Open Research Online

The Open University's repository of research publications
and other research outputs

Intelligent Side Information Generation in Distributed Video Coding

Thesis

How to cite:

Akinola, Mobolaji Olukunle (2015). Intelligent Side Information Generation in Distributed Video Coding. PhD thesis The Open University.

For guidance on citations see FAQs.

© 2015 Mobolaji Akinola

Version: Version of Record

Copyright and Moral Rights for the articles on this site are retained by the individual authors and/or other copyright owners. For more information on Open Research Online's data policy on reuse of materials please consult the policies page.

oro.open.ac.uk

Abstract

Distributed video coding (DVC) reverses the traditional coding paradigm of complex encoders allied with basic decoding to one where the computational cost is largely incurred by the decoder. This is attractive as the proven theoretical work of Wyner-Ziv (WZ) and Slepian-Wolf (SW) shows that the performance by such a system should be exactly the same as a conventional coder. Despite the solid theoretical foundations, current DVC qualitative and quantitative performance falls short of existing conventional coders and there remain crucial limitations. A key constraint governing DVC performance is the quality of *side information* (SI), a coarse representation of original video frames which are not available at the decoder. Techniques to generate SI have usually been based on *linear motion compensated temporal interpolation* (LMCTI), though these do not always produce satisfactory SI quality, especially in sequences exhibiting non-linear motion.

This thesis presents an intelligent *higher order piecewise trajectory temporal interpolation* (HOPTTI) framework for SI generation with original contributions that afford better SI quality in comparison to existing LMCTI-based approaches. The major elements in this framework are: (i) a cubic trajectory interpolation algorithm model that significantly improves the accuracy of motion vector estimations; (ii) an *adaptive overlapped block motion compensation* (AOBMC) model which reduces both blocking and overlapping artefacts in the SI emanating from the block matching algorithm; (iii) the development of an empirical mode switching algorithm; and (iv) an intelligent switching mechanism to construct SI by automatically selecting the best macroblock from the intermediate SI generated by HOPTTI and AOBMC algorithms. Rigorous analysis and evaluation confirms that significant quantitative and perceptual improvements in SI quality are achieved with the new framework.

Intelligent Side Information Generation in Distributed Video Coding

Thesis submitted in partial fulfilment for the degree of Doctor of Philosophy (PhD)

By

Mobolaji Olukunle Akinola

Faculty of Mathematics, Computing and Technology

Department of Computing and Communications

The Open University, Milton Keynes, UK.

June, 2014

Dedication

This thesis is dedicated to the Almighty God, and especially to my wife Sumbo and three boys, Wole, Bolu and Tise who supported me during the period of my PhD study

Acknowledgements

I cannot overstate my gratitude to both my Supervisors Professor L.S. Dooley and Dr. K.C.P. Wong. Without their guidance and support, I would not have completed this thesis. Throughout my studies, they provided encouragement, advice, suggestions, criticism and lots of academic ideas and critical thoughts that shaped this thesis. I would also, like to thank Dr. A. Poulton for his support and for undertaking the proof-reading of this thesis, which he completed in record time

I would like to extend my gratitude to the MCT Research Deanery at the OU who funded my PhD work and provided additional funds in times of financial difficulties, without which this thesis would not have been completed.

I would also like to thank all my colleagues in the XGMT Research Laboratories of the computing and communications department of the Open University for their friendship and support, especially, Dr. Faisal Tariq. Special thanks to Smarti and Parminder Reel who also proof read the final version of the thesis.

Finally, I would like to give special thanks to my wife whose endless love supported me to complete this work.

Abbreviations and Acronyms

3D	Three Dimension
AOBMC	Adaptive Overlapped Block Motion Compensation
AVC	Advance Video Coding
BER	Bit Error Rate
BMA	Block Matching Algorithm
BDME	Bi-directional Motion Estimation
CODEC	Coder-Decoder
CRC	Cyclic Redundancy Check
dB	Decibel
DCT	Discrete Cosine Transform
DS	Diamond Search
DSC	Distributed Source Coding
DISCOVER	DIStributed COding for Video sERvices
DISCUS	Distributed Source Coding Using Syndromes
DVC	Distributed Video Coding
EEG	Electroencephalographic
FLS	Fuzzy Logic System
GOP	Group of Picture
IEEE	Institute of Electronics and Electrical Engineering
IMS	Intelligent Mode Switching
ITU-T	International Telecommunication Union, Standard
JPEG	Joint Photographic Experts Group

JVT	Joint Video Team
LDPC	Low-Density Parity-Check
LDSP	Large Diamond Search Pattern
MATLAB®	Matrix Laboratory (proprietary Software)
MB	Macro Block
MC	Motion Compensation
MCI	Motion Compensated Interpolation
ME	Motion Estimation
MPEG	Moving Pictures Expert Group
MS	Mode Switching
MSE	Mean Square Error
MV	Motion Vector
NN	Neural Netwok
PRISM	Power-efficient, Robust, High-compression, Syndrome-based Multimedia coding
PSNR	Peak Signal to Noise Ratio
QCIF	Quarter Common Intermediate Format
RAM	Random Access Memory
RD	Rate Distortion
RSC	Recursive Systemic Convolution
RST	Rough Set Theory
SI	Side Information
SW	Slepian and Wolf
TSS	Three Step Search
TV	Television
WZ	Wyner-Ziv

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Introduction

Today, images and videos are everywhere and it is hard to imagine any area of life or industry where they do not play some role. The increasing processing power, miniaturization of chips by hyper-integration (Jian-Qiang, 2009) and storage capacity of computers (digital processors) has made it possible for mobile devices such as mobile phones, web cams and tablets to include video cameras making them widely available. The availability of these cameras has fuelled the increasing demand for visual communication technologies in all aspects of human life. Given that video capture, storage and transmission demands high bit rates, coupled with additional constraints in processing power and energy limitations for mobile devices, the compression of video data which is generally referred to as *video coding* has remained an important research area. *Video coding* refers to the exploitation of spatial-temporal correlation to remove redundancies leading to compression. While some compression techniques are lossless, such as arithmetic codes, others are lossy and includes techniques such as quantization, which results in the loss of visual information, which together reduces the quantity of data used to represent the information contained in the video.

Lossless coding techniques, such as arithmetic codes include *Low Density Parity Check* (LDPC) and Turbo codes which only remove redundancies. Lossy coding techniques on the other hand combine the removal of redundancies with dropping of visual information. For example, motion compensated prediction is used for removing temporal redundancy while the quantization process associated with *discrete cosine transform* (DCT) which in itself is completely reversible and therefore lossless, allows the hierarchical removal of spatial redundancies by making it possible for less sensitive visual information to be removed first.

1.1.1 History and Trends of Video Coding

Image and video compression has evolved from the 1990s to meet the various applications of images and videos as they develop with two groups being very prominent in their evolution. The present conventional state of the art coder-decoder (codec) is represented by the *International Telecommunication Union,* (ITU-T) *advanced video coding* (AVC) standard (Wiegand et al., 2003) and *Moving Picture Experts Group* (MPEG) - 4 (Gall, 1991) coding standards, which have been combined into one standard under the supervision of the Joint Video Team (JVT) (Hendrawan and Yusuf, 2012). Figure 1.1 shows the trend of the evolution of video compression from the matured conventional coding standards to the ones presently undergoing standardization, mostly through the efforts of the JVT (Zhao et al., 2013; Chen et al., 2009) and the yet to be standardized advances such as *Distributed Video Coding* (DVC).

While conventional video coding standards are efficient and fit well with the present day broadcasting paradigm, which has complex encoders allied with simple decoders (TV set top boxes), new and emerging application scenarios, such as wild life surveillance or transmission of medical images from remote villages to medical specialists in the cities have led to proposals for new architectures and paradigms.

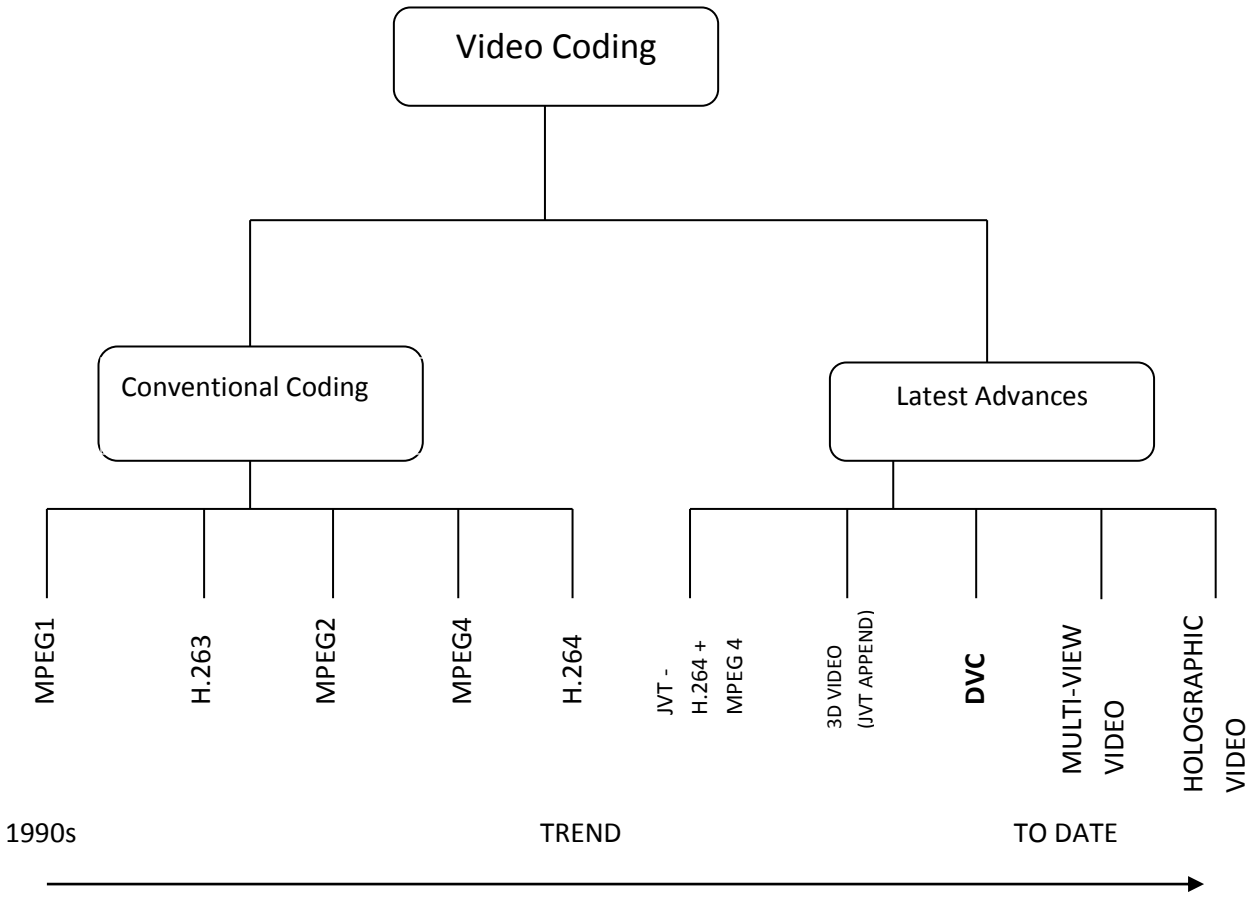


Figure 1.1 Pictorial Diagram showing the trend of video compression and standards.

These include DVC based on the theories of *Wyner-Ziv* (1973) (WZ) and *Slepian-Wolf* (1973) (SW) which reverses the broadcast paradigm to that which has simple encoders allied with more complex decoders.

State of the art conventional video encompasses techniques which work well when the original video frame is present, thus are usually implemented at the encoder. They combine together to make the encoder typically 5 to 10 times more computationally complex (Semsarzadeh et al. 2013; Ostermann et al., 2004; Saponara et al., 2003) than the decoder, while at the same time making the state of the art video codec highly efficient.

1.1.2 Conventional Video Coding Standard and High Complexity Encoding

One of the most successful conventional video coding standards that has crossed over to still feature prominently in the latest advances as can be seen from Figure 1.1, is the H.264 video coding standard (Hendrawan and Yusuf, 2012). The H.264 codec clearly depicts the asymetric nature of the conventional encoder-decoder, where the encoder is significantly more complex than the decoder. Some of the features in the encoder side of the H.264 that increase its complexity include:

(i) *Multiple Frames Referencing:* This is the use of multiple reference frames for *motion estimation* (ME) which greatly increases the efficiency of the codec while at the same time increasing overhead as the schemes with lowest overheads such as the *adaptive and fast multi-frame selection algorithm* consider at least five reference frames (Wiegand et al., 2003) during ME.

(ii) *Variable Block Size Motion Estimation:* This is the use of the optimum block size for each *macro-block* (MB) and this is determined using a block size prediction scheme known as the *MB frame selection* scheme (Ostermann et al., 2004) which is executed every time the block size is to be determined, making the overhead cost very high but increasing codec output quality.

(iii) *Integer Transforms:* This is the use of a scaling algorithm to convert all DCT coefficients to integers. This makes the implementation of DCT in hardware simpler as the tasks can solely be handled using additions and shifts (Ostermann et al., 2004) while at the same time avoiding the inverse transform mismatch problem of DCT. Most of the algorithms have sacrificially higher calculation overheads for the simplicity of implementation (Ostermann et al., 2004; Hendrawan and Yusuf, 2012).

(iv) *Context Adaptive Binary Arithmetic Coding:* This is the use of *context-based adaptive binary arithmetic coding* algorithm (Ostermann et al., 2004) in coding residual data in state of the art video codecs which allow the exploitation of context modelling to

remove redundancy in the symbol statistics as it varies over space and time as well as source material and coding conditions. Though this technique increases coding performance, giving 5-15% bit rate reduction for typical coding conditions (Ostermann et al., 2004), it however also increases overhead (Hendrawan and Yusuf, 2012).

1.1.3 Low Complexity Encoding, DVC and SI

The high complexity of the encoder in the conventional codecs makes it difficult for them to be deployed in the resource poor cameras that are now increasingly available as hand held mobile devices. The possibilities of futuristic applications that can use these cameras, such as free viewpoint TV, remote surveillance and telemedicine, necessitate a new paradigm for a less computationally intensive encoder allied with a more computationally intensive decoder. This encouraged researchers to return to the lossy joint decoding of statistically correlated sources with *side information* (SI) using the, WZ theorem (Wyner and Ziv 1973) and lossless SW (Slepian and Wolf 1973) theorems propounded in the seventies to produce a practical implementation which is referred to as DVC. Both WZ and SW produced theoretical proof that the exploitation of redundancies of correlated sources at the decoder gives the same performance as the exploitation of the redundancies at the encoder (Brites et al., 2013). While SW produced theoretical rate bounds for the lossless case, WZ produced theoretical rate bounds for given performance targets, thereby allowing for losses. The practical implementation of WZ and SW theorems is referred to as DVC mainly because it enables the exploitation of the correlation between sources that might not be physically located in the same place, as they can be independently transmitted but jointly decoded where the correlation is exploited.(Brites et al., 2013).

DVC is thus a paradigm shift from conventional coding standards as it allies simple encoders with complex decoders, allowing the deployment of simple, cheap, resource constrained encoders in portable mobile devices (Brites et al. 2013; Pereira et al. 2008).

This alternative DVC paradigm has restarted new research in video coding, producing interesting results and making feasible, a new set of futuristic applications including remote visible spectrum sensing and surveillance, three dimension (3D) TV, free viewpoint TV (Artigas et al. 2007; Li, 2008; Ye et al. 2009; Liu, Yue and Chen 2009; Brites et al. 2013).

DVC is non-predictive and though its practical implementation is difficult as evidenced by the WZ and SW theorems remaining as theory for more than thirty years after their publication, interest in its practical implementation and improvement has continued because it has a number of functional advantages over the conventional codecs including:

(i) For conventional schemes, correlation is jointly exploited at the encoder in a so called downlink scenario which is feasible with present day video broadcast schemes requiring cheap decoders like TV set top boxes. However, for emerging technologies, where different encoders transmit from different locations real-time, then practical communication between views becomes difficult, and possible joint exploitation of correlation at the decoder becomes important.

(ii) With the WZ codec, the computationally intensive algorithms are transferred to the decoder, thus reducing the complexity of the encoder, saving transmission computational overhead.

(iii) The WZ codec becomes flexible as the decoder controls the coding process acting as a joint decoder for various frames and decides what frame(s) and bit rates should be employed to give the best performance, for example, when the audience at a video trailer been viewed on 3G mobile phone decided that a bit rate of 160 kb/s for 176x144 frame resolution, sufficiently meets their viewing expectation (Knoche and Sasse, 2006), the bit rate can be set at this rate from the DVC decoder. Whereas, conventional schemes however, have their coding process controlled at the encoder. The encoder

predicts the number of bits required and sends it regardless of actual requirement at the decoder, making it an inflexible structure.

(iv) Conventional video coding employs temporal correlation to predict the next frame, which is based on the last predicted frame and is thus prone to drift losses if there is loss of a set of predictors as the next set of predictors becomes less effective. DVC on the other hand is not predictive. The availability of different frames at the decoder, jointly decoded, gives the possibility of further processing before output is given, and processing to remove occlusions, and use of other fusion algorithms are possible using the DVC paradigm.

All the early DVC architectures and practical implementations, while having low encoder overheads, had a quantitative and qualitative performance deficit compared to conventional codecs outlined in Section 1.2. Both *Peak Signal to Noise Ratio* (PSNR) and *rate-distortion* (RD) results (Brites et al., 2013; Ouret, Dufau and Ebrahimi 2009) using stationary, head and shoulder, synthetic and slow object motion sequences, consistently revealed that codecs, like H.264 still performed better in comparison with WZ-based DVC codecs.

A conceptualization of DVC highlighting the important role SI plays is shown in Figure 1.2 for three sample input frames. The idea is for frame #2 to be dropped by the encoder to reduce complexity so it is not transmitted to the decoder. Instead, adjacent frames #1 and #3 (which are known as *key frames*) are transmitted to a computationally more powerful decoder, usually using an H.264 intra coding without motion compensation, which simplifies the encoding process. The missing frame is then reconstructed at the decoder using the SI, which is initially generated by interpolating frames #1 and #3. Various algorithms can then be applied at the decoder to improve SI quality to approximate the best representation of the missing frame.

