34.71	36.00	37.17
JMD-TV ($\mathbf{M_R=M_{NR}}$)	30.10	32.52	34.17	35.46	36.68	37.84
JMD-TV ($\mathbf{M_R= 0.3}$)	31.46	33.59	34.93	35.92	36.96	37.99
JMD-TV ($\mathbf{M_R= 0.5}$)	**32.34**	**34.38**	**35.79**	**36.85**	**37.93**	**38.95**
			Monopoly			
Subrate	**0.05**	**0.1**	**0.15**	**0.2**	**0.25**	**0.3**
BCS-TV-AL3 [102]	25.57	27.92	30.20	32.19	33.99	35.45
JMD-TV ($\mathbf{M_R=M_{NR}}$)	26.42	28.65	30.95	32.89	34.67	36.07
JMD-TV ($\mathbf{M_R= 0.3}$)	28.14	29.80	31.95	33.75	35.17	36.60
JMD-TV ($\mathbf{M_R= 0.5}$)	**28.86**	**30.86**	**33.02**	**34.92**	**36.68**	**38.08**

Plastic						
Subrate	**0.05**	**0.1**	**0.15**	**0.2**	**0.25**	**0.3**
BCS-TV-AL3 [102]	38.68	43.05	44.33	45.49	46.69	47.93
JMD-TV (M_R=M_{NR})	41.03	44.15	45.27	46.35	47.54	47.94
JMD-TV (M_R= 0.3)	41.07	44.42	45.84	46.76	47.87	48.65
JMD-TV (M_R= 0.5)	**41.19**	**44.77**	**46.00**	**47.04**	**48.00**	**49.05**
Middle-Bury						
Subrate	**0.05**	**0.1**	**0.15**	**0.2**	**0.25**	**0.3**
BCS-TV-AL3 [102]	26.28	28.59	30.39	31.81	33.00	34.25
JMD-TV (M_R=M_{NR})	27.58	29.88	31.69	33.11	34.31	35.55
JMD-TV (M_R= 0.3)	28.72	30.55	32.25	33.55	34.83	35.61
JMD-TV (M_R= 0.5)	**29.14**	**31.50**	**33.13**	**34.25**	**35.36**	**36.40**

***Note*:** The bold values relates to the maximum PSNR (dB) reached for a given subrate and image

iii. Relationship between Proposed Scheme and Camera Separation

In this subsection, the effect of camera separation on the performance of proposed JMD-TV is evaluated and compared with independent BCS-TV-AL3. As shown in Figure 4.13, we have selected seven views from the datasets, and one of them (S_n) is chosen to serve as the reference view. Each view is separated by a specific distance i.e. approximately 15cm from its neighboring view [119]. Each time, one out of six remaining views (S_{n-3}, S_{n-2}, S_{n-1}, S_{n+1}, S_{n+2}, S_{n+3}) will be chosen to serve as the non-reference view and pair with S_n.

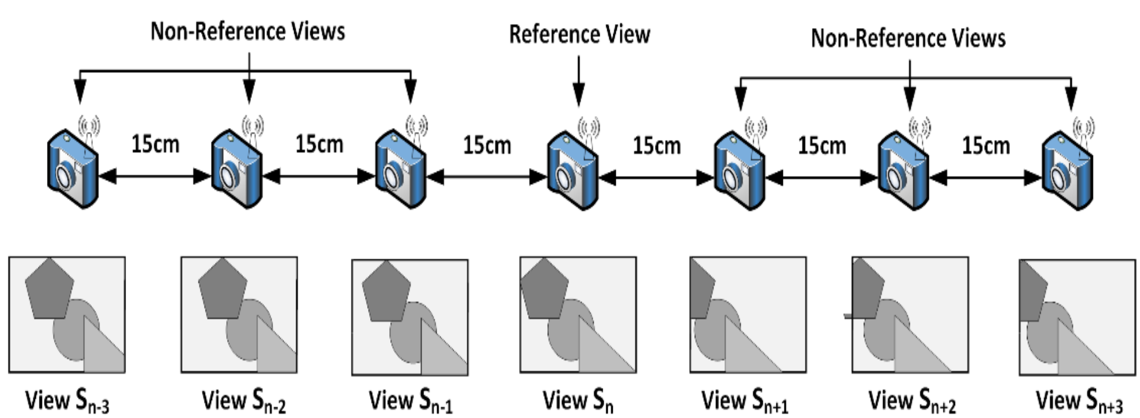


Figure 4.13: Joint reconstruction using one reference and one non-reference view

Because we observe that the reconstruction of using the left or right neighboring non-reference image (e.g. S_{n+1} or S_{n-1}) yields approximately the same results, the PSNR values obtained from using the left neighboring views are average up with their counterparts from the right for all dataset.

Table 4-8: PSNR (dB) comparison of the proposed JMD-TV and conventional BCS-TV-AL3 with various camera separations (percentage of overlap) for different multi-view test images

Aloe → Reference View S_n @ 0.5 = 41.05dB							
Non-Reference Views	**Subrate**	**0.05**	**0.1**	**0.15**	**0.2**	**0.25**	**0.3**
	BCS-TV-AL3	25.78	27.09	27.85	28.7	29.69	30.85
$S_{n\pm1}$	**JMD-TV**	**27.26**	28.44	29.09	29.91	30.87	31.93
$S_{n\pm2}$	**JMD-TV**	26.77	29.53	28.98	29.74	30.64	31.56
$S_{n\pm3}$	**JMD-TV**	27.08	**29.67**	**29.32**	**30.07**	**30.94**	**31.97**
Baby → Reference View S_n @ 0.5 = 41.75dB							
Non-Reference Views	**Subrate**	**0.05**	**0.1**	**0.15**	**0.2**	**0.25**	**0.3**
	BCS-TV-AL3	29.09	31.63	33.36	34.71	36	37.17
$S_{n\pm1}$	**JMD-TV**	**32.34**	**34.38**	**35.79**	**36.85**	**37.93**	**38.95**
$S_{n\pm2}$	**JMD-TV**	30.30	32.40	34.02	35.28	36.47	37.65
$S_{n\pm3}$	**JMD-TV**	29.97	32.2	33.72	35.14	36.19	37.36
Monopoly → Reference View S_n @ 0.5 = 41.95dB							
Non-Reference Views	**Subrate**	**0.05**	**0.1**	**0.15**	**0.2**	**0.25**	**0.3**
	BCS-TV-AL3	25.57	27.92	30.20	32.19	33.99	35.45
$S_{n\pm1}$	**JMD-TV**	**28.86**	**30.86**	**33.02**	**34.92**	**36.68**	**38.08**
$S_{n\pm2}$	**JMD-TV**	28.05	30.06	32.15	33.86	35.55	36.72
$S_{n\pm3}$	**JMD-TV**	26.78	29.11	31.12	32.72	34.61	35.83
Plastic → Reference View S_n @ 0.5 = 51.95dB							
Non-Reference Views	**Subrate**	**0.05**	**0.1**	**0.15**	**0.2**	**0.25**	**0.3**
	BCS-TV-AL3	38.68	43.05	44.33	45.49	46.69	47.93
$S_{n\pm1}$	**JMD-TV**	**41.19**	**44.77**	**46.00**	**47.04**	**48.00**	**49.05**
$S_{n\pm2}$	**JMD-TV**	39.84	44.18	45.37	46.32	47.78	48.13
$S_{n\pm3}$	**JMD-TV**	39.16	43.45	44.76	45.74	47.09	48.03
Middle-Burry → Reference View S_n @ 0.5 = 41.65dB							
Non-Reference Views	**Subrate**	**0.05**	**0.1**	**0.15**	**0.2**	**0.25**	**0.3**
	BCS-TV-AL3	26.28	28.59	30.39	31.81	33.00	34.25
$S_{n\pm1}$	**JMD-TV**	**29.14**	**31.50**	**33.13**	**34.25**	**35.36**	**36.40**
$S_{n\pm2}$	**JMD-TV**	29.01	31.17	32.84	33.95	35.16	36.34
$S_{n\pm3}$	**JMD-TV**	28.66	30.49	32.61	34.08	35.14	36.43

***Note*:** The bold values relates to the maximum PSNR (dB) reached for a given subrate and image

The simulation results in Table 4.8 show that as the separation between the views increases the gain decreases. For smaller separations (S_n-$S_{n\pm1}$) the proposed scheme provides an average gain of 1.5dB to 3dB, whereas for larger separation (S_n-$S_{n\pm3}$) the gain reduces to an average of ~0.5dB to1.5dB when moving from higher to lower subrates. As the distance between the reference and non-reference images increases, the correlation between them is reducing, leading to less accurate registration and fusion of the images.

4.4.3 Joint Reconstruction with Multiple Reference Nodes

In this subsection, the effect of using two or more reference nodes in the proposed scheme is evaluated and compared with BCS-TV-AL3. We only present the results obtained from the Baby, Monopoly, and Middle-bury datasets, because similar trends w.r.t. the gain was observed in remaining datasets (Aloe, Plastic, Bowling) that we have tested. As shown in Figure 4.14, we have selected seven views from the datasets, and one of them ($\mathbf{S_n}$) is chosen to serve as the non-reference view. Each time, two out of the six remaining views ($\mathbf{S_{n-3}}$+ $\mathbf{S_{n+3}}$, $\mathbf{S_{n-2}}$ + $\mathbf{S_{n+2}}$, $\mathbf{S_{n-1}}$ + $\mathbf{S_{n+1}}$) will be chosen to serve as the two reference views and pair with $\mathbf{S_n}$. E.g., in the first trial, $\mathbf{S_{n-1}}$and S_{n+1} were selected to work as the two reference views where the images captured by them are used to aid the reconstruction of images captured by $\mathbf{S_n}$.

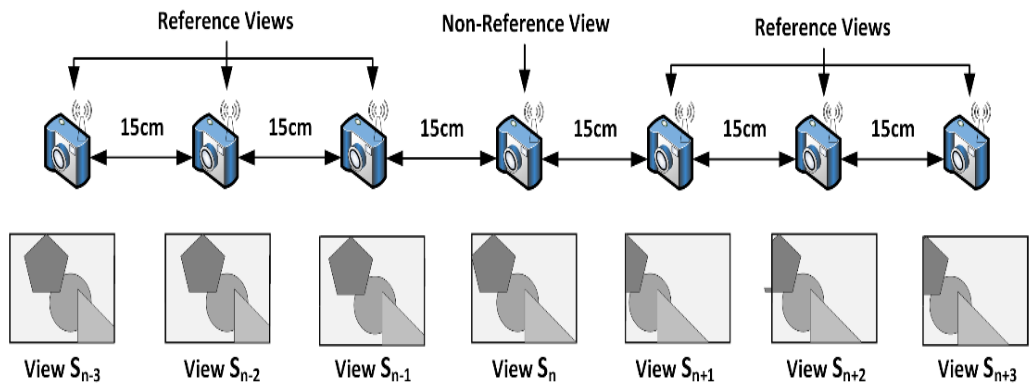


Figure 4.14: Joint reconstruction using two reference and one non-reference view

Table 4-9: R-D performance (dB) achieved using conventional BCS-TV-AL3 and the proposed JMD with multiple reference nodes (S_{n+1}-S_{n-1}, S_{n+2}-S_{n-2}, S_{n+3}-S_{n+3}) and one non-reference node for different multi-view test images (Baby, Monopoly, Middlebury)

Baby	**Subrate**	**0.05**	**0.1**	**0.15**	**0.2**	**0.25**	**0.3**
Reference Image @ 0.5 S_{n-1} = 41.79 S_{n+1} = 41.92	**BCS-TV-AL3**	29.09	31.63	33.36	34.71	36.00	37.17
	JMD-TV	33.38	35.21	36.5	37.42	38.47	39.46
	Gain	**4.29**	**3.58**	**3.14**	**2.71**	**2.47**	**2.29**
Reference Image @ 0.5 S_{n-2} = 41.77 S_{n+2} = 41.82	**BCS-TV-AL3**	29.09	31.63	33.36	34.71	36.00	37.17
	JMD-TV	32.27	34.27	35.64	36.6	37.72	38.72
	Gain	**3.18**	**2.64**	**2.28**	**1.89**	**1.72**	**1.55**
Reference Image @ 0.5 S_{n-3} = 41.77 S_{n+3} = 41.90	**BCS-TV-AL3**	29.09	31.63	33.36	34.71	36	37.17
	JMD-TV	31.06	33.27	34.83	36.00	37.06	38.01
	Gain	**1.97**	**1.64**	**1.47**	**1.29**	**1.06**	**0.84**
Monopoly	**Subrate**	**0.05**	**0.1**	**0.15**	**0.2**	**0.25**	**0.3**
Reference Image @ 0.5 S_{n-1} = 41.18 S_{n+1} = 42.16	**BCS-TV-AL3**	25.57	27.92	30.20	32.19	33.99	35.45
	JMD-TV	30.11	31.95	33.89	35.71	37.4	38.68
	Gain	**4.54**	**4.03**	**3.69**	**3.52**	**3.41**	**3.23**
Reference Image @ 0.5 S_{n-2} =41.68 S_{n+2} = 41.08	**BCS-TV-AL3**	25.57	27.92	30.20	32.19	33.99	35.45
	JMD-TV	28.82	30.77	32.74	34.64	36.33	37.61
	Gain	**3.25**	**2.85**	**2.54**	**2.45**	**2.34**	**2.16**
Reference Image @ 0.5 S_{n-3} = 41.66 S_{n+3} = 41.87	**BCS-TV-AL3**	25.57	27.92	30.20	32.19	33.99	35.45
	JMD-TV	27.69	29.81	31.8	33.59	35.26	36.46
	Gain	**2.12**	**1.89**	**1.6**	**1.4**	**1.27**	**1.01**
Middlebury	**Subrate**	**0.05**	**0.1**	**0.15**	**0.2**	**0.25**	**0.3**
Reference Image @ 0.5 S_{n-1} = 41.18 S_{n+1} = 42.162	**BCS-TV-AL3**	26.28	28.59	30.39	31.81	33.00	34.25
	JMD-TV	30.52	32.68	34.15	35.21	36.18	37.07
	Gain	**4.24**	**4.09**	**3.76**	**3.40**	**3.18**	**2.82**
Reference Image @ 0.5 S_{n-2} = 41.68 S_{n+2} = 41.08	**BCS-TV-AL3**	26.28	28.59	30.39	31.81	33.00	34.25
	JMD-TV	29.40	31.70	33.27	34.5	35.46	36.36
	Gain	**3.12**	**3.11**	**2.88**	**2.69**	**2.46**	**2.11**
Reference Image @ 0.5 S_{n-3} = 41.66 S_{n+3} = 41.87	**BCS-TV-AL3**	26.28	28.59	30.39	31.81	33.00	34.25
	JMD-TV	28.94	31.09	32.77	34.01	34.95	35.76
	Gain	**2.66**	**2.50**	**2.38**	**2.20**	**1.95**	**1.51**

Note**:** The bold values relates to the performance gain (dB) of the proposed JMD over independent BCS-TV-AL3.

Table 4.9 presents the average PSNR obtained across all the views for each multi-view dataset. The results show that the JMD-TV is better than independent BCS-TV-AL3 on average by 2.5dB to 4dB from higher to lower subrates. Even when the camera separations are larger the proposed JMD still managed to provide a gain of ~1dB to 2dB.

In fact, we have also evaluated the reconstruction by using three and four reference views. From what we can observe, the gain is higher when more reference views were used. For example, the gain can increase to an average of 0.4dB to 0.8dB for higher to lower subrates when three reference views were adopted as shown in Table 4.10.

Table 4-10: R-D performance (dB) achieved using the proposed JMD with three reference nodes (S_{n+1}, S_{n-1}, S_{n+2} or S_{n-2}, S_{n-1}, S_{n+1}) and one non-reference node (Sn) for different multi-view test images (Baby, Monopoly, Middlebury)

Baby	**Subrate**	**0.05**	**0.1**	**0.15**	**0.2**	**0.25**	**0.3**
Reference Image @ 0.5	**BCS-TV-AL3**	29.09	31.63	33.36	34.71	36.00	37.17
	JMD-TV	33.99	35.75	37.00	37.85	38.87	39.83
	Gain	**4.90**	**4.12**	**3.64**	**3.14**	**2.87**	**2.66**
Monopoly	**Subrate**	**0.05**	**0.1**	**0.15**	**0.2**	**0.25**	**0.3**
Reference Image @ 0.5	**BCS-TV-AL3**	25.57	27.92	30.20	32.19	33.99	35.45
	JMD-TV	31.05	32.75	34.61	36.31	37.94	39.12
	Gain	**5.48**	**4.83**	**4.41**	**4.12**	**3.95**	**3.67**
Middle-Bury	**Subrate**	**0.05**	**0.1**	**0.15**	**0.2**	**0.25**	**0.3**
Reference Image @ 0.5	**BCS-TV-AL3**	26.28	28.59	30.39	31.81	33.00	34.25
	JMD-TV	31.16	33.39	34.93	36.15	37.16	38.11
	Gain	**4.88**	**4.8**	**4.54**	**4.34**	**4.16**	**3.86**

***Note*:** The bold values relates to the performance gain (dB) of the proposed JMD over independent BCS-TV-AL3.

However, the increment is not as significant when we used four reference views. In this case, the gain over three reference views is only limited to an average of 0.1dB to ~0.3dB. In addition to this, we also noticed that increasing the number of reference views does not help to improve the performance when the camera separation remained large e.g. when the S_{n-3} and S_{n+3} were used.

4.4.4 Joint Reconstruction of Multiple Non-Reference with Two Reference Nodes

In this section, the multiple non-reference views are reconstructed from two fixed reference view. In order to analyze the outcomes, five different views (S_{n-2}, S_{n-1}, S_n,S_{n+1}, S_{n+2}) of the same scene are selected from the image datasets that we used. To simplify the explanation, we consider S_{n+1}, S_n, S_{n-1} as the non-reference views and S_{n-2} and S_{n+2} as their left and right neighboring reference view as shown in Figure 4.15.

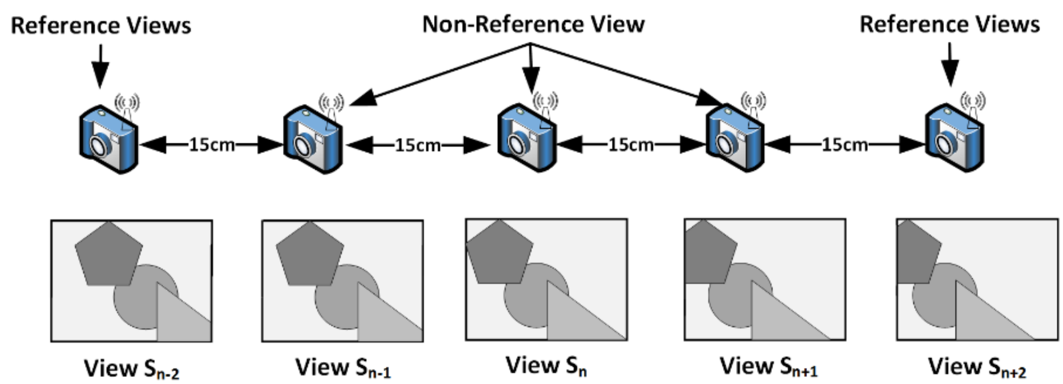


Figure 4.15: View S_{n+1}, S_n, S_{n-1} are the non-reference view, whereas view S_{n-2} and S_{n+2} are considered the left and right neighboring reference views of them

Table 4.11 presents the average PSNR obtained for each non-reference view (S_{n+1}, S_n, S_{n-1}). From the results, it can be noticed that the non-reference views that are nearer w.r.t any one of their left or right neighboring reference views provides better gains than that having both left or right neighboring reference views staying further away. This is due to the higher correlation between the views that are sitting closer to each other.

Table 4-11: R-D performance (dB) achieved by using the proposed scheme with fixed neighboring reference views and variable non-reference views for multi-view (Baby, Monopoly, Middlebury) test images

Baby		**Subrate**	**0.05**	**0.1**	**0.15**	**0.2**	**0.25**	**0.3**
Reference Image @ 0.5 S_{n-2} = 41.79 S_{n+2} = 41.62	S_{n-1}	**BCS-TV-AL3**	28.96	31.51	33.29	34.67	35.93	37.13
		JMD-TV	**32.53**	**34.72**	**36.26**	**37.31**	**38.44**	**39.49**
	S_n	**BCS-TV-AL3**	29.09	31.63	33.36	34.71	36.00	37.17
		JMD-TV	32.27	34.27	35.64	36.6	37.72	38.72
	S_{n+1}	**BCS-TV-AL3**	29.14	31.62	33.35	34.72	36.06	37.23
		JMD-TV	32.61	34.59	36.07	37.06	38.25	39.23
Monopoly		**Subrate**	**0.05**	**0.1**	**0.15**	**0.2**	**0.25**	**0.3**
Reference Image @ 0.5 S_{n-2} = 41.39 S_{n+2} = 41.80	S_{n-1}	**BCS-TV-AL3**	25.08	27.63	29.90	31.78	33.58	35.30
		JMD-TV	**28.93**	**31.21**	**33.00**	**34.79**	**36.47**	**37.77**
	S_n	**BCS-TV-AL3**	25.57	27.92	30.20	32.19	33.99	35.45
		JMD-TV	28.82	30.77	32.74	34.64	36.33	37.61
	S_{n+1}	**BCS-TV-AL3**	25.50	27.86	30.14	32.17	33.95	35.69
		JMD-TV	29.17	31.39	33.17	35.01	36.64	38.01
Middle-Bury		**Subrate**	**0.05**	**0.1**	**0.15**	**0.2**	**0.25**	**0.3**
Reference Image @ 0.5 S_{n-2} = 41.39 S_{n+2} = 41.80	S_{n-1}	**BCS-TV-AL3**	26.15	28.52	30.36	31.74	33.09	34.26
		JMD-TV	**29.44**	**31.93**	**33.54**	**34.69**	**35.68**	**36.79**
	S_n	**BCS-TV-AL3**	26.28	28.59	30.39	31.81	33.00	34.25
		JMD-TV	29.20	31.70	33.27	34.50	35.46	36.36
	S_{n+1}	**BCS-TV-AL3**	26.51	28.78	30.6	32.14	33.47	34.79
		JMD-TV	29.54	32.02	33.68	34.78	35.84	36.96

***Note*:** The bold values relates to the maximum PSNR (dB) reached for a given subrate and image

4.4.5 Comparison of Proposed Scheme with Other Multi-view CS Compression Schemes

In this subsection, the proposed JMD is compared with several standard multi-view CS joint reconstruction schemes. In this case, we focused on the performance at lower subrates because we think that it is not feasible for a battery-powered device to always encode and send the captured images at high subrate. Higher subrate will produce more measurements and hence will take more time and energy to transmit. All the simulation results that we obtained are summarized in Table 4.12, presented in terms of gain i.e. joint reconstruction over independent reconstruction.

Table 4-12: R-D performance gain (dB) comparison of the proposed JMD with standard multi-view reconstruction schemes for various multi-view test images

			Aloe			
Subrate	**0.05**	**0.1**	**0.15**	**0.2**	**0.25**	**0.3**
JMD-TV	**1.87**	**1.79**	**1.68**	**1.66**	1.58	1.47
JMD-DDWT	1.30	1.17	1.13	1.04	1.00	0.95
DC-BCS-SPL [76]	0.48	0.73	-	1.64	**-**	**2.33**
M-CS-Residual [73]	0.35	0.57	0.95	1.20	1.28	1.38
k-t FOCUSS [74]	0.33	0.57	0.92	1.11	1.20	1.25
			Baby			
Subrate	**0.05**	**0.1**	**0.15**	**0.2**	**0.25**	**0.3**
JMD-TV	**4.29**	**3.58**	**3.14**	**2.71**	**2.47**	2.29
JMD-DDWT	2.60	2.02	1.92	1.9	1.85	1.72
DC-BCS-SPL [76]	0.78	1.13	-	1.99	**-**	**2.47**
M-CS-Residual [73]	0.53	0.93	1.23	1.39	1.55	1.68
k-t FOCUSS [74]	0.50	0.84	1.12	1.29	1.39	1.55
			Monopoly			
Subrate	**0.05**	**0.1**	**0.15**	**0.2**	**0.25**	**0.3**
JMD-TV	**4.54**	**4.03**	**3.69**	**3.52**	3.41	3.23
JMD-DDWT	2.93	2.88	2.77	2.65	2.63	2.60
DC-BCS-SPL [76]	0.68	1.11	-	3.33	**-**	**4.30**
M-CS-Residual [73]	0.50	0.90	1.68	2.23	2.37	2.54
k-t FOCUSS [74]	0.48	0.90	1.52	1.99	2.27	2.49
			Plastic			
Subrate	**0.05**	**0.1**	**0.15**	**0.2**	**0.25**	**0.3**
JMD-TV	**2.51**	**1.72**	**1.67**	**1.55**	**1.31**	**1.12**
JMD-DDWT	0.67	0.38	0.13	-0.15	-0.27	-0.32
DC-BCS-SPL [76]	0.20	-0.34	-	-1.04	-	-0.75
M-CS-Residual [73]	0.23	0.1	-0.23	-0.50	-0.47	-0.32
k-t FOCUSS [74]	0.1	-0.20	-0.33	-0.66	-0.60	-0.48
			Bowling			
Subrate	**0.05**	**0.1**	**0.15**	**0.2**	**0.25**	**0.3**
JMD-TV	**2.83**	**2.68**	**2.49**	**2.35**	2.14	2.09
JMD-DDWT	0.64	0.58	0.37	0.20	0.11	0.06
DC-BCS-SPL [76]	0.50	0.96	-	2.02	**-**	**2.93**
M-CS-Residual [73]	0.44	0.78	0.99	1.13	1.23	1.35
k-t FOCUSS [74]	0.40	0.74	0.89	1.02	1.17	1.23

Note**:** The bold values relates to the maximum PSNR (dB) gain reached for a given subrate and image

The results for Modified-CS-Residual [73] and k-t FOCUSS [74] are implemented by modifying their available code [125], [126] respectively, with respect to the experimental setup discussed in Section 4.4. The results of DC-BCS-SPL using DDWT are directly obtained from [76], as the implementation was not readily available at the time of writing. The DDWT version is chosen because it gives the best performance as reported by [76].

From the simulation and analysis, it can be concluded that the proposed scheme provides substantial reconstruction gain at lower subrates when compared with other standard multi-view reconstruction schemes. This is because as the subrate increases, larger set of measurements is used to represent I_{NR}. The additional measurements help to reduce the prediction errors and improve the reconstruction of I_{NR}, even before the proposed scheme is applied as required in DC-BCS-SPL. Hence, the quality of neighboring images is not much better than I_{NR}, and this limited the gain that can be achieved by making use of the correlation information of the neighboring images.

Moreover, we also noticed that in datasets that do not contain any distinct object, such as Bowling, our JMD-DDWT performance gain decreases dramatically. In the case like this, the I_r obtained does not show major variation features than the I_{NR} i.e. it does not contain the important information such as edges and high frequency. Furthermore, the correlation between neighboring pixels within the image is too high.

4.4.6 Comparison of Proposed Scheme with Different Numbers of Reference Node

In this subsection, we compared various setups of JMD to find out the one that provides better performance. In this case, the reconstruction quality of I_{NR} using different setups was evaluated. The different setups selected for the evaluation are described in Table 4.13 and discussed as follows.

Table 4-13: Different setups

Setup	Subrate
1	**Single Reference View + Single Non-Reference View**
2	**Multiple Reference (Same Subrates) + Single Non-Reference View**
3	**Multiple Reference (Different Subrates) + Single Non-Reference View**

In Setup 1, only two views are selected, where one is considered as the reference image I_R, which is used to reconstruct it's neighboring non-reference image I_{NR}. Then in Setup 2; three views are selected. In this case, two views referred to as I_{R1} and I_{R2} (left and right neighbors of I_{NR}) are used to aid the reconstruction of a single I_{NR}. It should be noted that in these two setups, I_{NR} is transmitted at a lower subrates (0.05 to 0.3) when compared with I_R / I_{R1}/I_{R2} (0.5). Overall, the idea is to improve the reconstruction of I_{NR} using I_R / I_{R1}/ I_{R2}. Setup 3 is similar to Setup 2; the only difference is that I_{R1} is transmitted at subrate of 0.5 while I_{R2} at subrate same as $I_{NR.}$ Other settings remain the same.

Table 4.14 presents the average PSNR for dataset Baby, Monopoly and Middlebury using the setups mentioned in Table 4.13. The results show that the reconstruction of $\mathbf{I_{NR}}$ in Setup 2 and 3 provides better results than that of Setup 1. This is because, as the number of reference views increases, more correlation between the images can be exploited, and leading to more accurate registration and fusion.

Table 4-14: R-D performance (dB) comparison of the proposed JMD with different setups for different multi-view test images

			Baby			
Subrate	**0.05**	**0.1**	**0.15**	**0.2**	**0.25**	**0.3**
BCS-TV-AL3	29.09	31.63	33.36	34.71	36.00	37.17
Setup1	32.34	34.38	35.79	36.85	37.93	38.95
Setup2	**33.38**	**35.21**	**36.5**	**37.42**	**38.47**	**39.46**
Setup3	32.87	34.87	36.21	37.25	38.29	39.26
			Monopoly			
Subrate	**0.05**	**0.1**	**0.15**	**0.2**	**0.25**	**0.3**
BCS-TV-AL3	25.57	27.92	30.20	32.19	33.99	35.45
Setup1	28.86	30.86	33.02	34.92	36.68	38.08
Setup2	**30.11**	**31.95**	**33.89**	**35.71**	**37.40**	**38.68**
Setup3	29.48	31.35	33.42	35.24	36.96	38.29
			Middle-Bury			
Subrate	**0.05**	**0.1**	**0.15**	**0.2**	**0.25**	**0.3**
BCS-TV-AL3	26.28	28.59	30.39	31.81	33.00	34.25
Setup1	29.14	31.50	33.13	34.25	35.36	36.40
Setup2	**30.52**	**32.68**	**34.15**	**35.21**	**36.18**	**37.07**
Setup3	29.73	31.99	33.55	34.65	35.68	36.67

***Note*:** The bold values relates to the maximum PSNR (dB) reached for a given subrate and image

On average, Setup 2 provides a gain of 0.5dB-1dB over Setup 1 for higher to lower subrates. However, it should also be noted that Setup 2 uses two reference views for the reconstruction of $I_{NR.}$ Assuming that a fixed number of sensor nodes is to be deployed, an increase in a number of reference nodes will result in fewer non-reference nodes. This leads to an increase in data transmission as reference nodes are transmitting images at higher subrates (0.5).

To overcome the issue mentioned above, we consider the same scenario as described in Setup 2, but rather than transmitting both I_{R1} and I_{R2} at the higher subrate, I_{R1} is transmitted at 0.5 whereas I_{R2} is transmitted at the same subrate of $I_{NR.}$ By doing so, although there are two reference nodes, but only one reference node is required to transmit the images at higher subrate (0.5). From the results, it can be seen that it is able to provide an average gain of ~0.3-0.5 dB over Setup 1. From the discussion above, it can be concluded that Setup 2 is an optimum choice for the multi-view scenario when compared to Setup 1. However, there is a trade-off between the data transmission and image quality. Hence, one can consider using Setup 3, which also outperform Setup 1, but with the similar data transmission.