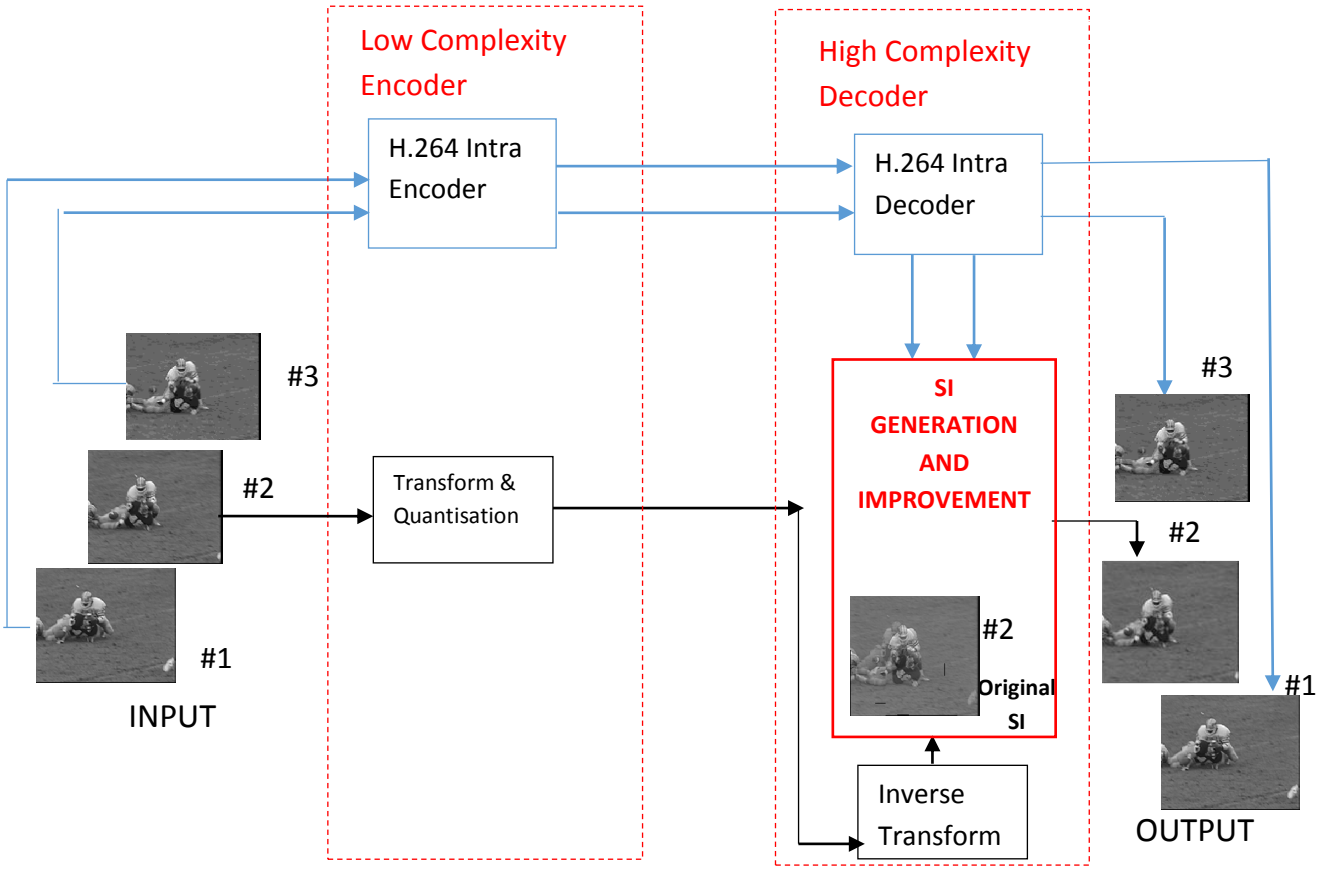


Figure 1.2 Basic DVC Conceptualization and role of SI

The final version of frame #2 is created normally using a DCT and quantization which is transmitted to the decoder to enable a reconstruction of the missing frame as shown in Figure 1.2. The DVC codec thus exploits a statistical dependence between the original input and SI via this lossy channel (usually a Gaussian correlation model).

As Figure 1.2 shows, the output codec quality is dependent on the SI representation, making this the key DVC module. Since SW and WZ theories state DVC should have the same coding performance as conventional codecs, because the initial SI is based on a coarse approximation of the missing frame, (Brites et al. 2013; Artigas et al. 2007; Li,

2008; Ye et al. 2009; Liu, Yue and Chen 2009), all confirm that any improvement in SI quality will lead to better WZ codec output quality. This was the key motivation for the research work in this thesis as highlighted by the shaded *SI Generation and Improvement Framework* block in Figure 1.2.

Additionally, *linear motion compensated temporal interpolation* (LMCTI) has been used in SI generation (Artigas et al. 2007; Li, 2008; Pereira et al. 2008), with proposed enhancements including hierarchical temporal interpolation (Liu, Yue and Chen 2009) and spatially-aided SI generation (Ye et al. 2009). These SI generation techniques use linear interpolation allied with various temporal and spatial refinements. While LMCTI provides reasonable SI quality for sequences with slow-to-medium object motion, it tends not to generally be so successful for sequences exhibiting non-linear and fast motion (Ye et al. 2009).

1.2 Motivation, Research Problem Statement and Objectives

DVC provides a paradigm shift in video coding that reverses the traditional coding paradigm of complex encoders allied with basic decoding, to one where the computational cost is largely incurred by the decoder. This opens up the use of DVC on resource-poor mobile devices such as mobile phones, web cams, palmtops and tablets, to meet the ever increasing demand for high quality visual communication technologies in all aspects of human life and in every conceivable location. Since the theoretical proof exists from SW and WZ theorems that DVC performance should match conventional codecs in quality (Wyner and Ziv 1973; Slepian and Wolf 1973), the hope that futuristic applications such as free viewpoint TV and remote surveillance and remote, disaster response, telemedicine has the potential to achieve high quality outputs acts as motivation to pursue DVC research.

The basic challenge and problem of DVC remains the persistent performance gap between contemporary DVC implementations and conventional codecs such as H.264 (Brites et al. 2013; Artigas et al. 2007; Li, 2008; Ye et al. 2009; Ortega, 2007; Pereira et al. 2008). Thus allowing for this research to set the reduction of the persistent performance gap as overarching objective. Therefore, by tackling one of the most important DVC performance bottlenecks the overall DVC output quality may be improved. This means that the development of new and effective solutions to remedy the bottleneck that has hitherto kept DVC from reaching the theoretical performance similar to that of the conventional codecs is a major motivation for the research presented in this thesis.

As discussed in section 1.1, SI quality is an important element in terms of the overall DVC output quality and narrowing the coding performance gap with conventional codecs. To meet this challenge, SI improvement framework was proposed which addresses four key objectives in the thesis. They are as follows:

(i) Investigate and appraise higher-order interpolation models to improve SI quality.

Justification: Existing linear-based interpolation algorithms used for SI generation tend to degrade when video contains features like fast moving objects, multiple objects and/or complex motion. New high-order SI models will be designed to better reflect different types of motion in such sequences and their corresponding performance critically evaluated on both SI and WZ output quality.

(ii) To understand and explain how the *block matching algorithms* (BMA) which create SI produce various visual artifacts and examine methods to minimize their impact.

Justification: The interpolation process to generate the initial SI can lead to artefacts due to fast BMA blocking and overlapping effects. Similar artefacts appear in other interpolation

based techniques like video frame rate up conversion (FRUC) and de-interlacing, so an investigation will be undertaken to assess the schemes used to minimize their effects. SI specific solutions will be then constructed and integrated to lower the perceptual impact of BMA artefacts on output quality.

(iii) To develop empirical techniques for switching SI block-based algorithms.

Justification: In evaluating the solution proposed in (ii), not all MBs in an SI frame necessarily provide improved quality compared with the higher-order algorithm developed in (i), because parameter settings for the SI improvement algorithms vary between different video sequence, frames and even MBs. SI frames constructed by combining the better MB option from the two algorithms will thus be investigated and an empirical MB switching strategy developed with its impact on overall SI quality analysed.

(iv) To construct an automated mechanism to manage block switching and parameter settings in the SI generation framework.

Justification: Manual block switching necessitates the fixing of parameters which may not be appropriate for all sequences. To automatically manage block switching and key parameter selection based upon input video characteristics, new artificial intelligence-based mechanisms will be examined and a proof-of-concept implementation developed for SI generation improvement, with corresponding output quality analysed.

To achieve all four research objectives, this thesis will presents an original *SI Generation and Improvement Framework* shown in Figure 1.3 with original scientific contributions in four constituent blocks via a suite of rigorously analysed algorithms. These are respectively:

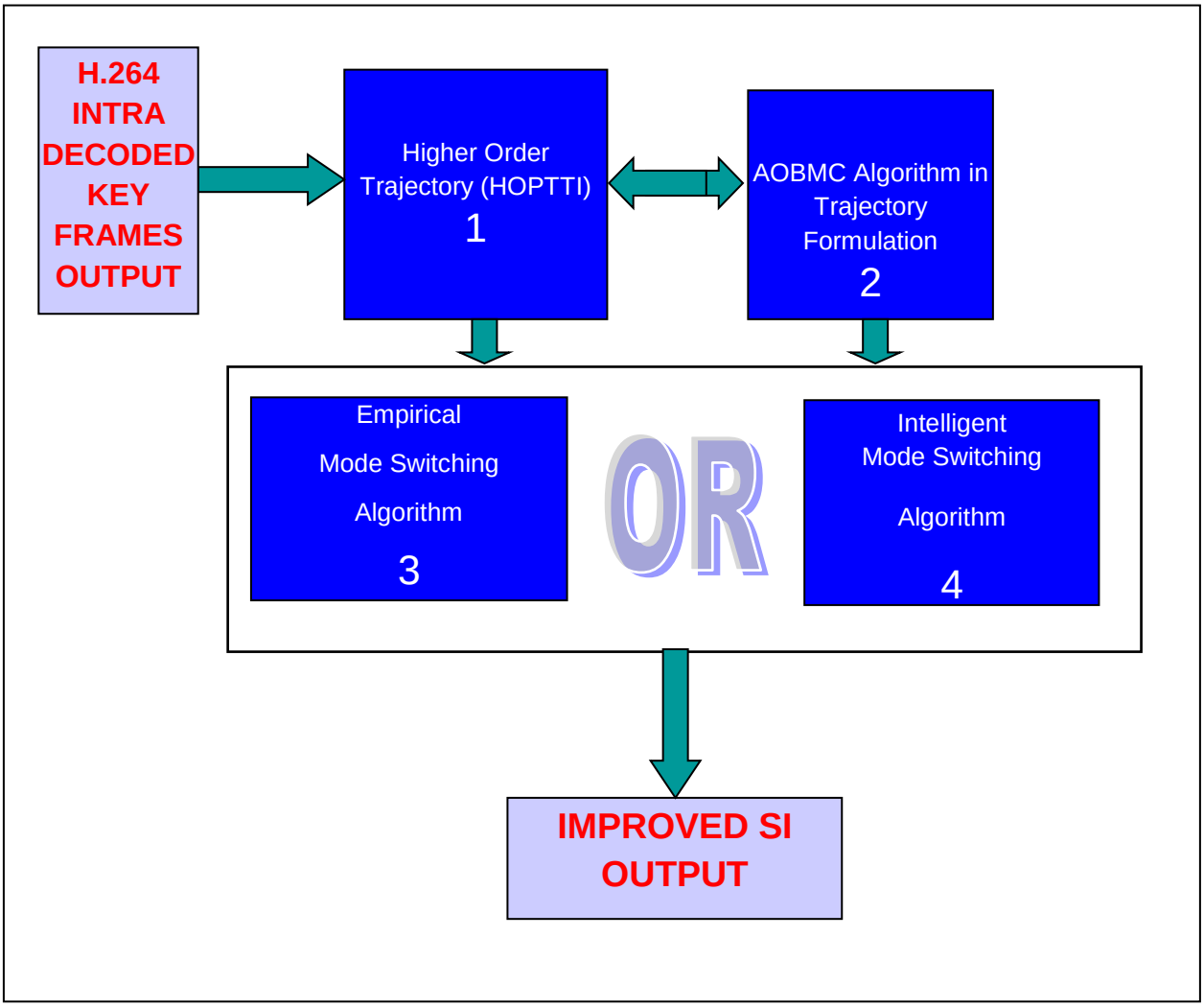


Figure 1.3 Block 1-4 of the *SI Generation and Improvement Framework.*

BLOCK 1 comprises the *higher order piecewise trajectory temporal interpolation* (HOPTTI) algorithm which introduces higher-order trajectories into the SI generation scheme. This trajectory model solution fulfils objective (i) by including acceleration and jolt rather than the linear interpolation approaches traditionally used to create SI. RD results for a range of video test sequences show an SI improvement of up to 5 *Decibels* (dB) at typical rates employed by the community, compared with existing SI techniques. This work has been published in (Akinola, Dooley and Wong 2010).

BLOCK 2 addresses the blocking and overlapping artifact issues identified in objective (ii), by integrating the AOBMC algorithm within the SI framework to improve output

quality. Both numerical and perceptual results confirm SI improvements of up to 3.6dB and a minimum of 1dB PSNR improvement in SI achieved using this new technique for a wide range of test sequences. Part of this work was published in (Akinola, Dooley and Wong 2011)

BLOCK 3 introduces and develops an empirical *mode switching* (MS) algorithm that uses spatial-temporal video content parameters to switch between AOBMC and HOPTTI MBs in the frames to improve SI generation and increase overall DVC RD performance. Part of this work was published in (Akinola, Dooley and Wong 2011). Empirical MS makes use of video content to generate more accurate SI representations to fulfill objective (iii) of the thesis. SI results confirm this empirical MS strategy can consistently provide up to 1.5dB PSNR improvement in SI quality.

BLOCK 4 extends the MS algorithm to include intelligent MB switching based on *Rough Set Theory* (RST) in a proof-of-concept implementation, to automatically switch between HOPTTI and AOBMC MBs in the frame. The solution automatically sets certain thresholds which were previously manually determined, with results confirming that this intelligent MS improved the RD performance of SI by on average, 1.5dB making RD results to outperform even H.264 Intra, noteworthy particularly at low bit rates.

Summarising, the conventional video codecs typified by H.264, has an encoder that has 5–10 times overhead than the decoder which makes it difficult to employ in the emerging resource poor devices and possibilities of futuristic applications that can use them, such as free view point TV and remote surveillance and telemedicine. This necessitates a new paradigm for a less computationally intensive encoder allied with a more computationally intensive decoder, supported as feasible by the WZ and SW theorems. The above scenario gave rise to DVC which is the practical implementation of WZ and SW theorems with the publication in 2002 (Puri and Ramchandran 2002; Aaron, Rane and Girod 2004; Pradhan

and Ramchandran 2003) of the earlier architectures. The coding performance even for stationary, slow and single object video sequences is low compared to H.264 standard video codec. However, in literature (Brites et al., 2013; Ouret, Dufau and Ebrahimi 2009; Pradhan, Chou and Ramchandran 2003; Puri and Ramchandran 2002; Aaron, Rane and Girod 2004; Pradhan and Ramchandran 2003; Varodayan, Aaron and Girod 2006; Ortega 2007; Pereira et al. 2008), it has emerged that one of the most important limitations to DVC performance is the SI, generated traditionally by the use of LMCTI based algorithms at the decoder where the original frames are not available. Therefore there is scope for DVC improvement and original contribution to knowledge as shown in the shaded block of DVC *SI Generation and Improvement Framework* in the DVC conceptualization of Figure 1.2. A framework for SI improvement and increase in overall DVC performance quality is further presented with for contributions that show that the perpetual performance gap between DVC and conventional codecs have been narrowed.

1.3 Organization of the Thesis

This thesis is organized as follows:

A survey of the origins and development of DVC including its information theoretic underpinnings from the SW and WZ theorems alongside a critical evaluation of the various DVC architectures in literature is presented in Chapter 2. This Chapter also presents the DVC architecture employed for the test bed in this thesis.

Chapter 3 presents a review of the SI generation strategies that have been implemented in DVC. A discussion of the importance of SI in DVC as well as its overarching status as a fundamental bottleneck that is militating against DVC performance is presented. The gap that has been identified in SI generation is highlighted.

Chapter 4 discusses the research methodology that is applied, spanning from the problem formulation and solutions strategy to the quantitative and qualitative performance analysis techniques. The Chapter presents the software implementation of the test bed used and the validation of the proposed solutions alongside an analysis of the video sequence dataset employed for testing.

Chapter 5 introduces the HOPTTI framework that aims to provide a more accurate method of SI estimation and why this framework has the potential of providing a substantially improved SI for DVC. Furthermore, the improvement in SI as the trajectory order increases is shown.

Chapter 6 develops and integrates AOBMC as to tackle overlapping and blocking artifacts that stem from block matching algorithms utilized in the generation of the higher order trajectories. Showing that AOBMC using only spatial window does not always perform well, the algorithm is further enhance by a MS algorithm that combines video content temporal parameter in order to empirically switch between the AOBMC generated SI and the one originally generated by HOPTTI.

Chapter 7 presents an RST based intelligent enhancement to the MS algorithm which improves SI generation while at the same time automates the adaptive video content parameters used for spatial-temporal switching. Furthermore, due to the novel nature of this intelligent algorithm, a benchmark is established to effectively analyze the contribution of the RST switching enhancement.

Chapter 8 explores some feasible future works that will exploit the outcomes from this thesis and Chapter 9 provides some conclusions with regards to the research.

Chapter 2

Survey of DVC

2.1 Theoretical Background

This Chapter presents DVC from its theoretical foundations and literature to show how it represents a paradigm shift from conventional coding standards by allowing the deployment of simple, resource constrained encoders in portable mobile devices (Brites et al., 2013; Pereira et al., 2008). One major characteristic of DVC is that individual frames and sequences are coded independently (i.e. there is no communication between the sensors), but then transmitted to a central base station to be jointly decoded. In addition, computationally complex algorithms usually performed at the encoder such as motion compensation and disparity estimation are transferred to the joint decoder. This is based on the SW and WZ theorems and the DVC architecture models a lossy virtual correlation channel providing SI at the decoder.

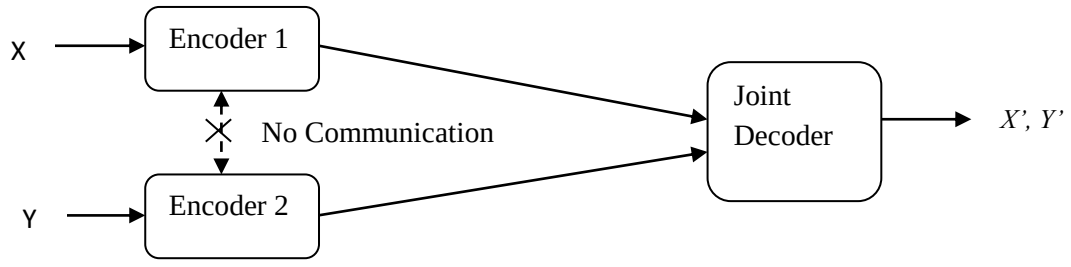


Figure 2.1 Basic Characterization of DVC Theorem as independent compression of statistically dependent sequences *X* and *Y*.