4.4.7 Number of Measurements

In this subsection, we observe the number of measurements that could be reduced by using the proposed JMD, when compared with the independent schemes. Table 4.15 tabulates the rate saving percentage at different reconstructed qualities for various multi-view image datasets. From the results, it can be noticed that measurements savings vary for different reconstruction qualities i.e. for higher reconstruction quality the saving rate is ~30%, whereas for lower reconstruction quality the saving rate is ~66%. This is because the effect of the SPI at higher measurement rate is lower than when at lower measurements. The SPI contains a larger set of measurements to represent $\mathbf{I_{NR}}$ that reduces the prediction errors and improve the reconstruction of $\mathbf{I_{NR}}$, even before the proposed scheme is applied. Hence, the quality of neighboring images is not much better than $\mathbf{I_{NR}}$, and this limited the gain that can be achieved by making use of the correlation information of the neighboring images.

Table 4-15: Compression performance comparison of the proposed JMD with independent scheme in terms of measurement saved for various multi-view test images at different reconstruction quality (PSNR)

PSNR	Measurements BCS-TV-AL3	Measurements JMD-TV	Measurements Saved	Measurement Saving (%)	Measurement (Average %)
			Aloe		
~27.26	39004	13056	25948	66	
~29.09	65156	39424	25732	39	44
~30.87	90648	64936	25712	28	
			Baby		
PSNR	**Measurements BCS-TV-AL3**	**Measurements JMD-TV**	**Measurements Saved**	**Measurement Saving (%)**	**Measurement (Average %)**
~33.38	39424	13056	26368	66	
~36.00	65456	39224	26232	40	45
~38.47	91648	65536	25912	30	
			Monopoly		
PSNR	**Measurements BCS-TV-AL3**	**Measurements JMD-TV**	**Measurements Saved**	**Measurement Saving (%)**	**Measurement (Average %)**
~30.20	39424	13056	26368	66	
~33.99	65456	39424	26032	40	45
~36.70	91648	64936	26312	29	
			Middle-Bury		
PSNR	**Measurements BCS-TV-AL3**	**Measurements JMD-TV**	**Measurements Saved**	**Measurement Saving (%)**	**Measurement (Average %)**
~30.39	39424	12956	26468	67	
~33.00	65536	39024	26512	41	46
~35.50	91648	60480	27168	31	

On average, the number of measurement saved by the proposed scheme against the independent scheme is ~45% for all multi-view images at different reconstruction qualities.

4.4.8 Execution Time

In this subsection, the computation time of the proposed scheme and the other multi-view reconstruction schemes are evaluated. All the schemes are implemented using MATLAB ver. 8.3.0.532 (R2014a) on an Intel(R) Xeon(R), CPU E5-1620 desktop computer with 3.6 GHz processor and 8GB RAM. However, it is important to note that all the implementations have not been particularly optimized for execution speed. Table 4.16

presents the average execution times required to obtain the final reconstruction image at various subrates of 5 independent trials.

Table 4-16: Average Execution Time (sec) comparison of the proposed JMD with standard multi-view reconstruction schemes for different multi-view test images at various subrates

Samples	Subrate	0.05	0.1	0.15	0.2	0.25	0.3
	Schemes	**Execution Time (Sec)**					
Aloe	**JMD-TV**	**40.35**	**38.51**	**36.44**	**35.10**	**34.29**	**33.69**
	kt-Focuss[74]	63.21	59.51	57.01	55.84	52.94	49.50
	M-CS-Residual[73]	765	741	724	709	693	676
Baby	**JMD-TV**	**35.37**	**33.63**	**32.98**	**31.13**	**30.48**	**28.54**
	kt-Focuss[74]	57.17	55.87	52.12	49.15	47.12	44.01
	M-CS-Residual[73]	739	725	711	697	679	661
Monopoly	**JMD-TV**	**37.06**	**35.88**	**34.08**	**33.39**	**32.11**	**30.87**
	kt-Focuss[74]	59.51	58.07	56.87	53.74	51.84	48.97
	M-CS-Residual[73]	747	731	719	703	689	673

Note: The bold values relates to the minimum Execution Time (sec) reached for a given subrate and image

The experimental results presented in Table 4.16 show that the execution time of the proposed scheme is shorter than other schemes. This is because the BCS that is less computationally complex and capable of providing fast initial reconstruction is employed in our proposed scheme. Moreover, we use less complex registration and fusion methods to project the reference image ($\mathbf{I_R}$) onto the non-reference image ($\mathbf{I_{NR}}$). It can be observed that all the schemes take more time to reconstruct a view when lower subrates are adopted because to find a good prediction from a more limited amount of measurements is harder and hence more time consuming.

4.5. Conclusive Remarks

In this chapter, a multi-view compression based on BCS and joint-decoding is proposed. The encoding is performed by using BCS to reduce the hardware complexity. The block based approach simplifies the implementation and storage on the visual node, and provides significantly faster reconstruction. At the decoder, a SPI is generated. The SPI is the outcome of exploiting the inter-view redundancies present in the multi-view images captured by different visual nodes. It works well for both near-field and far-field images and could handle the parallax and occlusion issues. Furthermore, it does not require motion estimation or motion compensation as in most of the conventional compression scheme. Experimental results show that the proposed scheme can be applied to images with low, medium and high texture variations. It can outperform the different independent BCS compression by a margin of 1.5dB to 3dB at various subrates. Furthermore, when compared with other standard multi-view CS compression scheme the proposed scheme shows a gain of 1.5dB-2 dB at lower subrates, and the reconstruction speed is also 30%-40% shorter.

Chapter 5
SINGLE-VIEW VIDEO COMPRESSION AND RECONSTRUCTION

The conventional video compression schemes are usually based on complex-encoder simple-decoder paradigm. Generally, the encoding of videos frames involves using Motion Estimation and Compensation (ME/MC) to exploit the spatial and temporal redundancies among the frames. In contrast, the decoding of such compressed videos is much simpler. The conventional paradigm is not suitable for Visual Sensor Network (VSN) where complex encoding is to be performed by visual nodes with primitive hardware while simple decoding is performed by powerful server with advance hardware. Hence, using Compressive Sensing (CS) is one of the better solutions.

In this chapter, we show that how the proposed scheme described in Chapter 4 can be modified and extended to replace conventional video compression. The focus is on using the proposed scheme to reduce the redundancies between video frames at the decoder. The relationship between frames correlation and compression performance is also exploited. Different ways of arranging the frames have been investigated to determine the one that yields better results. In addition to this, a quantization approach is proposed to transform the CS measurements produced by the visual nodes into bits. This allows us to compare the proposed scheme with other conventional video compression scheme.

This chapter is organized as follows. Section 5.1 presents an overview of using the proposed scheme for video compression. Section 5.2 explain in detail on how the proposed scheme was modified and extended for video compression, as well as the proposed quantization approach for CS measurements that incorporates Scalar Quantization (SQ) with Adaptive Differential Pulse Code Modulation (ADPCM). All the experimental results are presented and discussed in Section 5.3. Section 5.4 concludes this chapter.

5.1. Single-view Video Compression and Reconstruction Model

Consider the model shown in Figure 5.1, where a visual node **S** is recording a video. In this case, each frame is encoded using Block-based Compressive Sensing (BCS) and transmitted to the server independently. At the server, we consider a set of J consecutive frames received from the sensor node S as a Group of Pictures (GoP). Since the video is continuous, we also assume that another GoP tails the current GoP. The GoP consists of a key frame F_K (the first), and J-1 non-key frames F_{NK}. The F_K and F_{NK} are encoded at subrate of M_K and M_{NK} respectively, with $M_K > M_{NK}$.

At the sensor node, each video frame F_x is first divided into a small block of size 16×16, where x represents the frame number and F_0 is equivalent to F_K. Each block within a frame will then be sampled with respect to the sensing matrix Φ_x as presented in Eq. (5.1) to produce a set of measurements (Y_x) as defined in Eq. (5.2).

$$\Phi_x = \begin{bmatrix} \Phi_x & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \Phi_x \end{bmatrix} \qquad \textbf{(5.1)}$$

$$Y_x = \Phi_x F_x \qquad \textbf{(5.2)}$$

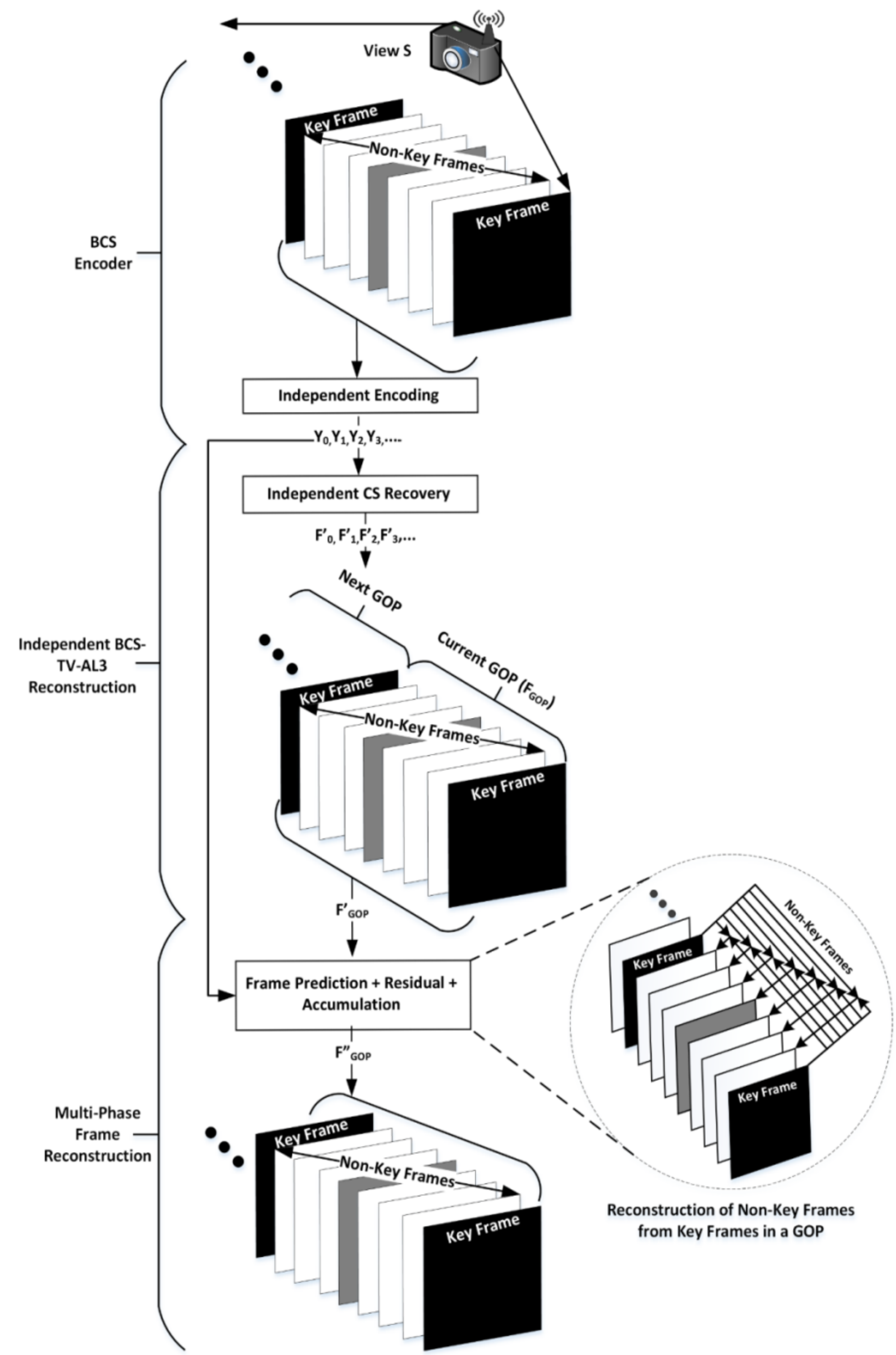


Figure 5.1: System architecture for single-view video compression and reconstruction using the proposed JMD

The frames are encoded and transmitted independently. The measurements ($\mathbf{Y_x}$) of each frame received by the server are first decoded independently in a frame by frame manner by solving the TV minimization [90] problem as given in Eq. (5.4) till a complete GoP is obtained.

$$\mathrm{TV(F)} = \sum_{i,j} \left| F_{i+1,j} - F_{i,j} \right| + \left| F_{i,j+1} - F_{i,j} \right| \quad \textbf{(5.3)}$$

$$F = \mathrm{argmin}_F \, |y - \Theta F|_{\ell_2} + \lambda \, \mathrm{TV(F)} \quad \textbf{(5.4)}$$

Once the complete GoP is obtained, the proposed JMD is then applied to the GoP to exploit the spatial and temporal redundancies among the frames. As illustrated in Figure 5.1, the first frame (key frame) of the current and next GoP serve as the reference frames for the JMD to generate some predicted and residual frames for improving the quality of (**J-1**) non-key frames $\mathbf{F_{NK}}$ of the current GoP.

5.2. Modified Joint Multi-phase Decoding for Video Compression

In this section, the decoding process of the proposed JMD is discussed in detail. Generally, the proposed JMD can be divided into three major phases as shown in Figure 5.2. The detail explanation of each phase is provided in the following subsections.

The proposed JMD for video compression is different from the image compression version in the registration and fusion process. The main differences are highlighted as follows.

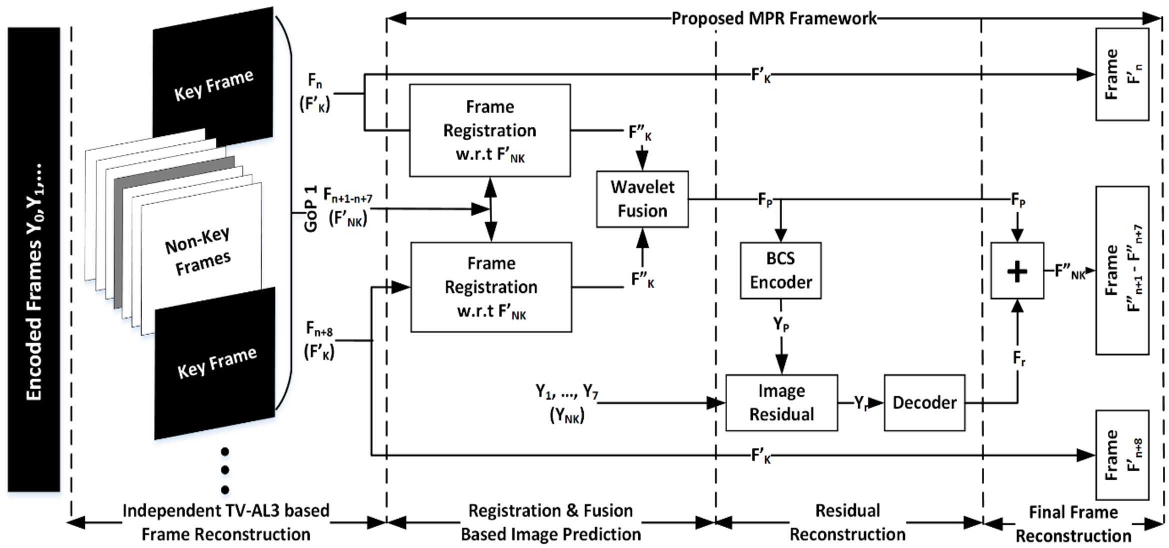


Figure 5.2: Proposed Joint Multi-phase Decoding (JMD)

i. <u>Registration Approach</u>

- **<u>Transformation</u>**: In this approach rather than using affine transform to align the frames simple translation transform is used. The reason is that affine transform makes sense in the case when multiple images are not on the same plane and are to be rectified. Whereas, in the case of video sequence each frame is usually on the same plane so it would be sufficient to consider translation transform.
- **<u>Optimization</u>**: Evolutionary optimizer is used instead of gradient descent optimizer because frames in the video sequence usually have the similar orientation that better facilitates the OE optimizer than the GD. Unlike, GD the OE optimizer iterates to find a set of parameters that produce the best possible registration result (ability to step out of from non-optimal minima to maxima) rather than adjusting the transformation parameters in the direction of the extrema. OE optimizer does this by disturbing, or modifying, the parameters from the last iteration (the parent). If the

new (child) parameters yield a better result, then the child becomes the new parent whose parameters are disturbed aggressively, else the parent parameters are continued to distribute but less aggressive. The optimizer is considered to be one of the important parameters of registration that states the methodology to maximize the achieved similarity metric M to produce a final output.

ii. <u>Fusion Approach</u>

- **Fusion**: In image-based prediction, the max and mean operations were used for approximation and detail coefficient respectively, whereas in frame based prediction the mean operation for both approximation and detail coefficient is used. The reason is that in frame based prediction the approximation coefficient among the frames within the video sequence are not significantly improved from each other. While in image-based prediction the approximation coefficients of the neighboring can produce better approximation than each other.

5.2.1. Phase 1- Frame Prediction

In this phase, a frame prediction method based on registration and fusion approaches is proposed. The aim is to predict the J-1 number of non-key frame (F'_{NK}) within the GoP from the key frames (F'_{K}) by exploiting the correlations in them. The proposed method helps to exploit the spatial and temporal correlations among the frames and generate a set of predicted non-key frames. However, the proposed frame prediction method differs from image prediction method in terms of registration and fusion approaches as discussed above.

The proposed prediction method is initialized by performing intensity based registration on the two independently reconstructed key frames F'_K within the GoP. The registration process projects the F'_K onto the same plane of F'_{NK} i.e. aligning F'_K to F'_{NK} and exploiting the temporal correlation among them. An initial transformation matrix between the F'_K and F'_{NK} frames is first calculated by using phase correlation that helps to find the gross alignment. Next, F'_K is aligned w.r.t. $F'_{NK}$by using translation transformation to produce transformed F'_K that is called F''_{KT}. Then, the transformed frame F''_{KT} is passed through a similarity metric and optimization function to estimate the registration accuracy and produce the final registered image F''_K as shown in Eq. (5.5).

$$\mathbf{F''_K} = \operatorname{argmin}_N \left\| \sum_{y \in L_{F'_{KT}}} \sum_{x \in L_{F'_{NK}}} P_{F'_{KT} F'_{NK}}(y, x; a) \log_2 \left(\frac{P_{F'_{KT} F'_{NK}}(y, x; a)}{P_{F'_{KT}}(x) P_{F'_{NK}}(y; a)} \right) \right\| \quad \mathbf{(5.5)}$$

The mutual information and one evolutionary are used in the similarity metric and optimizer respectively.

Once both the key frames F''_K are registered, a wavelet based fusion process is applied on them to produce the predicted frame F_P. The fusion process decomposes the registered key frames F''_K into their respective approximation (A) and detail (D) coefficients maps with three levels of decomposition using symlet 4-tap filter. The A and D coefficients in the two decomposition maps are then fused together using point-to-point operations. We empirically set the A and D coefficients as shown in Eq. (5.6) and Eq. (5.7) respectively.

$$A_{Mean}(x,y)= (A_K\ (x,y)+A_{NK}\ (x,y))\ /\ 2 \quad \textbf{(5.6)}$$

$$D_{Mean}(x,y)= (D_K\ (x,y)+D_{NK}\ (x,y))\ /\ 2 \quad \textbf{(5.7)}$$

For each A and D coefficients from the same coordinate of the two decomposition maps, the average magnitudes are computed. The average value of the both the A and D coefficients then serves as the output in the fused map. After fusing all the A and D coefficients from the two decomposition maps, inverse transformation are applied to the fused map to reconstruct the predicted frames F_P. The proposed frame prediction method estimates the object motions and creates predicted frames F_P.

5.2.2. Phase 2- Residual Reconstruction

After the predicted frames F_P is generated, the projection of F_P onto the measurement basis $Y_P= \Phi_x I_P$ is performed. Then, the difference between the given measurements Y_x and Y_P is determined as expressed in Eq. (5.8) and the output is known as the residual measurement Y_r.

$$Y_r = Y_x - Y_P \quad \textbf{(5.8)}$$

To obtain the residual frames $\mathbf{F_r}$, the residual measurements are then decoded by solving Eq. (5.4) using BCS-TV-AL3 reconstruction.

5.2.3. Phase 3 - Final Frame Reconstruction

In order to produce the final reconstructed frames $\mathbf{F''_{NK}}$ within the GoP, the $\mathbf{F_r}$ and $\mathbf{F_P}$ are added together. It is a normal point-to-point addition that is expressed in Eq. (5.9). By doing so, uniformity in terms of frame measurements (**Y**) is achieved i.e. the measurements computed for $\mathbf{F''_{NK}}$ is to some extent equal to the measurements $\mathbf{Y_{NK}}$.

$$F''_{NK} = F_r + F_P \qquad \textbf{(5.9)}$$

After the key frames, F'_K (F_0 and F_J from Y_0 and Y_J) are reconstructed using BCS-TV-AL3, they are used as the reference frames for the reconstruction of the non-key frames F'_{NK} between them.

The proposed scheme produces the non-key frame F''_1 from Y_1, $F_{0,}$ and F_J in the same way as F''_2 are produced from Y_2, F_0 and F_J. The process continues for all the remaining non-key frames. We expect the reconstruction quality to drop when reconstructing non-key frames that are far from the key frames. Hence, the reconstruction quality may deteriorate more as the GoP size (J) increases.

5.2.4. Proposed SQ-ADPCM Quantization Framework

In order to evaluate the proposed schemes based on bitrate instead of subrate, the CS measurements must be quantized. From our review, it is observed that applying Scalar Quantization (SQ) directly to CS measurements is highly inefficient in terms of rate-distortion performance when compared with traditional coding schemes as discussed in [104]. Many [105-110] have focused on improving the rate-distortion performance of and it is still an open research problem. Most of the efforts are based on either quantizer optimization or the reconstruction, or both. While, few of the work is based on combining the simple uniform SQ with differential modulation.

In this context, we proposed to use uniform SQ with Adaptive Differential Pulse Code Modulation (ADPCM) for quantizing the CS measurements and it is referred to as SQ-ADPCM. The ADPCM is a variant of Differential Pulse-Code Modulation (DPCM) that can vary the size of the quantization step, to allow further reduction of the required bandwidth for a given signal-to-noise ratio and provides greater levels of prediction gain than simple DPCM.

The proposed SQ-ADPCM framework is applied to BCS. At the encoder side, a SQ is applied on the residual measurements rather than directly on each block of CS measurements. The residual measurements are achieved by subtracting the current block from the predicted block in the measurement domain. It should be noted that unlike DPCM, ADPCM is based on adaptive prediction approach that results in better prediction levels. At the reconstruction side of the system, the same prediction is added onto the de-quantized residuals to produce the CS measurements ready for BCS-based reconstruction. The advantage of such quantization scheme is that the CS encoder and decoder operations need not to be modified accordingly and results in better reconstruction quality. The complete architecture of proposed quantization with BCS is shown in Figure 5.3.

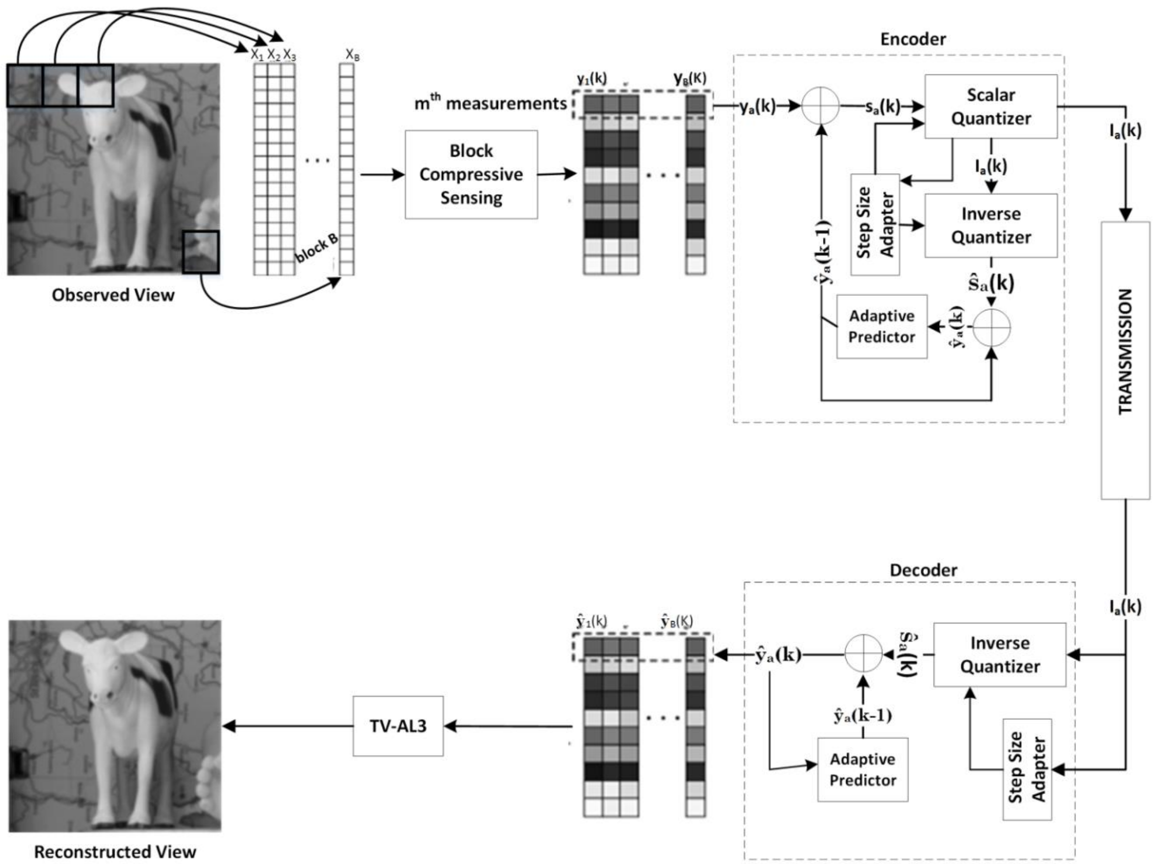


Figure 5.3: Complete architecture of the proposed SQ-ADPCM with BCS

On the encoder side, the BCS measurements are acquired using B ×B blocks from the original image, producing M-dimensional measurement vector for block k of the image, x(k) as shown in Eq. (5.10).

$$y(k) = [y_1(k)\ \ y_2(k) \ldots y_a(k) \ldots y_{Mn}(k)]^T = \Phi_B\, x(k) \qquad \textbf{(5.10)}$$

Let us consider a measurement vector $y_a(k)$, a residual $s_a(k)$is achieved by subtracting the prediction measurements from $y_a(k)$. The prediction measurements of $y_a(k)$ are generated by using the previously processed block $\hat{y}_a(k-1)$ of the corresponding vector. The residual is given as in Eq. (5.11).

$$s_a(k) = y_a(k) - \hat{y}_a(k-1) \qquad \textbf{(5.11)}$$

The achieved residual measurements are then scalar-quantized to produce quantization index $i_a(k)$. The encoder not only transmits the quantization index $i_a(k)$ to the decoder but also uses it as an input for the ADPCM feedback loop. The feedback loop first de-quantize the $i_a(k)$, producing the quantized residual $\hat{s}_a(k)$. Finally, the prediction is implemented with an adaptive predictor and is given as:

$$\hat{y}_a(k) = \hat{s}_a(k) + \hat{y}_a(k-1) \qquad \textbf{(5.12)}$$

It should be noted that the set of measurements in the first block is processed in the same manner and the predictor and quantizer step-size are initialize to zero.

The decoding process is the inverse of the encoding process. It uses the ADPCM value to update the inverse quantizer, which produces a difference $\hat{s}_a(k)$. The difference is then added to the predicted $\hat{y}_a(k-1)$to produce the output measurement vector $y_a(k)$. Once the measurements are obtained the BSC based proposed JMD is applied to reconstruct the final image.

The proposed method not only helps to reduce the amount of bits needed to represent the image but also shows significant reconstruction improvements (1dB-2dB) when compared with independent SQ and SQ-DPCM using various video sequences as presented in Table 5.1. Detail experimental analysis of the proposed SQ-ADPCM framework is presented in section 5.3.5.

Table 5-1: PSNR performance in dB at various bitrates for different video sequences

			Hall Monitor			
Subrate	**0.1**	**0.2**	**0.3**	**0.4**	**0.5**	**0.6**
SQ [104]	19.20	20.11	21.52	22.03	23.24	23.86
SQ-DPCM [110]	20.75	21.91	23.10	24.03	25.05	25.41
SQ-ADPCM	**21.98**	**23.01**	**24.40**	**25.36**	**26.25**	**26.70**
			News			
Bit rate	**0.1**	**0.2**	**0.3**	**0.4**	**0.5**	**0.6**
SQ [104]	19.85	20.55	20.99	21.84	23.02	23.96
SQ-DPCM [110]	21.06	21.72	22.77	23.95	24.98	25.64
SQ-ADPCM	**22.78**	**23.35**	**24.26**	**25.10**	**26.41**	**27.88**
			Mobile			
Subrate	**0.1**	**0.2**	**0.3**	**0.4**	**0.5**	**0.6**
SQ [104]	16.64	17.02	17.52	18.32	18.78	19.39
SQ-DPCM [110]	18.20	19.35	20.01	21.11	21.99	22.21
SQ-ADPCM	**19.66**	**20.86**	**21.65**	**22.50**	**23.35**	**23.95**

5.3. Experimental Results

In the following, the evaluation of the proposed scheme coupled with TV-AL3 referred to as JMD-TV is presented. It is applied to a set of standard grayscale CIF [128] video sequences with a frame size of 352×288 to evaluate its performance. The selected video sequences involve slow to fast motion contents. Table 5.2 present the list of video sequences used in the evaluation of proposed scheme along with the details of the sequence in terms of no. of frames and content type.

The video sequence with low, medium and high contents have low, medium and high spatial details as well as slow, medium and fast camera and object movement, respectively. The experimental setup involves the implementation of the proposed JMD-TV with different GoP (J) sizes i.e. 3, 5, and 8. The purpose is to evaluate the performance at different variations. The evaluation is carried out by recording the Peak Signal to Noise Ratio (PSNR) at different sampling rate (subrate). Additionally, we also performed evaluation based on Structural Similarity Index (SSIM) which is considered more accurate and consistence with human visual perception than PSNR. Due to the random Φ, the image quality may vary. Hence, all PSNR and SSIM values represent an average of 5 independent trials.

A block size of **16x16** rather than 32x32 and 64x64 is adopted, because smaller block size leads to less memory usage as discussed in chapter 4. All the non-key frames within a GoP are encoded at lower subrates (0.05, 0.1, 0.15, 0.2, 0.25, 0.3) with the key frames encoded at a fixed subrate of 0.5. Before evaluating the proposed scheme in detail, a correlation estimation of the CS measurements among the adjacent frames is provided.

Table 5-2: Several standard grayscale CIF & HD Video Sequences

	Video Sequence	No. Frames	Content Type
CIF Video Sequence (Size 352×288)	**Hall Monitor**	300	Low
	Mother Daughter	300	Low
	Coast Guard	300	Medium
	Foreman	300	Medium
	Mobile Calendar	300	High
	Stefan	300	High

5.3.1. Relationship between Proposed JMD -TV and GoP

In this subsection, the effect of GoP on the Rate-Distortion (R-D) performance of proposed JMD-TV is evaluated and compared with independent BCS-TV-AL3. Table

5.3, 5.4 and 5.5 present the R-D (dB) results of proposed JMD with three different GoP sizes i.e. 3, 5, and 8 at various subrates for various video sequences.