The scenario in Figure 2.1 depicts the independent encoding rate of two statistically dependent sources *X* and *Y* which are jointly decoded. This contrasts with the traditional source compression paradigm which deals with two statistically dependent sources *(X* and *Y)* that are jointly encoded and jointly decoded as shown in Figure 2.2. The scenario depicted by Figure 2.1 has been examined by using SW and WZ theorems which are further discussed in Sections 2.1.1 and Sections 2.1.2.

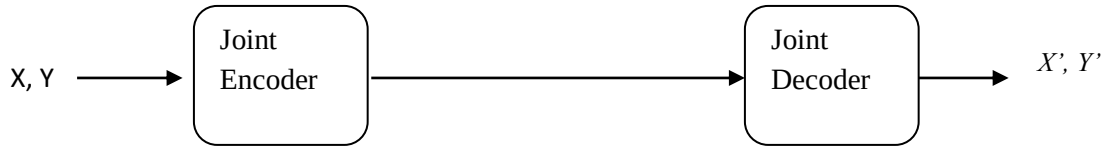


Figure 2.2 Traditional coding paradigm joint encoding and joint decoding of statistically dependent sequences *X* and *Y*.

2.1.1 SW and WZ Theorems

In Slepian and Wolf (1973), SW established the lossless coding information theoretic bounds for Distributed Source Coding (DSC) which refers to correlated random sources that are independently encoded but are jointly decoded by exploiting their statistical dependencies.

Consider a typical situation as depicted by SW in which *X* and *Y* are two binary correlated memory-less sources to be encoded in such a way that the encoder of each source is constrained to operate without knowledge of the other source, while the decoder has available both encoded binary message streams as shown in the Figure 2.1.

The lossless rate for the case when both *X* and *Y* are jointly encoded and jointly decoded can be approached with a vanishing error probability for long sequences. The achievable rate region can be defined as $R_X \geq H(X|Y)$, $R_Y \geq H(Y|X)$ *and* $R_X + R_Y \geq H(X, Y)$, where $H(X|Y)$ *and* $H(Y|X)$ denote the conditional entropies between the two sources. Considering the particular case where *Y* has been encoded at its own entropy rate $R_Y = H(Y)$ then

according to the SW theorem, X can be losslessly decoded at the rate $H(X|Y)$, if the sequence length tends to infinity. The minimum total rate for the two sources is thus $H(Y) + H(X|Y) = H(X, Y)$. Fig 2.3 shows graphically the rate bounds with vanishing error probability until there are no errors.

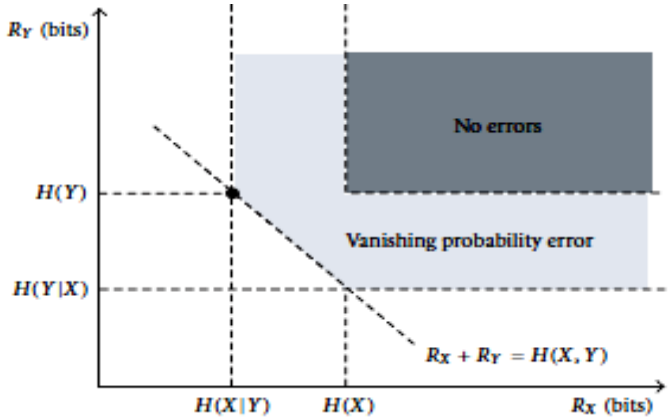


Figure 2.3 Achievable rate regions defined by SW bounds (Ouret, Dufaux and Ebrahimi 2009).

Practical implementations of the SW theorem popularly employed in DVC are two lossless channel codes namely, the TURBO code and the LDPC codes. The one used in this thesis is the LDPC code as it allows DVC to achieve better RD performance with respect to Turbo codes, especially for low motion video sequences (Varodayan, Aaron and Girod 2006) and is therefore briefly introduced in Section 2.1.3.

2.1.2 Practical Implementation of DVC

In Wyner and Ziv (1973), WZ considered the problem of coding two statistically correlated sources X and Y, with respect to a fidelity criterion. They established the rate-distortion (RD) function $R*X|Y(D)$ for the case where the SI Y is perfectly known to the decoder only. For a given target distortion D, $R*X|Y(D)$ in general verifies $RX|Y(D) \leq R*X|Y(D) \leq RX(D)$, where $RX|Y(D)$ is the rate required to encode X if Y is available to both the encoder and the decoder, and RX is the minimal rate for encoding X without SI. WZ have shown that, for correlated Gaussian sources and a mean square error distortion measure, there is

no rate loss with respect to joint coding and joint decoding of the two sources, i.e., $R*_{X|Y}(D) = R_{X|Y}(D)$. This no rate loss result has been extended in (Pradhan, Chou and Ramchandran 2003) to the case where X and Y does not need to be Gaussian, i.e. where X and Y can follow any arbitrary distribution, since these authors proved the theoretic duality between source coding with SI at the decoder and channel coding with SI at the encoder. Practically, code constructions based on the WZ theorem thus naturally rely on a source code quantizer, followed by a SW coder (channel code), such that the fidelity criterion determined by the quantizer remains the same so far as the channel model is matched perfectly and there is no loss in the channel.

2.1.3 Channel Codes and Practical Implementation of WZ Theorem

In the practical implementation of the WZ theory, the duality between source coding and channel coding with SI that has been established by Pradhan, Chou and Ramchandran (2003) was employed and two channel codes that have been utilized in DVC are LDPC and Turbo Codes. Channel coding was first introduced into DVC in one of the earliest architectures discussed in Section 2.2 called *Distributed Source Coding Using Syndromes* (DISCUS) (Pradhan and Ramchandran 2003), where coset codes are employed to encode a source X by calculating the corresponding syndromes. In the DISCUS framework, the decoder which receives the syndromes and has access to the correlated Y source (i.e. the SI) reconstructs X with the value closest (in terms of hamming distance) to the source. The LDPC and Turbo codes are subsequently discussed in more detail.

a) LDPC Codes

The LDPC codes form the basics of the SW channel coding part of the DVC codec employed in this thesis and it ensures that the necessary bits are transferred through the channel without loss. The LDPC codes are linear lossless block codes discovered by Gallager in 1960 (Gallager 1962) that are obtained from bipartite graphs such as shown in

Figure 2.4. such that if the graph *G* has *n* left nodes (also referred to as message nodes) and *m* right nodes (referred to as check nodes), the graph gives a linear block code with length *n* and dimension of at least *n* – *m*. The LDPC codes allow DVC to achieve better RD performance with respect to Turbo codes, especially for low motion video sequences (Varodayan, Aaron and Girod 2006). Overall, LDPC codes have been found to have lower complexity and better performance.

The code given by the bipartite graph is analogous to a matrix representation whereby *H* is a binary *m* x *n* matrix in which the entries defined by the indices (*i*, *j*) is 1; if and only if the *i*th check node is connected to the *j*th message node in the graph. The LDPC code generated by the graph is then the set of vectors $C = (c_1, c_2, c_3 \ldots\ldots c_n)$ such that $H.C^T = 0$.

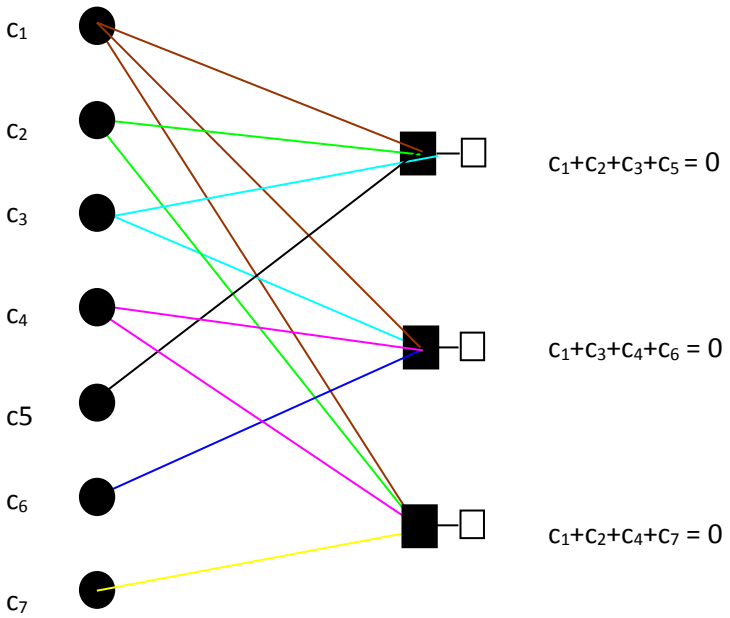


Figure 2.4 An LDPC code from a bipartite graph G

The matrix representation of the above bipartite graph would then be:

$$H = \begin{bmatrix} 1 & 1 & 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 & 0 & 0 & 1 \end{bmatrix} \qquad (2.1)$$

$$C^T = \begin{bmatrix} c_1 \\ c_2 \\ c_3 \\ c_4 \\ c_5 \\ c_6 \\ c_7 \end{bmatrix} \tag{2.2}$$

$$\begin{aligned} c_1 + c_2 + c_3 + c_5 &= 0 \\ c_1 + c_3 + c_4 + c_6 &= 0 \\ c_1 + c_2 + c_4 + c_7 &= 0 \end{aligned} \tag{2.3}$$

From the foregoing, we can infer that any linear code has a representation as a code associated to a bipartite graph, though not every linear code can be represented by a sparse bipartite graph. When the code can be represented by this graph, it is called an LDPC code.

b) Turbo Codes

Turbo codes are one of the other popular channel codes in the practical implementation of DVC and have been applied by (Ascenso, Brites and Pereira 2006; Aaron, Zhang and Girod 2002) and others.

The turbo encoder consists of parallel concatenation of identical *recursive systematic convolution* (RSC) encoders. The RSC usually has a pseudo-random inter-lever between them shown by the illustration in Figure 2.5.

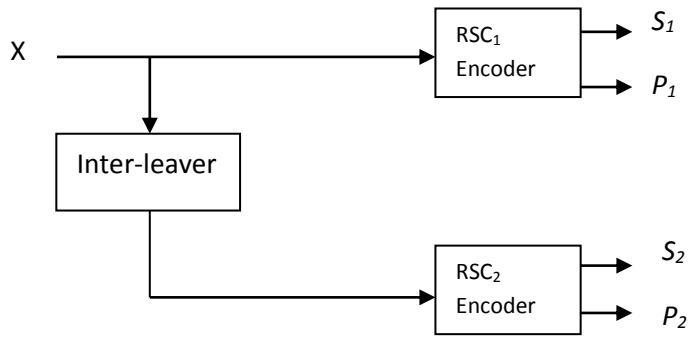


Figure 2.5 A Turbo encoder with two RSCs and inter-leaver (Brites 2005).

Given an input sequence X, each RSC encoder enclosed by the turbo encoder generates a parity sequence P_j with j=[1, 2] corresponding to the sequence at its input which is X for RSC_1 but an inter-leaved version of X for RSC_2, depending on the RSC generator matrix. The RSCs also generate a systematically recursive copy of the sequence which is at its input, represented by S_j with j=[1, 2] in Figure 2.5. According to a given pattern, the input sequence is shuffled so that the inter-leaver could de-correlate the output parity sequences of both RSC which then allows turbo codes with very good bit error rate (BER) performance.

Using a convergence criterion, usually the bit plane error probability (Brites 2005), an estimated X^* of the sequence X at the input of the turbo encoder, is obtained when the convergence criterion is reached. The more random the parity sequence at the output of the RSC, and the longer the inter-leaver length, the better the BER performance of the turbo coder.

2.2 DVC Architectures and Bottlenecks

The first DVC architectures started appearing in 2002 (Puri and Ramchandran, 2002; Aaron, Rane and Girod, 2004; Artigas et al., 2007). The first of these (Puri and Ramchandran, 2002) is an architecture called, *Power-efficient, Robust, high compression Syndrome based Multimedia coding* (PRISM), proposed at UC Berkeley which has the architecture in Fig 2.6. These authors have earlier published a simpler distributed source coding precursor called DISCUS (Pradhan and Ramchandran 2003) which is discussed first due to its simplicity and the fact that it shows the link of DVC to distributed source coding.

2.2.1 Distributed source coding using syndromes (DISCUS)

DISCUS (Pradhan and Ramchandran 2003) is a binary source coding architecture where X and Y are correlated binary sources that can be generalized to the continuous-valued

sources *X* and SI *Y* where *X* is only present at the encoder but *Y*, the SI is present at the decoder. Using the known correlation condition between *X* and the SI *Y*, observed at the decoder, the decoder is able to reconstruct *X"* which is equivalent to *X* by sending only two bits representing the index. To illustrate, suppose *X* and *Y* are 3-bits binary words where the hamming distance between *X* and *Y* is 1, and the correlation conditions are:

$$X, Y \in \{0,1\}^3 \qquad d_H(X,Y) \leq 1 \qquad (2.4)$$

Where $X, Y \in \{0,1\}^3$ signifies that *X* and *Y* are members of the binary set {0, 1} with a word length equivalent to the index 3. $d_H(X,Y) \leq 1$ signifies that the Hamming distance between X and Y is less than or equal to 1. Using the correlation condition $X, Y \in \{0,1\}^3$ in 2.4, the decoder builds a set of possible cosets of the 3- bit repetition code for *X* with Hamming distance of 3 in a look up table as $\{000,111\}$, $\{001,110\}$, $\{010,101\}$, $\{100,011\}$. Thus the index transmitted from encoder tell the decoder that X is for example, the set $\{010,101\}$, and the decoder knows the SI *Y* to be 011, then *X* is decoded to be 010 as hamming distance between *X* and *Y* must not be greater than 1.

In the above illustration and in Pradhan and Ramchandran (2003), we have the distributed source coding paradigm where the SI, *Y* known at only the decoder, is employed to decode *X*, in a codec which is shown to operate above the rate limit for the WZ fidelity criterion when zero mean Gaussian random variables are used as sources.

2.2.2 PRISM architecture

The PRISM architecture which was proposed by the University of California at Berkeley (and therefore also called the *Berkeley Architecture*), shown in Figure 2.6, is based on block classification and classifies 8X8 or 16X16 blocks into three sets, viz: WZ coded, not coded and intra-coded, having pre-defined rates that follow a Laplacian distribution.

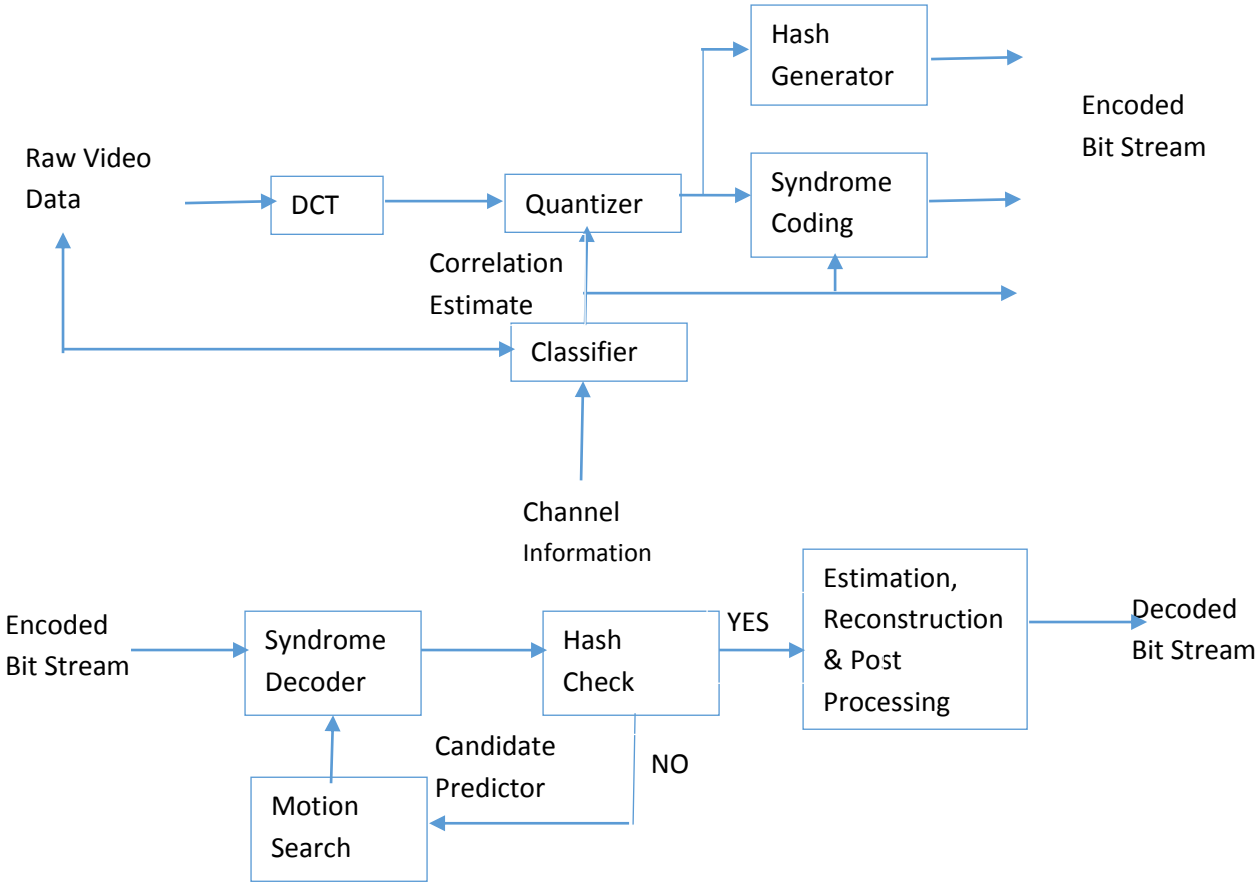


Figure 2.6 PRISM Architecture (Puri and Ramchandran 2002)

All blocks are transformed using DCT and the WZ data are quantized and encoded using trellis code. For the blocks that fall into the class to be coded, only the least significant bits of the quantized DCT coefficients are coded (called syndrome bits), since it is assumed that the most significant bits can be inferred from the SI. For each block, the encoder sends a 16-bit *Cyclic Redundancy Check* (CRC) signature needed to select the best candidate SI block. A motion search is performed to generate SI candidate blocks using half-pixel displacement in a window around the block to be decoded.

2.2.3 Stanford Architecture

Stanford Architecture (Aaron, Zhang, and Girod 2002) appeared simultaneously with the PRISM architecture. The Stanford Architecture was proposed by Stanford University; Fig 2.7 shows the details.