Table 5-3: Average R-D (dB) performance achieved by using the conventional BCS-TV-AL3 and the proposed scheme to encode various low motion content video sequences at different GoP=3,5,8

Low Motion Content Video Sequence	**Mother Daughter**					
	Subrate	**0.05**	**0.1**	**0.15**	**0.2**	**0.25**	**0.3**
	BCS-TV	24.88	30.36	32.35	33.73	35.55	37.15
	MPR-TV GoP3	**36.35**	**40.10**	**41.03**	**41.41**	**41.94**	**42.41**
	MPR-TV GoP5	34.16	35.55	36.73	38.43	39.88	40.72
	MPR-TV GoP8	32.00	34.26	35.81	37.65	38.63	39.61
	Hall Monitor						
	Subrate	**0.05**	**0.1**	**0.15**	**0.2**	**0.25**	**0.3**
	BCS-TV	21.20	23.71	25.26	26.68	28.18	29.71
	MPR-TV GoP3	**32.31**	**32.70**	**33.28**	**33.93**	**34.45**	**35.08**
	MPR-TV GoP5	29.52	31.56	32.10	32.98	33.64	34.26
	MPR-TV GoP8	28.98	30.65	31.77	32.81	33.83	34.85

***Note*:** The bold values relates to the maximum PSNR (dB) reached for a given subrate and video sequence

Table 5-4: Average R-D (dB) performance achieved by using the conventional BCS-TV-AL3 and the proposed scheme to encode various medium motion content video sequences at different GoP=3,5,8

Medium Motion Content Video Sequence	**Forman**						
	Subrate	**0.05**	**0.1**	**0.15**	**0.2**	**0.25**	**0.3**
	BCS-TV	22.75	25.27	26.94	28.79	30.89	32.62
	MPR-TV GoP3	**28.71**	**30.83**	**31.20**	**32.48**	**33.03**	**34.70**
	MPR-TV GoP5	**25.19**	**28.79**	**29.95**	**31.73**	**32.48**	**33.92**
	MPR-TV GoP8	**25.18**	**26.75**	**28.94**	**30.31**	**31.85**	**33.50**
	Coast Guard						
	Subrate	**0.05**	**0.1**	**0.15**	**0.2**	**0.25**	**0.3**
	BCS-TV	20.75	22.87	23.90	25.04	26.07	26.88
	MPR-TV GoP3	**28.07**	**28.62**	**29.11**	**29.79**	**30.40**	**30.89**
	MPR-TV GoP5	**24.32**	**25.73**	**26.40**	**27.49**	**28.16**	**28.91**
	MPR-TV GoP8	**23.42**	**24.50**	**25.58**	**26.49**	**27.27**	**28.03**

***Note*:** The bold values relates to the maximum PSNR (dB) reached for a given subrate and video sequence

Table 5-5: Average R-D (dB) performance achieved by using the conventional BCS-TV-AL3 and the proposed scheme to encode various high motion content video sequences at different GoP=3,5,8

High Motion Content Video Sequence	Mobile Calendar						
	Subrate	**0.05**	**0.1**	**0.15**	**0.2**	**0.25**	**0.3**
	BCS-TV	16.68	18.60	19.58	20.66	21.60	22.50
	MPR-TV GoP3	**23.43**	**24.27**	**24.71**	**25.38**	**26.09**	**26.65**
	MPR-TV GoP5	**21.35**	**22.38**	**23.15**	**23.84**	**24.60**	**25.28**
	MPR-TV GoP8	**20.15**	**21.28**	**22.24**	**22.97**	**23.79**	**24.65**
	Stefan						
	Subrate	**0.05**	**0.1**	**0.15**	**0.2**	**0.25**	**0.3**
	BCS-TV	18.81	20.65	21.78	23.03	24.12	25.31
	MPR-TV GoP3	**23.19**	**25.95**	**26.02**	**27.60**	**28.42**	**29.33**
	MPR-TV GoP5	**22.57**	**24.23**	**25.48**	**26.76**	**27.41**	**28.65**
	MPR-TV GoP8	**22.28**	**23.97**	**25.16**	**25.96**	**26.82**	**28.03**

Note**:** The bold values relates to the maximum PSNR (dB) reached for a given subrate and video sequence

The R-D (dB) results presented are averaged over only the non-key frames attained for each complete video sequence. It can be seen that the proposed scheme shows a notable gain over independent BCS-TV-AL3 for all the video sequences. For low-motion videos, the gain on average is 3.5dB- 7dB higher than the independent BCS-TV-AL3 for all GoP sizes. Whereas, for medium and high-motion videos the gain on average is 3dB-5dB and 2dB-4dB respectively for all GoP sizes. The low-motion videos reconstruction gain is better than medium and high-motion videos as it shows higher correlation measurements among the frames than medium and high-motion videos. The higher correlation measurements result in more accurate frame prediction and residual reconstruction.

It should also be noted that as the GoP size increases the gain in terms of PSNR decreases. This is because the proposed scheme makes use of key frames to reconstruct the non-key frames. Thus, the non-key frames sitting nearer to the key frame have a higher degree of correlation than those further away. For smaller GoP size (J=3) the proposed scheme provides an average gain of 3dB to 6dB, whereas for larger GoP size (J=8) the average gain is of ~2dB to 4dB. Furthermore, the performance gain decreases when the subrate increases. As mentioned earlier, F_K are the key frames that are transmitted at a higher subrate than that of F_{NK}. Hence, F_K produces a larger set of measurements, which superimposes the correlated smaller set of measurements encompasses by F_{NK}. This in result reduces the prediction errors of F_{NK} that occurs due to smaller set measurements and produce an improved version of F_{NK}.

We have also tested the visual quality using SSIM metric. The SSIM curves of six different video sequences for three different GoP sizes at various subrates are shown in Figure 5.4. The selected video sequence represents all the three motion content types. The graph clearly shows improvement in visual quality and significant gain of proposed framework over the independent framework for all GoP sizes at various subrates. A similar trend as of PSNR can be observed for SSIM metric.

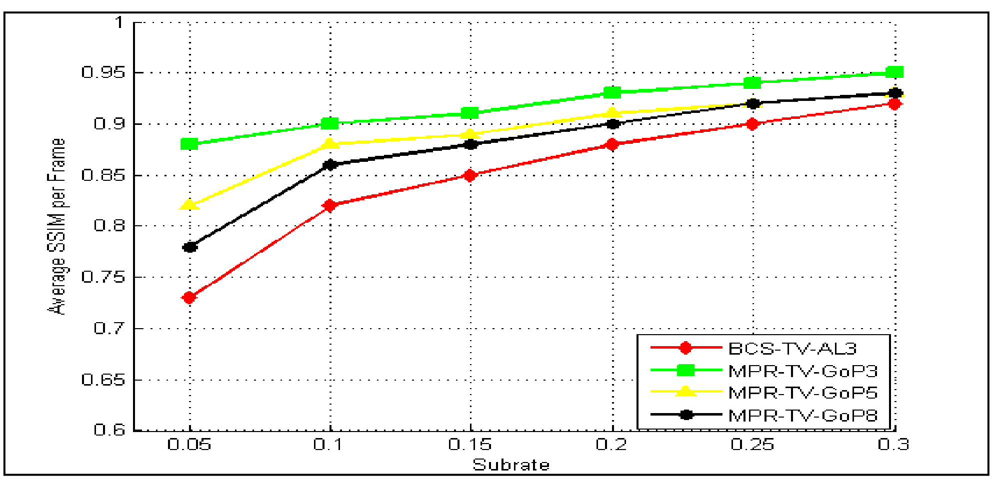


Figure 5.4a. Forman Video Sequence

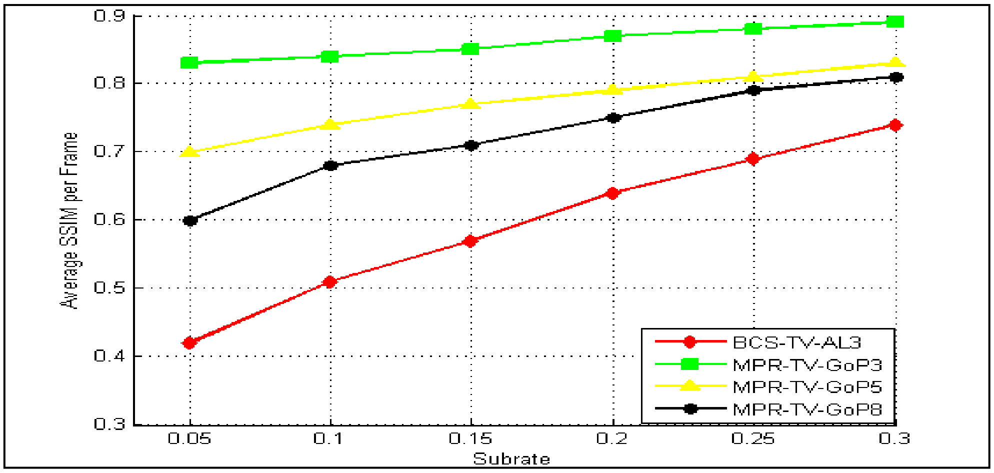


Figure 5.4b. Coastguard Video Sequence

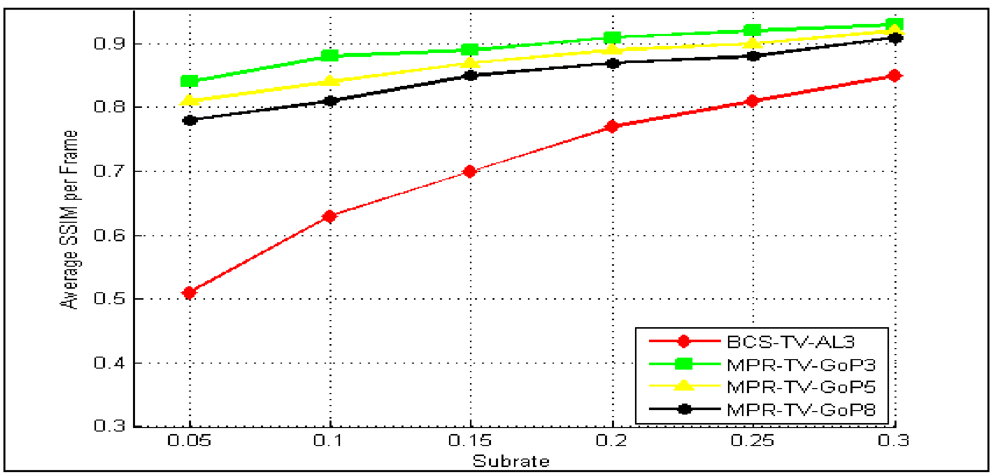


Figure 5.4c. Stefan Video Sequence

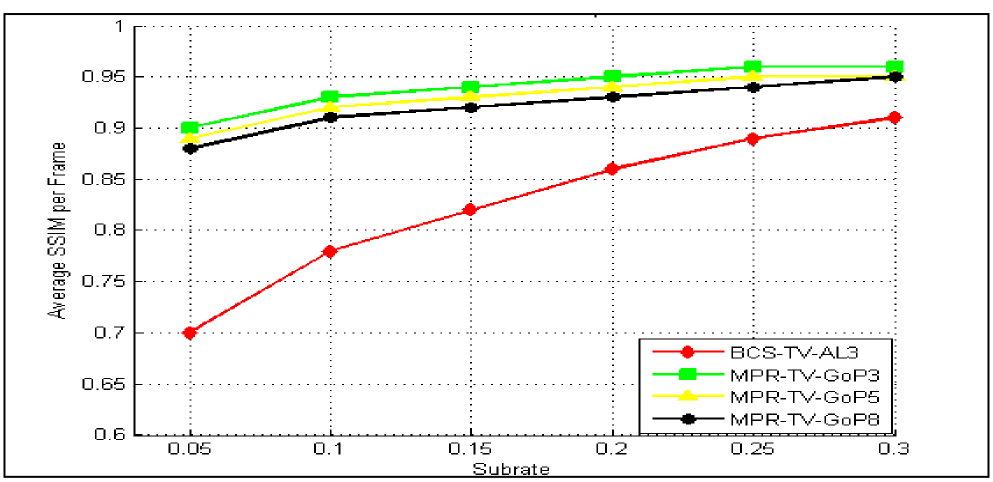


Figure 5.4d. Hall Monitor Video Sequence

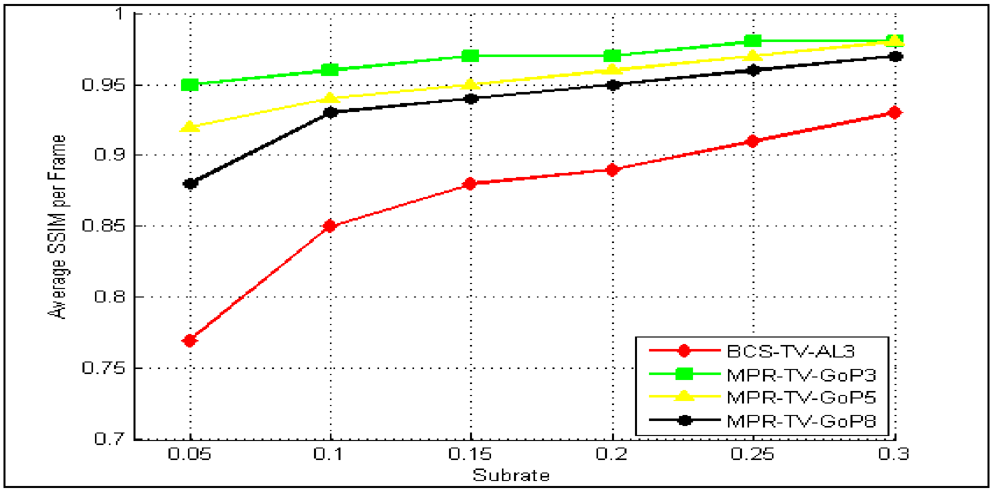


Figure 5.4e. Mother-Daughter Video Sequence

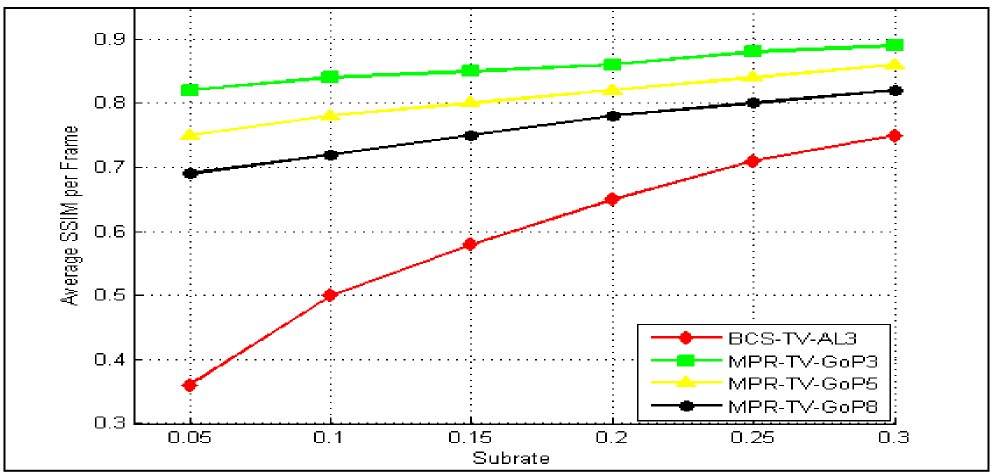


Figure 5.4f. Mobile Calendar Video Sequence

Figure 5.4: Average SSIM comparison of various video sequences with GoP sizes 3, 5, and 8 at various subrates using conventional BCS-TV-AL3and proposed JMD.

From the above evaluation, we also observed that the GoP size 3 and 5 provides better reconstruction gains than GoP size 8. Additionally, we tested the proposed scheme with GoP size 16, but the reconstruction gains were not significant enough to be reflected. Considering the case of VSN it's not feasible to transmit the key frames frequently as in the case with GoP size 3 and 5; it will increase the computational burden at the encoder.

Thus, GoP size 8 is considered as a more balance point among all the GoP sizes and will opt in later experiments.

5.3.2. Subrate

In this subsection, the effect of key frames (F_K) subrate on the reconstruction of non-key frames (F_{NK}) using the proposed JMD-TV is evaluated. From the previous evaluation, GoP size 8 is considered as a balance point among all the GoP sizes and thus opted in this experiment. Two different setups are used for evaluation. In the first setup, both F_K and F_{NK} are transmitted at the same subrate i.e. M_K=M_{NK}. In the second setup, F_K is transmitted at a fixed subrate of M_R=0.5 and F_{NK} are transmitted at different subrates that range from M_{NK}=0.05 to 0.3 with an interval of 0.05 between each subrate.

From the simulation results presented in Table 5.6, noticed that the F_K subrate has a greater effect on the reconstruction of F_{NK} such that in the first setup (M_K =M_{NK}) the gain is lower than that of the second setup. When F_K and F_{NK} are transmitted at the same rate, the reconstructed F_K does not contain information that could significantly help the reconstruction of F_{NK}.

Table 5-6: R-D (dB) performance comparison of the proposed JMD and conventional BCS-TV-AL3 with different subrate setups for various video sequences at GoP=8

	Hall Monitor						
Low Motion Content Video	**Subrate**	**0.05**	**0.1**	**0.15**	**0.2**	**0.25**	**0.3**
	BCS-TV-AL3	21.70	23.53	24.76	26.44	28.16	29.77
	JMD-TV (M_R=M_{NR})	22.88	25.54	27.37	29.34	30.84	32.62
	JMD-TV (M_R= 0.5)	**28.98**	**30.65**	**31.77**	**32.81**	**33.83**	**34.85**
	Mother Daughter						
	Subrate	**0.05**	**0.1**	**0.15**	**0.2**	**0.25**	**0.3**
	BCS-TV-AL3	24.88	30.36	32.19	33.73	35.55	37.15
	JMD-TV (M_R=M_{NR})	29.02	32.15	33.81	35.57	37.83	39.15
	JMD-TV (M_R= 0.5)	**32.00**	**34.26**	**35.81**	**37.65**	**38.63**	**39.61**

	Coast guard						
	Subrate	**0.05**	**0.1**	**0.15**	**0.2**	**0.25**	**0.3**
Medium Motion Content Video	**BCS-TV-AL3**	20.96	22.65	23.87	24.87	25.96	26.77
	JMD-TV (M_R=M_{NR})	21.21	23.79	25.17	26.41	27.67	28.58
	JMD-TV (M_R= 0.5)	**23.42**	**24.50**	**25.58**	**26.49**	**27.27**	**28.03**
	Forman						
	Subrate	**0.05**	**0.1**	**0.15**	**0.2**	**0.25**	**0.3**
	BCS-TV-AL3	22.75	25.14	26.79	28.66	30.89	32.62
	JMD-TV (M_R=M_{NR})	23.92	26.39	28.10	29.84	31.35	33.35
	JMD-TV (M_R= 0.5)	**25.18**	**26.75**	**28.94**	**30.31**	**31.85**	**33.50**
	Mobile Calendar						
High Motion Content Video	**Subrate**	**0.05**	**0.1**	**0.15**	**0.2**	**0.25**	**0.3**
	BCS-TV-AL3	16.68	18.58	19.58	20.52	21.58	22.50
	JMD-TV (M_R=M_{NR})	17.99	19.43	20.65	21.80	22.88	24.03
	JMD-TV (M_R= 0.5)	**20.15**	**21.28**	**22.24**	**22.97**	**23.79**	**24.65**
	Stefan						
	Subrate	**0.05**	**0.1**	**0.15**	**0.2**	**0.25**	**0.3**
	BCS-TV-AL3	18.81	20.65	21.78	23.03	24.12	25.31
	JMD-TV (M_R=M_{NR})	19.23	21.65	23.25	24.59	26.12	27.50
	JMD-TV (M_R= 0.5)	**22.28**	**23.97**	**25.16**	**25.96**	**26.82**	**28.03**

Note: The bold values relates to the maximum PSNR (dB) reached for a given subrate and video sequence

5.3.3. Visual Result Comparison

Since JMD-TV performs better at lower subrate, it is important to ensure that the subrate used is sufficient to produce visually recognizable frame. Two different video sequences are selected that represents medium and high motion contents. The results shown in Figure 5.5 are of the center frame of each GoP reconstructed by using JMD-TV and BCS-TV-AL3 at different subrate of 0.05, 0.1 and 0.2. It should be reminded that the key frames $\mathbf{F_K}$ used in the reconstruction of $\mathbf{F_{NK}}$ are reconstructed at a subrate of 0.5.

By comparing the visual results presented in Fig. 5.5(a, b), it can be noticed that the frame reconstructed by using the proposed JMD-TV improves the blurring effect present in the frame reconstructed using BSC-TV-AL3. Moreover, by comparing the highlighted regions (White dotted boxes), it can be noticed that frame reconstructed using JMD-TV looks much sharper than BCS-TV-AL3.

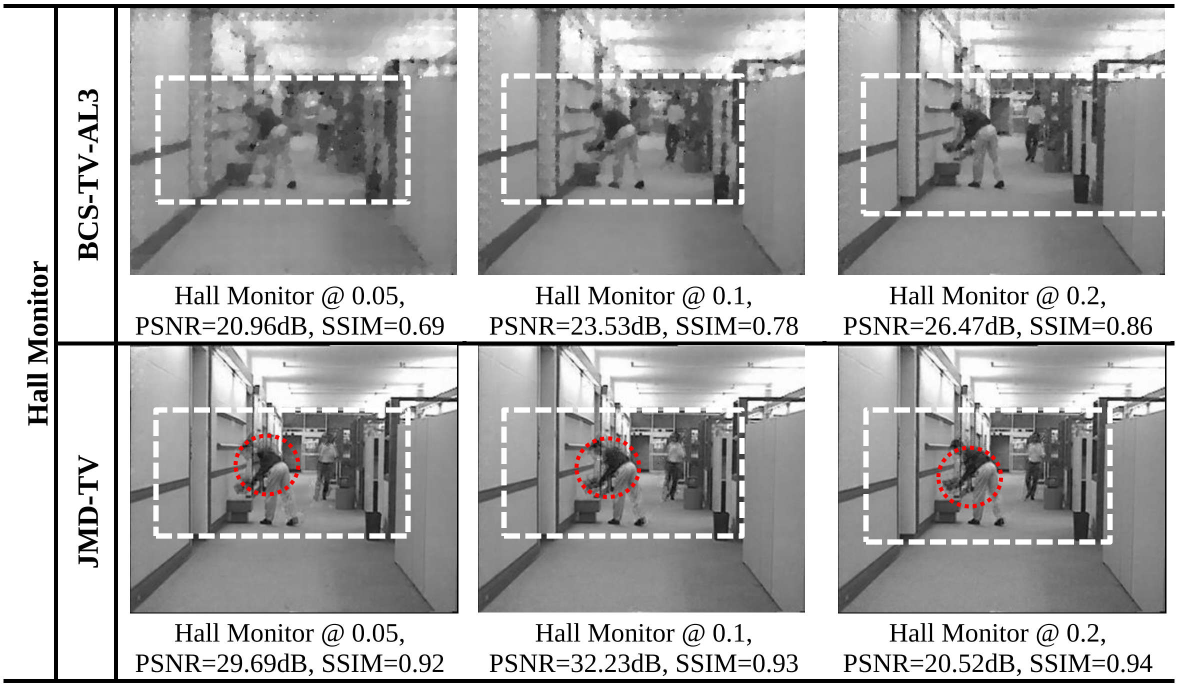


Figure 5.5a. Reconstruction of frame# 105th of Hall Monitor video at different subrates

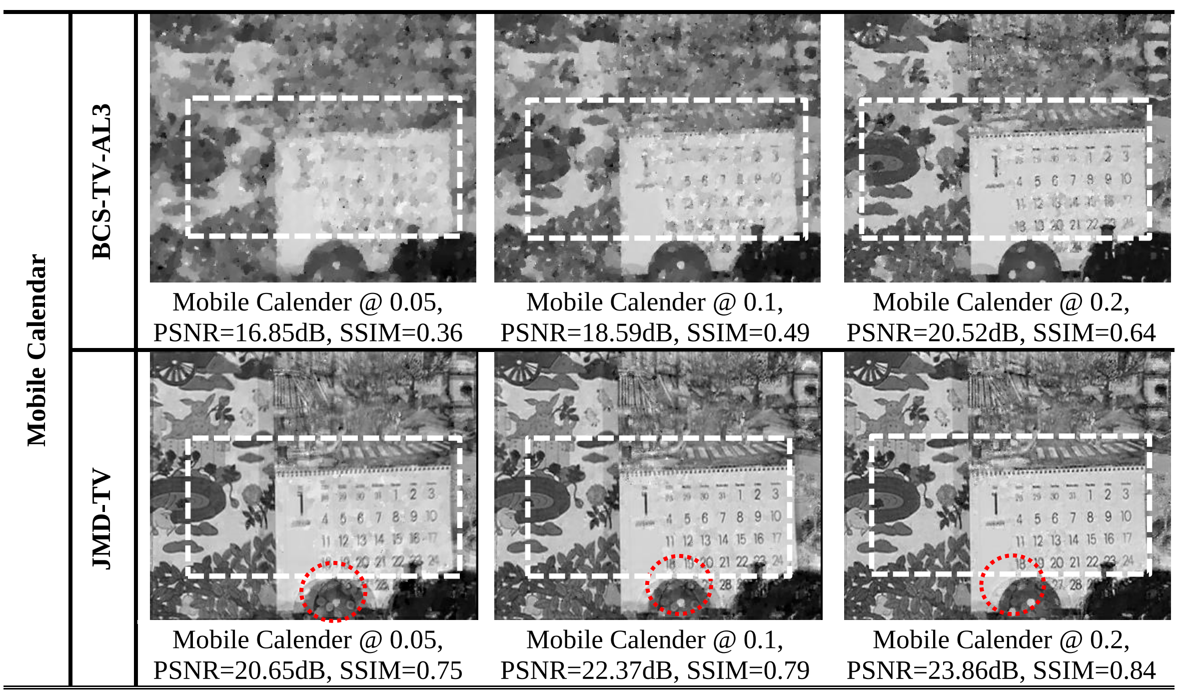


Figure 5.5b. Reconstruction of frame# 65th of Mobile Calendar video at different subrates

Figure 5.5: Visual quality comparison for the reconstruction of various video sequences at various subrates using independent BCS-TV-AL3 and proposed JMD-TV

The performance of the proposed JMD-TV is higher for medium motion video contents (Hall Monitor) due to more accurate frame prediction and residual reconstruction. However, at lower subrate (0.05) some noise as highlighted by the red dotted circle is observed due to inadequate prediction of motion. Similarly, for the video containing fast moving objects (Mobile Calendar), the JMD-TV is exposed to certain noise as highlighted by the red dotted circle.

5.3.4. Comparison of Proposed JMD-TV with other CS Video Compression Schemes

In this section, the proposed JMD-TV is compared with conventional CS video compression that we discussed in section 2.5 of Chapter 2. This includes MS-Residual [73], k-t FOCUSS [74], and MC-BCS-SPL [81].

All the simulation results that we obtained for the first 100 frames are summarized in Table 5.7, presented in terms of gain i.e. proposed final reconstruction over independent reconstruction.

The results for MS-Residual, k-t FOCUSS, and MC-BCS-SPL were obtained after modifying their available code [125, 126, 129] respectively, with respect to the experimental setup described in Section 4. The block size is 16x16, and the GoP size is 8. From the simulation results, it can be seen that the proposed JMD provides substantial gain at lower subrates when compared with MS-Residual, k-t FOCUSS and MC-BCS-SPL for various type of video.

Table 5-7: R-D (dB) performance gain (dB) comparison of the proposed JMD with MS-Residual [73], k-t FOCUSS [74], and MC-BCS-SPL [81] for various video sequences

			Hall Monitor			
Subrate	**0.05**	**0.1**	**0.15**	**0.2**	**0.25**	**0.3**
JMD-TV	**6.88**	**6.75**	**6.68**	**6.51**	**6.47**	**6.33**
MC-BCS-SPL	2.64	3.88	4.96	5.77	5.98	6.21
kt-Focuss	1.50	1.96	2.45	3.21	3.97	4.05
MS-Residual	0.85	1.06	1.55	2.17	2.88	3.35
			Mother Daughter			
Subrate	**0.05**	**0.1**	**0.15**	**0.2**	**0.25**	**0.3**
JMD-TV	**5.96**	**5.83**	**5.69**	**5.57**	**5.41**	**5.27**
MC-BCS-SPL	2.17	3.04	**3.77**	4.30	4.85	5.14
kt-Focuss	1.08	1.87	2.65	3.34	3.99	4.75
MS-Residual	0.38	0.77	1.27	1.83	2.59	3.05
			Coast Guard			
Subrate	**0.05**	**0.1**	**0.15**	**0.2**	**0.25**	**0.3**
JMD-TV	**2.76**	**2.65**	**2.55**	**2.42**	**2.31**	**2.22**
MC-BCS-SPL	0.95	1.35	1.49	1.74	1.98	2.15
kt-Focuss	0.45	1.01	1.21	1.40	1.60	1.89
MS-Residual	0.25	0.54	0.75	0.95	1.20	1.49
			Forman			
Subrate	**0.05**	**0.1**	**0.15**	**0.2**	**0.25**	**0.3**
JMD-TV	**5.56**	**4.96**	**4.26**	**3.69**	**2.14**	**2.08**
MC-BCS-SPL	0.90	2.01	2.81	3.44	3.57	3.74
kt-Focuss	0.45	0.67	0.81	1.07	1.23	1.59
MS-Residual	0.15	0.24	0.55	0.77	0.95	1.19
			Mobile Calendar			
Subrate	**0.05**	**0.1**	**0.15**	**0.2**	**0.25**	**0.3**
JMD-TV	**3.73**	**3.59**	**3.47**	**3.32**	3.19	3.06
MC-BCS-SPL	0.96	1.82	2.15	2.95	**3.81**	**4.55**
kt-Focuss	0.66	1.02	1.89	2.52	3.12	3.86
MS-Residual	0.26	0.55	0. 95	1.52	2.02	2.78
			Stefan			
Subrate	**0.05**	**0.1**	**0.15**	**0.2**	**0.25**	**0.3**
JMD-TV	**4.17**	**4.04**	**4.01**	**3.96**	**3.87**	**3.73**
MC-BCS-SPL	0.24	0.44	0.82	1.35	2.07	2.8
kt-Focuss	0.12	0.30	0.56	0.99	1.42	1.95
MS-Residual	0.05	0.10	0.25	0.44	0.87	1.04

***Note*:** The bold values relates to the maximum PSNR (dB) gain reached for a given subrate and video sequence

5.3.5. Comparison of Proposed JMD-TV with Conventional Video Compression Schemes

In this section, the proposed JMD-TV is compared with state-of-the-art video compression schemes. CS is based on simple-encoder complex-decoder paradigm, which is the opposite of conventional video compression schemes. The comparison is to investigate the Rate-Distortion (R-D) performance of the proposed scheme against the conventional DISCOVER [38], H.264 [131] and H.263 [132] video reconstruction schemes. Two different coding selections are used for H.263 and H.264 in the experiment (i.e. H.263 (intra), H.263 (I-P-P) and H.264 (intra), H.264 (I-P-P), respectively). All the Rate-Distortion (R-D) performance (i.e., PSNR (dB) vs. Bitrate (bpp)) results that we obtained for the first 100 frames are presented in Figure 5.6. The GoP size of 3 and block size is 16x16 is selected for all implementations.