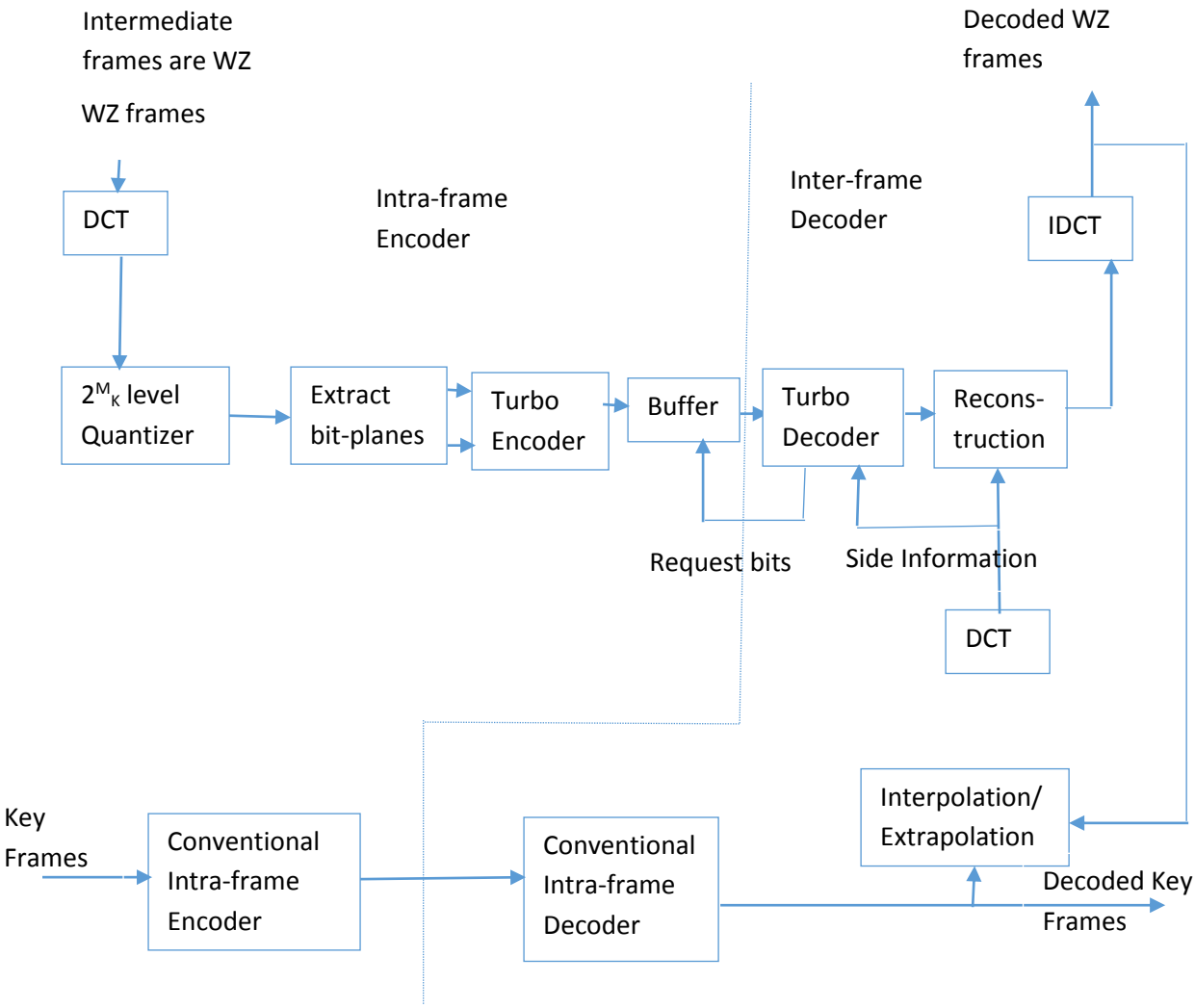


Figure 2.7 Stanford WZ Video Architecture (Aaron, Zhang, and Girod 2002)

The Stanford architecture coding decision is taken at the frame level, the sequence is structured into *groups of pictures* (GOP) in which selected frames called key frames are intra-coded using a standard Codec (H264 or JPEG-2000 for example) and intermediate

frames are WZ coded. The coding architecture was first proposed for the pixel domain (Aaron, Zhang, and Girod 2002) and later extended to the block transform domain where WZ frames are DCT transformed, quantized and fed into a punctured turbo coder. The SI is constructed using motion compensated interpolation of previously decoded key frames. The SI for each WZ frame is taken as a noisy version of the original and the more accurate the SI, the less error the turbo decoder has to correct.

2.2.4 DISCOVER Architecture

The Distributed Coding for Video Services (DISCOVER) architecture is one of the most popular and well researched architectures (Artigas et al. 2007), it proposes an architecture similar to (Aaron, Zhang, and Girod 2002) in which a coding decision is made at the frame level but with many techniques added or improved to enhance the performance of the basic building blocks. This is therefore an enhanced form of the Stanford architecture. The frame level decision to decide which frames would be WZ frames and which frames would be key frames, for example, have been made adaptive by the inclusion of simple but powerful activity measures based on histogram functions, so that the GOP sizes depend on the motion activity within the frames. Other notable changes in the architecture include the introduction of the rate compatible LDPC discovered by (Gallager 1962) and discussed in Section 2.1.3, which was shown by (Li, 2008; Varodayan, Aaron and Girod 2006) to give better channel capacity for the DVC virtual channel model compared to the turbo codes. The DISCOVER architecture was followed by other implementations mostly based on the DISCOVER architecture but producing better functionality or improving performance. One notable implementation that improved performance is that which introduced source classification and advanced temporal interpolation for generating SI (Varodayan, Aaron and Girod 2006). The DISCOVER (Artigas et al. 2007) architecture is shown in Fig 2.8.

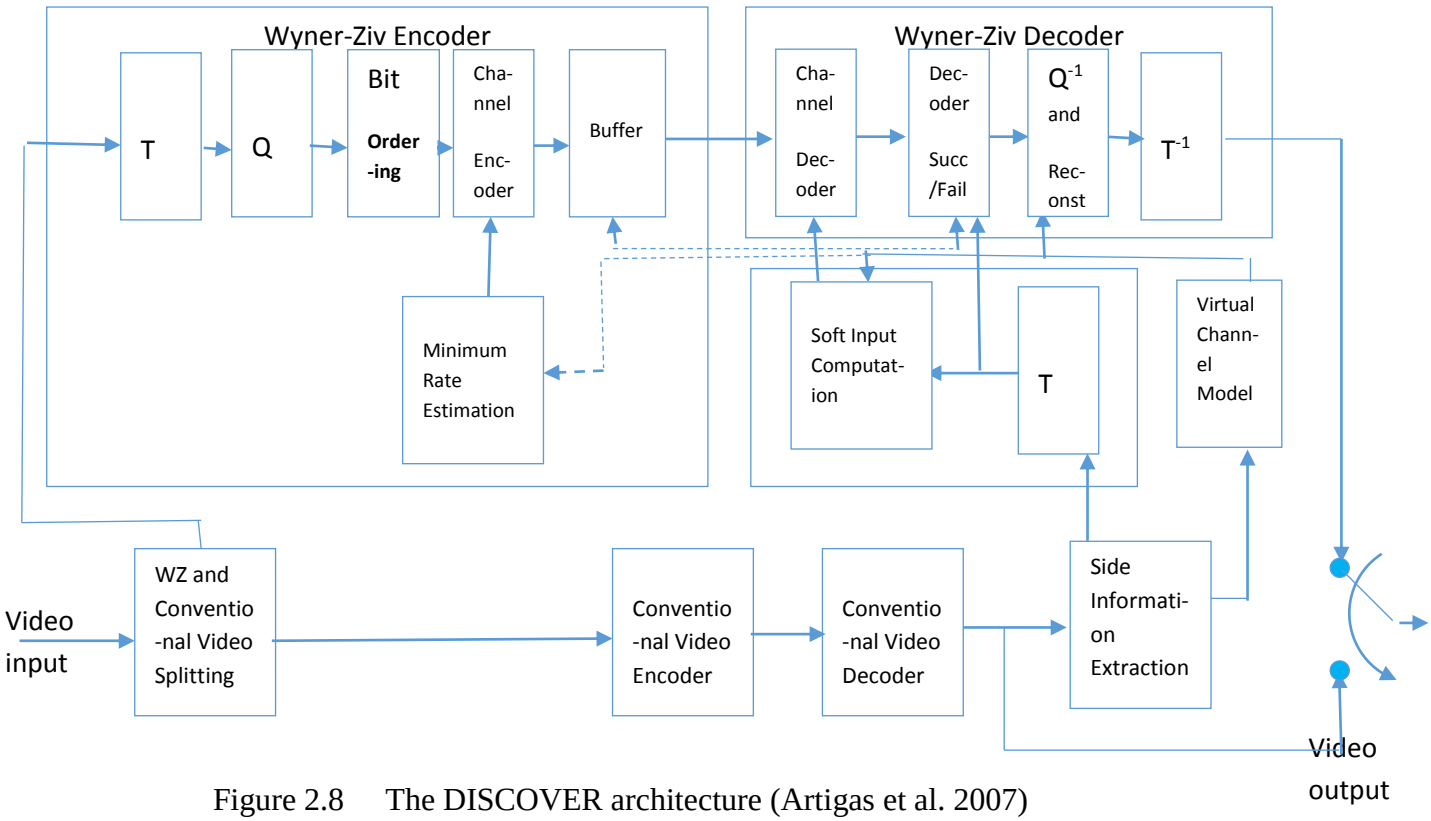


Figure 2.8 The DISCOVER architecture (Artigas et al. 2007)

2.2.5 Li's DVC Codec and Architecture

The codec (Li, 2008) has an architectural framework which is also similar to the Stanford model (Aaron, Zhang, and Girod 2002) in which a coding decision is made at the frame level but with improvements to enhance the performance of the codec such as source classification and classification information algorithms. Li thus illustrated the impact of motion characteristics on the codec performance showing that when motion was slow temporal interpolation achieves high accuracy but as motion becomes faster, interpolation accuracy decreases.

The codec is developed in a modular fashion using scripts which was published and is available for testing. This enable the results as published in Li, 2008 to be tested and for the codec to form a substantial part of the ground truth employed in this thesis as further explored in chapters 4 and 5.

2.2.6 Comparing and Contrasting the Features of Various Architectures

The architectural frameworks proposed by Stanford University discussed in Section 2.2.3 and that proposed by University of California at Berkeley are the most widely accepted DVC frameworks. While the other architectural framework proposals are improvements on one or the other, others combine the various features of the two architectures in order to achieve an improved framework.

The major features both in the encoder and decoder of the architectural frameworks are presented in Table 2.1, enabling a detailed comparison of the features in the Stanford, Berkeley and other architectures.

Table 2.1 FEATURES OF THE DVC PRACTICAL IMPLEMENTATION ARCHITECTURES

Features	***Stanford Architecture***	***PRISM Architecture (Berkeley)***	***Other Architectures***
Encoder			
Frame Splitting	Video divided into key frames and WZ frames with key frames periodically inserted to form GOP sizes. Key frames derived from intra coded conventional codecs without exploiting temporal redundancy e.g. H.264 Intra.	The architecture works on block level and each video frame is divided into 8x8 pixel blocks. Block classification is performed at the encoder to take advantage of the classification gain.	The Stanford architecture frame level splitting with periodic GOP has been mostly adopted in literature including by the popular DISCOVER codec.
Transform	Block based Transform e.g. DCT to WZ frames forming coefficient	DCT is applied to each 8x8 pixel block in the	DCT is still by far the most widely employed transform including

	bands.	frame.	DISCOVER. The wavelet transform is gaining in popularity especially to take advantage of the classification gain, e.g. (Li, 2008)
Quantization	DCT bands are quantized with levels depending on target quality (Aaron, Rane and Girod 2004).	A scalar quantization is applied to the DCT coefficients also depending on set targets similar to the Stanford Architecture.	Transform bands are usually quantized and rates are controlled depending on set targets.
Channel Coding	Turbo codes are employed for coding the bit planes of DCT and sent to a buffer from where they are sent in packets to the decoder upon request through feedback loop.	Syndrome codes are employed for blocks classified in the syndrome code classes where it is assumed that the most significant bits can be inferred from the SI.	Both Turbo codes and LDPC codes are most important channel codes in literature but LDPC have been shown to give better RD performance.
Decoder			
SI generation	Decoder generates SI by motion compensated interpolation .The SI is taken as a noisy version of original WZ frame and 'errors' are corrected by employing bits from the turbo decoder.	The candidate blocks in the syndrome codes are the SI.	SI have been generated mostly by linear motion compensated interpolation, though it has been acknowledged that natural video motion is seldom linear and linear models are only appropriate for videos with slow motion.

Correlation noise model	Residual statistics between WZ frame, DCT coefficients and SI is assumed to be modeled by a Laplacian distribution.	There is no correlation model in the Berkeley architecture and residual statistics is not exploited.	The channel model mostly employed in literature is the Laplacian distribution and it has been adopted by the popular DISCOVER codec.
Channel decoding	The DCT bit planes are turbo decoded as soon as the SI, DCT coefficients and residual statistics are known.	The cosets are identified from the syndrome codes (SI) and then soft decoded using the codeword within the codet identified.	The LDPC codes have been mostly adopted by other DVC architectures.
Frame reconstruction	The frame is finally reconstructed using the decoded WZ bits and SI DCT bands for bands not transmitted.	CRC sum is generated for each decoded quantized block and compared to the CRC sum received from encoder until all blocks in the frame are matched and recovered from the corresponding SI.	The frame reconstruction using decoded WZ bits and SI has been widely adopted in literature.

2.2.7 The Architecture and Development of HOPTTI Test bed

The architectural framework for the DVC test bed that is employed in the trialing of the ideas presented in this thesis is shown in Figure 2.9. The codec implementation is modular, which allows for the easy replacement of each module or sub-module as the case may be for new algorithm implementation.

The modularity is exemplified by the fact that there is a DVC main M-file (Matlab file) from where all the other sub-modules are called from. Thus a sub-module called YUVread in another M-file helps to load the original video in the YUV format, where all that is required to load another video format is to replace this sub-module with a different M-file with the code required to load that video format into matlab.

Also, the architecture implements the latest advances in DVC architecture that are present in literature so that the new ideas presented will be building on the advances already made. The major features of the codec are hereby discussed and detailed:

a) Channel Coding

Following the proof of the duality between source coding and channel coding with SI by (Pradhan, Chou and Ramchandran 2003), the same authors introduced the coset codes in DVC in the DISCUS paper (Pradhan and Ramchandran 2003) as discussed in Section 2.1.3. Other researchers have since proposed different channel codes and by far the most efficient of the channel codes is the LDPC codes by (Fresia and Vandendorpe 2009). Therefore, the LDPC codes have been adopted in the HOPTTI codec.

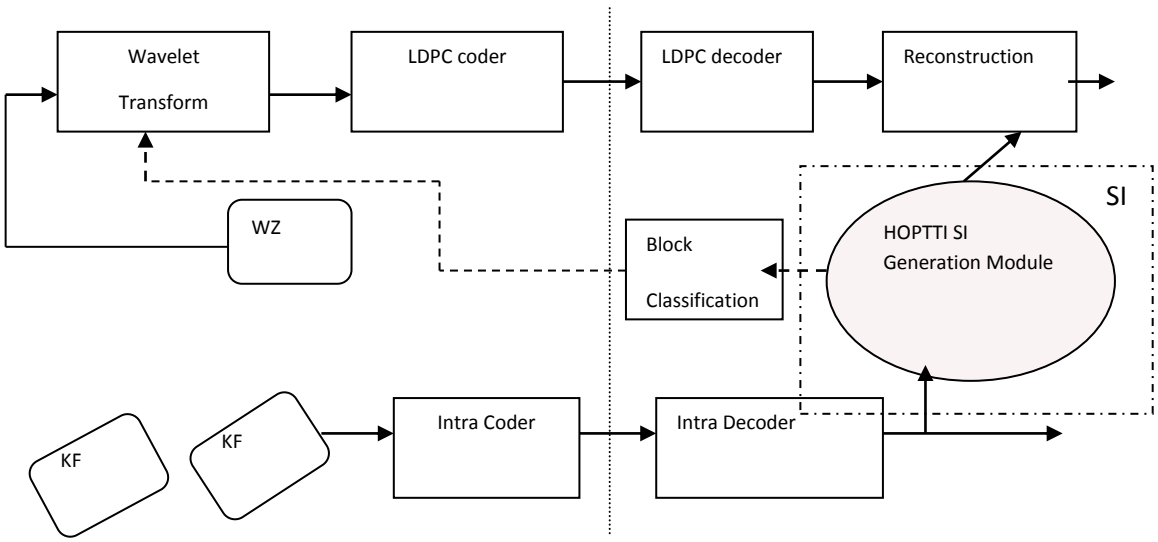


Figure 2.9 Architecture of Codec highlighting SI module

b) Block Classification

Li (2008) recognized that there is a non-negative classification gain and it will always improve the codec. Block classification is therefore adopted in the architectural framework of this thesis in line with Li's codec where blocks are classified into four classes based on spatial-temporal correlation where bits are allocated to the class of significant wavelet coefficients. This requires a feedback channel which is kept at a minimum of 1 bit/block, using key frames only and scale invariant assumption of the temporal correlation structure which keeps the encoder unchanged (low overhead) but the decoder becomes more flexible though at a slightly increased cost.

c) Wavelet Transform

The wavelet transform rather than the DCT is employed in the test bed following Li (2008) to exploit the spatial correlation as this has been employed successfully by other classification based approaches (Shapiro, 2000). The transform coefficients are quantized and reorganized into wavelet trees and the significant coefficients are coded and transmitted as per the classification previously determined at the decoder and fed back to the encoder.

d) SI Generation

The SI generation process in DVC is central to RD performance because it determines the amount of 'errors' that the frame reconstruction process needs to correct through the LDPC decoder and thus the higher the quality of SI that can be generated the fewer 'errors' and the better the overall performance of the codec (Brites et al. 2013). Bi-directional motion compensated temporal interpolation (MCTI) is adopted in the test-bed similar to the one used in (Varodayan, Aaron and Girod 2006). The SI generation module formed the basis of the major contributions in this thesis and the Higher Order Piece-wise Temporal Trajectory

Interpolation was implemented (Akinola, Dooley and Wong 2010) which is formulated based on a cubic trajectory model discussed in Chapter 5 of this thesis.

2.3 DVC Application Scenarios

DVC practical implementation has continued to generate interest due to its possible effective deployment for certain application scenarios that include:

(i) Wireless video cameras for surveillance

(ii) Mobile document scanners

(iii) Mobile video mail

(iv) Free View-point TV

This list of applications includes real time and non- real time applications as well as mono-view and multi-view applications with differing complexities and power requirements. The following Section will briefly describe the applications and how the features of the DVC architecture discussed earlier fit into the application scenario (Brites 2005).