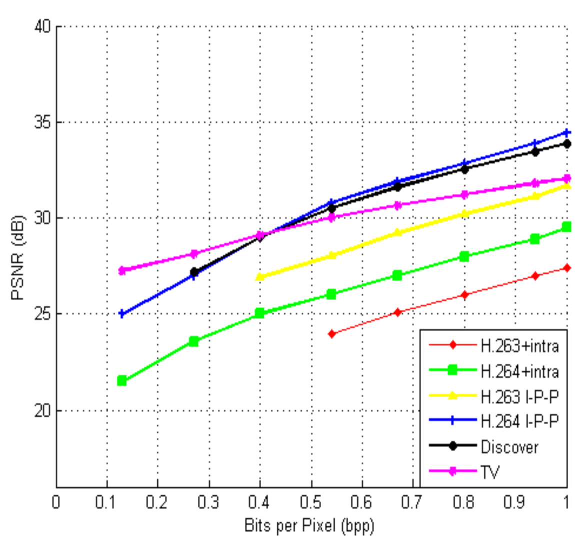


Fig. 5.6a. Coastguard

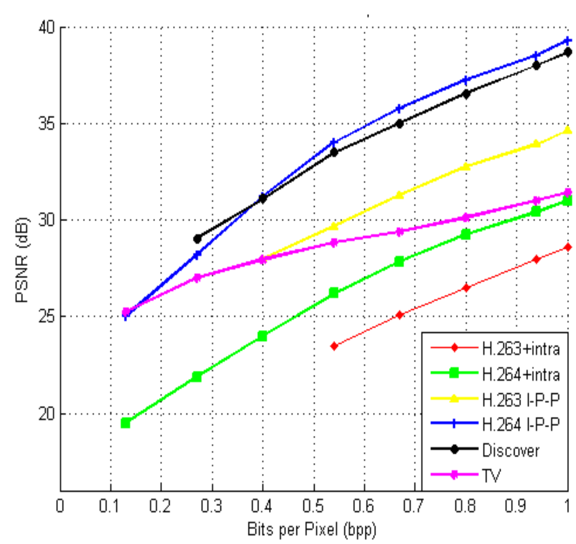


Fig. 5.6b. Hall Monitor

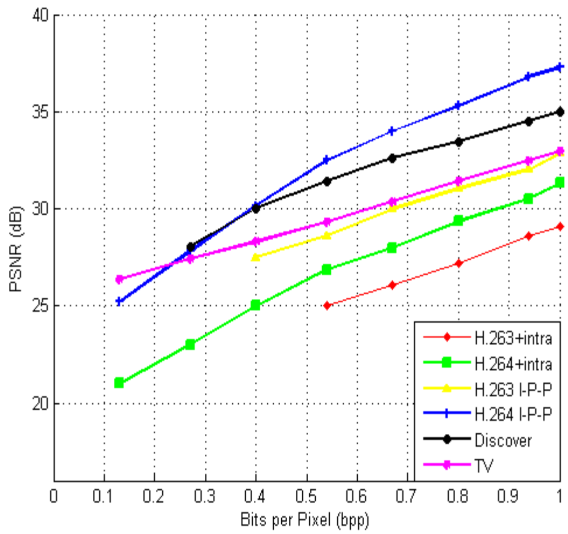


Fig. 5.6c. Foreman

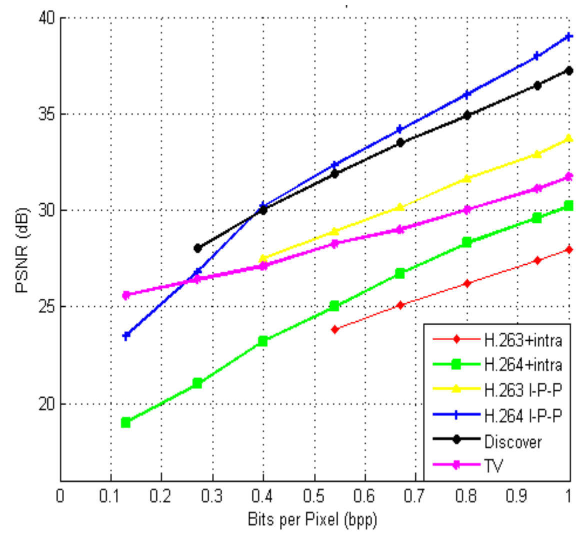


Fig. 5.6d. News

Figure 5.6: Bitrate vs PNSR for various video sequence at GoP = 3

From the simulation results, it is observed that the proposed scheme performs better than H.263 (intra) and H.264 (Intra) for all video sequences at various bitrates. For Foreman and Coastguard video sequence the performance of the proposed scheme is better than H.263 (I-P-P) at various bitrates. Whereas, at lower bitrates the performance of the proposed scheme is better than DISCOVER and H.264 (I-P-P). It should also be noted that both the DISCOVER and H.264 scheme uses feedback channel to improve the (WZ/key) frames, respectively. To the best of our knowledge, all the CS based video scheme reported till date in the literature performs noticeably lower than the conventional schemes (i.e. CS based image/video schemes are still in early development phase).

5.3.6. Number of Bits

In this subsection, we observe the bit rate savings between the proposed JMD-TV and the BCS-TV-AL3 scheme for various video sequences at the different reconstruction qualities (PSNR). The Table 5.8 shows the PSNR performance and bit saving of four different video sequences. From the results, it can be noticed that bits saving varies for different reconstruction qualities i.e. for higher reconstruction quality the saving rate is ~40%, whereas for lower reconstruction quality the saving rate is ~66% for all videos. This is because the proposed JMD provides better reconstruction quality at lower measurement rates. On average, the number of measurement saved by the proposed scheme against the independent scheme is ~50% for all video sequences at different reconstruction qualities.

Table 5-8: Coding performance comparison of the proposed JMD with BCS-TV-AL3 scheme in terms of Bit saved for various video sequences at different reconstruction quality (PSNR).

		Hall Monitor			
PSNR	**Bits BCS-TV-AL3**	**Bits JMD-TV**	**Bits Saved**	**Bits Saving(%)**	**Bits (Average%)**
~24.03	13068	1980	11088	84	
~25.05	17424	5148	12276	72	75
~25.57	20196	7128	13068	68	
		Coast Guard			
PSNR	**Bits BCS-TV-AL3**	**Bits JMD-TV**	**Bits Saved**	**Bits Saving(%)**	**Bits (Average%)**
~23.42	12276	5940	6336	52	
~24.02	14256	9108	5148	37	40
~24.53	17424	12276	5148	30	
		Mother Daughter			
PSNR	**Bits BCS-TV-AL3**	**Bits JMD-TV**	**Bits Saved**	**Bits Saving(%)**	**Bits (Average%)**
~29.37	9108	3960	5148	58	
~30.20	13068	7128	5940	45	49
~31.59	17424	9908	7516	43	
		Mobile Calendar			
PSNR	**Bits BCS-TV-AL3**	**Bits JMD-TV**	**Bits Saved**	**Bits Saving(%)**	**Bits (Average%)**
~18.32	9108	3168	5940	66	
~18.78	12276	5148	7128	58	58
~19.39	14256	7128	7128	50	

5.3.7. Execution Time:

In this subsection, the average reconstruction time of the proposed JMD-TV and other conventional CS scheme with GoP size 8 at different subrates for various video sequence is presented. All the schemes are implemented using MATLAB (R2014a) running on a computer with an Intel(R) Xeon(R) E5-1620 3.6 GHz CPU and 8GB RAM. We measured the average execution time (in second) required to reconstruct a single frame at various subrates.

The results in Figure 5.7 show that the average execution time of the proposed JMD-TV , MS-Residual [73], k-t FOCUSS [74], and MC-BCS-SPL [81] ranges from **6.39s – 12.06s, 24.78s – 30.01s, 63.54s – 69.51s** and **187s- 198s** respectively.

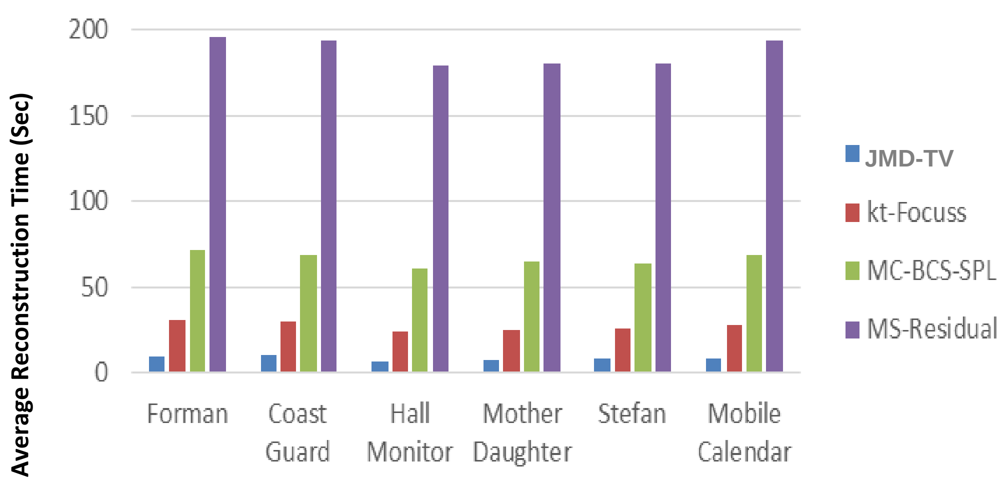


Figure 5.7: Average execution time (sec) comparison for various video sequences at GoP=8

At lower subrate, all the four schemes take a longer time to find a better reconstruction due to a small number of received measurements. Overall, the proposed

JMD-TV takes approximately **2-3 times** shorter interval than MC-BCS-SPL and kt-Focuss. Moreover, the proposed JMD-TV shows much better results over the MS-Residual. This is due to the less complex BCS-TV-AL3 and the simplified process of predicting the non-key frames using the proposed JMD-TV. However, it is important to note that all the implementations above have not been optimized for execution time.

5.4. Conclusive Remarks

In this chapter, we show how the proposed scheme can be used to replace conventional video compression. The proposed scheme is able to generate an approximation of the non-key frames in shorter time when compared to MC/ME methods. Additionally, it does not require any feedback channel or motion estimation as required by most of the conventional video coding schemes. The frames are arranged in different ways on the basis of GoP's. The proposed scheme is investigated with three different GoP size of 3, 5, and 8. The results shows that smaller GoP = 3 provides 1dB-2dB better reconstruction gains as compared to larger GoP = 8. In addition to this, a quantization approach is proposed to transform the CS measurements produced by the sensor nodes into bits. This allows us to compare the proposed BCS based JMD with other conventional video compression scheme. The detailed simulation analysis proves that the proposed JMD-TV can outperform the independent BCS-TV-AL3 scheme by a margin of 3dB to ~5dB at different subrates for various video sequences with low, moderate and high motion contents. When compared with conventional CS video reconstruction schemes, the proposed JMD-TV shows a gain of 2dB - 4dB in terms of reconstruction quality and takes approximately **2-3 times** shorter interval for execution.

Chapter 6
MULTI-VIEW VIDEO COMPRESSION AND RECONSTRUCTION

In the previous two chapters, inter-view and inter-frame correlation for image and video are exploited to reconstruct views and frames respectively. In this chapter, the ideas presented in chapter 4 (i.e. to exploit the inter-view correlation among the multi-view image from adjacent views) and chapter 5 (i.e. to exploit the spatial and temporal correlation within the video sequence) are combined and extended. To the best of our knowledge, very few have investigated on exploiting the inter-view correlation present among videos captured from different viewpoints.

The proposed scheme uses the concept of exploiting the correlation among the adjacent frames to predict the target frames. Not only that it exploits the inter-view correlation, but also the inter-frame (temporal, spatial) correlation within the successive frames. But different from the setup adopted in chapter 5, the use of Group of Pictures (GOP) is not required as the results show that the use of the four adjacent frames is sufficient. This helps to simplify the registration and fusion process. The process is also modified to accommodate the need of dealing with both inter-view and inter-frame correlations.

Similarly, the video captured by the different visual sensor in a Visual Sensor Network (VSN) is first compressed using the Block-based Compressive Sensing (BCS). All the videos are encoded independently at different subrates and transmitted to a server for reconstruction. Then, the proposed Joint Multi-phase Decoding (JMD) is applied to improve the reconstruction of videos frames encoded at lower subrate. In this case, four adjacent frames are used to produce the Side Projection Information (SPI), which serves as a prediction of the counterpart frames in lower subrate videos. Two of the four frames are extracted from neighbouring left and right views whereas, the remaining two are the temporal frames before and after the current view frame.

The rest of this chapter is organized as follows. Section 6.1 presents the model of using the proposed scheme for multi-view video compression. The scheme is explained in details in Section 6.2. All the experimental results are presented in Section 6.3 and the chapter is concluded in Section 6.4.

6.1. Multi-view Video Compression and Reconstruction Model

The overall mode is shown in Figure 6.1. In this case, we consider a VSN that consists of **S** number of visual nodes. Each visual node monitors a scene from different viewpoints. The captured data is then encoded and transmitted to the server independently. All the frames captured by the non-reference node(s) are encoded at a lower subrate, whereas, frames from the nearest left (s-1) and right (s+1) nodes are encoded at a higher subrate. At the server, the frames extracted from all the correlated visual nodes are used to produce the SPI, which will, in turn, be used to improve the reconstruction of frames captured by the non-reference node.

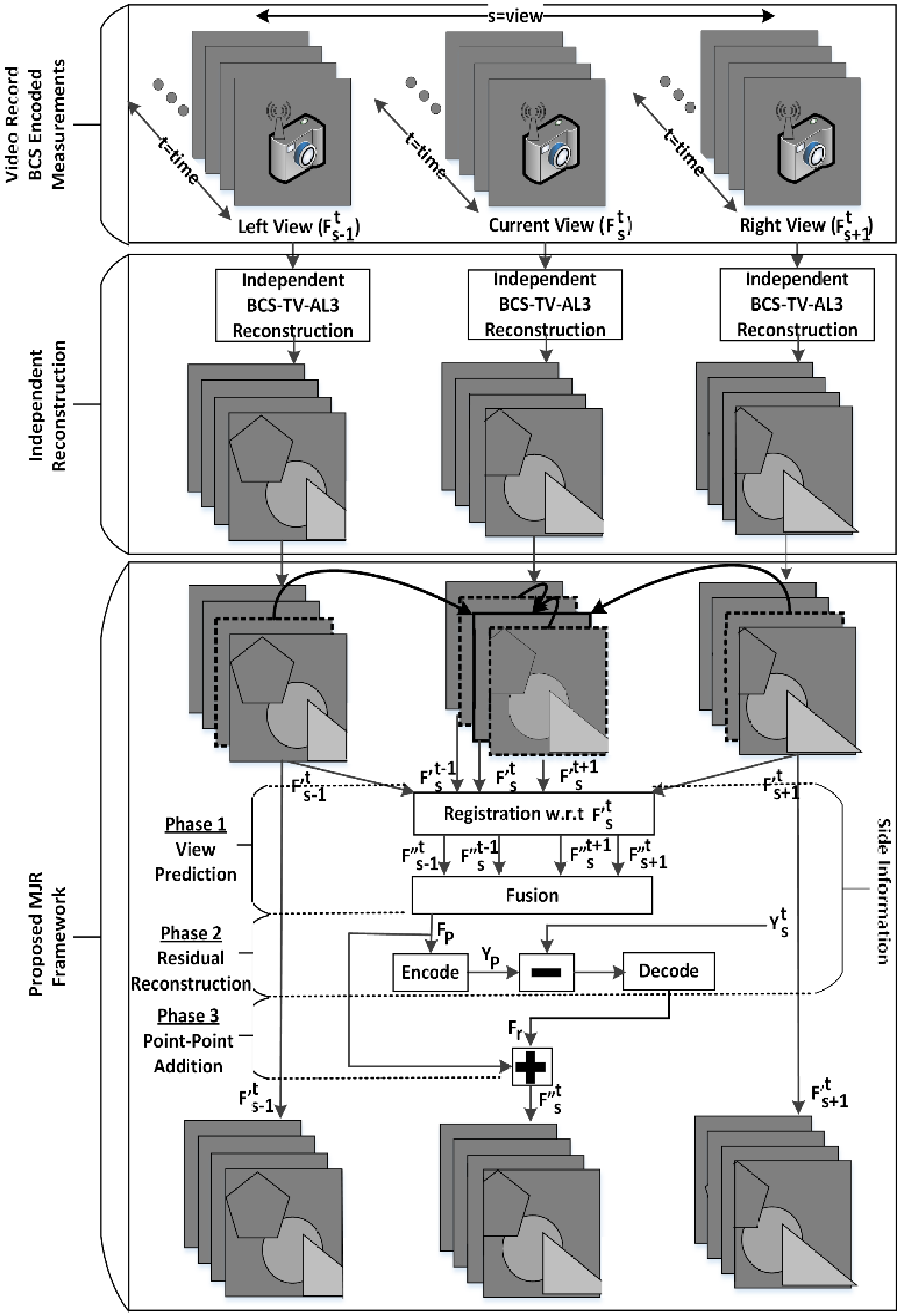


Figure 6.1: Block diagram of proposed Joint Multi-phase Decoding (JMD) for multi-view videos

6.2. Proposed Reconstruction Scheme

6.2.1. Encoding: Block-based Compressive Sensing

Each frame in each of the views F_x^y is first divided into small blocks of size 16×16. Next, each block is sampled with respect to the sampling matrix Φ_x^y. This produces a set of measurements Y_x^y as defined in Eq. (6.1).

$$Y_x^y = \Phi_x^y F_x^y \tag{6.1}$$

Where, **x** and **y** represents the view (… s-1, s, s+1 …) and time (… t-1, t, t+1 …) respectively, such that $0 \leq \mathbf{x} < \mathbf{S}$, $0 \leq \mathbf{y} < \infty$.

6.2.2. Decoding: Independent + Joint Multi-phase Decoding (JMD) Scheme

Initially, the encoded measurements $\mathbf{Y_x^y}$ received by the server are decoded independently using the TV-AL3. The proposed JMD is then applied to decode and improve the reconstruction quality of frames captured by the non-reference node.

- **Step 1: Frame Prediction:**

For each reconstructed key frame F'^{t}_{s} at time t captured by the non-reference node s, a prediction of it is generated by applying image registration and fusion on the four reconstructed adjacent frames (F'^{t+1}_{s}, F'^{t-1}_{s}, F'^{t}_{s+1},F'^{t}_{s-1}). They are the frames highlighted with the black dotted lines in Figure 6.1. The aim is to project and align the adjacent frames to the perspective of the key frame. This produces four aligned adjacent frames that can be fused later to create a SPI that better resembles the key frame.

Registration: In this case, an intensity-based method is adopted, because as mentioned in the previous chapter, it requires less amount of pre-processing and can achieve better alignment than that of feature-based methods.

The registration process initiates with the generation of an initial transformation matrix between the F'^{t}_{s} and F'^{t+1}_{s}, F'^{t-1}_{s}, F'^{t}_{s+1}, F'^{t}_{s-1} frames by using phase correlation. Next, frames F'^{t+1}_{s}, F'^{t-1}_{s}, and frames F'^{t}_{s+1}, F'^{t}_{s-1} are aligned to F'^{t}_{s} by using translation and affine transformation respectively to produce transformed frames. Two different transformations are used because we are dealing with frames having different perspectives. For example, the frames F'^{t}_{s+1},F'^{t}_{s-1} obtained from neighbouring left and right nodes of F'^{t}_{s} are aligned using affine transform while the frames F'^{t+1}_{s}, F'^{t-1}_{s} before and after the F'^{t}_{s} are aligned using translation transform. This is because it is noted from the analysis that affine transform produces better alignment when multiple frames not on the same plane are to be rectified. Whereas, when the frames are on the same plane then the translation transform produces better results.

The transformed frames are then passed through a similarity metric (SM) and optimization function to estimate the registration accuracy. The mutual information is used in the SM while One Evolutionary (OE) and Gradient Descent (GD) are used for optimization of SM. The optimizer is considered to be one of the important steps of registration. The aim is to maximize the SM. We evaluated the two optimizers independently and noted that OE works well when the frame orientations are similar while GD works well for different orientations. Thus, both the optimizers are used.

The OE optimizer is used for F'^{t+1}_{s} and F'^{t-1}_{s} frames from the same non-reference node having similar orientation. On the one hand, OE optimizer iterates to find a set of parameters that produce the best possible registration result rather than adjusting the transformation parameters in the direction of the extrema. It increases the ability to step out of the range of non-optimal minimal range to maximum range due to the random nature of the parameter variation. On the other hand, GD is better for registration of neighboring left and right F'^{t}_{s+1} and F'^{t}_{s-1} frames having different orientations. It adjusts the transformation parameters so that the optimization follows the gradient of the similarity metric in the direction of the maxima. The registered version of F'^{t+1}_{s}, F'^{t-1}_{s}, F'^{t}_{s+1}, F'^{t}_{s-1} are then referred to as F''^{t+1}_{s}, F''^{t-1}_{s}, F''^{t}_{s+1}, and F''^{t}_{s-1} respectively.

Fusion: Once the frames are registered, the fusion process is performed using wavelets to preserve the quality and detail information of the frames. First, the registered frames are decomposed into respective decomposition maps using a Symlet-4 wavelet filter with 3 level of decomposition. Each map contains a set of the approximation (A) and detail (D) coefficients. Next, the decomposition maps are merged using point-to-point operations. In the case of frames F''^{t}_{s+1}, and F''^{t}_{s-1}, the mean of detail (D) coefficients of the two decomposition maps is calculated and taken as the output of the fusion, and for the approximation (A) coefficients the highest magnitude (max) are selected after comparing the coefficients from the two decomposition maps. While in the case of frames F''^{t+1}_{s} and F''^{t-1}_{s}, the mean of both approximation (A) and detail (D) coefficients of the two decomposition maps is calculated respectively and taken as the output. The reason is that the approximation coefficient between the frames F''^{t+1}_{s} and F''^{t-1}_{s} within the same view are not significantly improved from each other. While the approximation coefficients of the neighboring frames F''^{t}_{s+1} and F''^{t}_{s-1} can produce better approximation than each other. The two fused decomposition maps obtained from the frames F''^{t}_{s+1}, F''^{t}_{s-1} and F''^{t+1}_{s}, F''^{t-1}_{s} are then merged using max and mean operations for approximation (A) and detail (D) to produce a fused decomposition map.

After fusion, the inverse transformation is applied to the fused decomposition map to produce prediction frame F_P.

- **Step 2: Residual Reconstruction:**

The projection of F_P onto the measurement basis $Y_P = \Phi^{t}_{s} F_P$ is performed. In other words, BCS is applied to F_P to obtain its representation in terms of CS measurements, Y_P. Then, the difference between Y_P and the measurements of the current frame Y^{t}_{s} (received by the server) is calculated. The reason for doing this at the measurement level is to ensure maximum correlation with minimum prediction errors as compared to the spatial level. The output is known as the residual measurement Y_r as depicted in Eq. (6.2).

$$Y_r = Y^{t}_{s} - Y_P \tag{6.2}$$

To obtain the residual frame F_r, the residual measurement Y_r is decoded by using the TV-AL3. Generally, the reconstruction of the residual yields better results when similar blocks exist in both frames. In the multi-view videos that we have tested, occlusions take place due to depth discontinuity (i.e. overlapping of objects in the frames). The residual of such occluded blocks exhibits features different from the other blocks of the frame and often result in higher correlation.

- **Step 3: Point-to-Point Addition:**

The residual frame F_r and the prediction frame F_P are added together to produce the final reconstruction F''^t_s which is an improved version of the initially reconstructed F'^t_s. It is a normal point-to-point addition as expressed in Eq. (6.3).

$$F''^t_s = F_r + F_P \tag{6.3}$$

6.3. Experimental Results

In the following subsections, the evaluation of the proposed scheme coupled with TV-AL3 referred to as JMD-TV is presented. The proposed scheme is applied to various standard grayscale multi-view video sequences [130, 133, 134] shown in Table 6.1. The selected video sequences are categorized into three types, namely low, medium and high. They are categorized with respect to the amount of variations and motions. For example, “Love Birds” is consider as low motion video because the background is mostly static and the entire video only involves some minor facial and hand movements.

The evaluation is carried out by recording the Peak Signal to Noise Ratio (PSNR) and Structural Similarity Index (SSIM) at different subrates. Due to the random Φ, the reported values represent the average of 5 independent trials. A block size of 16x16 is adopted for each video. Each frame from the neighboring left and right nodes (F'^t_{s+1}, F'^t_{s-1}) is encoded at higher subrate of 0.5 (fixed). Then each frame from the non-reference node (F'^t_s) is encoded at lower subrates that range from 0.05-0.3.

Table 6-1: Several standard grayscale Multi-view Video Sequences

Video Sequence = Break Dancer No. Frames = 300 Content Type =High Size = 1024x768 Fps = 15 and 30	
Video Sequence = Ballet No. Frames = 300 Content Type =High Size = 1024x768 Fps = 15 and 30	
Video Sequence = Book Arrival No. Frames = 300 Content Type =Medium Size = 1024x768 Fps = 15 and 30	
Video Sequence = Newspaper No. Frames = 300 Content Type =Medium Size = 1024x768 Fps = 15 and 30	
Video Sequence = Love Birds No. Frames = 300 Content Type =Low Size = 1024x768 Fps = 15 and 30	
Video Sequence = Exit No. Frames = 250 Content Type =Medium Size = 640x480 Fps = 15 and 30	

6.3.1. Proposed JMD-TV and BCS-TV-AL3

In this subsection, the Rate-Distortion (R-D) performance of the proposed scheme is evaluated. The proposed scheme (JMD-TV) is compared to independent BCS-TV-AL3. At the same time, the effect of key frames (F_K) subrate on the reconstruction of the non-key frame (F_{NK}) using the proposed scheme is evaluated. Two different setups are used for evaluation. In the first setup, all the key frames (i.e. the four adjacent frames) and non-key frame (F'^{t}_{s}) are transmitted at the same subrate (i.e. $M_{K=}$ M_{NK}). In the second setup, the key frames from the neighbouring left and right nodes (F'^{t}_{s+1}, $F'^{t}_{s-1}$$F'^{t+1}_{s}$,, $F'^{t}_{s-1}$$F^{t}_{s+1}$, F^{t}_{s-1}) are transmitted at a fixed higher subrate of $M_K = 0.5$ and the key frames within the non-reference view (F'^{t+1}_{s}, F'^{t-1}_{s}) are transmitted at the same subrate as of and non-key frame (F'^{t}_{s}) range from $M_{NK} = 0.05$ to 0.3. The results presented in Table 6.2 are the average value of the first 50 frames in the first three views of each multi-view video sequence. Thus, a total of 150 frames per dataset is used.

The results presented in Table 6.2 shows that the proposed scheme on average is about 1dB to 2.5 dB better than BCS-TV-AL3 from higher to lower subrate. For low motion video (Love Bird), the gain is higher than moderate Book Arrival and Newspaper and high motion videos (Ballet, Break Dancer, Exit). The proposed scheme performs better when the variation and object motion lower. In such case, the intensity-based registration is able to register the frames more accurately. This is because most of the objects' intensity and perspective remain unchanged when moving from one frame to another. Accurate registration creates better SPI that can be used to improve the non-reference frame.

Table 6-2: PSNR (dB) achieved by the proposed scheme for different multi-view videos

Break Dance						
Subrate	**0.05**	**0.1**	**0.15**	**0.2**	**0.25**	**0.3**
BCS-TV-AL3	25.39	29.7	31.42	33.25	35.02	36.49
JMD-TV (M_k=M_{Nk})	27.60	31.00	32.70	34.16	35.77	36.88
JMD-TV (M_k= 0.5)	**28.65**	**31.85**	**33.32**	**34.74**	**36.23**	**37.52**
Ballet						
Subrate	**0.05**	**0.1**	**0.15**	**0.2**	**0.25**	**0.3**
BCS-TV-AL3	25.44	28.99	30.55	31.92	33.01	34.03
JMD-TV (M_k=M_{Nk})	26.85	30.11	31.52	32.72	33.58	34.52
JMD-TV (M_k= 0.5)	**27.58**	**30.81**	**32.19**	**33.21**	**33.99**	**34.86**
Book Arrival						
Subrate	**0.05**	**0.1**	**0.15**	**0.2**	**0.25**	**0.3**
BCS-TV-AL3	23.27	26.82	28.19	29.65	30.73	31.98
JMD-TV (M_k=M_{Nk})	25.69	29.15	30.35	31.77	32.85	34.02
JMD-TV (M_k= 0.5)	**28.21**	**31.65**	**32.76**	**34.01**	**35.01**	**36.00**
Newspaper						
Subrate	**0.05**	**0.1**	**0.15**	**0.2**	**0.25**	**0.3**
BCS-TV-AL3	18.45	21.64	23.00	24.47	25.89	27.02
JMD-TV (M_k=M_{Nk})	21.36	24.25	25.71	27.15	28.55	29.69
JMD-TV (M_k= 0.5)	**23.68**	**26.03**	**27.5**	**28.94**	**30.23**	**31.38**
Lovebird						
Subrate	**0.05**	**0.1**	**0.15**	**0.2**	**0.25**	**0.3**
BCS-TV-AL3	19.49	23.23	24.31	25.46	26.46	27.37
JMD-TV (M_k=M_{Nk})	22.31	25.67	26.75	27.86	28.84	29.94
JMD-TV (M_k= 0.5)	**25.83**	**27.81**	**28.86**	**29.93**	**30.81**	**31.89**
Exit						
Subrate	**0.05**	**0.1**	**0.15**	**0.2**	**0.25**	**0.3**
BCS-TV-AL3	24.04	27.98	29.53	31.14	32.42	33.74
JMD-TV (M_R=M_{NR})	26.27	29.41	31.22	32.42	33.69	35.18
JMD-TV (M_k= 0.5)	**26.92**	**30.53**	**31.78**	**33.17**	**34.32**	**35.52**

Note: The bold values relates to the maximum PSNR (dB) reached for a given subrate and video sequence

Additionally, Table 6.3 shows the visual comparison results (SSIM) of BCS-TV-AL3 with JMD-TV (JMD-TV(0.5)) because from Table 6.2 we observe that JMD-TV(0.5) produces better output as compare to JMD-TV(MR=MNR). Further our main scheme focuses on (JMD-TV (0.5)). The JMD-TV (MR=MNR) was evaluated only for testing. The SSIM shows similar trend when compared to PSNR. Overall, it is noticed that the gain decreases when the subrate increases. This is because when the subrate of F_s^t increases, a larger set of measurements is used to represent F_s^t. Thus, it reduces the prediction errors of F_s^t even before the proposed scheme is applied. Since the quality of

F^t_{s+1} and F^t_{s-1} is not much better than F^t_s, this limited the gain that can be achieved by projecting and making use of the correlation information of F^t_{s+1} and F^t_{s-1}.

Table 6-3: SSIM results of the proposed scheme for reconstruction of various multi-view video sequences

Break Dance						
Subrate	**0.05**	**0.1**	**0.15**	**0.2**	**0.25**	**0.3**
BCS-TV-AL3	0.74	0.83	0.87	0.89	0.91	0.92
JMD-TV	**0.82**	**0.89**	**0.92**	**0.94**	**0.96**	**0.96**
Ballet						
Subrate	**0.05**	**0.1**	**0.15**	**0.2**	**0.25**	**0.3**
BCS-TV-AL3	0.73	0.81	0.85	0.9	0.91	0.92
JMD-TV	**0.81**	**0.87**	**0.9**	**0.94**	**0.94**	**0.95**
Book Arrival						
Subrate	**0.05**	**0.1**	**0.15**	**0.2**	**0.25**	**0.3**
BCS-TV-AL3	0.67	0.78	0.83	0.86	0.89	0.91
JMD-TV	**0.88**	**0.93**	**0.94**	**0.95**	0.96	**0.96**
Newspaper						
Subrate	**0.05**	**0.1**	**0.15**	**0.2**	**0.25**	**0.3**
BCS-TV-AL3	0.57	0.71	0.77	0.82	0.85	0.88
JMD-TV	**0.81**	**0.87**	**0.89**	**0.93**	**0.94**	**0.95**
Lovebird						
Subrate	**0.05**	**0.1**	**0.15**	**0.2**	**0.25**	**0.3**
BCS-TV-AL3	0.50	0.64	0.7	0.75	0.8	0.83
JMD-TV	**0.80**	**0.85**	**0.88**	**0.9**	**0.92**	**0.94**
Exit						
Subrate	**0.05**	**0.1**	**0.15**	**0.2**	**0.25**	**0.3**
BCS-TV-AL3	0.69	0.80	0.83	0.87	0.89	0.91
JMD-TV	**0.79**	**0.89**	**0.91**	**0.93**	**0.94**	**0.95**

6.3.2. Inter-view and Inter- frame Correlations

In this subsection, the effect of neighboring frames (inter-view) and adjacent temporal frames (inter-frame) on the non-reference node is evaluated. The analysis is carried out based on the three scenario define in Table 6.4.

Table 6-4: Different correlation cases

Case	Description
1	Only the temporal frames correlations are used with proposed JMD.
2	Only the inter-view frames correlations among the frames are considered only with proposed JMD.
3	Both temporal and inter-view correlations are combined together in the proposed JMD.

In case 1, only the inter-frame correlations are considered. While, in case 2 only the inter-view correlations are considered. In case 3, both inter-frame and inter-view frame correlations are used to improve the reconstruction of the non-reference frame.

Table 6-5: Average PSNR (dB) achieved for independent and joint exploitation of inter-frame and inter-view frame correlations with proposed scheme for various multi-view videos

			Break Dancer			
Subrate	**0.05**	**0.1**	**0.15**	**0.2**	**0.25**	**0.3**
BCS-TV-AL3 [102]	25.39	29.7	31.42	33.25	35.02	36.39
JMD-TV (Temporal)	27.03	30.55	32.17	33.87	35.63	36.79
JMD-TV (Interview)	28.30	31.14	32.76	34.27	35.85	36.96
JMD-TV (Joint)	**28.65**	**31.85**	**33.32**	**34.74**	**36.23**	37.52
			Ballet			
Subrate	**0.05**	**0.1**	**0.15**	**0.2**	**0.25**	**0.3**
BCS-TV-AL3 [102]	25.44	28.99	30.55	31.92	33.01	34.03
JMD-TV (Temporal)	26.11	29.76	31.17	32.54	33.25	34.21
JMD-TV (Interview)	26.85	30.38	31.63	32.92	33.56	34.41
JMD-TV (Joint)	**27.58**	**30.81**	**32.19**	**33.21**	**33.99**	**34.86**
			Newspaper			
PSNR Subrate	**0.05**	**0.1**	**0.15**	**0.2**	**0.25**	**0.3**
BCS-TV-AL3 [102]	18.45	21.64	23	24.47	25.89	27.02
JMD-TV (Temporal)	21.78	24.76	26.03	27.47	28.84	29.9
JMD-TV (Interview)	22.93	25.73	26.93	28.36	29.65	30.65
JMD-TV (Joint)	**23.68**	**26.03**	**27.5**	**28.94**	30.23	**31.38**
			Love Birds			
Subrate	**0.05**	**0.1**	**0.15**	**0.2**	**0.25**	**0.3**
BCS-TV-AL3 [102]	19.49	23.23	24.31	25.46	26.46	27.37
JMD-TV (Temporal)	22.82	25.99	27.06	28.15	29.11	29.92
JMD-TV (Interview)	24.45	26.51	27.97	29.07	30.05	30.95
JMD-TV (Joint)	**25.83**	**27.81**	**28.86**	**29.93**	**30.81**	**31.89**
			Book Arrival			
Subrate	**0.05**	**0.1**	**0.15**	**0.2**	**0.25**	**0.3**
BCS-TV-AL3 [102]	23.27	26.82	28.19	29.65	30.73	31.98
JMD-TV (Temporal)	26.04	29.53	30.66	32.01	33.07	34.22
JMD-TV (Interview)	27.25	30.68	32.03	33.46	34.31	34.93
JMD-TV (Joint)	**28.21**	**31.65**	**32.76**	**34.01**	**35.01**	**36.00**
			Exit			
Subrate	**0.05**	**0.1**	**0.15**	**0.2**	**0.25**	**0.3**
BCS-TV-AL3 [102]	24.04	27.98	29.53	31.14	32.42	33.74
JMD-TV (Temporal)	25.79	28.99	30.57	31.94	33.23	34.44
JMD-TV (Interview)	26.22	29.93	31.18	32.6	33.78	34.99
JMD-TV (Joint)	**26.92**	**30.53**	**31.78**	**33.17**	**34.32**	**35.52**

***Note*:** The bold values relates to the maximum PSNR (dB) reached for a given subrate and video sequence

As the proposed scheme combine both temporal and interview redundancies to improve the reconstruction of frame. In table 6.5 we try to analysis the effect of both the redundancies independently and jointly on the reconstruction of the frame. The results shows the effect of frames captured by neighboring left and right nodes on the non-reference view is more significant than the adjacent temporal frames. This is due to the fact that the neighboring nodes frames are encoded at a higher subrate than the adjacent temporal frames that helps to generate a better prediction of the frame. Additionally, from our observation, SSIM measurement exhibits the same trend when compared to the case of PSNR measurement.

6.3.3. Visual Results

Since the proposed scheme performs better at lower subrate, it is important to ensure that the subrate used is sufficient to produce visually recognizable frame. The results shown in Figure 6.2 are sample frames reconstructed by using JMD-TV and BCS-TV-AL3 at different subrates. It the case of using the proposed scheme, the adjacent left and right frames are encoded at a subrate of 0.5 and the other frames are transmitted at the subrate of 0.05 to 0.3.

The visual results show that the JMD-TV has alleviated the blurring effect presented in the frame reconstructed using BCS-TV-AL3. By comparing the regions highlighted in the white dotted boxes, it can be noticed that the frame reconstructed by using JMD-TV looks much sharper when compared to that of BCS-TV-AL3. Moreover, it can be noticed that the JMD-TV reconstructions look much sharper for medium (Newspaper, Book Arrival, Exit) and low (Lovebirds) content videos as compared to high motion content videos (Ballet, Break-dancer).

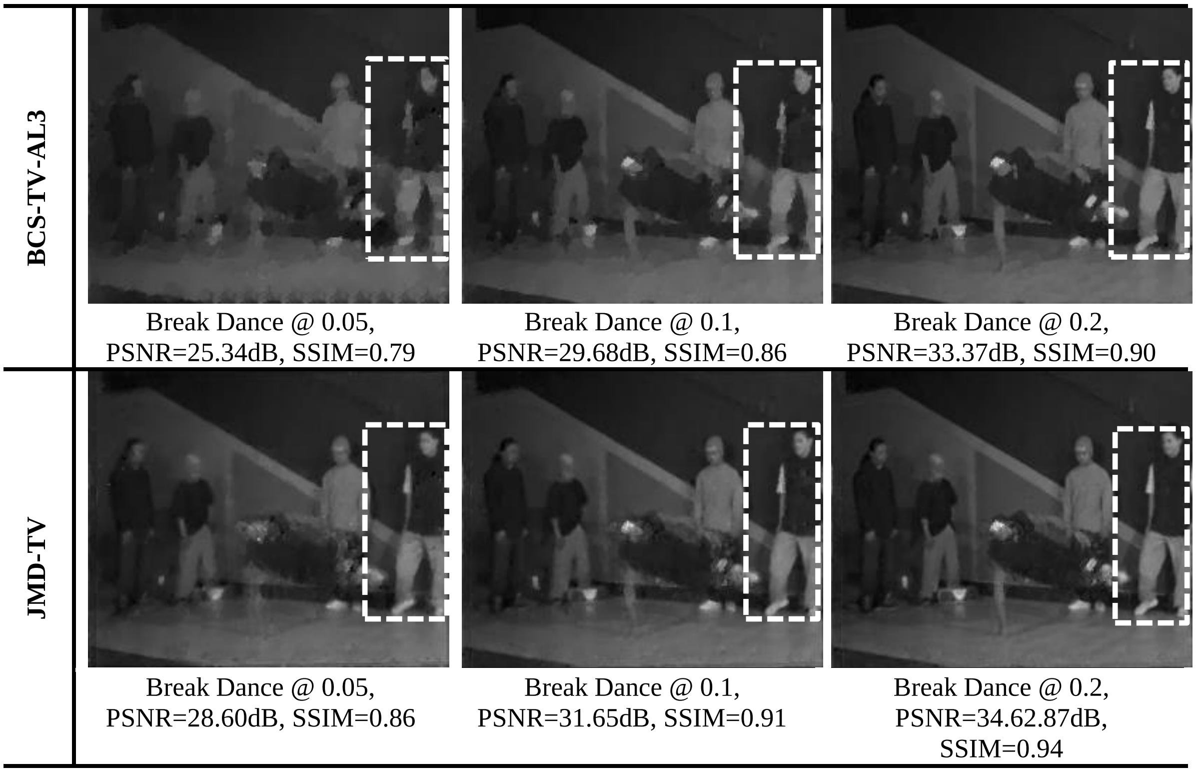


Figure 6.2a: Reconstruction of frame# 8th of Break Dancer video at three different subrates (0.05, 0.1, 0.2)

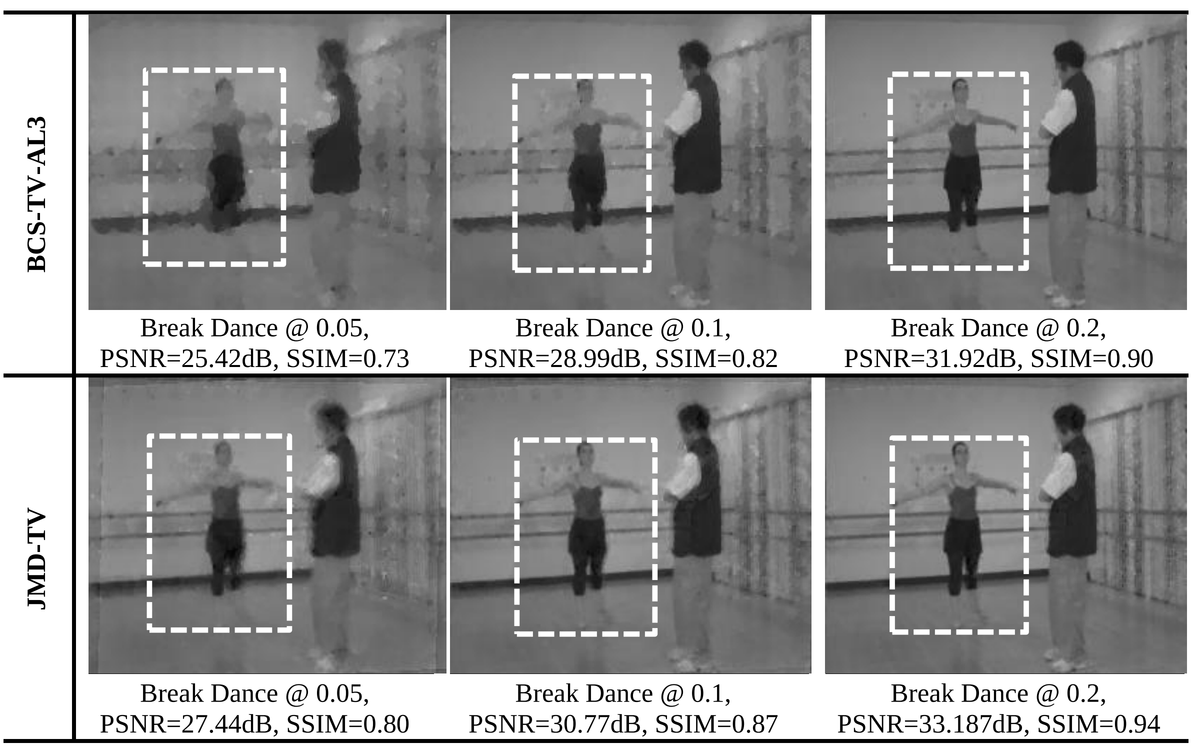


Figure 6.2b: Reconstruction of frame# 31 of Ballet video at three different subrates (0.05, 0.1, 0.2)

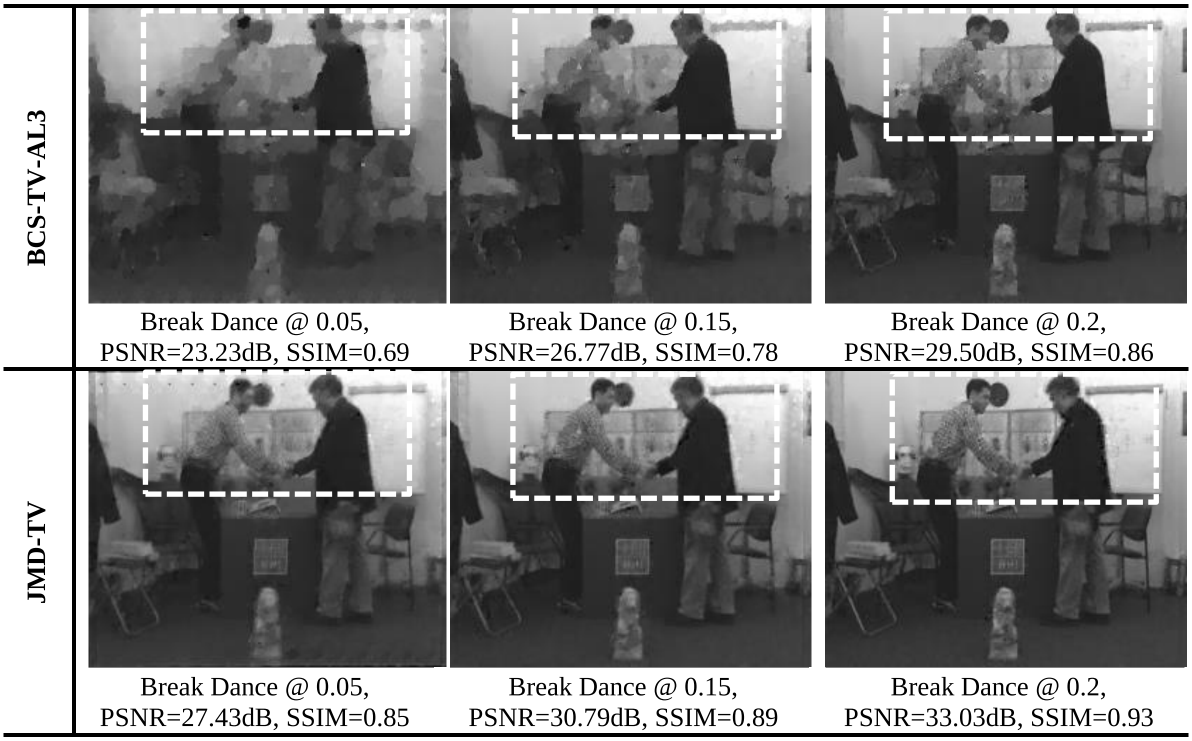


Figure 6.2c: Reconstruction of frame# 50 of Book Arrival video at three different subrates (0.05, 0.1, 0.2)

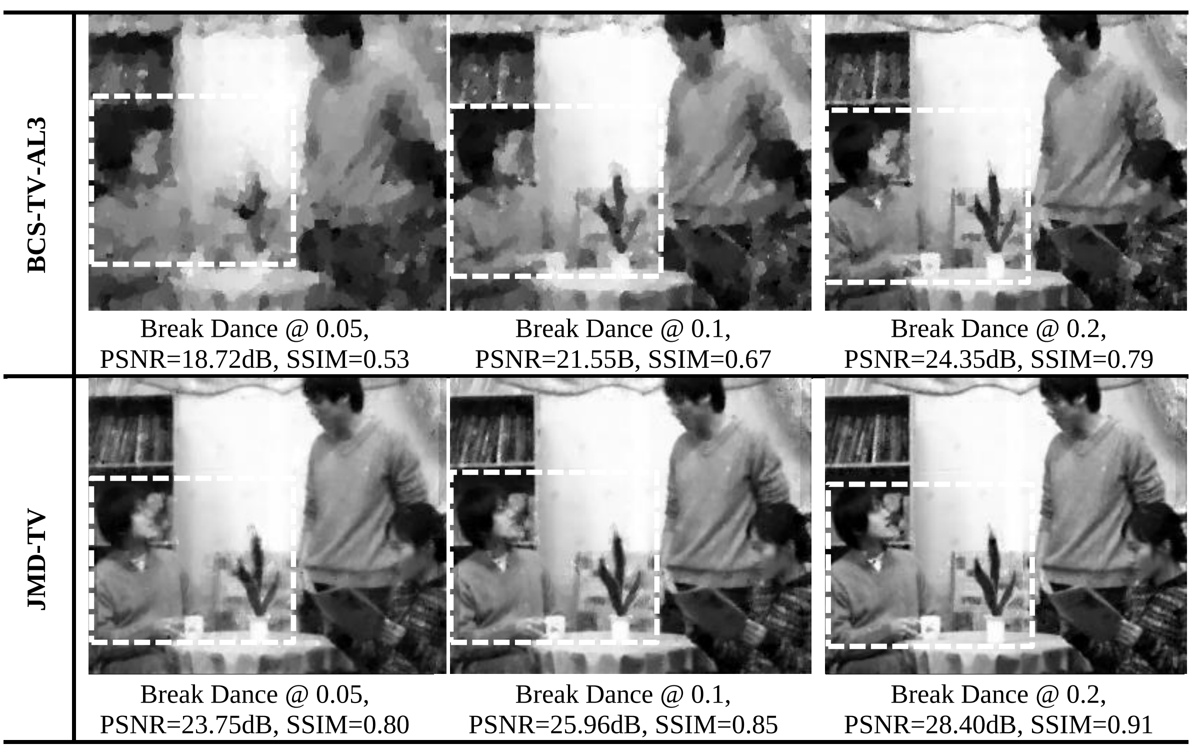


Figure 6.2d: Reconstruction of frame# 188 of Newspaper video at three different subrates (0.05, 0.1, 0.2)

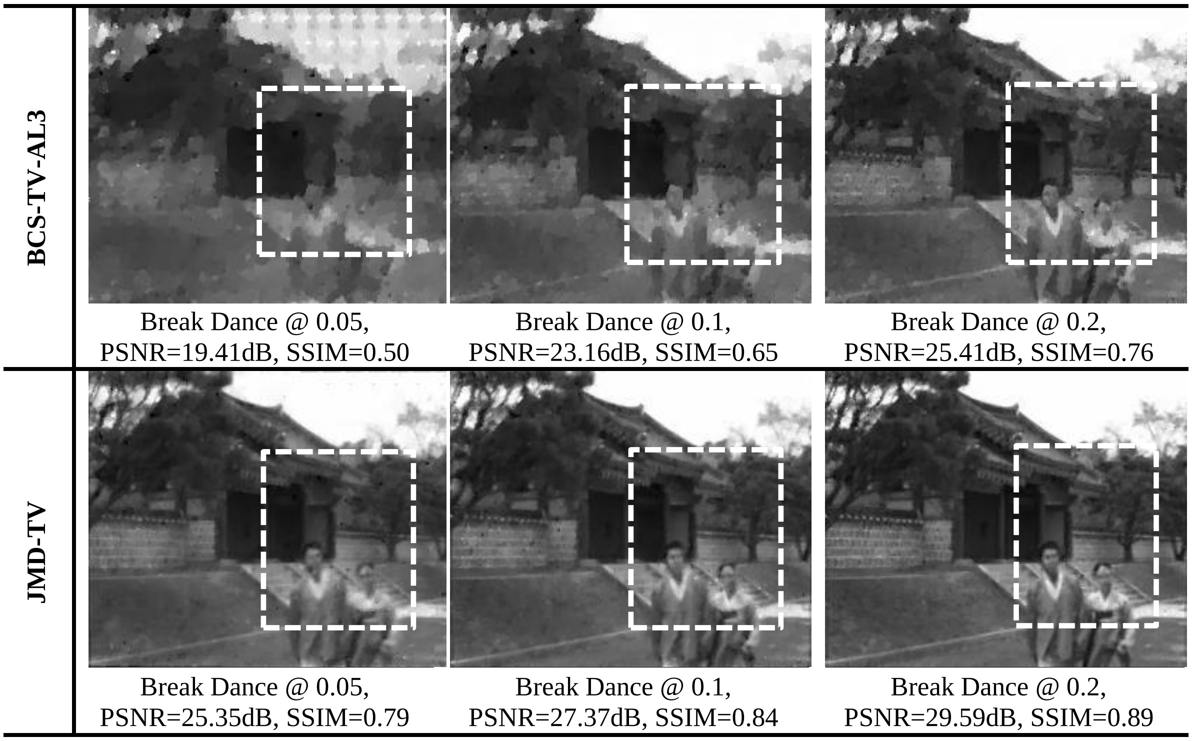


Figure 6.2e: Reconstruction of frame# 100 of Love Birds video at three different subrates (0.05, 0.1, 0.2)

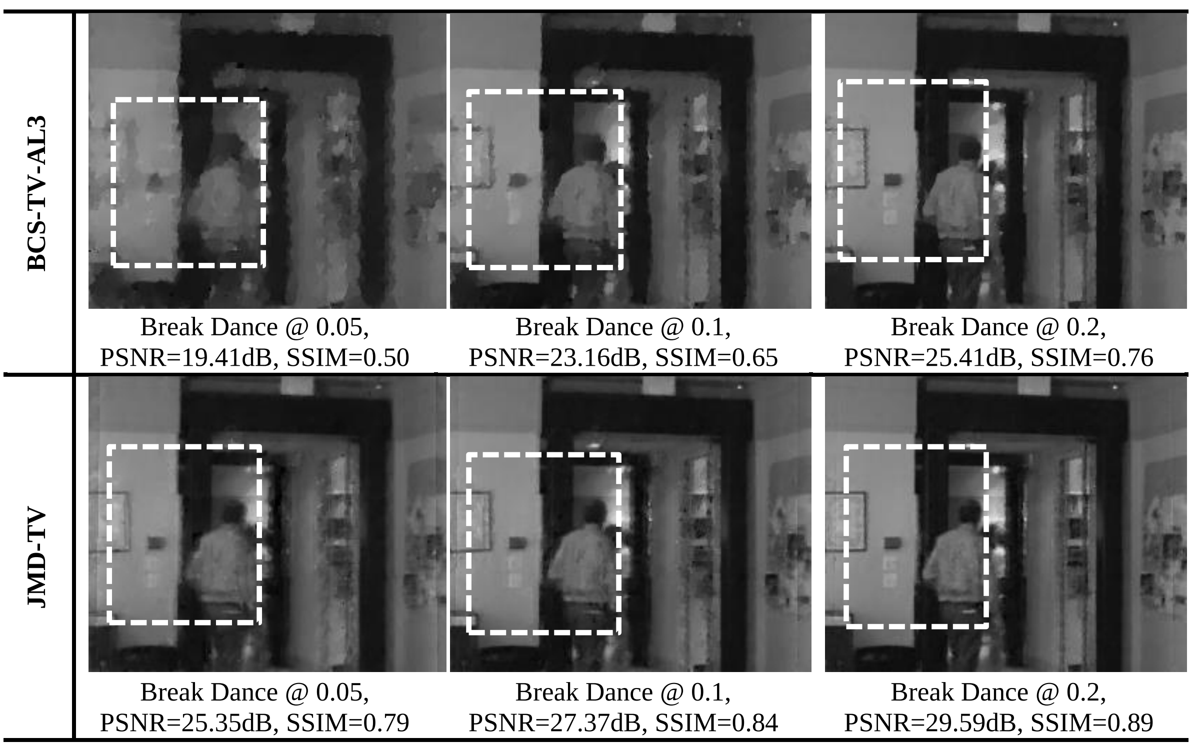


Figure 6.2e: Reconstruction of frame# 5 of Exit video at three different subrates (0.05, 0.1, 0.2)

Figure 6.2: Visual quality comparison of different multi-view video frames at three different subrates (0.05, 0.1, 0.2) using conventional BCS-TV-AL3 and proposed JMD

6.3.4. Comparison of Proposed Scheme with other CS-based Compression Schemes

In this subsection, the proposed scheme is compared with the Motion Compensation-Joint Decoding (MC-JD) and Disparity Compensation – Total Variation (DC-TV) scheme proposed in [82, 83]. The results of both MC-JD and DC-TV were directly obtained from the literature, as the implementation was not readily available at the time of writing. The setup used by MC-JD and DC-TV is adopted for the evaluation. However, only the performance at lower subrates (i.e. 0.05-0.3) is presented, as it is difficult for a battery-powered device to always encode and send the captured frames at high subrate ($\geq$ 0.35). The MC-JD setup involves all the frames of the three views of each multi-view video as well as encoding of key frames at subrate of 0.6, block size =16, and pixel resolution of 320x240. While DC-TV adopts the first five frames from the first five views of each multi-view video and all the frames are encoded at the same subrate.

All the simulation results presented in Table 6.6 are in terms PSNR(dB). From the simulation results, it can be observed that the performance gain of the proposed JMD-TV is 1.5dB – 2.5dB higher than MC-JD at various subrates whereas, when compared with DC-TV the gain is better at lower subrates.

Table 6-6: R-D Performance gain (dB) comparison of the proposed scheme with standard CS video reconstruction scheme for different multi-view video sequences

Subrate	0.05	0.1	0.15	0.2	0.25	0.3
Break Dancer						
JMD-TV	**2.66**	**2.15**	**2.06**	1.78	1.54	1.28
DC-TV[83]	1.92	2.02	-	**2.08**	-	**2.11**
MC-JD [82]	-	-	-	-	-	-
Ballet						
JMD-TV	**2.24**	**2.02**	**1.84**	1.69	1.28	1.03
DC-TV[83]	1.38	1.59		**2.22**		**2.31**
MC-JD [82]	-	-	-	-	-	-
Book Arrival						
JMD-TV	**3.92**	**3.53**	**3.44**	**3.36**	2.86	2.69
DC-TV[83]	1.25	1.60	-	2.7	-	**3.74**
MC-JD [82]	-	-	-	-	-	-
Exit						
JMD-TV	**2.88**	**2.55**	**2.25**	**2.03**	**1.9**	**1.78**
DC-TV[83]	-	-	-	-	-	-
MC-JD [82]	1.25	1.43	-	1.51	-	1.60
Ballroom						
JMD-TV	**2.01**	**1.92**	**1.64**	**1.35**	**1.12**	**1.01**
DC-TV[83]	-	-	-	-	-	-
MC-JD [82]	0.29	0.45	-	0.55	-	0.70

***Note*:** The bold values relates to the maximum PSNR (dB) gain reached for a given subrate and video sequence

The performance of the DC-TV is better than proposed JMD-TV at higher subrates because the DC-TV is based on motion estimation and compensation approach. Such approach depends on the number of measurements to facilitate the motion estimation and compensation required for the frame prediction. The larger the number of measurements the better will be the prediction and will result in improved reconstruction quality at higher subrate. While, in the case of proposed scheme the larger measurements help to reduce the prediction errors and improve the reconstruction of $\mathbf{F}_s^t$, even before the proposed scheme is applied as required in DC-TV. Hence, the quality of adjacent frames is not much better than $\mathbf{F}_s^t$, and this limited the gain that can be achieved by making use of the correlation information of the adjacent frames.

6.4. Conclusive Remarks

In this chapter, the proposed JMD scheme was investigated for multi-view video compression. The proposed scheme exploits the correlation among the adjacent frames to predict the non-reference frames. Not only that it exploits the inter-view correlation, but also the inter-frame (temporal, spatial) correlation within the successive frames. The use of Group of Pictures (GOP) is not required as the results show that the use of the four adjacent frames is sufficient. The experimental results show that the proposed scheme outperforms the independent BCS-TV-AL3 by 3dB to ~5dB. Moreover, it also provides better reconstruction in terms of PSNR than other CS-based multi-view video compression schemes such as MC-JD and DC-TV by 1dB to 2dB.

Chapter 7
HARDWARE IMPLEMENTATION OF THE PROPOSED COMPRESSION SCHEME FOR VISUAL SENSOR NETWORK (VSN)

In the previous chapters, the simulation results show that Compressive Sensing (CS) has the potential to serve as an efficient compression for Visual Sensor Network (VSN), due to the simple-encoder complex-decoder paradigm, which is the inverse of traditional compression. However, one of the main challenges in using CS for compression is on reconstructing the images from a very small sample set of data. To the best of our knowledge, there is no practical evaluation on the quality of reconstructed images compressed using CS, as well as the energy consumption, memory utilization, and execution time. In this chapter, a practical VSN platform is developed to evaluate the aforementioned criteria.

Although there are existing VSN platform [135-145], most of them do not have an efficient compression implemented on the visual node. In this regard, a prototype is developed to implement and evaluate the proposed scheme. The chapter is organized as follows. Section 7.1 describes the hardware and software components used to construct the VSN platform. This is followed by the experimental setup in section 7.2. Then in section 7.3 the evaluation results are presented and discussed. Finally, this chapter is concluded in Section 7.4.

7.1. BCS Visual Sensor Platform Overview

A VSN platform primarily consists of hardware and software components. The hardware component includes the camera, processing unit and transmission module that work together to create a visual node that is capable of capturing and sending the data to the workstation for further processing. Whereas, the software component includes image acquisition, encoding process and communication protocol that helps to compress and packetized the data before transmission. As shown in Figure 7.1 is an example of how devices in VSN are typically connected. The development of the platform also aim to create a simple, flexible and low-cost VSN platform integrated with energy efficient compression.