2.3.1 Wireless Video Cameras for Surveillance

Wireless video cameras for surveillance has the important distinction of consisting of video cameras in different locations, some of which might be remote and have size constraints. DVC exploits the spatial-temporal redundancies of independent sources that do not communicate with each other at the source in the encoder, thus it fits perfectly well the application scenario. Furthermore, DVC's simplicity at the encoder makes it the perfect choice for an application whose members could be remotely located and may not have ready access to electrical power and other resources.

These cameras can be employed for wildlife monitoring, traffic monitoring, home monitoring (Brites 2005) and control systems etc.

2.3.2 Mobile Document Scanners

In recent times a lot of people do business from locations that are far away from their offices, especially working from home and sites. Also, global travelers oftentimes need to transmit documents during their trip and the ability to use mobile document scanners would come in handy. The application requires mobile scanners that record parts of the document as video frames. The video frames are then sent to a central server for processing which includes registration and super resolution techniques that are too complex to be undertaken at the document scanner. This application scenario readily lends itself to DVC which has a low cost encoder and a powerful decoder.

2.3.3 Mobile Video Mail

Increasingly, people want to convey the videos of what is happening in their environment to their friends and relatives instead of typing a text which in most cases can be inadequate. They therefore become mobile broadcasters giving real time news from remote locations or even disaster areas. In most cases the person they are messaging are in urban locations where they have access to powerful computers for decoding the coded video message. This application scenario fits the DVC paradigm.

2.3.4 Free View-point TV

This is a futuristic TV viewing paradigm where the view to be watched by the viewer is determined by the viewer, thus making it necessary for additional processing to be undertaken at the decoder. An application scenario where the decoder actively participates in the coding process by selecting what to see is not envisaged by the conventional coder-decoder which expects the decoder to be a "slave" that faithfully reproduces whatever it

gets. DVC on the other hand readily accommodates this application scenario, since in the DVC paradigm the decoder is where the central processing unit and viewer preferences can easily be incorporated at the decoder.

2.4 Summary

This Chapter presented the theoretical foundations of DVC, which has been proposed by SW and WZ in the 70's whose practical implementation has only recently been pursued due to advances in video coding requiring more views and transmission from remote locations which mandate a paradigm shift from the costly encoder and cheap decoder conventional coding standards to a cheap encoder which accommodates a slightly more expensive decoder.

The architectural frameworks for the practical implementation of DVC that have been presented in literature were then presented, comparing and contrasting their various features, pointing out what works well and does not. This led to the presentation of the HOPTTI test bed architecture that was developed for the trialing and testing of ideas for this thesis highlighting its features and how it leverages from the existing architectures. Thus the HOPTTI framework is a highly modular and flexible architecture that has the various component parts of HOPPTI implemented as sub-modules called from the main DVC M-file which enables the testing and trialing of the ideas that form the major contributions in this thesis.

From the theoretical foundations and architectural frameworks presented, it was shown that SI generation is central to the improvement of DVC if it is to produce high quality output that compares favourably with the conventional coder-decoders. Therefore, a more comprehensive survey of SI generation in DVC is pursued in the next Chapter of this thesis

in order to present the case for the ideas presented in this thesis which were then tested and the results presented in subsequent Chapters.

Finally, examples of application scenarios that readily fits into the DVC paradigm was listed and how they fit into the DVC context is highlighted.

Chapter 3

Survey of SI Generation in DVC

3.1 Introduction to SI Generation

DVC and the practical implementation of the WZ theory has received a huge interest from the research community because of its perceived advantages in emerging technologies such as Multi-sensor Surveillance, Free Viewpoint TV (FVTV) and Three Dimensional Television (3DTV) over conventional coding represented by the International Telecommunication Union (ITU-T) Standard of AVC (Wiegand et al., 2003) and the Moving Picture Experts Group (MPEG) (Gall, 1991) coding standard. The availability of wearable webcams, mobile document scanners, mobile (wireless) video conferencing devices, etc, with limited resources in mobile phones and hand held devices is a major factor driving the above mentioned emerging technologies where these resource limited devices need to stream videos from remote locations.

While the conventional coding standards and implementation have been quite successful in the quality of their outputs, it is agreed by many researchers

(Brites et al. 2013; Puri, and Ramchandran, 2002.; Aaron, Rane and Girod, 2004; 2000; Holman, 2000; Ortega, 2007; Pandit, Vetro and Chen, 2008; Yeo and Ramchandran, 2007; Guo et al., 2008; Ouret, Dufaux and Ebrahimi, 2007; Artigas et al., 2007; Puri et al., 2006; Aaron, Zhang and Girod 2002) that the algorithms they employ are too resource intensive for use by these portable resource limited devices that are now widely available.

The theories of SW and WZ provide the theoretical foundations for the paradigm shift to move the resource intensive algorithms used by conventional codecs to exploit the correlation in videos (code video) from the encoder to the decoder i.e. away from the resource limited encoder to a powerful decoder, theoretically without compromising the quality of the output as discussed in Chapter 2.

The exploitation of correlation at the decoder where the actual video frames (source information) are not available is, however, not trivial and this is the root of the challenge posed by the new DVC paradigm. Though there had been a lot of previous work, the first set of successful practical implementation of WZ codecs and their architectures presented with detailed comparison in Chapter 2, were reported more than thirty years after the theories were published (Artigas et al., 2007; Puri, et al., 2006; Aaron, Zhang and Girod, 2002; X. Li, 2008; Varodayan, Aaron and Girod, 2006; Ascenso, Brites and Pereira 2006; Guillemot et al., 2007).

A review of the reported performance of the practical implementations of WZ DVC codecs shows that there is a big performance gap between the qualities of the output both qualitatively (visual) and quantitatively, when compared to conventional codecs. This gap had persisted over time making reviewers (Brites et al., 2013; Ortega 2007; Guillemot et al., 2007; Pereira et al., 2008) ask for radical new thoughts and approaches if this gap is to be closed. The reviewers identified quality of the SI as a major bottleneck to the performance of the WZ codec and it is a consensus in literature as discussed in Chapter 2

that SI generation is central to DVC performance. While there are surveys on DVC generally where the overall improvement in DVC is discussed, the growing body of work that focuses primarily on SI generation has shown it as the single most important bottleneck of DVC that deserves particular attention and as such will benefit from a separate and in-depth review. An in-depth review is critical to identifying the gaps of what is left undone and modifications of what is not being done properly that continue to make SI generation a bottleneck to DVC performance. Furthermore, in this chapter, presentation is made of the tools that would enable this SI generation bottleneck to be overcome.

3.2 Theoretical Background of SI Generation

The theoretical underpinnings of DVC SI generation at the decoder can be traced to the information theoretic proofs for the WZ (Wyner and Ziv 1973) and SW (Slepian and Wolf 1973) theorems as discussed in Section 1.1.2 and Section 2.1. In order to prove "faithful reproduction" (Slepian and Wolf 1973) of correlated sources, in the SW theory certain assumptions were made, one of the fundamental assumptions is that there are two encoders drawing independently from a bivariate distribution, however the ability of the joint decoder to decode the encoded streams depends on further assumptions of the information available to it both in itself and from the encoder. In Slepian and Wolf (1973), sixteen different classes of information made available for the faithful reproduction were examined, noting some interesting combinations and their effect.

WZ further introduced the concept of the fidelity criterion where the reproduction might not be exactly the same as the source, but only satisfies the criterion that it produces a decoded message that is compatible with the required fidelity. Therefore, WZ clearly relaxes further the requirements for the information necessary to produce a lossy version of the correlated sources at the decoder.

From the foregoing, it can be seen that the ability of the decoder to reproduce the so called "faithful reproduction" (Slepian and Wolf 1973) as well as a copy meeting the fidelity criterion (Wyner and Ziv 1973) depends hugely on the information (same as SI) available at the decoder. The importance of SI even from the theoretical underpinnings of DVC can therefore not be over-emphasized.

3.2.1 SI and Practical Implementation of DVC

The earliest practical implementations of DVC that discusses SI are the works of A. Aaron et al (Aaron, Zhang and Girod 2002; Jagnohan, Sehgal and Ahuja 2002) where the simplest scenario of taking the adjacent video frame as the SI was considered in the implementation of SI generation (Jagnohan, Sehgal and Ahuja 2002) as shown in Figure 3.1

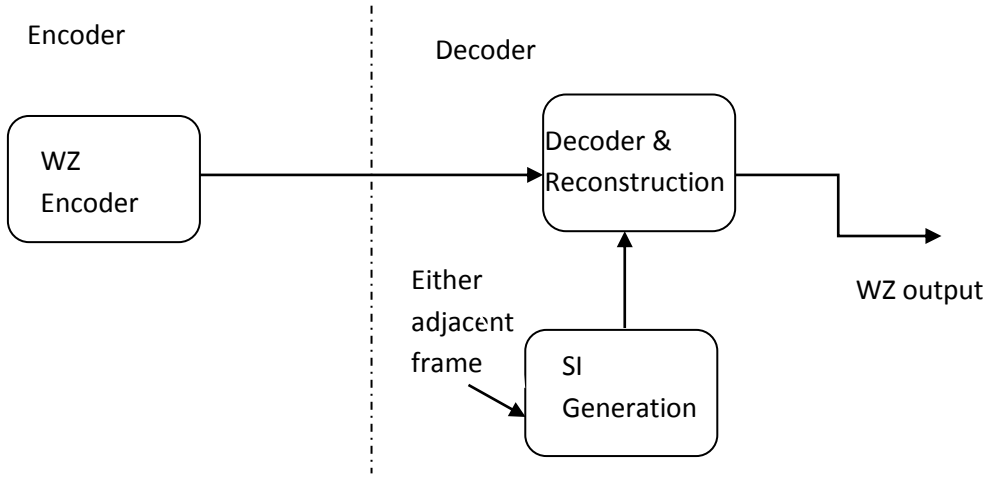


Figure 3.1 Illustration of one of the simplest SI generation scenarios (Jagnohan, Sehgal and Ahuja 2002)

Then the idea of "smart" implementation of SI generation was also discussed where the SI that is closest to the original is generated, for instance, which adjacent frame is to be chosen, is it the previous adjacent frame or the later adjacent frame.

Furthermore, various means of generation of SI that would improve the quality of SI can thus be flexibly considered with Aaron, Zhang and Girod (2002) and Jagnohan, Sehgal and Ahuja (2002) considering bi-directional interpolation.

3.2.2 DVC Bottlenecks and SI

The SI generation problem is fundamental to DVC theoretical foundations. Information theoretic proofs assumes that the decoder is capable of generating multiple cosets which is the SI, with one of the cosets being the original. The decoder essentially then has to predict in a one step problem the right coset (Jagnohan, Sehgal and Ahuja 2002). However, the problem in practical implementation of DVC is that the SI can be generated by a decoder through various means but none will be the original. Therefore, the problem practically becomes a WZ coding problem which has a two-step solution in which in the first step the best coset is generated (but can never be as good as the original) and the second step is a refinement process using more bits from the encoder to reach pre-determined fidelity criterion or the upper bound. While (Jagnohan, Sehgal and Ahuja 2002) recognized this problem, they did not present any practical solution as noted in (Aaron, Zhang and Girod 2002).

The two steps in the WZ problem depend on each other as can be seen above. The more fundamental bottleneck is the generation of the SI, as the bits for refinement can be exhausted for poorly generated SI before the required SI fidelity is attained.

3.3 SI Generation

In DVC literature, the generation of SI has always played a prominent role. The quality of the SI as the building block for the final decoder output has been noted (Brites et al., 2013; Ouret, Dufau and Ebrahimi 2009; Pradhan, Chou and Ramchandran 2003; Puri and Ramchandran 2002; Aaron, Rane and Girod 2004; Pradhan and Ramchandran 2003; Varodayan, Aaron and Girod 2006; Ortega 2007; Pereira et al. 2008). In the earliest architectures described in Chapter 2, starting with DISCUS (Pradhan and Ramchandran

2003) that uses binary sources, the generation of the SI, which in this case are the cosets at the decoder plays a prominent role in the failure (error) rates. In fact the results presented (Pradhan and Ramchandran 2003) shows that with the employment of more sophisticated trellis codes, DISCUS gains 3 – 4dB for increasing the possibility of generating of generating error free cosets by the use of channel codes.

In PRISM (Puri and Ramchandran 2002), the authors described an SI generation scheme that uses a half pixel motion search in the window surrounding the block under consideration in order to generate high quality SI and in turn have a high quality output at the decoder. This becomes necessary especially as the sources being investigated move from the binary sequences to more complex and natural sources. The DISCOVER (Artigas et al. 2007) architecture, one of the most prominent DVC implementations and well documented research platforms, also acknowledged the importance of SI. DISCOVER introduced the famous advance temporal interpolation scheme for SI generation which is the same as the linear *motion compensated temporal interpolation* (LMCTI) that has become the most widely used in SI generation (Pereira et al. 2008; Artigas et al. 2007; Li 2008), with a lot of proposed enhancements including hierarchical temporal interpolation (Liu, Yue and Chen 2009) and spatially-aided SI generation (Ye et al. 2009). LMCTI provides reasonable SI quality for sequences with slow-to-medium object motion; it tends not to be so successful for sequences exhibiting non-linear motion (Ye et al. 2009). Other enhancements to LMCTI includes the 3-D content *adaptive recursive search* (3DCARS) (Borchert et al., 2007) which employs a quarter-pixel *Motion Vector* (MV) search implying 16 more MV searches than other MB based search algorithms. The *pixel-domain WZ* (PD-WZ) codec (Tagliasacchi et al. 2006) also improved the pixel domain SI generation algorithm by spatial smoothing.

Moving away from LMCTI, higher-order trajectories (Chahine, 1995; Chahine and Konrad 1995) for temporal variations have been modeled, leading to more accurate sequence reconstructions, at the cost of greater computational complexity at the decoder. Though this is of less consideration in DVC generally, it might become critical depending on the application. One of the major drawbacks of temporal SI generation is that the fast, block based MV generation algorithms, which AOBMC has been employed in a number of variants to tackle and to improve SI quality, bring about artifacts and overlapping (ghosting). Examples include *motion compensated frame interpolation and adaptive object block motion compensation* (MCFI-AOBMC) (Choi et al. 2007) for instance, where bilateral LMCTI is applied to overcome both hole and overlapping problems by coupling AOBMC with an object segmentation and MV clustering technique. Also, in *improved side information generation for distributed video coding* (ISIG-DVC) (Huang and Forchhammer, 2008), AOBMC is combined with a variable block-size refinement algorithm to produce improved SI, while the *low complexity motion compensated frame interpolation* (ALCFI) (Zhai, Yu and Li 2005) utilizes AOBMC together with MV smoothing. These AOBMC-based algorithms all attempt, to some degree, to address the blocking artifacts and overlapping issues caused by BMA by using AOBMC. A related issue is the fact that depending on the spatial-temporal characteristics of the video, some frames and blocks are not improved by AOBMC therefore it might not be necessary to expend further computational overhead in applying AOBMC to such frames or blocks. This led to the investigation of a combination of the higher-order HOPTTI algorithm with AOBMC in a way to both enhance the quantitative and perceptual SI quality by increasing peak signal-to-noise ration (PSNR) and reduce BMA induced artifacts while at the same time improving the efficiency of the codec.

3.3.1 Pixel Domain SI Generation Scheme

One of the earliest DVC implementation was the pixel domain DVC introduced by A. Aaron et al (Aaron, Rane and Girod 2004) where the average pixel value of the two adjacent key frames to the WZ frame is used to generate the SI. In (Aaron, Rane and Girod 2004), this SI generation scheme is referred to as *average interpolation* which averages the two pixel values from the same position of the adjacent frames. If for instance the pixel at location j of the previous adjacent frame is X_{2i-1j} while the pixel at location j of the future adjacent frame is X_{2i+1j}, then the SI for location j will be calculated as $\frac{1}{2}(X_{2i-1j} + X_{2i+j})$.

More sophisticated spatial, pixel domain interpolation have been used and they include the edge directed interpolation scheme introduced by Li (2008). The edge directed interpolation scheme, while giving reasonably high quality SI is computationally and memory intensive thereby necessitating intermediate storage of output.

Though some DVC architectures continue to use the pixel domain encoder, they have opted to generate the SI by additionally employing the use of motion compensated interpolation which is a temporal rather than spatial, pixel domain SI generation scheme. One of such DVC architectures is the one introduced by Tagliasacchi et al. (2006), where the pixel based SI referred to as Y^{S}_{2i} is combined with the temporal SI generated by motion compensated interpolation referred to as Y^{T}_{2i}. The combination of the pixel domain SI and the temporal domain SI is achieved on a pixel by pixel basis by examining the one that gives the closest estimate to the original, referred to as X_{2i}. Therefore, a selection is made based on the SI that gives the minimum difference to the original and the SI that emerges is then referred to as the spatial-temporal SI Y^{ST}_{2i}. However the problem that arises is the fact that in DVC, the original is not known at the decoder and an estimate of the original is made based on potentially erroneous data at the decoder.

3.3.2 Temporal Domain SI Generation Scheme

By far the most versatile and widely used SI generation technique is the temporal domain SI generation schemes. The temporal SI generation schemes involve the modeling of the movement of pixels between frames and the assumption or calculation of a trajectory which enables the estimation of the location of pixels in intermediate frames. The major tools used in the temporal domain SI generation are *motion estimation* (ME) and *motion compensation* (MC). The method of estimation of motion and its subsequent use to locate the MBs in the intermediate frame is illustrated in Figure 3.2.

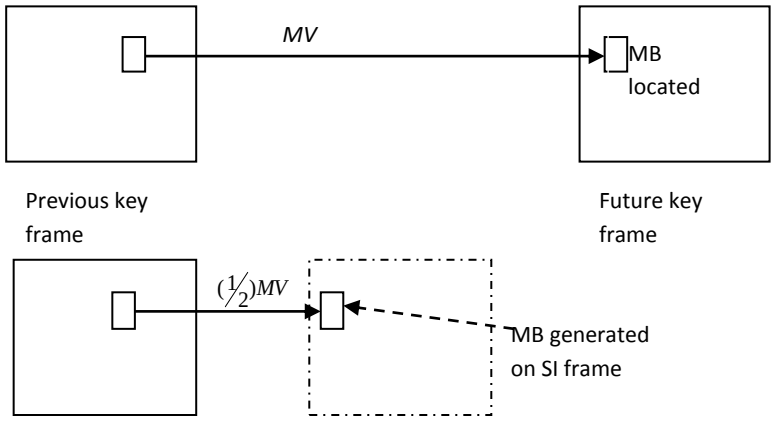


Figure 3.2 Temporal domain SI generation

In Figure 3.2, the previous frame is used as reference and the MV represents the location best match for the MB in the future frame. Also, it is assumed that the MB motion is linear and the position in the intermediate frame is thus given by half of the MV between the previous and future frames.