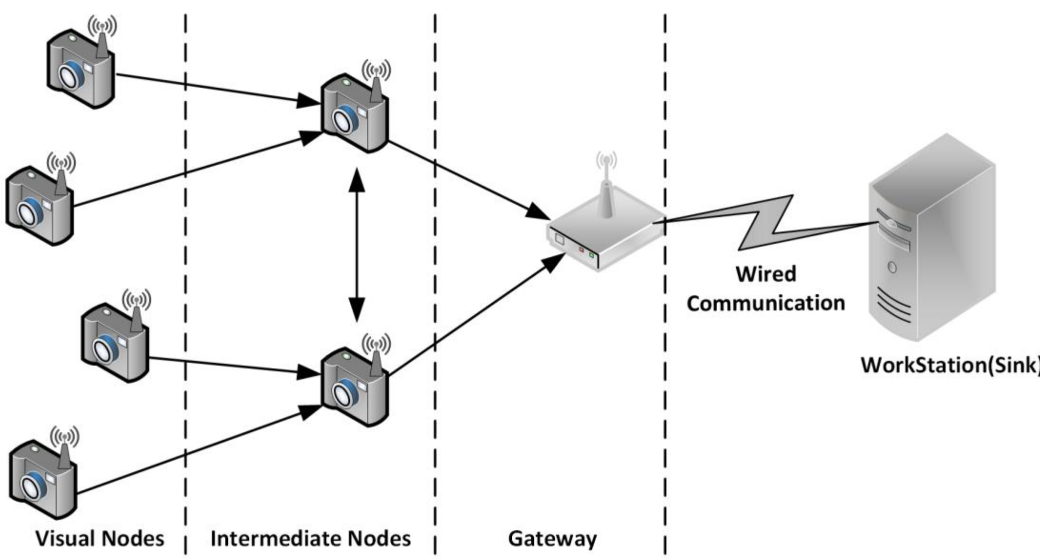


Figure 7.1: Overall architecture of a VSN

The main motivations in designing the proposed platform are:

- to have an off-the-shelf solution that is easily reproducible using existing low cost and widely available hardware components. We create a visual sensor by combining Arduino board, with an external uCAM-II camera and XBee transmission module. The uCAM-II camera is used to capture image data that will be processed and compressed on the Arduino board, before they are transmitted via the XBee transmission module.
- to implement BCS on the visual sensor to reduce the amount of data that needs to be processed and transmitted. BCS is adopted to create a simple-encoder complex-decoder paradigm that is preferable for VSN. This shifted most of the complex computation to the server and helps to prolong the lifetime of the devices that are powered by batteries.
- to implement and evaluate the proposed compression scheme using real-world data.

Details of the hardware and software components used to implement the proposed scheme are provided in the following subsections.

7.1.1. Hardware Components

As shown in Figure 7.2 is a visual node that consists of an Arduino Due board [146], a CMOS uCAM-II camera [147] and an XBee transmission module [148].

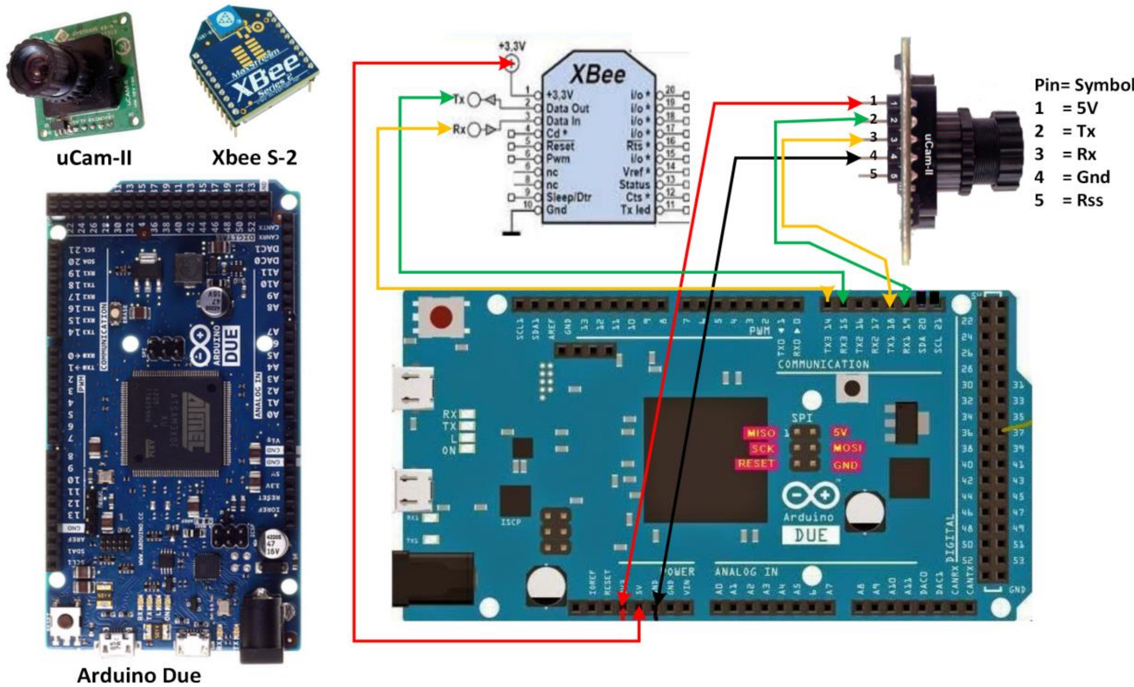


Figure 7.2: Standalone visual node built using Arduino Due, uCAM-II and XBee.

i. Arduino Due Board:

Although there are a number of other microcontrollers available, Arduino is a low-cost card-size board that offers sufficient processing power and memory for simple computation tasks. Moreover, its functionalities can be extended by connecting to many other peripherals (or shields), the code developed for one model can be reprogrammed and run on other Arduino board with minimum modifications. In the development of the proposed BCS visual node, an Arduino Due board [146] is selected. It is equipped with an Atmel SAM3X8E ARM Cortex-M3 micro-controller running at 84MHz, 96KB of SRAM memory, and 512 KB of flash memory. In addition to this, it also comes with several URAT interfaces that can be used to communicate with other external components. The reason of selecting Due over other Arduino boards is that it uses less energy (runs at 3.3V), higher computing performance (clock speed of 84 MHz), and has more SRAM and flash memory. Overall, it is difficult to implement image processing task on other Arduino boards due to the limited amount of memory.

ii. uCAM-II CMOS Camera:

Among the many low-power, low-cost CMOS cameras [149-153], the uCAM-II by 4D schemes is selected [169] for the development of the BCS visual node. Unlike the other available cameras that only provide images in JPEG format, the uCAM-II is capable of providing images in both RAW and JPEG formats. Furthermore, uCAM-II can capture images at resolution ranges from 80x60 to 640x480. Moreover, the uCAM-II is also compatible with lenses of different viewing angles. These include the standard 56 degree lens that comes together with uCAM-II, as well as the 76 degree lens and the 116 degree lens can be purchased as additional components. It operates on normal 5V DC supply and no external DRAM is required for storing the images. The uCAM-II is connected to the Arduino Due board through one of the UART interfaces at 115200 bauds.

iii. XBEE Wireless Module:

Wireless communication between the visual node and the server is performed by using a XBee module. It can send and receive data via the 2.4GHz or 900MHz band at a relatively low power. They can be used to set up a simple point-to-point link by using the transparent mode, or to form a complex self-healing network that spread over a large area when using the API mode [154]. For the development of the BCS visual node, the XBee module is configured to operate in the API mode. In this case, the visual data is enclosed in a packet before transmission takes place. The XBee module is connected to the

Arduino Due board through another UART interface. However, 125000 bauds is used because the communication between the XBee module and the Due board is not reliable at 115200 bauds given the Due's clock frequency of 84MHz [154].

7.1.2. Software Components

In this context, the software architecture is built using modular design. As shown in Figure 7.3, the platform consists of data preprocessing in the sensor side, control protocol during the transmission, and stream management in the server. We will summarize several key components in the rest of this section.

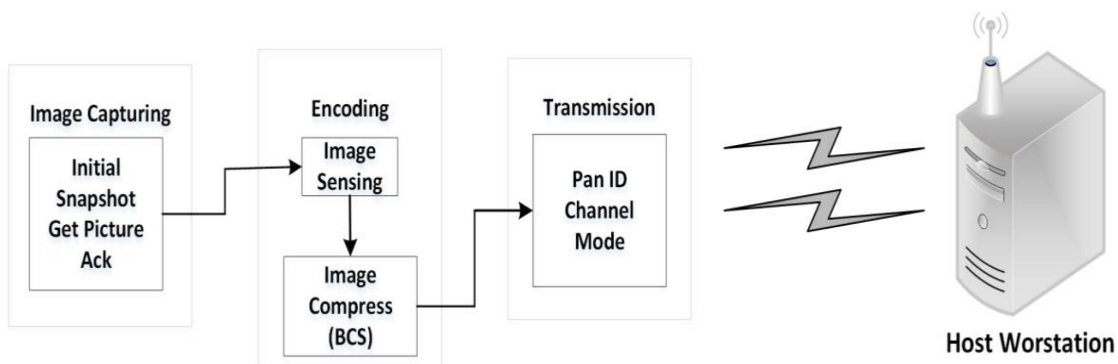


Figure 7.3: Software components associated to the visual node.

i. Image Capture

In our implementation, we capture an 8-bit gray scale RAW image and store the image data in the Arduino flash memory for further processing. As Arduino Due have a larger flash memory than SRAM, it is better first to store the large image data into flash memory using PRGMEM variable modifier and then read the data from flash memory back into SRAM using a block-by-block approach.

In order to start the communication process, a connection between the host and the uCAM-II must be established. As shown in Figure 7.4, this is started by synchronizing the host with the uCAM-II via SYNC command. The host sends the SYNC command continuously until an acknowledgement ACK and SYNC command is received from the uCAM-II. After the response is received by the host it should reply with the ACK command to confirm the synchronization process.

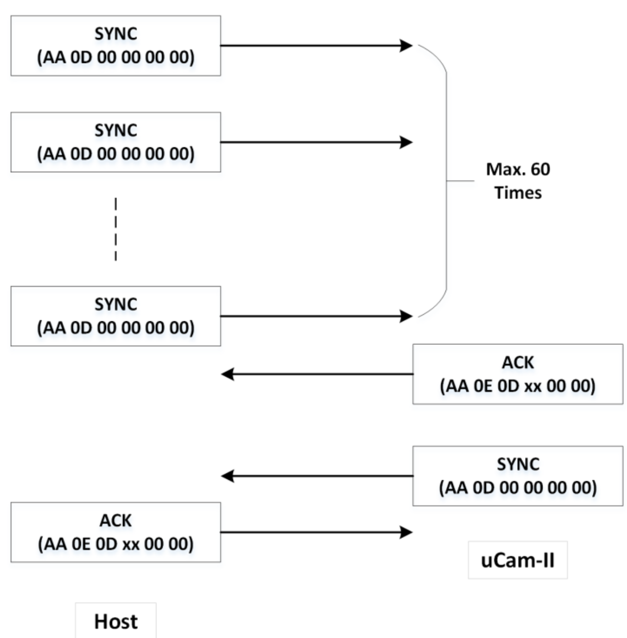


Figure 7.4: Synchronisation process between uCAM-II and host.

After the communication link is established, uCAM-II is ready to capture images. To capture a RAW image, the following commands have to be sent from the host to the uCAM-II.

a) **INITIAL** is first used to configure the image size and image format.
b) **SNAPSHOT** is to instruct uCAM-II to capture an image and store it in buffer.
c) **GET PICTURE** is used to request an image from the uCAM-II.
d) **ACK** is sent to indicate the end of the last operation.

The overall process of capturing an 8-bit grayscale RAW image with resolution of 128x128 RAW is shown in Figure 7.5. This resolution is selected because Arduino Due has limited SRAM of 96KB.

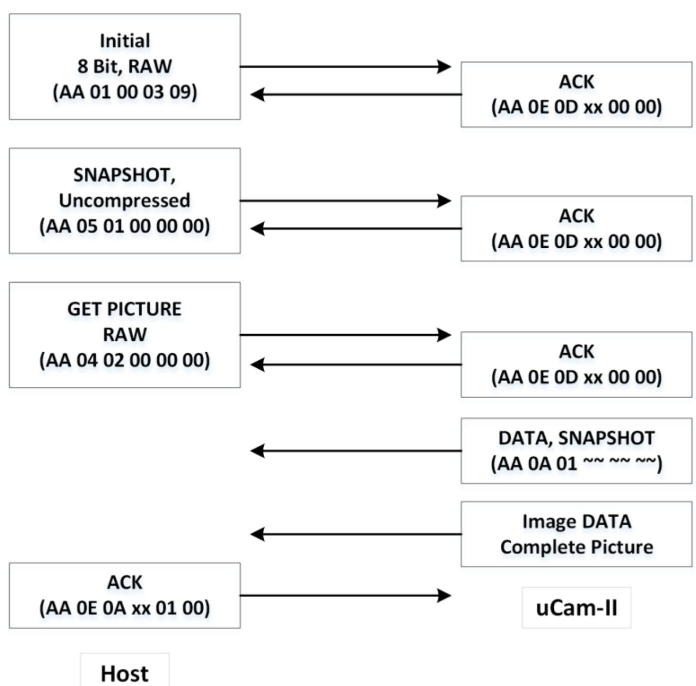


Figure 7.5: Process of capturing an 8-bit 128x128 RAW image.

ii. Encoding Process

The image obtained from uCAM-II is first stored into the flash memory. The BCS is applied to encode the image in a block-by-block basis. The encoding process can be divided into two parts, namely image sensing and image compression as shown in Figure 7.6.

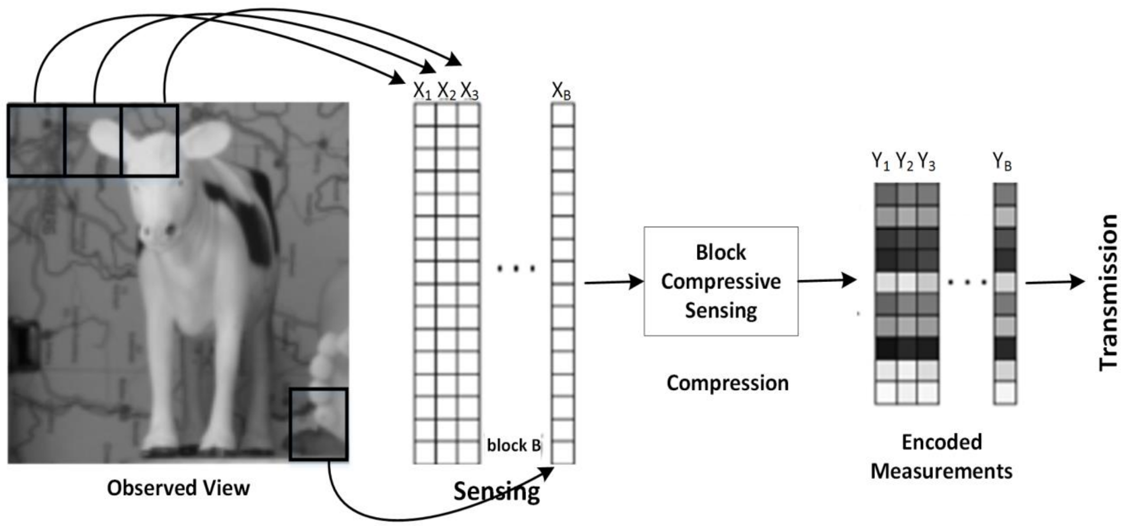


Figure 7.6: Encoding process of BCS

In the first part, the RAW image of resolution 128x128 is first divided into small 16x16 independent blocks, and each block is rearranged into a vector with 256 pixel values. This produces a matrix of size 256x64, and this is denoted as the sensed measurement, I. Next, I is sampled by random measurement matrix Φ. The measurement matrix Φ used in the proposed scheme is a constrained structure (block diagonal) matrix that is incoherent with any sparsity basis with a very high prospect. This also reduces the memory required to store the measurements when it is implemented as a dense matrix.

The size of the measurement matrix Φ is determined based on the block size and sampling rate. For example, if the block size is 16x16 and the sampling rate is 0.2, then the Φ generated is of size 51x256. Then Φ is multiplied with I to obtain the encoded measurement matrix Y. All the encoded measurement will then be transmitted to the server via the XBee module. But before transmission, the encoded measurements are quantize using uniform quantization. Each measurement value is converted to a signed 16 bit binary vector. From our analysis, the measurement value can exceed the range of -128 to +128 because the signed 8 bit binary vector is not sufficient to fit the encode value for each image measurement. The signed 16 bit binary vector was used instead.. Hence, it is not sufficient to fit the value into a signed 8 bit binary vector.

iii. Wireless Communication

Two Series-2 XBee modules are used. One is connected to the Arduino Due and the other is connected to the server. The former is configured as the end device that in charge of sending data, whereas the latter is configured as coordinator that in charge of setting up the network and receiving data. It is also necessary to ensure that they are operating under the same PAN ID and channel number. All these parameters have to be configured in prior to forming a wireless network. The API mode is used over AT mode to emulate the transmission pattern of a VSN. In API mode, the input data will be packetized into many API frames before transmit within the wireless network. The API frame structure is shown in Figure 7.7.

Start Delimiter	**Length**		**Frame data**				**Check sum**
1	**2**	**3**	**4**	**5**	**6-8**	**9…n(<100)**	**n+1**
0x7E	MSB	LSB	Frame Type	Frame ID	Source/Destination Address	Data	Single Byte

Figure 7.7: API frame structures for Xbee transmit and receive request

In every API frame, the first byte is a start delimiter that is used to indicate the beginning of each API frame. The value is always 0x7E allowing easy detection of a new incoming frame. The next field indicates the length of the frame. The length is of 16 bits value and is divided into MSB (most significant bits) and LSB (least significant bits). After the length is the frame type, frame ID, source or destination address and the payload (data). The frame type indicates how the information is organized in the data field. The frame ID is used to enable a form of acknowledgement that indicates the result of the transmission. Source or destination address is a 64-bit value that indicates either the source or the destination of the packet. The data field contains the information to be transmitted and is dependent on the frame type.

The value in each field may vary according to transmit or receive request. For transmit request the frame type, frame ID, 64-bits source or destination address values are 0x10, 0x01, 0x000000000000FFFF (destination address) respectively whereas, for receive request the values are 0x91, 0x00, 0xFFFFFFFFFFFFFFFF (source address) respectively. The last field of the API frame is the checksum that is used to test the data integrity. The checksum is calculated by first adding all the bytes in the frame excluding the start delimiter and length, then subtract the lowest 8 bits of the result from 0xFF.

The transmission process between a visual node (end device) and the server (coordinator) is shown in Figure 7.8. After coordinator has setup the network, other end devices will be able to automatically join the network. Initially, the server will broadcast a packet contains an ‘I’ character via the coordinator to all the visual nodes. This initialization step helps to determine the number of visual nodes in the network, and to know the number of images that are going to be received. This is followed by broadcasting two more packets containing character ‘C’ and ‘T’ in respective order.

The visual node is always looking for packets transmitted from the server. Once a packet is received, the visual node will process the information acquired from the packet. If the received packet contains an ‘I’, the same packet will be transmitted back to the server for acknowledgement purpose, whereas if the received packet contains ‘C’, the node will capture and encode the images using BCS. The reason of doing this is to synchronize the image capturing process of different visual nodes. This is to ensure that the images are captured at approximately the same time to ensure maximum correlation. Furthermore, this also allows the server to control when the capturing should take place.

Once a packet that contains a ‘T’ is received the visual node will packetize the encoded measurements into numbers of API frame, and each frame has a payload size of 72 bytes. All the data will be continuously transmitted to the server until there is no more data to transfer. Then, a packet that carries a value of zero is sent. The purpose of this frame is to inform the server that the previous packet was the end. The proposed Joint Multi-phase Decoding (JMD) is applied to the received encoded data at the server to recreate the captured images.

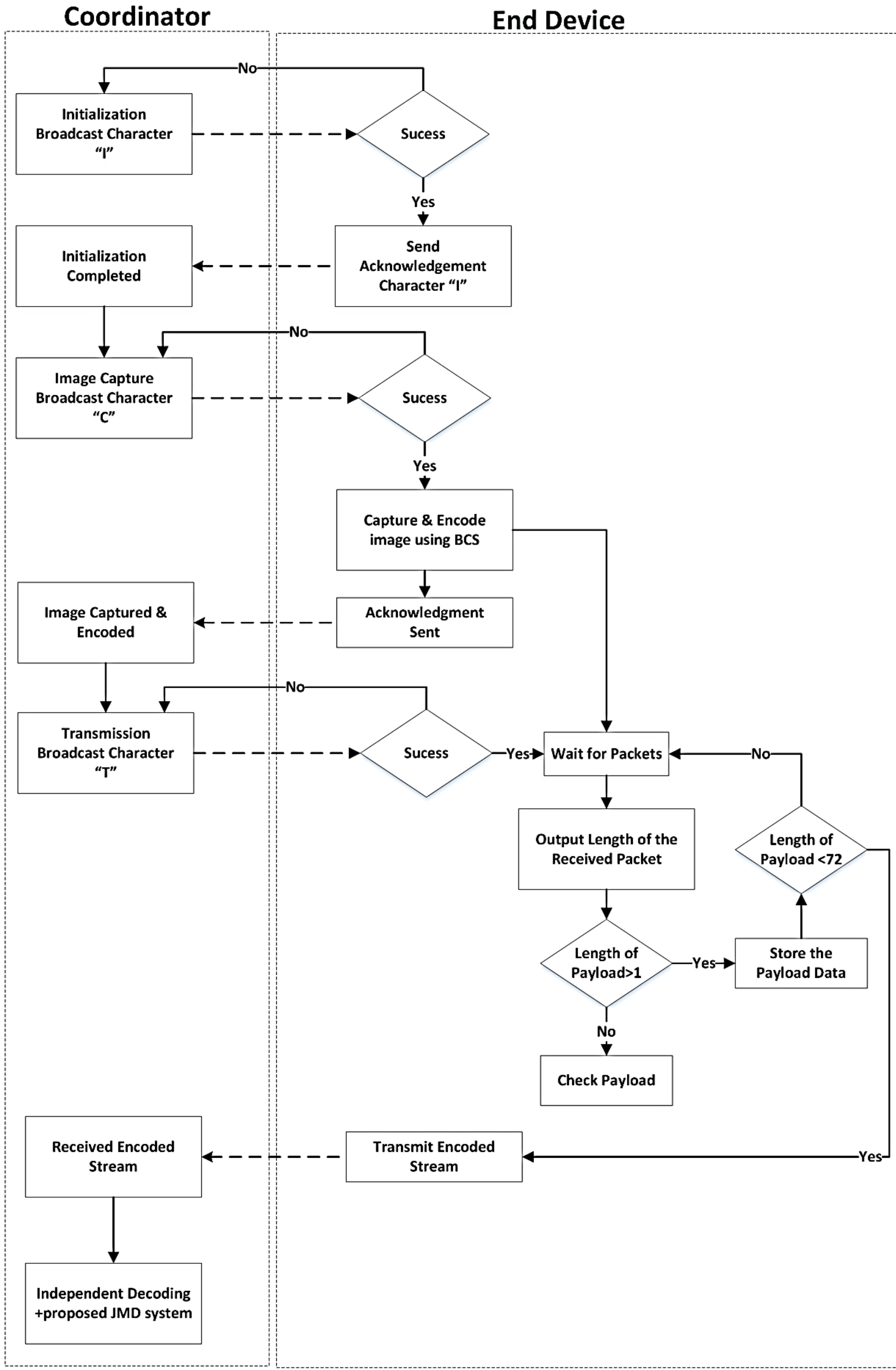


Figure 7.8: Data transmission between a visual node (end device) and the server (coordinator)

7.2. Experimental Setup

To simplify the evaluation process, two visual nodes are deployed in a horizontal setup, and each visual sensor is separated by a specific distance from its neighbor as shown in Figure 7.9. However, the setup can be extended by adding more visual nodes. In this case, the proposed compression scheme described in chapter 4 is implemented. Hence, one of the visual nodes is configured as the non-reference node and the other as reference node. The images captured are of 8-bit grayscale format with the resolution of 128x128. All the images are encoded independently using the BCS. Images captured by the non-reference node are encoded at lower subrates range from 0.05 to 0.3. The idea is to improve the images captured by the non-reference node with the help of images captured by the reference node. The encoded measurements from the two visual nodes are then transmitted using XBee module to the server for reconstruction.

The server is equipped with an Intel(R) Xeon(R) E5-1620 CPU running at 3.6 GHz and 8GB of RAM. The server is used to reconstruct the encoded measurements by using the proposed JMD. It is implemented using MATLAB ver. 8.3.0.532 (R2014a). Because the server will be receiving images from different visual nodes, it is important to differentiate the origin of the data. To achieve this, the server will refer to the source address embedded in the received packet.

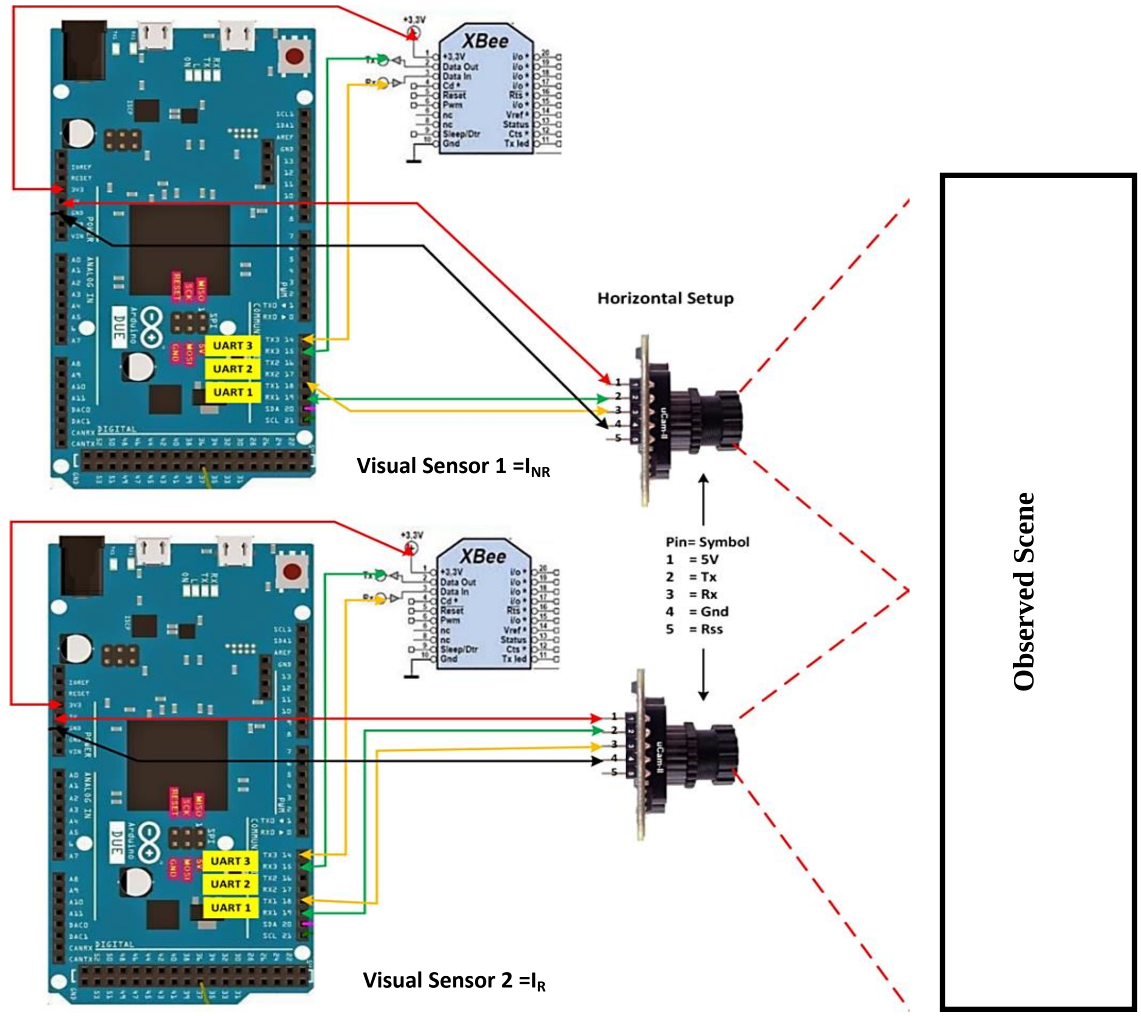


Figure 7.9: Proposed multi-visual setup

7.3. Experimental Results

The evaluation is carried out by measuring the execution time and energy consumption for capturing, encoding and transmission of visual data in Seconds (Sec) and Joules (J) respectively at various sampling rates. Moreover, to validate the effectiveness of the proposed scheme in terms of visual quality, the Peak-Signal to Noise-Ratio (PSNR (dB)) and Structural Similarity Index Metric (SSIM) are also measured. All the images captured by the non-reference node are encoded at lower subrates of 0.05, 0.1, 0.15, 0.2, 0.25, 0.3, whereas images captured by the reference node are encoded at a fixed subrate of 0.3. In addition to this, the effect of using bock size 8x8 and 16x16 for BCS is also compared.

In addition, the energy consumption is not measured for the proposed JMD system as the proposed JMD system is implemented at the receiver side. The energy consumption is measured for the encoder (battery powered node). As our main focus is to measure the performance of the BCS at encoder side, so we are measuring only the power consumption of the node, not the entire system, as it is important to evaluate how the scheme would perform under battery powered nodes (encoder), not the receiver.

Further, we have to also consider the power and time of capturing, encoding and transmission. The energy consumption at different stages is measured by taking the product of measured power and measured time (Energy = Power * Time).

7.2.1. Execution & Transmission Time Analysis

The total time required to perform the capturing, encoding and transmission of visual data is presented in Table 7.1. The image capturing time and sensing time for both block size are about the same. It is noted that the image encoded with block size 8x8 is 3-4 times faster in terms of execution time than block size 16x16. This is due to the extra bytes produced by using larger block size. Subsequently, image encoded with block size 8x8 takes 6.72% - 12.24% less transmission time than block size 16x16. However, using larger block size produces more encoded measurements in total when compared to a smaller block size.

Table 7-1: Time taken to complete the encoding and transmission at various block sizes and subrates

Sub rate	Size in Bytes	Image Capture Time (Sec)	Total Encoding Time (Sec)		Total Transmission Time (sec)	Total Encoding+ Transmission Time (sec)
			Sensing Time	Compression Time		
Block Size =8						
0.05	1183	1.41	0.011	0.111	0.460	0.582
0.10	2233	1.41	0.011	0.222	0.965	1.198
0.15	3913	1.41	0.011	0.370	1.575	2.056
0.20	5010	1.41	0.011	0.481	2.167	2.659
0.25	6243	1.41	0.011	0.592	2.649	3.252
0.30	7273	1.41	0.011	0.702	3.167	3.887
Block Size=16						
0.05	1288	1.41	0.011	0.478	0.524	1.013
0.10	2653	1.41	0.011	0.957	1.107	2.075
0.15	4276	1.41	0.011	1.398	1.661	3.070
0.20	5423	1.41	0.011	1.877	2.244	4.132
0.25	6343	1.41	0.011	2.356	2.805	5.172
0.30	7875	1.41	0.011	2.834	3.383	6.228

7.2.2. Energy Consumption Analysis

The energy consumption at different stages is measured by taking the product of measured power and measured time (Energy = Power * Time). The time required is already measured in Table 7.1. The power is assessed by measuring the current drain at each stage independently, whereas the voltage remains constant at 3.3V. The results obtained are shown in Table 7.2. The stages include standby, capturing, encoding, and transmission. In the standby stage, the visual nodes are waiting for instructions from the server. Capturing stage refers to the capturing of an image. Encoding stage is the sensing and compression of the captured image. Lastly, the transmission stage refers to the transmission of encoded bits stream from visual node to the workstation. All the measurement was done by using the Unity True RMS Multi-meter. All the power values are presented in Watt (W).

Table 7-2: Power consumed at different stages.

Operating stages	Voltage (V)	Total Current (mA)	Average Current (mA)	Average Power VxI (W)
Standby	3.3	107.5-108.2	107.80	0.350
Image Capture	5.0	80.17-85.10	82.63	0.410
Encoding	3.3	15.50-15.90	15.70	0.0521
Transmission	3.3	37.7-37.9	37.8	0.1221
Standby + Encoding	3.3	122.9-124.1	123	0.407
Standby + Encoding +Transmission	3.3	159.8-160.4	160.1	0.528

The results show that the power required for encoding of an image is 0.05 watt that is 52.2% - 62.4% less than the power required for transmission of the encoded bit stream, which is 0.122 watt. The power difference between encoding and transmission is recoded by first calculating the average power values for both and then applying the following formula:

$$\text{Percentage Difference} = \frac{|E1 - E2|}{(E1 + E2)/2} \times 100 \qquad \textbf{(7.1)}$$

Where,

E1= first value, E2= second value

Moreover, the power consumption during standby is 0.35W and the total power consumption with encoding and transmission is 0.52W.

Table 7.3 presents the energy consumed at different stages using block size 8x8 and 16x16 for various subrates. The results show that the energy consumed during encoding when using block size 8x8 is 2-3 times less than block size 16x16. Subsequently, the transmission also consumed 8.4% - 13.4% less energy. The transmission energy difference between both the 8 and 16 size blocks at each subrate (0.05-0.3) is calculated by using eq. (7.1).

Table 7-3: Energy consumption using block size 8x8 and 16x16 at various subrates

Sub rate	Idle State (J)	Image Capture (J)	Encoding (J)	Transmission (J)	Total Encoding + Transmission (J)
			Block Size 8		
0.05	1.43	0.58	0.006	0.056	0.062
0.10	1.43	0.58	0.012	0.118	0.130
0.15	1.43	0.58	0.019	0.205	0.224
0.20	1.43	0.58	0.025	0.265	0.290
0.25	1.43	0.58	0.031	0.323	0.354
0.30	1.43	0.58	0.036	0.386	0.424
			Block Size 16		
0.05	1.43	0.58	0.025	0.064	0.089
0.10	1.43	0.58	0.050	0.135	0.185
0.15	1.43	0.58	0.072	0.203	0.275
0.20	1.43	0.58	0.097	0.274	0.371
0.25	1.43	0.58	0.122	0.342	0.464
0.30	1.43	0.58	0.147	0.418	0.560

The energy required for encoding is 40%-60% less than the energy required for transmission. This validates the statement in [20] that transmission of data requires more energy when compared to processing. It should also be noted that the energy consumed by the visual node when in standby is 1.43J. Standby energy is the energy consumed by Arduino Due once the camera is out into sleep mode (happen after 4.1s of being idle). We calculated this energy consumption by first calculating the power and time of standby state.

At standby state:

V = 3.3V,

I = 107.80 mA,

P = V x I = 0.35 watt

Time = 4.1sec required by Arduino Due to wake from idle state

E = P x T =0.35x4 =1.43J

The standby energy is 1.43J, which is a bit on the higher side. However, the value presented is not the exact appearance of the energy consumption as it should be noted that power saving strategy such as putting the micro-controller in deep sleep mode or lower frequency, or performing ADC reduction, or powering off the radio module was not used. It is expected that the idle state consumption can be further reduced to a greater extent by applying all these power management strategies.

7.2.3. Visual Quality Analysis

The visual nodes are placed horizontally aligned side by side, and each visual node is separated by a specific distance from its neighbour as shown in Figure 7.10. One of the visual node is configured as the non-reference node and the other is configured as the reference node. The observed scenes are shown in Table 7.4.

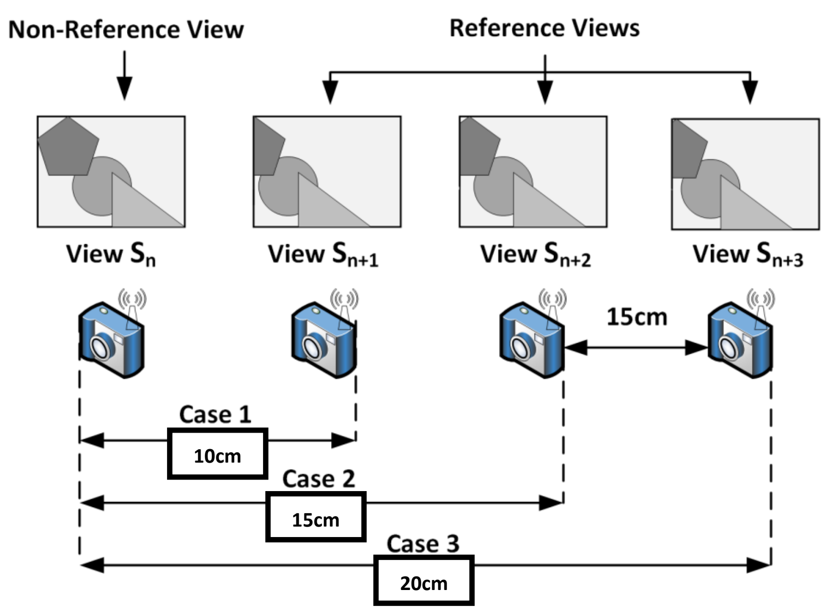


Figure 7.10: Different separation setups

Table 7-4: Sample images captured by the visual nodes.

	Non-Reference Image	Reference Image @ 10cm Separation	Reference Image @ 15cm Separation	Reference Image @ 20cm Separation
Building				
Books				
Park				

The results of comparing the proposed scheme (JMD-TV) with independent BCS (BCS-TV-AL3) are presented in Table 7.5. For smaller separations (**10cm**) the proposed JMD-TV provides an average gain of 1.5dB to 2.5dB, whereas for larger separation (**20 cm**) the gain reduces to an average of 1dB to 2dB when moving from higher to lower subrates. Generally, the proposed scheme produces poor results if the camera separation is too large. As the distance separation increases, the correlation between them is reducing, leading to less accurate registration and fusion of the images. Furthermore, it is also noticed that larger block size generates 0.5dB to 0.8db better reconstruction than the smaller block size.

Table 7-5: Performance (PSNR (dB)) comparison of using the proposed compression scheme with different camera separations, block sizes, and subrates

			Building					
	Reference Views	**Subrate**	**0.05**	**0.1**	**0.15**	**0.2**	**0.25**	**0.3**
Block Size 8		**BCS-TV-AL3**	19.93	21.23	23.30	24.85	25.42	26.43
	$S_{n\pm1}$ =10cm	**JMD-TV**	**22.34**	**23.56**	**25.56**	**26.95**	**27.47**	**28.32**
	S_{n+2}=15cm	**JMD-TV**	21.75	22.95	24.98	26.41	26.89	27.74
	S_{n+3}=20cm	**JMD-TV**	21.18	22.45	24.41	25.79	26.28	27.22
Block Size 16		**BCS-TV-AL3**	21.24	22.97	23.99	25.02	26.02	26.69
	$S_{n\pm1}$ =10cm	**JMD-TV**	**23.76**	**25.27**	**26.11**	**27.12**	**27.99**	**28.52**
	S_{n+2}=15cm	**JMD-TV**	23.12	24.70	25.65	26.54	27.45	27.98
	S_{n+3}=20cm	**JMD-TV**	22.64	24.20	25.08	25.95	26.91	27.44
			Park					
	Reference Views	**Subrate**	**0.05**	**0.1**	**0.15**	**0.2**	**0.25**	**0.3**
Block Size 8		**BCS-TV-AL3**	17.84	19.49	20.66	22.13	22.76	23.92
	$S_{n\pm1}$ =10cm	**JMD-TV**	**20.67**	**22.24**	**23.27**	**24.43**	**25.00**	**25.98**
	S_{n+2}=15cm	**JMD-TV**	20.12	21.67	22.75	23.97	24.51	25.47
	S_{n+3}=20cm	**JMD-TV**	19.71	21.11	22.21	23.52	23.97	24.93
Block Size 16		**BCS-TV-AL3**	18.59	20.12	21.04	22.07	22.99	24.02
	$S_{n\pm1}$ =10cm	**JMD-TV**	**21.75**	**23.27**	**23.78**	**24.68**	**25.5**	**26.31**
	S_{n+2}=15cm	**JMD-TV**	21.22	22.70	23.17	24.05	24.95	25.83
	S_{n+3}=20cm	**JMD-TV**	20.66	22.12	22.63	23.54	24.44	25.31
			Book					
	Reference Views	**Subrate**	**0.05**	**0.1**	**0.15**	**0.2**	**0.25**	**0.3**
Block Size 8		**BCS-TV-AL3**	18.41	20.99	22.84	24.75	25.61	26.54
	$S_{n\pm1}$ =10cm	**JMD-TV**	**21.09**	**23.73**	**25.44**	**27.07**	**27.87**	**28.50**
	S_{n+2}=15cm	**JMD-TV**	20.82	23.36	25.14	26.89	27.56	28.22
	S_{n+3}=20cm	**JMD-TV**	20.15	22.67	24.45	26.33	27.05	27.88
Block Size 16		**BCS-TV-AL3**	19.58	22.31	23.72	25.32	26.66	27.66
	$S_{n\pm1}$ =10cm	**JMD-TV**	**22.53**	**25.20**	**26.38**	**27.81**	**28.78**	**29.61**
	S_{n+2}=15cm	**JMD-TV**	22.25	24.88	26.09	27.35	28.55	29.33
	S_{n+3}=20cm	**JMD-TV**	21.98	24.49	25.78	26.98	28.17	29.10

***Note*:** The bold values relates to the maximum PSNR (dB) reached for a given subrate and image

Some samples of the reconstructed images are shown in Figure 7.11. By comparing the highlighted regions (white dotted boxes), it is noticed that the proposed JMD-TV reduces the blurring effect presents in the images reconstructed using BSC-TV-AL3 and the reconstructed image looks much sharper.

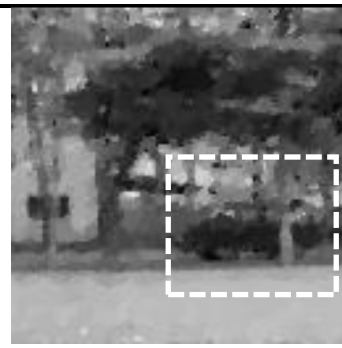
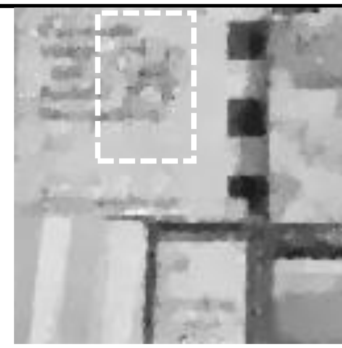
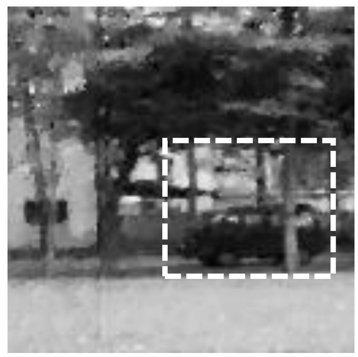
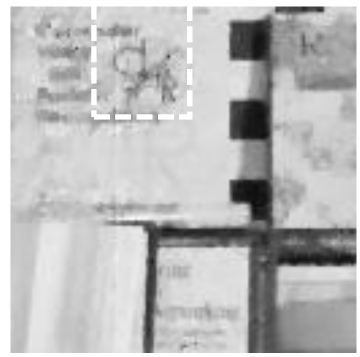
	Building	**Park**	**Book**
BCS-TV-AL3	 Building @ 0.2, PSNR=25.02dB, SSIM=0.72	 Park @ 0.2, PSNR=22.08dB, SSIM=0.65	 Books @ 0.2, PSNR=25.32dB, SSIM=0.78
JMD-TV	 Building @ 0.2, PSNR=26.54dB, SSIM=0.81	 Park @ 0.2, PSNR=24.08dB, SSIM=0.77	 Books @ 0.2, PSNR=27.35dB, SSIM=0.84

Figure 7.11: Visual quality comparison of images encoded using independent BCS-TV-AL3 and proposed JMD-TV at subrates 0.2, block size 16x16, and visual nodes separation of 15cm

7.2.4. Complexity and Energy Consumption Comparison

The computational complexity and energy consumption of using BCS with different block sizes is compared with the case no compression (RAW) and the case of using JPEG compression (JPEG). In each case, the time and energy taken to encode and transmit an image is measured. When BCS is applied to encode the images, block size (B) of 8x8 and 16x16 are evaluated. In both situations, subrate (M) of 0.3 is used.

Table 7-6: Comparison of computational complexity and energy consumption with and without using compression

Size of Raw Image= 128x128						
Image Type	**Encoding Time (Sec)**	**Encoding Power (W)**	**Encoding Energy(J)**	**Transmission Time (Sec)**	**Transmission Power (W)**	**Transmission Energy (J)**
RAW	-	-	-	8.20	0.122	1.004
JPEG	3.015	0.052	0.156	6.39	0.122	0.781
BCS B = 8x8 M = 0.3	**0.713**	0.052	**0.037**	**3.16**	0.122	**0.385**
BCS B = 16x16 M =0.3	2.845	0.052	0.105	3.38	0.122	0.412

The results show that the transmission of image without compression requires more time and energy. It can be noted that JPEG compression consumes 30% less energy and BCS consumes 50%-60% less energy than the case of uncompressed RAW image. At subrate of 0.3, BCS with block size 8x8 and 16x16 require 60% and 10% less encoding time respectively when compared to JPEG compression. In terms of transmission time and energy consumption, the BCS outperforms JPEG by a margin of 30%-40%.

7.4. Conclusive Remarks

A visual node prototype has been developed to evaluate the proposed scheme. BCS is implemented on the visual node to encode the captured image before transmission. The evaluations show that the energy taken to transmit an image is 50% higher than that of compressing the image. Hence, it is wise to compress the image before transmission takes place. When compared with the case of no compression and when JPEG is used to compress the captured image, the total energy consumption (encoding + transmission) is 40% to 60% lower when block size of 8x8 is used whereas for block size 16x16 the energy consumed by the proposed scheme is 10% to 20% lower.

Chapter 8
CONCLUSION AND FUTURE WORK

8.1. Thesis Achievements

In this thesis, a single-view and multi-view visual compression scheme that focuses on improving the reconstruction of measurements encoded using Compressive Sensing (CS) for Visual Sensor Network (VSN) is presented. The encoder must rely only on CS data acquisition techniques; no conventional scheme should be required. It is anticipated that using CS for multi-view visual compression will reduce the power consumption, memory utilization as well as processing time. The main focus of the research is on the joint reconstruction of CS encoded single and multi-view visual data by effectively exploiting the correlation among the multiple views, since it is the most crucial components that affect the quality of the recovered visual data. Moreover, the recovery of the encoded data from such small measurements is not directly possible. The correlation is mainly driven by the displacement of objects. Overall the research aims to develop a low-complexity single-view and multi-view Visual (image and video) compression scheme that gives better performance at low bitrate. The key highlights of the thesis are as follow:

First, a single and multi-view visual compression based on BCS and joint-decoding is proposed. We describe how the multi-view images, single-view video, and multi-view videos can be compressed by using the proposed compression scheme. As opposed to conventional compression scheme, the use of CS creates a simple-encoder complex-decoder paradigm that is more suitable for the VSN. On the one hand, the visual nodes, which serve as the encoders, are only required to quantize and transmit the measurements produced by CS. On the other hand, the server, which acts as a joint-decoder, will perform the complex task of exploiting the correlations and redundancies of information collected by different visual nodes. This reduces the amount of processing to be done on the encoder as well as the energy consumption. In the proposed scheme, certain visual nodes are configured to encode and transfer the information (I_{NR}) at subrates lower than others. Image registration and fusion are used to generate projected image (I_P) that closely resembles I_{NR}. This procedure is approximately 2-3 times shorter than the use of Motion Estimation and Compensation. The difference between I_P and I_{NR} at the measurements level is determined, and the difference is added to I_P to produce the final reconstructed output.

Experimental results show that the proposed scheme is approximately 30% - 40% shorter than the use of Motion Estimation and Compensation schemes. In addition to this, it outperforms other compression schemes at lower subrates by up to 2dB - 3dB when it was applied to the case of multi-view images. Furthermore, when applied to the case of single and multi-view video, the proposed scheme shows a gain of 1.5dB - 3dB in terms of reconstruction quality and takes approximately 2-3 times shorter interval for execution than the other single and multi-view CS-based video compression schemes.

In addition to this, a quantization approach is also proposed to transform the CS measurements produced by the sensor nodes into bits. This allows us to compare the proposed BCS based JMD with other conventional video compression scheme. The simulation results shows that the proposed scheme show better results at lower bitrate when compared with conventional video compression schemes

Next, a visual node prototype has been developed to evaluate the proposed scheme. BCS is implemented on the visual node to encode the captured image before transmission. The evaluations show that the energy taken to transmit an image is 50% higher than that of compressing the image. Hence, it is wise to compress the image before transmission takes place. When compared with the case of no compression and when JPEG is used to compress the captured image, the total energy consumption (encoding + transmission) is 40% to 60% lower when block size of 8x8 is used whereas for block size 16x16 the energy consumed by the proposed scheme is 10% to 20% lower.

Finally, a low-complexity symmetric encryption algorithm based on Feistel structure, denoted as Secure Force (SF), is proposed. The encryption process is implemented using a simple architecture that only consists of basic mathematical operations. The more complex key expansion process is shifted to the server to reduce the burden on the encoder. SF is implemented and evaluated using Field Programmable Gate Array (FPGA). The evaluations for security parameters show that the proposed algorithm shows 15% better results for avalanche effect when compared with AES algorithm. While, for image histogram test the images encrypted by using the proposed algorithm shows linear histogram that is considered good. In terms of area utilization efficiency the proposed algorithm requires 40% - 50% less area for implementation as compared to different AES implementations. Moreover, the power consumed by SF implementation outperforms the AES implementation results by a margin of 1.63/293.49/269.59mW in static, dynamic and total power consumption respectively.

8.2. Future Directions

This thesis considers the different correlational and representational challenges related to the processing of visual information collected by multiple sensors. It should be noted that to provide a better representation of a scene, efficient handling of correlation that exist between the images is important. We have developed a low complexity single-view and multi-view compression scheme that focuses on improving the reconstruction of measurements encoded using Compressive Sensing (CS) by exploiting the correlations. The scheme presented in this thesis opens new exciting directions for further research.

1- Development of a framework for estimating the correlation between multiple correlated images given in the form of linear measurements without implementing explicit image reconstruction steps. The correlation can be estimated in the compressed domain by jointly processing the linear measurements. The framework must be able to handle the geometric correlation in the case of different camera angles, larger camera separations, and movements.

2- Development of efficient correlation estimation algorithms and joint representation of multiple correlated images captured by different sensing methodologies, e.g., planar, omnidirectional and compressive sensing (CS) sensors. For each sensor type the geometry of the 2D visual representation and the acquisition complexity vary. Therefore, the specific geometric nature of the captured images needs to be considered while developing distributed representation algorithms.

3- Designing of an efficient quantization scheme that helps towards the improvement of rate-distortion performance of quantized CS measurements which is also an open research problem. Quantization is one of the important elements of the encoding process and it should be designed in accordance with the concerned signal. For CS based schemes, the process is quite different from the conventional compression processes as it generates a set of linear measurements by using a sensing matrix rather than applying transformation on the image. Moreover, as discussed earlier the number of measurements obtained from CS are much small than original signal. In order to quantize such small measurements, a direct solution is to only apply scalar quantization (SQ) to each of the CS measurements obtained. However, from analysis it is observed that such quantization solution is highly inefficient in terms of rate-distortion performance as compared to traditional coding schemes

4- The development of CS based camera is hard and still under research and development. Many researchers have focused their work on the development of CS based camera, but no actual implementation of CS cameras for VSN is performed. Moreover, in literature to author's knowledge till date the practical evaluation of CS based on energy consumption, memory utilization, execution time and visual quality is not presented.

References

1- E. Yoneki, J. Bacon, 2005. “A Survey of Wireless Sensor Network Technologies: Research trends and Middleware's Role”. Technical Report UCAM-CL-TR646, University of Cambridge.2005.

2- I.F Akyildiz, W.Su*, Y. S.Subramaniam, E.Cayirci, 2002. “Wireless sensor networks: a survey”. International journal of computer networks, Vol.38, pages 29.2002.

3- N. Ahmad, K. Khursheed, M. Imran, N. Lawal, and M. O'Nils. 2013. “Modeling and verification of a heterogeneous sky surveillance visual sensor network”. Int. Journal of Distributed Sensor Networks, Vol. 2013, Article ID 490489 (2013), 11 pages. DOI:10.1155/2013/490489

4- S. Soro and W. Heinzelman, “A survey of visual sensor networks,” Advances in Multimedia, vol. 2009, Article ID 640386, 21 pages, 2009.

5- Y.Gao, K.Wu, F. Li, 2003. “Analysis on the redundancy of wireless sensor networks,” in Proceedings of WSNA’03.

6- J.Yick, B.Mukherjee, D.Ghosal, 2008. “Wireless sensor network survey”, Computer Networks 52, 2292–2330.

7- R. I. Hartley and A. Zisserman, 2004. “Multiple View Geometry in Computer Vision”. Cambridge University Press, 2004.

8- Y. Ma, S. Soatto, J. Kosecka, and S. Sastry, 2003. “An Invitation to 3-D Vision: From Images to Geometric Models”. Springer, 2003.

9- M.A. Razzaque, C. Bleakley, and S. Dobson. 2013. Compression in wireless sensor networks: A survey and comparative evaluation, ACM Transactions on Sensor Networks (TOSN), Vol.10, No.1 (Nov. 2013), 44 pages. DOI: 10.1145/2528948

10- M. Flierl and B. Girod, 2007. “Multiview video compression,” IEEE Signal Processing Magazine, vol. 24, no. 6, pp. 66–76, 2007.

11- P. Merkle, A. Smolic, K. Muller, and T. Wiegand, 2007. “Efficient prediction structures for multi-view video coding,” IEEE Transactions on Circuits and Systems for Video Technology, vol. 17, no. 11, pp. 1461–1473, 2007.

12- S. A. Hussain, M. I.Razzak, A. A. Minhas, M.Sher, G. R. Tahir, 2009. "Energy Efficient Image Compression in Wireless Sensor Networks," in International Journal of Recent Trends in Engineering, Vol. 2, No. 1.2009

13- M. Ebrahim, C.W. Chong. 2014. “A comprehensive review of distributed coding algorithms for visual sensor network (VSN)”, International Journal of Communication Networks and Information Security (IJCNIS), Vol. 6, No. 2 (August 2014), 13 pages.

14- M. F. Duarte, M. A. Davenport, D. Takhar, J. N. Laska, T. Sun, K. F. Kelly, and R. G. Baraniuk. 2008. “Single pixel imaging via compressive sampling”. IEEE Signal Process Magazine, Vol. 25, No. 2 (March 2008), 9 pages. DOI: 10.1109/MSP.2007.914730

15- M. Lustig, D. L. Donoho, J. M. Santos, and J. M. Pauly. 2008. “Compressed Sensing MRI”. IEEE signal processing magazine. Vol 25, no.2, pp 72-82, 2008.

16- M. Lustig, D. L. Donoho, J. M. Santos, and J. M. Pauly.2007. “Sparse MRI: The Application of compressed sensing in Rapid MRI imaging”. Magzine Reson Med. Vol58, no. 6, pp.1182-95. 2007

17- W.tang, J. Ma, F.J. Herrmann,” Optimized CS for curve-let based seismic data reconstruction”

18- D. Donoho. 2006. Compressed sensing. IEEE Transaction on Information Theory, Vol.52, No.4 (April 2006), 17 Pages. DOI: 10.1109/TIT.2006.871582

19- E. Candes, and M.B. Wakin. 2008. “An introduction to compressive sampling”. IEEE Signal Processing Magazine, Vol.25, No. 2 (March 2008), 10 pages.

20- V. Raghunathan, C. Schurghers, S. Park, and M. Srivastava. (2002) “Energy-aware wireless microsensor networks”. IEEE Signal Processing Magazine. 40-50.

21- G. J. Pottie and W. J. Kaiser, "Wireless integrated network sensors," Communications of the ACM 43, vol. 5, pp. 551-558, 2000

22- A. Subramanya, 2001. “Image Compression Technique,” Potentials IEEE, Vol. 20, Issue 1, pp 19-23.

23- Y. Sulema, S.E. Kahou, 2010. “Image compression: Comparative analysis of basic algorithms”, Design & Test Symposium, East-West - EWDTS.
24- A. Mammeri, B. Hadjou, A. Khoumsi, “A Survey of Image Compression Algorithms for Visual Sensor Networks”, ISRN Sensor Networks, Article ID 760320, 19 pages, 2012.
25- O.Ghorbel, W. Ayedi, M.W. Jmal, 2012. “Dct & Dwt Images Compression Algorithms in Wireless Sensors Networks: Comparative Study and Performance Analysis”, International Journal of Wireless & Mobile Networks (IJWMN) Vol. 4, No. 6.
26- C. B. Margi, K.Obraczka, 2006. “Energy consumption tradeoffs in visual sensor networks,” in Proceedings of the 24th Brazilian Symposium on Computer Networks, Curitiba, Brazil.
27- J. Oliver, M. P. Malumbres,.2001. “An Implementation of the EZW. Algorithm,” Universidad Jaume I (SNRFAI 2001), pp. 37-42.
28- M.Akter, M. B. I. Reaz, M.Yasin, F.Choong. 2008. “A modified-set partitioning in hierarchical trees algorithm for real-time image compression,” Journal of Communications Technology and Electronics, vol. 53, no. 6, pp. 642–650.
29- C. C Chang, S.G. Chen, J.C.Chiang, 2007. “Efficient encoder design for JPEG2000 EBCOT context formation,” in Proceedings of the15th European Signal Processing Conference (EUSIPCO ’07), Poznan, Poland.
30- G Boopathy, S.Arockyasamy. (2010). "Implementation of Vector Quantization for Image Compression", Global Journal of Computer science & Technology, Vol. 10.
31- N.A. Koli, M.S.Ali, 2008, “A survey on the Fractal Image Compression Key Issues”. Information Technology Journal. Vol. 7(8), pp. 1085--1095.
32- Z.Xiong, A.D. Liveris, S. Cheng, 2004. “Distributed Source Coding for Sensor Networks Compression via channel coding”, IEEE Signal Processing Magazine.
33- F.Oldewurtel , M.Foks, P. Mähönen, 2008. “On a Practical Distributed Source Coding Scheme for Wireless Sensor Networks”. VTC Spring 08, pp.228-232.
34- C. Han, L. Sun, J. Guo, “A Coding and Post-processing Framework of Multiview Distributed Video for Wireless Video Sensor Networks,” International Journal of Distributed Sensor Networks, Article ID 209318, 10 pages, doi:10.1155/2013/209318, 2013.
35- W.Song, “Strategies and Techniques for Data Compression in Wireless Sensor Networks”, TELKOMNIKA Indonesian Journal of Electrical Engineering, 11 pages, (2013).
36- N. Gehrig, “Distributed Source Coding of Multi-View Images”, PhD Thesis, Communications and Signal Processing Group, Depart. of Electrical and Electronic Engg., Imperial College, University of London, 2007.
37- Z. Liveris, A.Dand Xiong and C. Georghiades, “Compression of binary sources with side information at the decoder using LDPC codes,” IEEE Communication Letter, vol. 6(1), pp. 440–442, 2002.
38- X. Artigas, J. Ascenso, M. Dalai, S. Klomp, D. Kubasov, and M. Ouaret, “The DISCOVER codec: Architecture, techniques and evaluation,” in Proc. Picture Coding Symposium, 2007.
39- R. Puri, A. Majumdar, and K. Ramachandran, “PRISM: A video coding paradigm with motion estimation at the decoder,” IEEE Trans. on Image Processing, vol. 16, no. 10, pp. 2436–2448, 2007.
40- Z. Xiong, A. D. Liveris, and S. Cheng, “Distributed source coding for sensor networks,” IEEE Signal Processing Magazine, vol. 21, pp. 80–94, 2004.
41- S. S. Pradhan, J. Kusama, and K. Ramachandran, “Distributed compression in a dense microsensor network,” IEEE Signal Processing Magazine, vol. 19, no. 2, pp. 51–60, 2002.
42- D. Slepian and J. K. Wolf, “Noiseless coding of correlated information sources,” IEEE Transactions on Information Theory, vol. 19, pp. 471 – 480, 1973.
43- D.Schonberg, R. Kunnun, P. Sandeep, “Distributed Code Constructions for the Entire Slepian-Wolf Rate Region for Arbitrarily Correlated Sources”, Data Compression Conference-DCC, pp. 292 – 301, 2004.
44- M.Grangetto, E.Magli, G. Olmo, “Distributed Arithmetic Coding for the Slepian-Wolf problem”, IEEE Transactions on Signal Processing - TSP , vol. 57, no. 6, pp. 2245-2257, 2009.
45- J. Chen, Da-Ke, H. En-Hui-H, “On the linear codebook-level duality between slepian–wolf coding and channel coding. Information theory and applications workshop”, pp. 84 – 93, 2007.
46- S. Shenvi, S. Jaggi, B.K. Dey, M. Langberg, “Real Slepian-Wolf Codes”, ISIT, Toronto, Canada, 2008.