3.3.3 Reducing Encoder Complexity in DVC

In video coding where we deal with the removal of redundancies either to improve storage efficiencies or transmission efficiencies, it is necessary to encode only frames or blocks that vary while deliberately not coding other video frames.

In DVC, in order to reduce encoder side complexity, frames are deliberately split into WZ frames and key frames (KFs). In the simplest case, every other frame is dropped, the so called *Group of Picture* (GOP) of 2.

In order to regenerate the WZ frames knowing the location of the pixels in available (key) frames is not enough. The need therefore arises for a method to estimate the location of the various pixels in between the key frames. The knowledge of how the pixels traveled between key frames, referred to as the trajectory therefore becomes all important in order to accurately regenerate the dropped frames.

The simplest trajectory model is to assume that there is no motion between key frames. This model just replaces WZ frames with available KFs. While this might be adequate for some video sequences that contain objects that are mostly stationary such as (i) a person standing in front of a door phone or (ii) someone reading the news in front of a stationary background, the quality of such models deteriorates very fast immediately there is the slightest motion. As a matter of fact in the news scenario where the video includes sound it is quickly apparent as it seems as if the news is not been read by the newscaster based on the perception from the movement of the mouth region which incidentally is the region where most viewers would concentrate their gaze. Figure 3.3 shows two possible implementations of this scenario, where Figure 3.3 a) makes use of the (t+1)th KF as the intermediate frames that replaces the missing or dropped frames and Figure 3.3 b) makes use of the previous (t)th frame as the intermediate frame.

Another trajectory model which is by far the most widely used model (Aaron, Setton and Girod 2003; Aaron, Zhang and Girod 2002; Pereira et al. 2008; Artigas et al. 2007; Li 2008; Liu, Yue and Chan 2009; Ye et al. 2009), is the one that assumes that video objects have linear motion. This model assumes uniform motion (which implies constant velocity). The major advantage of this model is its algorithmic simplicity as intermediate

frames can be readily generated by finding an average distance that objects have moved between available key frames. A simple illustration of the linear trajectory model is shown in Figure 3.4 where the intermediate frame is generated by assuming linear trajectory and is simply executed by averaging the MV between the key frames. While the linear trajectory model is algorithmically attractive, it is not an adequate motion trajectory model. It is therefore not surprising that a raft of algorithms to improve this trajectory model exists in DVC literature as noted in Section 3.1.

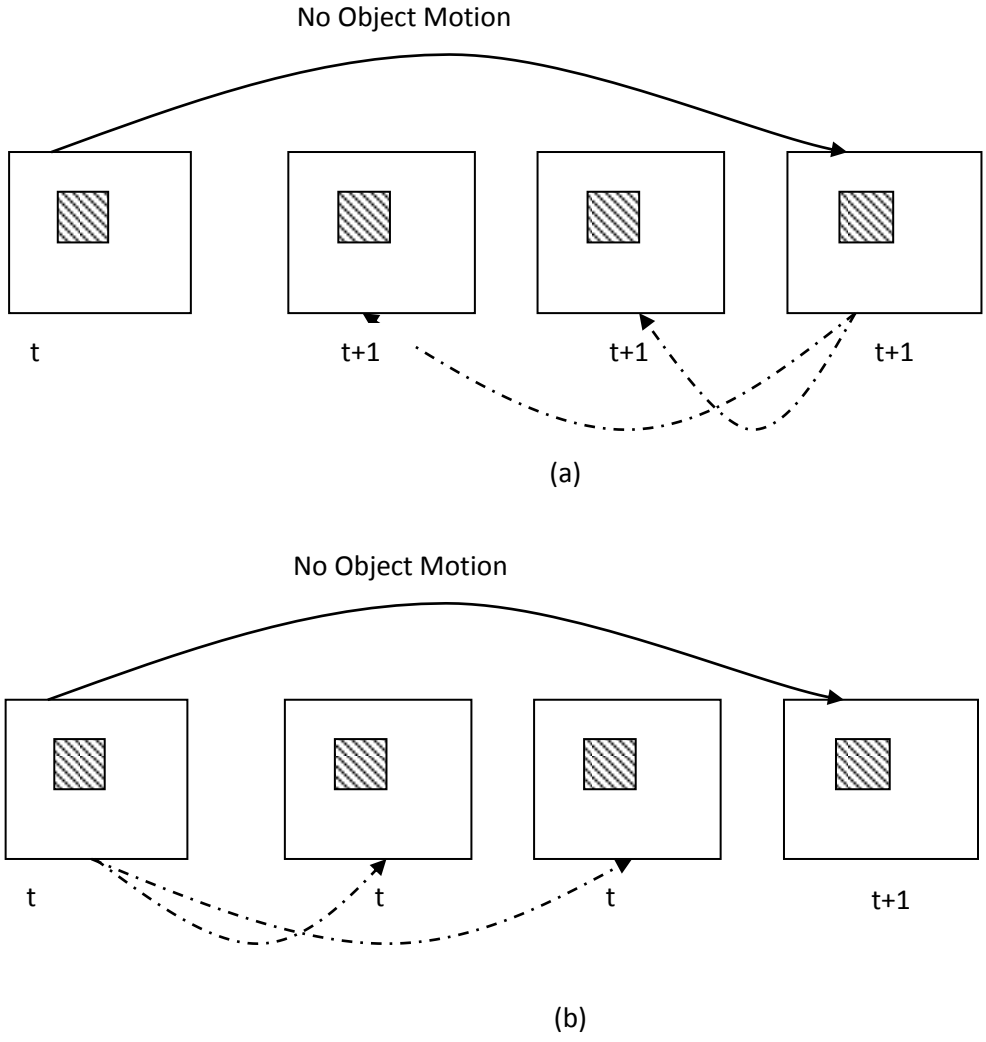
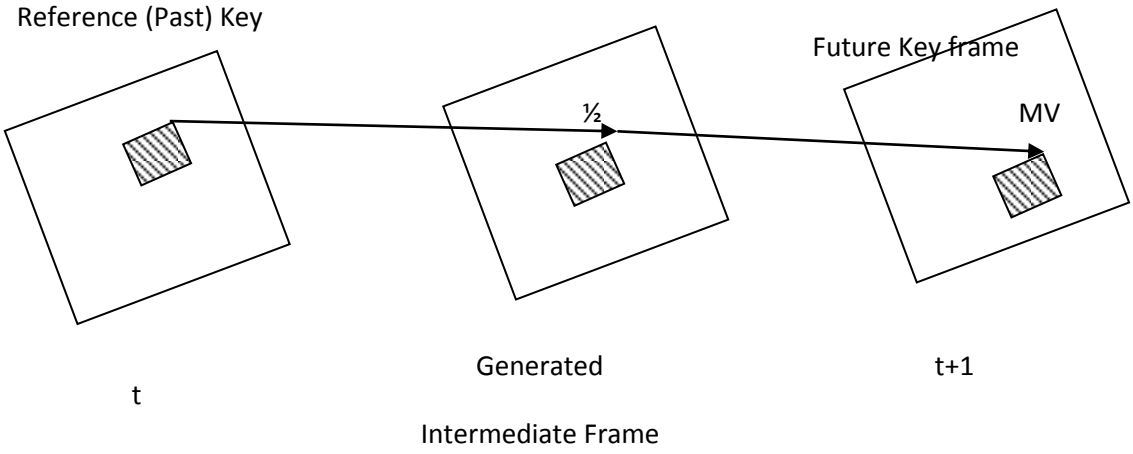
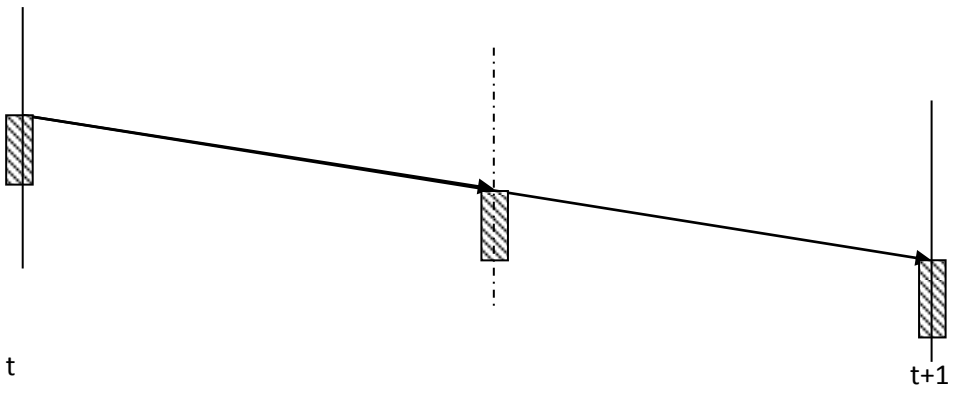


Figure 3.3 Scenario assuming no object motion. (a) the (t+1)th frame used as Intermediate frames (b) the (t)th frame used as Intermediate frames

Literature, (Aaron, Setton and Girod 2003; Aaron, Zhang and Girod 2002; Pereira et al. 2008; Artigas et al. 2007; Li 2008; Liu, Yue and Chan 2009; Ye et al. 2009) shows that it is essential to use higher order trajectories in order to be able to regenerate intermediate frames with reasonably high quality, as is needed for DVC to be competitive as conventional coding, which has the advantage of coding (exploitation of redundancies) at the camera side where the original is present as stated in Chapter 1.



(a)



(b)

Figure 3.4 Linear Trajectory (a) front view and (b) cross-sectional view

3.3.4 Practical Linear Motion Compensated Temporal Interpolation (LMCTI)

In the practical implementation of the ME and MC algorithms for temporal interpolation schemes of SI generation several different underlying problems have arisen and different solutions have been proposed to ameliorate the problems.

First, the problem of MV search which leveraged from the BMA of conventional video coding paradigms that have developed lightweight, fast and reliable ME. Conventional coding solutions were able to have access to the residual information from the original frames, while DVC does not have access to such and the effectiveness of the BMA algorithms was compromised leading to the observation of serious artifacts and holes in the SI subsequently generated (Wu et al. 2009).

3.3.5 Improvements to Linear Motion Compensated Temporal Interpolation (LMCTI)

Due to the perceived inadequacies of LMCTI for SI generation coupled with the growing realization of the importance of SI quality to overall DVC performance, various researchers have proposed improvements to it. It is worth looking at a few of them, signposting the developments in DVC SI generation.

a) Hierarchical Linear Motion Compensated Temporal Interpolation (LMCTI)

In this model introduced by Liu et al. (2009), the MVs generated by LMCTI are refined for more accurate MVs by the use of the bi-direction motion estimation (BDME) technique on the basis of variable block size hierarchical motion estimation. BDME technique using smaller and smaller block sizes to estimate MV is employed in order to see if the MVs can be improved and accuracy improved. Furthermore, motion vector filters are used to correct false MVs due to partial similarities in the video sequences in conjunction with a decision mechanism to handle overlapping caused by varying the block sizes.

b) Spatially Aided LMCTI

The spatially aided LMCTI (Wu et al. 2009) scheme uses spatial information to aid partially decoded WZ frames from linear motion compensated temporal interpolation. The algorithms implemented involve MV refinement and smoothing in conjunction with mode selection for optimal compensation. This scheme also implements an error concealment algorithm which conceals errors in the final SI frames generated making their subjective appearance more pleasing to the eye.

c) 3-D content adaptive recursive search (3DCARS) LMCTI

The 3-D content adaptive search (3DCARS) LMCTI scheme (Borchert et al. 2007) introduces a more demanding MV search algorithm which utilizes quarter pixel search and thus produces more accurate MVs. This scheme was used in order to get true motion estimates for objects in video sequences. Though there is improvement in DVC performance, the fact that the true motion is obtained could not compensate adequately for the linear trajectory that was assumed.

d) Low Delay Linear Motion Compensated Temporal Extrapolation (LMCTE)

In low delay applications that employ the use of resource poor encoders, it might be necessary to reduce delay at the decoder at the expense of performance. Thus, extrapolation schemes that make use of only past frames to generate SI have been examined by Wu et al (Borchert et al. 2007) and (Wu et al 2009). Due to the fact that this scheme does not have to wait for future frames it inherently presents a low delay advantage though extrapolation being a subset of interpolation, its performance is always slightly below that of similar interpolation schemes that use the same scheme. In hierarchical schemes where the LMCTI and LMCTE is employed and this step has to be completed

before any other step can begin, considerable time delay could be incurred by LMCTI schemes that is absent in LMCTE based schemes.

e) Hybrid Block and Mesh Based Motion Estimation and Interpolation Approach

A very complex SI creation approach introduced by Kubasov et al. (2006) enhances the SI quality when spatial domain motion discontinuities and occlusions are present in the video sequence. Though experimental results show that this approach gives a better RD performance when compared with similar pixel domain DVC codec with only the block based approach of generating SI, the complexity associated with this method made the block based approach to be embraced by the DVC community rather than the mesh based approach or this hybrid method.

f) Motion Field Estimation SI Generation Scheme

Generation of SI by the use of MV smoothness constraint on the motion field estimation between two previously generated frames at time instants, t-2 and t-1 and extrapolating the MV for the current frame is an approach that was proposed by Aaron et al. (2004). This MV estimation approach was coupled with overlapped motion compensation using pixel values from the frame at t-1. Experimental results however, show that the SI generation by extrapolation technique leads to a significant RD performance loss when compared to the frame interpolation approach.

g) Context Based SI Pixel Generation

A method where each SI pixel is created as a weighted average of pixel values referred to context, at the same spatial location in previous reconstructed frames is proposed by Li et al. (2006). The initially reconstructed frames are generated by LMCTI and the weight corresponds to the number of occurrences of corresponding context in the four previously reconstructed frames. The proposers of this scheme claim that the pixel based DVC

decoding complexity is significantly reduced with respect to an equivalent video codec with pixel based motion compensated interpolation while having comparable overall coding performance for low motion video sequences.

h) Multiple SI Generation by both Interpolation and Extrapolation

A multiple SI generated by means of linear frame interpolation and frame extrapolation is proposed by Huang et al. (2009). A combination of the available SI estimates is achieved through the corresponding correlation noise models and used to provide more accurate soft input data to the LDPC decoder, reducing the required parity rate for the target decoded quality. The authors reported RD gains of 0.4dB at high bitrates compared to SI generation with single SI generated by interpolation techniques. This improvement is not a qualitatively visible improvement which requires PSNR increase of at least 0.5dB to be visible in the video codec output sequence (Girod, 1993; Yang et al. 2005).

i) Motion Compensated Refinement Techniques

Ascenso et al. (2005) propose generating SI by continuously enhancing the decoded frame quality as WZ bits are received. In their framework, the initial SI is also created from linear frame interpolation which is then continuously refined. The results show that there is improved RD performance over an equivalent video codec without refinement.

j) Spatial-Temporal Refinement Method

A spatial-temporal refinement technique to improve SI was proposed by Weerakkody et al. (2007). In this method, initial SI is generated by linear frame extrapolation which is then interleaved for error estimation and flagging. The de-interleaving process then fills the flagged bits with alternative bits generated by an iterative spatial-temporal prediction technique. Experimental results also show improved RD performance over similar pixel domain DVC codec without the proposed method.

k) Expectation Maximization Algorithm

An expectation maximization algorithm which performs unsupervised learning of the disparity at the decoder by exchanging soft information is proposed by Varodayan et al. (2007). The soft information is obtained from the LDPC decoder and a probabilistic motion estimator and used to iteratively refine the SI. Experimental evaluation of the approach by the authors shows an increasingly better RD performance for increasing GOP sizes.

l) Spatial-Temporal Refinement of SI by Mode Selection

Exploitation of the spatial-temporal correlation of partially decoded WZ frames and reference key frames by MV refinement and smoothing and mode selection was proposed by (Ye et al. 2009). The resulting enhanced SI frame is then employed to perform another reconstruction, reducing the distortion in the reconstruction process while at the same time increasing the overall RD performance of the DVC codec.

m) Block Classification Gain, Coded Information, implicit and explicit MC for SI generation

A method to exploit the so called "block classification" gain and coded information (CI) was proposed by Li (2008). In this approach the SI is generated based on two methods, the explicit method based on multi-hypothesis MC temporal interpolation with sub-pixel accuracy and second method based on implicit MC with the Least-Square approach. An average of the SI generated from both approaches is then taken as it was noted that the explicit method provides better improvement for fast moving video sequences such as *American Football*, while the implicit method provides better SI for slow moving video sequence such as *Container*. The final SI is then combined with CI which is received parity bits for final reconstruction of the video frame.

n) SI generation by Refinement Based on Learning Approach

A proposed method to generate SI by refinement where the DCT bands to be decoded are successively improved as the decoding process continues is presented by Martins et al. (2009), and (2010). In the technique, the initial SI alongside reconstructed frame for already decoded bands of DCT in the pixel domain is employed to decide which SI blocks should be refined. The already decoded DCT bands provide information about the original WZ data that was not available during initial SI generation. Refinement is performed using displaced blocks and as such no explicit updating of MV field occurs (Martin et al. 2009). In their subsequent paper (Martin et al. 2010) a learning process which defines the relevance of the SI displaced blocks considering the previously decoded DCT bands that explicitly updates the MV field was introduced. Experimental results show RD performance improvements for high motion content video sequences and long GOP lengths.

3.3.6 Analysis of the Surveyed SI generation Schemes

The fundamental commonality with the entire SI generation schemes surveyed is that they have employed the linear frame interpolation either in the spatial (pixel) domain or in the temporal domain. The schemes have tried to leverage from the simple linear frame interpolation to correct the poor quality of SI produced by the linear interpolator.

There is a general consensus that linear frame interpolation gives a better SI than the linear extrapolation as there is always a RD penalty on extrapolation based SI.

A number of fundamental directions for the study of the SI bottleneck emerging from the survey therefore include the following:

(i) LMCTI is predominantly the algorithm of choice for generating SI in current DVC literature. It is used for generating the SI because of inherent advantages in its easy

formulation and the use of fast block based MV algorithms already well researched and documented in literature. At the very least it is used as the initial starting point for SI generation.

(ii) A study of the frame interpolation process is paramount in order to be able to proffer solutions and overcome the SI bottleneck. The first inclination from the survey of literature (e.g. Li 2008) is to explore the possibility of higher order frame interpolation, as object motions in naturally occurring scenes are non-linear. The use of two past frames in extrapolating a present frame has been suggested in Aaron et al. (2004).

(iii) A fundamental understanding of the MV search algorithms is important. Variation of the search algorithm in Borchert et al. (2007) is an example. The creation of a set of smooth MVs and removal of outliers has been proposed.

(iv) The various video sequences with their spatial-temporal characteristics need to be studied as the various algorithms have shown that some SI generation algorithms are better with slow sequences while others are better when the video sequences contain fast moving objects.

(v) Low delay SI creation is assumed to be generated by extrapolation without visible experimental results showing better times and computational costs. The fact based conclusion that can be reached from the survey of literature is that there is a RD lowering cost to pay when extrapolation is employed compared to interpolation.