47- N. Gehrig, P.L. Dragotti, “Symmetric and a-symmetric Slepian-Wolf codes with systematic and non-systematic linear codes”, IEEE Communications Letters, vol. 9 (1), pp. 61-63, 2005.
48- W. Liu, L. Dong, W. Zeng, “Power-efficient rate allocation for Slepian-Wolf coding over wireless sensor networks”. ICASSP, pp. 2713-2716, 2008.
49- Z. Huang, J. Zheng, “A Slepian-Wolf coding based energy-efficient clustering algorithm for data aggregation in wireless sensor networks”, ICC, pp. 198-202, 2012.
50- P. Wang, C. Li, Z. Jun, “Distributed Data Aggregation Using Clustered Slepian-Wolf Coding, Wireless Sensor Networks”, Communications, IEEE International Conference-ICC, pp. 3616 – 3622, 2007.
51- A. Wyner, J. Ziv, “The rate-distortion function for source coding with side information at the decoder”, IEEE Trans. Inform. Theory, vol. 22, no. 1, 1–10, 1976.
52- M. Gastpar, “On Wyner-Ziv Networks, Signals, Systems and Computers” , Conference Record of the 37th Alomar, (2003). Volume 1, pp. 855 – 859, 2004.
53- W. Liu, L, Dong, W. Zeng, “Wyner-Ziv Video Coding with Multi-resolution Motion Refinement: Theoretical Analysis and Practical Significance”, SPIE Digital Library, 11 pages, 2008.
54- F. Pereira, C. Brites, J. Ascenso, M. Tagliasacchi, Wyner-Ziv video coding: A review of the early architectures and further developments-ICME, pp. 625-628, 2008.
55- J.J. Ahmad, H.A. Khan, S.A. Khayam, Energy Efficient Video Compression for Wireless Sensor Network, Information Sciences and Systems,-CISS, pp. 629 - 634, 2009.
56- C. Ling, S. Gao, J.C. Belfiore, “Wyner-Ziv Coding Based on Multidimensional Nested Lattices”, IEEE Transactions on Communications, pp.1328-1335, 2012.
57- A. Aaron, B. Girod, “Wyner-Ziv video coding with low-encoder complexity”, Proc. Picture Coding Symposium-PCS, San Francisco, CA, Dec. Invited paper, 2004.
58- X. Guo, Y. Lu, F. Wu, Zhao, D.W.Gao, “Wyner-Ziv-based multi-view video coding”, IEEE transaction on Circuits and Systems for Video Technology, vol. 18, no. 6, pp. 713-724, 2008.
59- R.G. Baraniuk, V. Cevher, M.F. Duarte, C. Hegde, “Model-based compressive sensing”. IEEE Transactions on Information Theory, 56(4), pp. 1982-2001, 2010.
60- Y. Weiss, H.S. Chang, W.T. Freeman, “Learning Compressed Sensing”, Hebrew University of Jerusalem, Allerton. 2007.
61- M. Balouchestani, K. Raahemifar, S.Krishnan, “Compressed Sensing In Wireless Sensor Networks: Survey”, Canadian Journal on Multimedia and Wireless Networks, Vol. 2, No. 1, 2011.
62- H. Zheng, S. Xiao, X. Wang, X. Tian, M. Guizani, “Capacity & Delay Analysis for Data Gathering with Compressive Sensing in WSNs”, IEEE Transactions on Wireless Communications, 12(2), pp. 917-927, 2013.
63- C. Luo, F. Wu, J.Sun, C.W. Chen, “Compressive Data Gathering for Large-Scale Wireless Sensor Networks”, in Proceedings of ACM Mobicom'09, Association for Computing Machinery, Inc. (2009).
64- C.T. Chou, Rana, R. Kumar, H. Wen, “Energy efficient information collection in wireless sensor networks using adaptive compressive sensing”, pp. 443 – 450, 2009.
65- C. Karakus, A.C.Gurbuz, B. Tavli, "Analysis of Energy Efficiency of Compressive Sensing in Wireless Sensor Networks", Sensors Journal, IEEE, pp. 1999 - 2008 Volume: 13, Issue: 5, 2013.
66- J. Hao, F. Tosato, R.J. Piechocki, “Sequential Compressive Sensing in Wireless Sensor Networks”. VTC Springer, pp.1-5, 2012.
67- S. Fulani, K.R. Rao, “Compressive Sensing of Image and Application to Communication Channel Coding”, Multimedia Processing lab, Department of Electrical Engineering, University of Texas at Arlington, 2007.
68- E. Candes, J. Romberg, “Sparsity and incoherence in compressive sampling”, Inverse Prob., vol. 23, no. 3, pp.969–985, 2007.i
69- X. Chen, P. Frossard, “Joint reconstruction of compressed multi-view images”. In: Proceedings of the international conference on acoustics, speech, and signal processing, pp. 1005–1008. Taipei, Taiwan, 2009.
70- X. Li, Z. Wei, L.Xiao, “Compressed sensing joint reconstruction for multi-view images”. Electronic Letter, 46(23), pp.1548–1550, 2010.

71- M.B. Wakin, “A manifold lifting algorithm for multi-view compressive imaging”. In: Proceedings of the picture coding symposium. Chicago, IL, 2009.
72- JY. Park, MB. Wakin, “A geometric approach to multi-view compressive imaging”. EURASIP, 2012:37. Doi: 10.1186/1687-6180-2012-37, 2012.
73- W. Lu and N. Vaswani. 2009. Modified compressive sensing for real-time dynamic MR imaging. In Proceedings of the International Conference on Image Processing (ICIP’09). IEEE, Cairo, Egypt, 3045-3048, DOI: 10.1109/ICIP.2009.5414208.
74- H. Jung, K. Sung,K.S. Nayak, E.Y. Kim, and J. C. Ye. 2009. k-t FOCUSS: a general compressed sensing framework for high resolution dynamic MRI. Magn. Reson. Med., vol. 61, No.1 (Jan 2009), 14 pages. DOI: 10.1002/mrm.21757
75- H. Jung, and J. C. Ye. 2010. Motion estimated and compensated compressed sensing dynamic magnetic resonance imaging: what we can learn from video compression techniques. Int. Journal of Imaging Systems and Technology, Wiley, Vol. 20, No.2 (May 2010), 18pages. DOI: 10.1002/ima.20231
76- M. Trocan, T. Maugey, E. W. Tramel, J.E. Fowler, and P. Popescu.2010. Compressed sensing of multi-view images using disparity compensation. In Proceeding of International Conference on Image Processing (ICIP’10). IEEE, Hong Kong, 3345-3348. DOI: 10.1109/ICIP.2010.5652767
77- K. Chang, T. Qin, W. Xu and A. Men. 2013. A joint reconstruction algorithm for multiview compressed imaging. ISCAS’13. IEEE, Beijing, 221-224, DOI:10.1109/ISCAS.2013.6571822.
78- M.B. Wakin, J.N. Laska, M.F. Duarte, D. Baron, S. Sarvotham, D. Takhar, K.F. Kelly, and R.G. Baraniuk, 2006. Compressive imaging for video representation and coding, in Picture Coding Symposium - PCS 2006, Beijing, China, April 2006.
79- R. Marcia, and R.Willet, 2008. Compressive coded aperture video reconstruction, in Proc. European Signal Processing Conf. (EUSIPCO), 2008.
80- J. Y. Park, and M. B. Wakin, 2009. A multiscale framework for compressive sensing of video, Proceedings of the Picture Coding Symposium, pp. 1–4, May 2009.
81- S.Mun, J. E.Fowler, 2011. Residual reconstruction for block-based compressed sensing of video,in Proceedings of the IEEE Data Compression Conference, J. A. Storer and M. W. Marcellin, Eds., , Eds., Snowbird, UT, March 2011, pp. 183–192.
82- N. Cen, Z. Guan, and T. Melodia. Joint decoding of independently encoded compressive multi-view video streams, in Proc. of Picture Coding Symposium (PCS), San Jose, CA, December 2013.
83- M.Trocan, E.W.Tramel, J.E.Fowler, and B.Pesquet. Compressed sensing recovery of multi-view image and video sequences using signal prediction. Multimedia Tools and Appl., 72(1): 95-12, 2014.
84- E. J. Candes and T. Tao, “Decoding by linear programming,” IEEE Transactions on Information Theory, vol. 51, no. 12, pp. 4203–4215, December 2005.
85- E. Candes and B. Recht, “Exact matrix completion via convex optimization,” Foundations of Computational Mathematics, vol. 9, no. 6, pp. 717-772, 2009.
86- S. Mallat and Z. Zhang, “Matching pursuits with time-frequency dictionaries,” IEEE Transactions on signal processing, vol. 41, no. 12, pp. 3397-3415, 1993.
87- M. A. T. Figueiredo, R. D. Nowak, S. Wright. 2007. Gradient projection for sparse reconstruction: Application to compressed sensing and other inverse problems. IEEE Journal of Signal Processing, Vol. 1, No.4 (Dec 2007), 12 pages. DOI: 10.1109/JSTSP.2007.910281
88- T. Blumensath, and M.E.Davies. 2009. Iterative hard thresholding for compressed sensing. Applied and Computational Harmonic Analysis, Elsevier, Vol. 27, No.3 (Nov. 2009), 10 pages. DOI: 10.1016/j.acha.2009.04.002
89- S. S. Chen, D. L. Donoho, and M. A. Saunders, “Atomic decomposition by basis pursuit,” SIAM Journal on Scientific Computing, vol. 20, no. 1, pp. 33–61, August 1998.
90- A. Chambolle, and P.L.Lions.1997. Image recovery via total variation minimization and related problems, Numerische Mathematik Electronic Edition, Springer, Vol. 76, No. 2 (1997), 21pages. DOI: 10.1007/s002110050258
91- T. F. Chan, S. Esedoglu, F. Park, and A. Yip. 2005. Total variation image reconstruction: overview and recent developments. Mathematical Models in Computer Vision: The Handbook. Springer, New York, Chapter 2.

92- J. A. Tropp and S. J. Wright, "Computational methods for sparse solution of linear inverse problems," Proceedings of the IEEE, vol. 98, no. 6, pp. 948–958, June 2010.
93- D. Donoho, "De-noising by soft-thresholding," Information Theory, IEEE Transactions on, vol. 41, no. 3, pp. 613-627, 2002.
94- J. Haupt, and R. D. Nowak. 2006. Signal reconstruction from noisy random projections, IEEE Trans. on Information Theory, Vol. 52, No. 49 (Sep. 2006), 12 pages. DOI: 10.1109/TIT.2006.880031
95- L. Gan, "Block Compressed Sensing of Natural Images", in 15th International Conference on Digital Signal Processing, pp.403 – 406, Cardiff, 2007.
96- M. Sungkwang, J.E. Fowler, Block Compressed Sensing of Images Using Directional Transforms' in Data Compression Conference-DCC, pp547, Snowbird, UT, 2010.
97- P. Indyk and M. Ruzic, "Near-optimal sparse recovery in the l1 norm, "in Foundations of Computer Science. IEEE 49th Annual IEEE Symposium on, pp. 199-207, IEEE, 2008.
98- R. Berinde, P. Indyk, and M. Ruzic, "Practical near-optimal sparse recovery in the l1 norm, "in Communication, Control, and Computing, 2008 46th Annual Allerton Conference on, pp. 198-205, IEEE, 2009.
99- R. Berinde and P. Indyk, "Sequential sparse matching pursuit, "in Communication, Control, and Computing, 2009. Allerton 2009. 47th Annual Allerton Conference on, pp. 36-43, IEEE, 2010.
100- D. Baron, S. Sarvotham, and R. Baraniuk, "Bayesian compressive sensing via belief propagation," Signal Processing, IEEE Transactions on, vol. 58, no. 1, pp. 269-280, 2009.
101- C. Li. 2010. An efficient algorithm for total variation regularization with applications to the single pixel camera and compressive sensing. Master's thesis, Rice University, Houston, Texas, United States. DOI: http://hdl.handle.net/1911/62229.
102- C. Li. 2013. Compressive sensing for 3d data processing tasks: applications, models and algorithms. PhD thesis, Rice University, Houston, Texas, United States. DOI: http://hdl.handle.net/1911/70314.
103- M. Sungkwang, J.E. Fowler, and E. Tramel. 2012. Block-based compressed sensing of images & video. Foundations & Trends in Signal Processing, Vol. 4, No.4 (2012), 132 pages. DOI: 10.1561/2000000033
104- V. K. Goyal, A. K. Fletcher, and S. Rangan, "Compressive sampling and lossy compression," IEEE Signal Processing Magazine, vol. 25, no. 2, pp. 48–56, March 2008.
105- J. Z. Sun and V. K. Goyal, "Optimal quantization of random measurements in compressed sensing," in Proceedings of the IEEE International Symposium on Information Theory, Seoul, Korea, June 2009, pp. 6–10.
106- P. T. Boufounos and R. G. Baraniuk, "1-bit compressive sensing," in Proceedings of the 42nd Annual Conference on Information Sciences and Systems, Princeton, NJ, March 2008, pp. 16–21.
107- L. Jacques, D. K. Hammond, and J. M. Fadili, "Dequantizing compressed sensing: When oversampling and non-Gaussian constraints combine," IEEE Transactions on Information Theory, vol. 57, no. 1, pp. 559–571, January 2011.
108- L. Wang, X. Wu, and G. Shi, "Progressive quantization of compressive sensing measurements," in Proceedings of the Data Compression Conference, J. A. Storer and M. W. Marcellin, Eds., Snowbird, UT, March 2011, pp. 233–242.
109- ——, "Binned progressive quantization for compressive sensing," IEEE Transactions on Image Processing, vol. 21, no. 6, pp. 2980–2990, June 2012.
110- M. Sungkwang, J.E. Fowler, " DPCM FOR QUANTIZED BLOCK-BASED COMPRESSED SENSING OF IMAGES", 20th European Signal Processing Conference (EUSIPCO 2012)
111- M. Fitzpatrick, Derek. Hill, and Calvin R. Maurer, Jr. 2008. Image Registration. Handbook of Medical Imaging (Vol.2), Med. Image Processing & Analysis, SPIE PRESS, Bellingham, Washington, Chapter 8.
112- Mathworks. 2014. Automatic Registration. Retrieved April 20, 2014 from http://www.mathworks.com/help/images/-automatic-registration.html.
113- D. Mattes, D. R. Haynor, H. Vesselle, T. K. Lewellyn, and W. Eubank, "Nonrigid multimodality image registration," vol. 4322, pp. 1609–1620, SPIE, 2001.
114- M. Styner, C. Brechbuhler, G. Szckely, and G. Gerig, "Parametric estimate of intensity in homogeneities applied to MRI," Medical Imaging, IEEE Transactions on, vol. 19, pp. 153–165, March 2000.

115- M. Misiti, Y. Misiti, G. Oppenheim and J.M. Poggi. 2007. Wavelets and their Applications. Antony Rowe Ltd, Chippenham, Wiltshire.
116- P. Zeeuw. 1998. Wavelet and image fusion. CWI research STW (Mar. 1998), Retrieved April 20, 2014 from Mathworks wfusimg: www.mathworks.com/help/wavelet/ref/wfusimg.html.
117- T. A.Frajka, and K. Zeger. 2002. Residual image coding for stereo image compression. In proceedings of Int. Conf. on Image Processing, Vol. 2, 271-275. DOI. 10.1109/ICIP.2002.1039926
118- C. Strecha, and R. Fransens L. V. Gool. 2006. Combined depth and outlier estimation in multi-view stereo. Vol. 2 (2006) 8 pages. Retrieved January 10, 2014, from Computer Vision Laboratory CVLAB: http://cvlab.epfl.ch/data/strecharmvs, DOI:10.1109/CVPR.2006.78
119- D. Scharstein, and R. Szeliski. 2003. High-accuracy stereo depth maps using structured light. In proceedings of International Conference on Computer Vision and Pattern Recognition (CVPR'03), IEEE, Madison, WI , Vol 1, 195-202. DOI: 10.1109/CVPR.2003.1211354.
120- L. M. Po. 2006. CityU Image Database. Retrieved January 12, 2014, from http://abacus.ee.cityu.edu.hk/imagedb/dbroot/Stereo_Image/
121- J. E. Fowler. 2009. BCS-SPL-block compressed sensing with smooth projected landweber reconstruction. Version 1.5-1 (Aug. 2012). Retrieved Dec. 20, 2013 from http://www.ece.msstate.edu/~fowler/BCSSPL/
122- C. Li, W. Yin, and Y. Zhang. 2009. TVAL3: TV minimization by Augmented Lagrangian and ALternating direction Algorithms. Version beta2.4 (Nov. 2010), Retrieved January 10, 2014 from http://www.caam.rice.edu/~optimization/L1/TVAL3/
123- Y. Yang, O. C. Au ; L. Fang ; X. Wen ; and W. Tang, "Perceptual compressive sensing for image signals", IEEE International Conference on Multimedia and Expo (ICME 09), July 2009, pp 89-92, DOI: 10.1109/ICME.2009.5202443
124- Z. Gao, C. Xiong ; C. Zhou ; H. Wang, "Compressive Sampling with Coefficients Random Permutations for Image Compression", IEEE International Conference on Multimedia and Signal Processing (CMSP), 2011, pp 321-324, DOI: 10.1109/CMSP.2011.71
125- W. Lu, N. Vaswani. 2009. Recursive reconstruction of sparse signal sequences (sequential compressed sensing). Ver. 2, Code for large-sized images (optimization code revised for 2D-DFT&DWT), (2009), Retrieved April 15, 2014 from http://home.engineering.iastate.edu/~luwei/modcs/
126- J. C. Ye. 2012. k-t FOCUSS. Version 1, (Nov. 2012), Retrieved April 15, 2014 from http://bisp.kaist.ac.kr/research_02.html.
127- S. G. Hoggar. 2006. "Mathematics of Digital Images: Creation, Compression, Restoration, Recognition", Cambridge University Press, the Edinburgh Building, Cambridge.
128- YUV Video Test Sequences, Retrieved January 15, 2015, author fangjia from http://www.codersvoice.com/a/webbase/video/08/152014/130.html
129- J.E. Fowler, 2009. BCS-SPL-block compressed sensing with smooth projected landweber reconstruction. Mc-bcs-spl Version 1.0-1 (Aug. 2012). Retrieved Dec. 20, 2013 from http://www.ece.msstate.edu/~fowler/BCSSPL/
130- Book Arrival, Newspaper, LoveBirds Multiview Video Sequence- Courtesy IRCCyN IVC DIBR Videos -from: http://ivc.univ-nantes.fr/en/databases/DIBR_Videos/
131- International Telecommunication Union (ITU), "H.264 : Advanced video coding for generic audiovisual services", ITU-T Recommendations for h.264, 2005
132- International Telecommunication Union (ITU), "H.263 : Video coding for low bit rate communication", ITU-T Recommendations for h.263, 2005
133- Book Arrival, Multiview Video Sequence- Courtesy IRCCyN IVC DIBR Videos -from: http://ivc.univ-nantes.fr/en/databases/DIBR_Videos/
134- Break Dance Multi-view Video Sequence-Courtesy Microsoft Research, from: http://research.microsoft.com/en-us/um/people/sbkang/3dvideodownload/.
135- M. Rahimi et al. "Cyclops: Insitu image sensing and interpretation in wireless sensor networks," in ACM SenSys, 2005.
136- S. Hengstler, D. Prashanth, S. Fong, and H. Aghajan, "Mesheye: A hybrid-resolution smart camera mote for applications in distributed intelligent surveillance," in IPSN, April 2007.
137- P. Chen et al., "Citric: A low-bandwidth wireless camera network platform," in ACM/IEEE ICDSC, Sept 2008.

138- R. Kleihorst, A. Abbo, B. Schueler, and A. Danilin, “Camera mote with a high-performance parallel processor for real-time frame-based video processing," in IEEE AVSS, Sept 2007.
139- Evidence Embedding Technology, “Seed-eye board, a multimedia wsn device. http://rtn.sssup.it/index.php/hardware/seed-eye,"accessed 20/12/2013.
140- A. Rodrguez et al., “The Eye-RIS cmos vision system," in Analog Circuit Design, Springer Netherlands, 2008.
141- W.-C. Feng et al., “Panoptes: Scalable low-power video sensor networking technologies," ACM TOMCCAP, vol. 1(2), May 2005.
142- A. Rowe, D. Goel, and R. Rajkumar, “Firey mosaic: A vision-enabled wireless sensor networking system," in IEEE RTSS, Dec 2007.
143- Evidence Embedding Technology, “cmuCam: open source programmable embedded color vision sensors. http://www.cmucam.org/," accessed 19/12/2014.
144- S. Paniga, L. Borsani, A. Redondi, M. Tagliasacchi, and M. Cesana, \Experimental evaluation of a video streaming system for wireless multimedia sensor networks," in IEEE/IFIP Med-Hoc-Net, 2011.
145- ArduCam, “Arducam. http://www.arducam.com," accessed 19/8/2015.
146- Arduino, “Arduino Due Board. “https://www.arduino.cc/en/Main/ArduinoBoardDue," accessed 19/8/2015.
147- 4D Systems, “uCamII. http://www.4dsystems.com.au/product/ucam ii/," accessed 19/8/2015.
148- Digi, “XBee / XBee-PRO ZB (S2) Modules”. http://www.digi.com/support/productdetail?pid=3430. accessed 19/8/2015.
149- Sparkfun,” LinkSprite JPEG Color Camera TTL Interface”, https://www.sparkfun.com/products/retired/10061
150- Sparkfun,” CMOS Camera Module - 728x488”, https://www.sparkfun.com/products/11745
151- Ada-fruit, “TTL Serial JPEG Camera with NTSC Video”, https://www.adafruit.com/product/397
152- CMUcam4,”Overview”, http://www.cmucam.org/projects/cmucam4
153- M.C.Pros,” CIF Camera Module with OV6620 CMOS Color Sensor, I2C, 352x288”, http://microcontrollershop.com/product_info.php?products_id=4765&osCsid=uic2p97diu91sslonn3knbo2f4
154- J. Foster, “Xbee cookbook issue 1.4 for series 1 with 802.15.4 firmware, www.jsjf.demon.co.uk/xbee/xbee.pdf”. Accessed 4/12/2013.

Appendix

A. BCS Arduino Code

```
#include <avr/pgmspace.h>

const int  BLOCK_SIZE_ROW =16;
const int BLOCK_SIZE_COLUMN = 16;
const int ROW = 128;
const int COLUMN = 128;
const int FINAL_ROW = BLOCK_SIZE_ROW*BLOCK_SIZE_COLUMN;
//const int FINAL_COLUMN = (int)(ceil((int)ROW/BLOCK_SIZE_ROW)
* ceil((int)COLUMN/BLOCK_SIZE_COLUMN));
const int FINAL_COLUMN = 64;
const int SUB_RATE = 0.1;
const int N = BLOCK_SIZE_ROW*BLOCK_SIZE_COLUMN;
//const int M = (int)((SUB_RATE * N) + 0.5);
const int M = 90;
//unsigned long time;

//Image Size 128x128
const PROGMEM  float image[ROW][COLUMN] = {{    94,   91,   100,
    91,   64,   40,   35,   58,   70,   95,   123,  129,  130,  118,
    135,  143,  141,  140,  131,  133,  140,  137,  131,  131,  126,
    133,  91,   127,  153,  152,  154,  151,  147,  142,  134,  134,
    151,  153,  138,  149,  156,  151,  157,  162,  146,  159,  152,
    139,  165,  133,  125,  139,  107,  101,  119,  139,  144,  154,
    163,  168,  159,  161,  154,  137,  124,  154,  157,  95,   80,
    98,   107,  116,  121,  123,  125,  126,  124,  124,  122,  121,
    118,  113,  105,  99,   105,  115,  114,  102,  117,  117,  108,
    130,  110,  97,   114,  133,  123,  127,  108,  109,  127,  117,
    109,  104,  109,  104,  108,  101,  121,  121,  112,  116,  115,
    101,  108,  114,  111,  119,  119,  119,  108,  105,  113,  109,
    103,  114,  111,  111},…

//the in between data is removed total data size is 128x128

{   172,  173,  172,  172,  172,  172,  173,  172,  171,  173,  173,
    172,  173,  173,  174,  176,  177,  179,  179,  177,  177,  176,
    175,  176,  175,  176,  174,  174,  175,  174,  176,  180,  178,
    176,  175,  176,  177,  174,  173,  173,  173,  175,  176,  175,
    174,  173,  175,  175,  173,  173,  172,  173,  174,  174,  175,
    174,  172,  173,  173,  172,  172,  171,  171,  171,  170,  170,
    170,  169,  170,  170,  169,  169,  170,  172,  173,  172,  172,
    172,  171,  171,  170,  169,  167,  166,  167,  166,  166,  166,
    165,  165,  164,  163,  162,  161,  162,  162,  161,  161,  162,
    162,  162,  161,  162,  161,  160,  160,  158,  158,  157,  155,
    156,  155,  154,  153,  151,  154,  152,  151,  152,  151,  152,
    152,  151,  151,  150,  150,  152,  151}};
```

```
//Phi @0.35
const PROGMEM float phi[M][N]={{     -1,   -1,   -1,   -1,   -1,
    -1,  1,    1,    -1,  1,    -1,  1,    -1,  1,    1,    1,
    1,   1,    1,    1,   -1,   -1,  1,    -1,  -1,   -1,   1,
    -1,  -1,   1,    1,   -1,   -1,  -1,   1,   1,    -1,   1,
    -1,  1,    1,    1,   1,    1,   1,    -1,  1,    1,    1,
    -1,  -1,   -1,   -1,  -1,   1,   -1,   -1,  -1,   -1,   -1,
    -1,  1,    1,    1,   -1,   -1,  -1,   1,   1,    -1,   1,
    -1,  1,    1,    -1,  1,    -1,  -1,   1,   1,    -1,   1,
    1,   -1,   1,    -1,  1,    1,   1,    1,   -1,   1,    1,
    1,   -1,   1,    -1,  1,    -1,  1,    -1,  1,    -1,   1,
    1,   1,    -1,   1,   1,    -1,  -1,   -1,  1,    1,    -1,
    -1,  1,    -1,   -1,  -1,   1,   -1,   1,   -1,   -1,   -1,
    -1,  -1,   1,    -1,  1,    1,   -1,   1,   1,    1,    1,
    1,   1,    -1,   -1,  1,    -1,  1,    -1,  -1,   -1,   1,
    1,   1,    -1,   -1,  1,    -1,  1,    -1,  1,    1,    -1,
    1,   -1,   1,    -1,  1,    -1,  -1,   -1,  -1,   1,    -1,
    1,   -1,   -1,   -1,  -1,   1,   -1,   -1,  -1,   -1,   1,
    1,   1,    -1,   1,   -1,   1,   -1,   -1,  1,    -1,   1,
    -1,  -1,   -1,   1,   1,    1,   1,    1,   1,    -1,   1,
    1,   -1,   1,    -1,  -1,   1,   -1,   -1,  1,    -1,   1,
    -1,  1,    -1,   -1,  -1,   1,   1,    -1,  1,    1,    1,
    1,   -1,   -1,   1,   1,    1,   -1,   1,   -1,   -1,   1,
    1,   -1,   1,    -1,  -1,   -1,  1,    -1,  1,    -1,   1,
    1,   1,    -1,   1,   -1,   1,   1,    -1,  1},…

//the in between data is removed total phi size for subrate 0.1
is 26x64

{   1,   -1,   1,    1,   -1,   1,   -1,   -1,  1,    -1,   -1,
    -1,  1,    -1,   -1,  1,    1,   -1,   -1,  1,    -1,   -1,
    1,   1,    1,    1,   1,    -1,  -1,   1,   -1,   1,    -1,
    -1,  -1,   -1,   -1,  1,    1,   -1,   1,   1,    -1,   1,
    1,   1,    -1,   1,   -1,   -1,  -1,   -1,  -1,   -1,   -1,
    1,   1,    -1,   1,   -1,   -1,  1,    1,   -1,   -1,   -1,
    -1,  1,    -1,   -1,  1,    1,   1,    -1,  1,    1,    -1,
    -1,  -1,   1,    -1,  1,    1,   1,    -1,  1,    -1,   -1,
    -1,  1,    -1,   -1,  1,    1,   1,    1,   -1,   1,    1,
    1,   1,    -1,   -1,  -1,   -1,  1,    1,   1,    1,    -1,
    1,   -1,   -1,   1,   -1,   1,   1,    -1,  1,    -1,   -1,
    -1,  1,    -1,   1,   1,    -1,  1,    1,   1,    -1,   1,
    -1,  1,    -1,   1,   1,    -1,  -1,   -1,  -1,   1,    1,
    -1,  -1,   -1,   1,   -1,   -1,  -1,   -1,  -1,   -1,   1,
    -1,  1,    1,    1,   1,    1,   -1,   1,   1,    1,    -1,
    -1,  1,    -1,   1,   1,    -1,  1,    1,   1,    1,    1,
    1,   -1,   1,    1,   1,    1,   -1,   1,   -1,   -1,   1,
    -1,  1,    1,    -1,  -1,   1,   1,    -1,  1,    1,    -1,
    -1,  1,    -1,   -1,  -1,   1,   -1,   -1,  -1,   -1,   1,
    1,   -1,   -1,   -1,  1,    1,   1,    1,   -1,   -1,   -1,
```

```
    -1,  -1,  -1,  1,   -1,  1,   -1,  -1,  1,   -1,  1,
    -1,  1,   -1,  1,   -1,  1,   -1,  -1,  1,   -1,  -1,
    1,   1,   1,   -1,  -1,  -1,  -1,  1,   -1,  1,   -1,
    -1,  1,   -1}};

float im2colValue[FINAL_ROW][FINAL_COLUMN];

float compress[M][FINAL_COLUMN];

void setup() {
  Serial.begin(9600);
  Serial3.begin(9600);
}

void loop(){
//unsigned long start = micros();
  Serial.println();
  Serial.print("Start: ");
  Serial.print(micros());
  im2col();
  //Print((float*)phi,M,N,"phi");

//Print((float*)im2colValue,FINAL_ROW,FINAL_COLUMN,"im2colValue
");
  Serial.print("/////////////////////");
  //Serial.print("End:");
  Serial3.write("End");


  Serial3.write(micros());
  Serial.println();
  Serial.print("Start: ");
  Serial.print(micros());


Multiply((float*)phi,(float*)im2colValue,M,N,FINAL_COLUMN,(floa
t*)compress);


  Serial.print("/////////////////////");
  Serial.print("End:");
  Serial.print(micros());

//Final Output
 Print((float*)compress,M,FINAL_COLUMN,"compress");
  //getPhiMatrix();
  while(1);
}
```

```
//------------------------------------------------------------
void im2col(){
    //int final_row = BLOCK_SIZE_ROW * BLOCK_SIZE_COLUMN;
    //int final_column = ceil((int)ROW/BLOCK_SIZE_ROW) *
ceil((int)COLUMN/BLOCK_SIZE_COLUMN);

    int i, j;

    for (i=0; i<FINAL_ROW; i++)
    {
        for (j = 0; j < FINAL_COLUMN; j++)
          {
             im2colValue[i][j] = 0;
          }
    }

    int ind_r = 0, ind_c = 0, temp_r, temp_c;

    for (i = 0; i < FINAL_COLUMN; i++)
    {
        temp_r = ind_r;
        temp_c = ind_c;
        for (j=0; j < FINAL_ROW; j++)
        {
            im2colValue[j][i] = pgm_read_float(
&image[ind_r][ind_c] );

            ind_r ++;
            if (ind_r % BLOCK_SIZE_ROW == 0)
            {
                ind_r = temp_r;
                ind_c++;
            }
        }
        ind_r = temp_r + BLOCK_SIZE_ROW;
        if (ind_r >= ROW)
        {
            ind_r = 0;
            ind_c = temp_c + BLOCK_SIZE_COLUMN;
        }
        else
        {
            ind_c = temp_c;
        }
    }

    /*
    for (int ii=0; ii < FINAL_ROW; ii++)
    {
        for (int jj=0; jj < FINAL_COLUMN; jj++){
          Serial.print(im2colValue[ii][jj]);
          Serial.print(" ");
```

```
        }

        Serial.println("");
    }
    */
}

/*
void getPhiMatrix(){

    Serial.println(M);
    Serial.println(N);

    delay(10000);

    for (int i = 0; i < M; i++)
    {
        for (int j = 0; j < N; j++)
        {
            Serial.println(phiMatrix[i][j]);
            phi[i][j] = phiMatrix[i][j];
        }
    }

    for (int ii=0; ii < M; ii++)
    {
        for (int jj=0; jj < N; jj++){
          Serial.println(phi[ii][jj]);
        }
    }
}
*/

  // Matrix Printing Routine
  // Uses tabs to separate numbers under assumption printed int
width won't cause problems
  void Print(float* A, int m, int n, String label){
    // A = input matrix (m x n)
    int i,j;
    Serial.println();
    Serial.println(label);
    //Serial3.write(label);
    for (i=0; i<m; i++){
      for (j=0;j<n;j++){
        Serial.print(A[n*i+j]);
        Serial3.print(A[n*i+j]);
        Serial.print("\t");
      }
      Serial.println();
    }
  }
```

```
//Matrix Multiplication Routine
// C = A*B
void Multiply(float* A, float* B, int m, int p, int n, float*
C)
{
  // A = input matrix (m x p)
  // B = input matrix (p x n)
  // m = number of rows in A
  // p = number of columns in A = number of rows in B
  // n = number of columns in B
  // C = output matrix = A*B (m x n)
  int i, j, k;
  for (i=0;i<m;i++)
    for(j=0;j<n;j++)
    {
      C[n*i+j]=0;
      for (k=0;k<p;k++)
        C[n*i+j]=
C[n*i+j]+pgm_read_float(&A[p*i+k])*B[n*k+j];
    }
}
```

II.