(vi) Intelligent learning algorithms needed to be explored as the various mode changes and learning based algorithms (Ascenso et al. 2005; Weerakkody et al. 2007; Martin et al 2009; Martin et al 2010) have shown there is a gain in learning and intelligently applying what is learnt in improvement of the SI.

(vii) The use of auxiliary data such as CI proposed by Li (2008) points to the inadequacy of the initial SI generation techniques (notably LMCTI). The incorporation of CI data in

the initial SI generation algorithms should be explored as SI generation is done at a powerful decoder that reduces the impact of increased overhead. At the same time, care must be taken so the decoder does not becomes several times more costly than overheads in the encoder of conventional codecs.

3.3.7 Higher Order Motion Compensated Temporal Interpolation

The inadequacies of LMCTI have been noted variously in literature, especially when working with real life video sequences with multiple objects that move with varying motion in the same sequence. This work incorporates improvements to LMCTI by using non-linear trajectory models for the generation of SI.

The earliest introduction of non-linear trajectory model in temporal interpolation was by Chahine (1995) and Chahine and Konrad (1995). Though they used a dense motion field which implies at least pixel by pixel MV generation and experimentation was with interlaced video, they showed that including an acceleration term can give substantial improvement over the linear interpolation model. Petrazzuoli, Cagnazzo and Pesquet-Popescu (2010) introduced the same quadratic (acceleration) trajectory model in DVC and showed improvement in DVC performance.

3.4 Exploiting Spatial-Temporal Redundancy in DVC

In order to achieve high video compression ratios, the temporal information contained in the video needs to be understood and exploited. Basically there are two sources of motion in video and they are (i) the motion of objects in the video and (ii) camera motion. While camera motion results in the global movement or displacement of all pixels in the frame, object motion is selective and differs between the objects in the video and then between objects and background. In order to exploit temporal redundancy for video compression, an

estimation of the motion present in the video must be carried out, and this is usually a computationally intensive step in video coding.

Video frames on the other hand which are the building block of video sequences are spatial two dimensional arrangements of pixels. The qualitative performance output of video processing which eventually determines viewer satisfaction is what matters and errors introduced into video sequences by transmission channels, storage mediums and rendition mediums eventually show up as artifacts (pixels being in wrong positions in the frame or being in the frame when they are not supposed to be there) which make the output different from the original video sequence.

This section therefore discusses the various algorithms and processing in SI generation that constitute fundamental architectural bottlenecks identified in the SI generation literature survey of Section 3.1 with the aim of eventually employing them to change SI generation and achieve major contributions to DVC performance.

3.4.1 Block Matching Algorithms For Motion Estimation

One of the points identified in the SI generation literature survey is the predominant use of block based ME for interpolation, and in order to effect fundamental changes in SI generation, it is necessary to understand how it works. Block based ME is one of the most versatile means of exploiting the temporal redundancies in video sequences that have been well researched and documented in literature. It is said to be the single most important (Ghanbari 1999) factor that accounts for the compression performance efficiency of conventional codecs as represented by the H.264/AVC standard, achieving up to 100:1 compression ratio (Ghanbari 1999). The underlying assumption in ME is that patterns corresponding to background and objects move within the frame to form corresponding backgrounds and objects in subsequent frames. Therefore, the current frame is divided into a matrix of 16x16 pixels called a macro-block (MB) that are then compared with a

corresponding block and its adjacent neighbours in the previous frame to create a motion vector (MV) that shows the movement of the MB from one location to the other. Figure 3.5 illustrates the search area for a good MB match, constrained up to *p* pixels on all the sides of the corresponding MB in the previous frame, where *p* is called the search parameter.

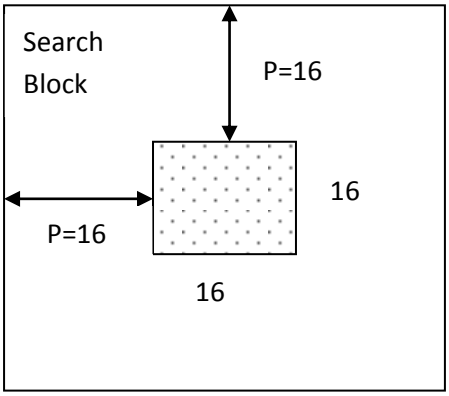


Figure 3.5 Illustration of Block Matching a MB of 16x16 pixels using a search parameter p=16

The selection of one MB in the reference frame with another in the current frame which best matches each other is based on the MB whose output of a cost function produces the least cost. Two such cost functions are mean absolute difference (MAD) represented by (3.1) and mean square error (MSE) represented by (3.2), where *N* is the size of the MB, C_{ij} and R_{ij} are pixels in the current frame and reference frames being compared, while i and j are the indices of the pixel in the MB.

$$MAD = \frac{1}{N^2}\sum_{i=0}^{N-1}\sum_{j=0}^{N-1}\left|C_{ij} - R_{ij}\right| \tag{3.1}$$

$$MSE = \frac{1}{N^2}\sum_{i=0}^{N-1}\sum_{j=0}^{N-1}(C_{ij} - R_{ij})^2 \tag{3.2}$$

PSNR makes use of MSE and is defined according to equation 3.3.

$$PSNR = \quad 10\log_{10}(255^2 / MSE) \qquad (3.3)$$

The value of 255 is the maximum possible pixel value in a gray-scale frame where pixels are represented by 8 bits.

The most basic block based ME using the BMA is the exhaustive search (ES) algorithm, this algorithm and improvements made to it as documented in literature is hereby presented:

a) Exhaustive Search

This ES algorithm is also known as the full search algorithm because it searches every possible location in the search window and calculates the cost function for everyone of such locations before deciding which one has the lowest cost function and thus the best match. The ES algorithm is highly computationally expensive, especially as the search window increases due to its exhaustive nature though it gives the best performance, usually measured in terms of PSNR of all the search algorithms. All other Fast BMAs try to achieve the same performance while reducing computation overhead compared to ES as much as possible. Some algorithms may include some scene specific weighting that though may not improve qualitative and quantitative performance but reduce overhead cost. For example, giving preference to blocks closer to the block under consideration during search will definitely reduce overhead cost but may not improve performance over ES, as trajectory continuity may suggest that parts of a solid object are not likely to suddenly move so far away from the other parts within the span of one frame difference.

b) Three Step Search (TSS)

The TSS is one of the earliest attempts to reduce the computational overhead of ES while keeping the quantitative performance as close as possible to the ES algorithm. The underlying principles of TSS is illustrated in Figure 3.6. Starting at the centre of the MB and setting up a "step size" S=4 for a usual search parameter of 7 it searches at eight locations +/- *S* pixels around the centre (0, 0).

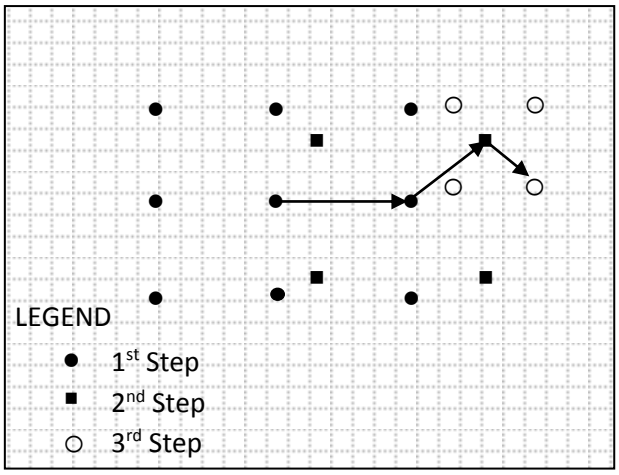


Figure 3.6 Illustration of the TSS algorithm (Li, Zeng and Liou 1994)

From the searches conducted in the first step described earlier, the one with the least cost is then selected to serve as the origin for a new search reducing the step size S, usually by half. The search is conducted for three such steps and the MB at the location of the point with the least cost is chosen as the best match. The TSS algorithm gives a flat rate computational cost reduction of a factor of 9.

Another variation of TSS is the New Three Step Search (NTSS) introduced by Li, Zeng and Liou (1994) and is one of the first BMA algorithms widely accepted into earlier conventional coding standards of MPEG 1 and H.261. While NTSS is similar to TSS in being centre biased, it has provision for a half way stop which further reduces computational cost.

c) Diamond Search

The diamond search (DS) algorithm (Zhu and Ma, 2000) is similar to the four step search algorithm (Po and Ma, 1996) with a diamond search pattern instead of the normal square pattern. In addition to the diamond search pattern, it has no limit to the number of steps that the search can take. DS employs two fixed patterns to undertake the search namely; the Large Diamond Search Pattern (LDSP) and the Small Diamond Search Pattern (SDSP). The DS procedure and the way the two fixed patterns interact are illustrated in Figure 3.7.

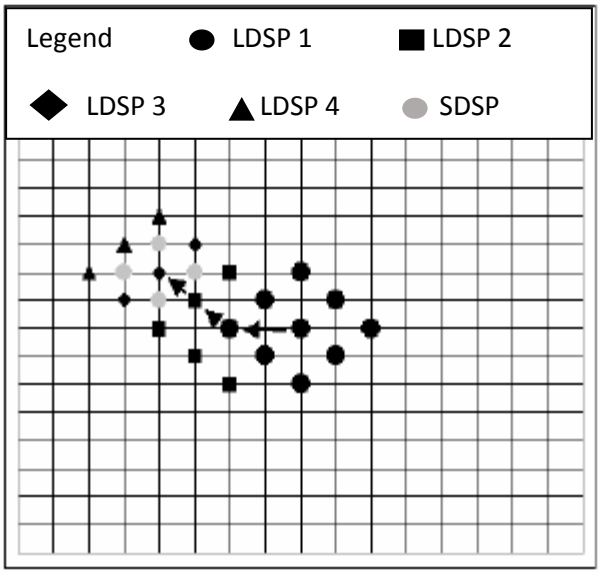


Figure 3.7 Illustration of the Diamond Search Algorithm (Zhu and Ma, 2000).

d) Adaptive Rood Pattern Search

The adaptive rood pattern search (ARPS) algorithm was developed by (Zhao et al. 2008) and it is the search algorithm that has been adopted in this thesis. It is therefore discussed in detail and its strengths and weaknesses highlighted. ARPS makes use of the fact that general motion in a frame is usually coherent, which implies that if the MBs around a particular MB moves in a particular direction, then the probability that the MB under discussion will as well move in the same direction is quite high. ARPS therefore employs

the MV of the MB to its immediate left to predict its own MV. The Figure 3.8 illustrates the use of the step size and the predicted vector in ARPS. The rood (cross-like) pattern search illustrated in Figure 3.9 is always the first step and this places the search in the area where probability of the best match being found is highest.

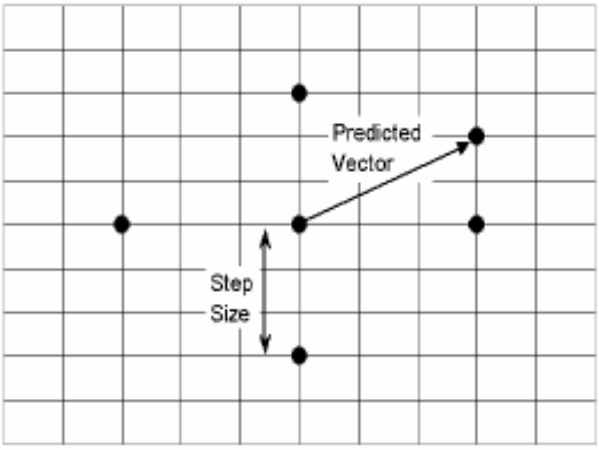


Figure 3.8 Illustration of the Adaptive Rood Pattern Search (ARPS) algorithm (Zhao et al. 2008)

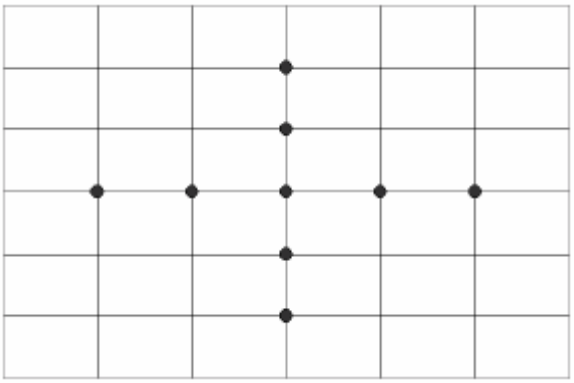


Figure 3.9 Illustration of the cross-like (rood) search pattern in ARPS

The main advantage of ARPS over DS is that if predicted MV is at the origin, it does not waste computational time and resources in a LDSP pattern search as in the DS but rather starts using SDSP straightaway. ARPS has been found to take a factor of 2 less computation compared to DS while PSNR performance is slightly better than DS. The fact that it gives a coherent MV set advocated in the literature review of Section 3.3.4 and

Aaron et al. (2004) and Borchert et al. (2007) and rejects outliers gives it a distinctive edge over the other MV search algorithms discussed in this section.

3.4.2 Motion Compensation

Using the set of MVs that are obtained during the motion estimation phase and the reference frame, a new frame with the blocks displaced can be generated and this is known as motion compensation. Usually, bidirectional motion compensation technique is employed in DVC and this makes use of both the current and future key frames as references and in most cases a simple average of the two frames from both directions is used to form the target blocks and frame.

Furthermore, in DVC, the motion estimation and compensation phases are carried out at the decoder where the original reference frame is not available and this means the residue which is the difference between original frames and the target frame is not available which is one major reason why the conventional codecs perform better (Brites et al., 2013; Semsarzadeh et al 2013).

3.4.3 The sources of visual artifacts in video

It is very important to identify the root causes of visual artifacts in image and video sequences in order to be able to proffer mitigating solutions that will reduce such artifacts or even eliminate it.

a) External Systems Artifacts and Sources

These artifacts are from sources external to the regeneration process and they include:

(i) artifacts existing in base video which are due to the fact that in image or video regeneration, the regenerated video is based on an existing image or video and the existing video might previously contain both latent and visible artifacts that becomes more acutely visible in the regenerated video. For example, in DVC, key frames used in

the regeneration of missing WZ frames are usually generated using some standard video codec such as H. 263 which might introduce its own artifacts

(ii) video acquisition and content creation artifacts which include artifacts caused by incorrect calibration or configuration of the camera (Cancellaro, Palma and Neri 2010)

(iii) artifacts from transmission medium noise and packet losses and

(iv) artifacts caused by the rendering medium, for example holographic displays can introduce their own artifacts while vacuum tube displays have also been known to introduce peculiar artifacts into video being viewed.

Other additional artifacts that are caused by external systems are; keystone distortions which distort the image dimensions, temporal mismatch, cardboard effect, jitter and color bleeding (Cancellaro, Palma and Neri 2010).

b) Computational Systems Artifacts and Sources

Image generation and regeneration involves a lot of computation and constitutes an important source of artifacts in images and video thus generated. In 1994, the classification of the sources of errors that lead to the introduction of artifacts in image generation and regeneration referred to as image synthesis was introduced by Arvo, Torrance, and Smits (1994). The sources of error were classified into three namely:

(i) errors due to the limitation of measurement or modeling, when models or measurements have to be approximated

(ii) errors due to digitization when analogue (continuous) data have to be digitized in order to make it possible to apply finite computer based operations on them

(iii) numerical precision errors when calculations are made with limited precision.

The computational errors enumerated above lead to different kinds of artifacts which include tessellation which is the repetitive visualization of some geometric form or shape, banding which includes blocky artifacts caused by quantization or coarse sampling and

aliasing which makes objects to appear jagged caused by over amplification of digitized objects in images (Arvo, Torrance, and Smits 1994).

c) Block Based Computational Artifacts

One of the most prominent root causes of computational artifacts in regenerated images is BMA and its assumption that a block of pixels can be handled as a single pixel. Also in DVC, computation artifacts in LMCTI were analyzed (Liu et al. 2010) by the use of texture and motion activity enabling the authors to locate the position of artifacts and to propose viable ways to remove the artifacts. Figure 3.10 shows the blocks of a frame from the *Foreman* sequence containing artifacts located after the texture and temporal analysis have been carried out.

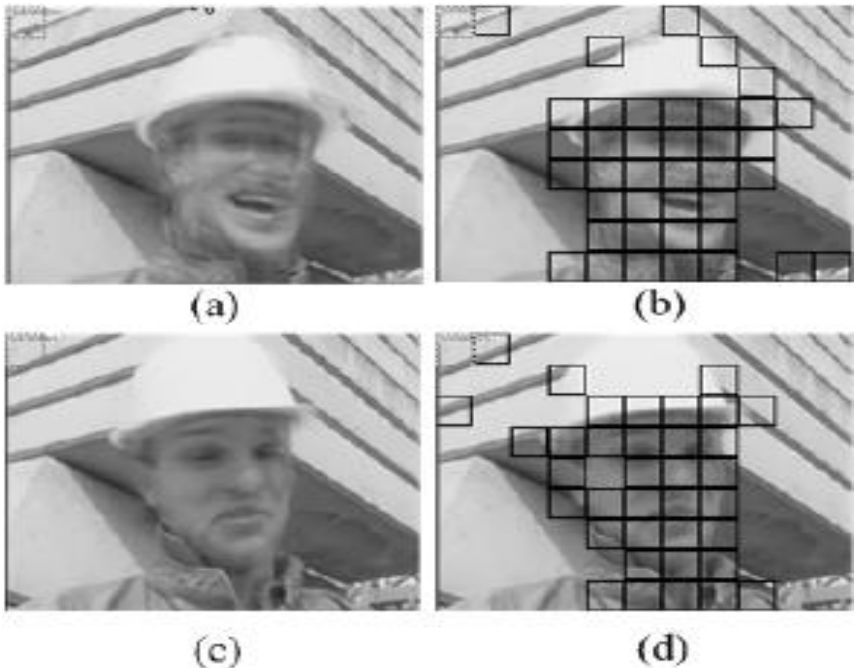


Figure 3.10 Sample DVC, LMCTI computational artifacts (a) frame #14 with (b) highlighting artifacts location (in blocks) and (c) frame #70 with artifacts location in (d). (Liu et al. 2010).

3.4.4 Review of Artifacts in DVC SI

In the earliest implementations of DVC, the need to show that it is possible to exploit correlation at the decoder was paramount and these DVC implementations experimented

with both synthetic and real life, no motion to very slow motion video sequences containing single objects. As interest in DVC increases, its performance compared to existing conventional codecs becomes important. Therefore, the need to experiment with real life sequences with medium to high motion content, containing multiple objects became important in order to enable adequate comparison with outputs of conventional codecs. Furthermore, the comparison was not limited to numerical (objective) comparison alone, but visual, qualitative (though subjective) evaluation needed to be carried out, thus the appearance of visual artifacts in DVC output had to be addressed.

Natario et al. (2005) reported that visual artifacts are prevalent in DVC SI and this affects DVC performance, so they proposed several SI artifact mitigating blocks in their SI generation module including motion field for overlapping and uncovered areas and blocks. The best methods to mitigate and remove artifacts in regenerated video sequences are those that are linked to the root causes of the artifacts. In Grouiller et al. (2007) artifacts are deliberately introduced into image sequences in such a way that that the causes of such artifacts are known and their characteristics studied (so called forward models) which enable artifact correction methods to be developed, ensuring that such correction methods do not introduce further artifacts of their own. Also, in Bosc et al (2011) thorough analysis of artifacts is performed before correction methods are proposed, and one notable finding of the analysis was that the type and location of certain artifacts are specific to the root cause of such artifacts, making it possible to develop generic method for mitigating such artifacts.

In DVC, various artifact mitigation solutions have been proffered which have a basis especially in the fact that artifacts begin to show up in DVC when there is motion in the sequence. One of the first sets of artifacts in DVC are mostly due to video capturing devices. Thus, no motion sequences such as head and shoulder news, performed very well but real life videos with camera movements such as panning and zooming exhibited

artifacts. Block 3 of the SI improvement framework (Figure 1.3), therefore develops a MS algorithm that employs the spatial-temporal characteristics to mitigate these artifacts.

3.4.5 Adaptive Overlapped Block Motion Compensation (AOBMC)

While higher-order interpolation with BMA for MV estimation has been shown promising results (Petrazzuoli, Cagnazzo and Pesquet-Popescu 2010), fast motion sequences with multiple objects still exhibited artefacts, signifying that there are issues to be resolved which are traceable to the use of block based computation in the higher order algorithms.

The AOBMC approach (Choi et al. 2007) allows the MV of a MB to be applied to larger groupings of pixels by using a raised cosine window. Specifically, situations where a MB either contains multiple objects with varying motions or one object traverses multiple MBs, so it is represented by different MVs, can be mitigated by this approach. The raised cosine window gives an enlarged window greater than the MB under consideration to allow the MV of the MB to be moderated by the MV of surrounding pixels in such a way that depends on the distance of the pixel from the afore-mentioned MB. This has led to AOBMC being employed in a number of variants to improve SI quality. In *motion compensated frame interpolation and adaptive object block motion compensation* (MCFI-AOBMC) (Choi et al. 2007) for instance, bilateral LMCTI is applied to overcome hole and overlapping problems by coupling AOBMC with an object segmentation and MV clustering technique. In ISIG-DVC (Huang and Forchhammer 2008), AOBMC is combined with a variable block-size refinement algorithm to produce improved SI, while *low complexity motion compensated frame interpolation* (ALCFI) (Zhai, Yu and Li 2005) also utilizes AOBMC, this time together with MV smoothing. These AOBMC-based algorithms all attempt to a varying degree, to address the restrictions caused by BMA by using LMCTI in SI generation.

3.5 Employing Video Content Characteristics, Processing Mode Changes and Artificial Intelligence for Improvement of SI Generation

In order to achieve high video compression ratios, the spatial-temporal information contained in the video need to be understood and exploited. Basically there are two sources of motion in video and they are (i) the motion of objects in the video while background remains static, these consist mainly of localized motion, (ii) the motion of background and objects simultaneously which consists mainly of global camera motion such as zooming and panning.

In Section 3.2, one conclusion is that intelligent learning algorithms needed to be explored as the various mode changes and learning based algorithms (Ascenso et al. 2005; Weerakkody et al. 2007; Martin et al 2009; Martin et al 2010) have shown there is an advantage in learning and intelligently applying what is learnt in improvement of the SI.

In order to incorporate learning, adaptation, reasoning and evolution capabilities into the SI generation paradigm, it is necessary to evaluate the various computational intelligence algorithms that are available that enable machines to be taught to interpret possible variations in video data such as variations in object movements and patterns.

Some of such computational intelligence algorithms that would be examined here will form a basis for the selection of intelligence algorithm in the major contribution Chapter 7 later in this thesis. We would examine Neural Network (NN), Fuzzy Logic System (FLS), Support Vector Machines (SVM) and RST.

3.5.1 Neural Network

One of the earliest machine learning algorithms is the NN which has biologically inspired roots and it has found widespread acceptance for being a robust system for information processing (Craven and Shavlik, 1997).

Basically, NNs are electronic simulation of the brain and how it functions, thus it comprises of several artificial neurons (AN) that are arranged in such a way and in such numbers so that it can be used to perform a task. An AN, illustrated in Figure 3.11 comprises of several inputs associated with its own weights illustrated by the red circles which can be adjusted during training, a nucleus represented by the black circle in our illustration and finally an output.

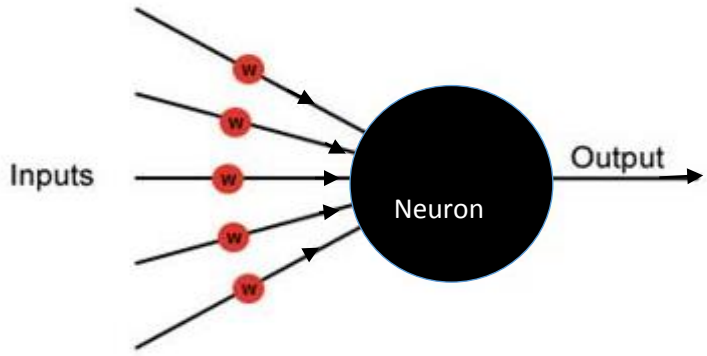


Figure 3.11 Illustration of an Artificial Neuron

As indicated above, these ANs can be arranged in different ways and numbers to form an NN. An illustration a *feed forward* arrangement to form an NN is in Figure 3.12. This arrangement is call a feed forward arrangement because of the way neurons feed their output forward to the next layer until a final output is obtained.

The optimal structure of the learning and organizational dynamics of NN is still not satisfactorily known, though there has been intensive research from many different fields applying NN, thus taking the myth away from the hidden layers of neurons and neural processing. One key drawback of NN is that biological systems are organized in a completely different way compared to artificial computing system. Thus it is usually difficult to adapt NN to computing systems. NN are powerful classifiers when some explicit prior knowledge of underlying probability distribution is known.

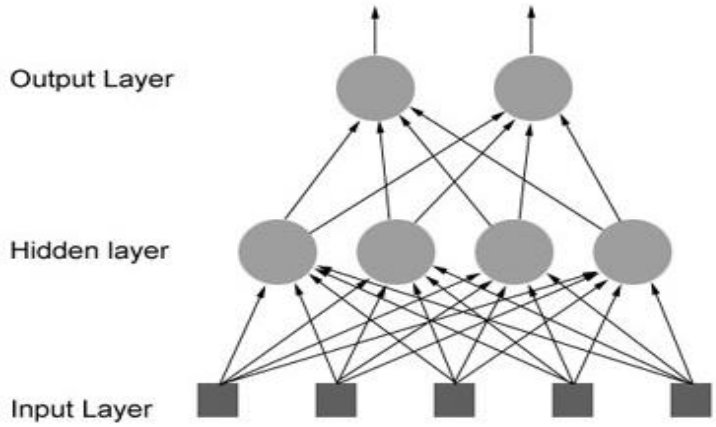


Figure 3.12 Illustration of a feed forward NN

3.5.2 Fuzzy Logic System

The concept of Fuzzy Logic System (FLS) was conceived by Lotfi Zadeh, a professor at the University of California at Berkeley (Zadeh 1996). FLS can be defined as the non-linear mapping of an input data set to a scalar output and usually consists of four components illustrated in Figure 3.13 which are; (i) Fuzzifier, (ii) Rules, (iii) Inference engine and (iv) Defuzzifier.

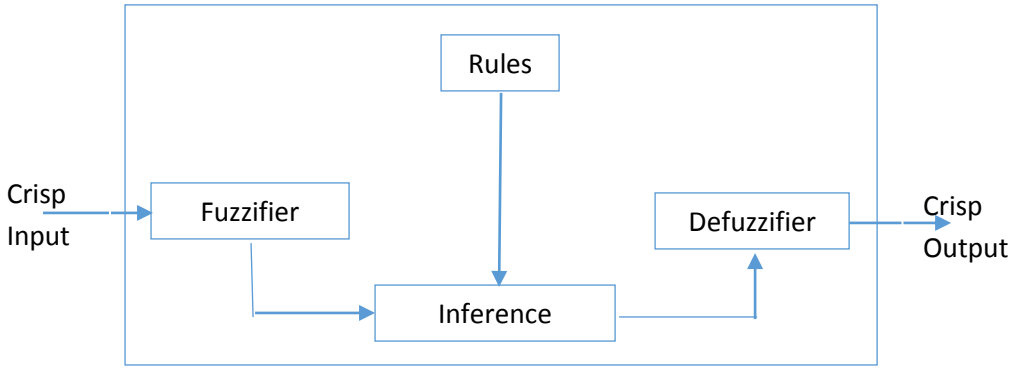


Figure 3.13 Illustration of the various components of the FLS

The FLS follows the following set of algorithm for its implementation which involves a four step cycle process of initialization, fuzzification, inference and defuzzification.

(i) Define the linguistic variables and terms (initialization)

(ii) Construct the membership functions (initialization)

(iii) Construct the rule base (initialization)

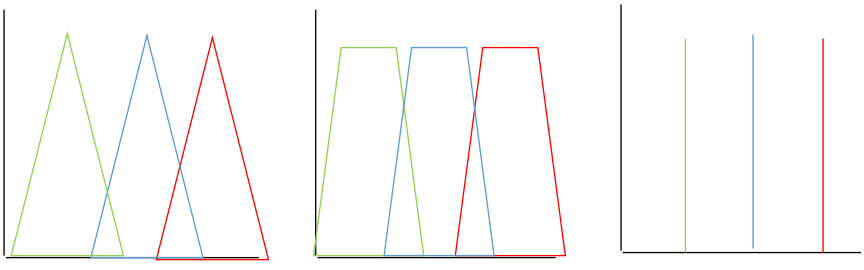
(iv) Convert crisp input data to fuzzy values using the membership functions (fuzzification)

(v) Evaluate the rules in the rule base (inference)

(vi) Combine the results of each rule (inference)

(vii) Convert the output data to non-fuzzy values (defuzzification)

Linguistic variables are the input or output variables of the system whose values are words or sentences from a natural language, instead of numerical values. A linguistic variable is generally decomposed into a set of linguistic terms. Membership functions, some illustration of which is shown in Figure 3.14 are used in the fuzzifcation and defuzzifcation steps of a FLS, to map the non-fuzzy input values to fuzzy linguistic terms and vice versa. A membership function is used to quantify a linguistic term. In a FLS, a rule base is constructed to control the output variable. A fuzzy rule is a simple IF-THEN rule with a condition and a conclusion. The evaluations of the fuzzy rules and the combination of the results of the individual rules are performed using fuzzy set operations. After evaluating the result of each rule, these results should be combined to obtain a final result. This process is called inference. The results of individual rules can be combined in different ways. After the inference step, the overall result is a fuzzy value. This result could be defuzzified to obtain a final crisp output. This is the purpose of the defuzzifier component of a FLS. Defuzzification is performed according to the membership function of the output variable.



(i) (ii) (iii)

Figure 3.14 Illustration of three different types of membership functions (i) Triangular (ii) Trapezoidal (iii) Singleton

3.5.3 Support Vector Machine

Support Vector Machines (SVM) is a machine learning algorithm introduced by V. N. Vapnik (Vapnik 1998; Vapnik 1995) which is based on statistical learning theory and has been successfully used for handwriting digit recognition (Cortes and Vapnik, 1995; Scholkopf, Burges and Vapnik, 1995; Scholkopf, Burges and Vapnik, 1996; Burges and Scholkopf, 1997), object recognition (Blanz et al., 1996), speaker identification which implies mouth movement recognition (Schmidt, 1996), face detection in images (Osuna, Freund and Girosi, 1997a), and text categorization (Joachims, 1997).

The simplest classifier is the linear classifier which separates into two classes and maximizes the distance between the two classes by computing the (n-1) -dimensional hyper plane with nearest data points on each side. These nearest data points are referred to as the support vectors.

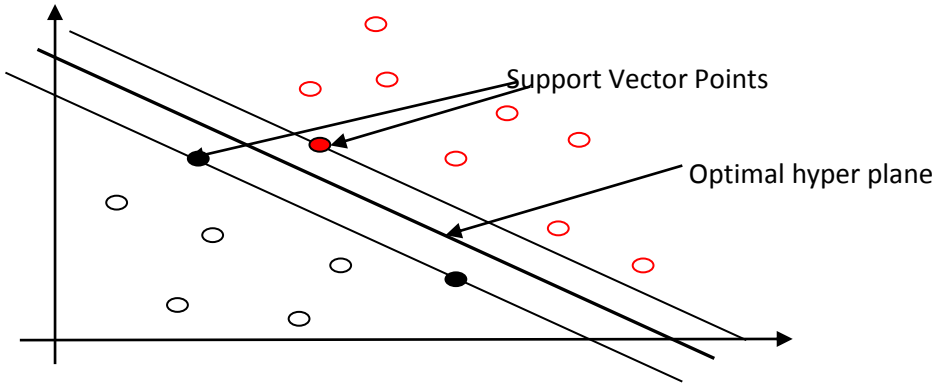


Figure 3.15 Illustration of optimal hyper plane and support vector points in the SVM algorithm.

The Figure 3.15 illustrates the principles and training of SVM using the linear classifier, where the optimal hyper plane is linear and is the one with the maximum distance from the nearest data points separating the two classes. The support vectors are those data points shown in solid dots (red and black) nearest to the optimal hyper plane.

Thus the complexity of SVM can be appreciated with a linear hyper plane, since the optimal location of the plane and the calculation of the support vectors are not trivial. More so, when a linear hyper plane does not fit the problem and quadratic, polynomial, radial or sigmoid basis functions has to be employed.

3.5.4 Rough Set Theory

RST was introduced by(Pawlak (1982) and was designed to be used for the classification of imprecise, uncertain and incomplete information similar to the problem posed by the generation of SI in DVC at the decoder, where the original is not available and as such the information for the generation of SI is noisy, imprecise and incomplete.

RST employs a Table containing information about the spatial-temporal characteristics of the video whose SI is to be generated, with each row of the Table representing a different unit of the video (MBs or frames are units that are prevalent in the DVC architectural framework) and each column containing different attributes that describe the MB or frame (e.g. MSE described in Section 3. 3 employed in exploiting the temporal characteristics of MBs).

RST provides the tools to arrange the information described above in a Table and determine the relationships between them in so called "equivalent classes". Then RST tools are then employed to construct a matrix which maps how these characteristics participate in the decision making process. The matrix is then used to generate functions and derive rules for generalization in a training scenario where the decisions have already been determined. Lastly, the outcomes from the training are employed in a generalized framework to analyze an unknown MB or frame and determine the decision that is best for the unknown MB or frame. Further exploration of RST in DVC is a major contribution in Chapter 7 of this thesis.

The advantages of the RST compared to other artificial intelligence methods considered in this section are:

(i) Unlike other closely related artificial intelligence methods, RST does not need any preliminary or additional information about the data (a priori knowledge), for example FL needs the value possibility in order to work effectively with provided data and Dumpster-Shafer theory requires basic probability assignments. RST just takes the given data and information provided and applies the RST processing algorithm on it.

(ii) RST provides all the tools and methods that efficiently find hidden patterns in the data as part of the overall RST algorithm. These tools and methods are deployed to find the hidden patterns in the spatial-temporal data of the various video sequences, MBs, segments and frames such that better decisions to improve SI can be reached in an intelligent manner.

(iii) RST allows for an automatic way to generate the decision making rules, by evaluating the objects, attributes and decisions in the information Table and deducing the minimal set that is important in the decision making process which is used to compile the decision making rules. RST thus effectively removes data that does not aid the decision making process, while compiling the decision making rules.

(iv) RST gives easy and straightforward results that are easy to interpret as the results are given in terms of the required decision. For example, in the use of RST for selecting the best methods of de-interlacing various sequences in Jeon et al. (2005). RST gives a decision which method will give better de-interlacing given the data under investigation.

Some drawbacks of the RST algorithms are also as follows:

(i) One major drawbacks of RST is the assumption that the information Table contains all the required data to deduce the rules needed for decision making and thus all that has to be done is deduce the rules.

(ii) Whenever the information Table does not contain all the rules, erroneous decisions would be made when predicting decisions.

3.6 Summary

This Chapter presents a survey of SI generation in DVC, starting with the theoretical foundations of SI generation and the earliest SI's employed in DVC. Improvements in the SI generation schemes highlighted the fact that LMCTI is the predominant SI generation algorithm. The inadequacy of LMCTI and the efforts in literature to improve SI generated by LMCTI is presented while the fundamentals of LMCTI SI generation is examined in detail including the ME search algorithms based on fast and efficient block based computations. The fact that errors in computation and algorithms finally show up as artifacts in video sequences causing an unpleasant viewing experience is highlighted, and the various sources of artifacts are also presented. Finally, it was shown that smart and intelligent schemes need to be exploited in order to explore the video characteristics both within video sequences and between various video sequences, making an algorithm to perform well in certain parts of a sequence while performing poorly in another part of the same sequence or an algorithm performing well with one video sequence but underperforming in another.

The shortcomings of the present SI generation schemes presented in this Chapter therefore form the basis of ideas to be presented in Chapters 6, 7 and 8 to improve SI generation in DVC which are then rigorously tested and showed to improve SI generation and subsequently, the entire DVC codec.

In order to show that the various ideas presented improves both SI generation and ameliorate this all important bottleneck in DVC, an underlying methodology for

investigating the outcomes of this survey, experimenting on the weaknesses discovered and showing that the solutions provided are valid improvements are thus presented in the next Chapter.

Chapter 4

Research Methodology

This Chapter discusses the research methodology employed to design, analyze and validate the algorithms which were introduced in the *SI Generation and Improvement framework* in Figure 1.2, with the overarching aim of narrowing the performance gap between conventional codecs and DVC. A model-based, simulation methodology has been chosen because it offers both a high degree of flexibility in designing DVC systems, and a realistic platform for testing the contributions. Such a model-based methodology is importantly a widely accepted development approach for DVC and has been used for example by Ghorbel et. al., 2014; Katsoyiannis and Breivik, 2014. The remaider of this Chapter will present:

(i) the framework for design and testing of proposed algorithms

(ii) performance measurement metrics

(iii) video datasets

(iv) and validation of software implementation

4.1 Framework for Design and Testing of Proposed Algorithms

The framework for modeling the DVC test-bed is the Stanford architecture whose detailed features were described in Section 2.2.3. The theories of SW and WZ which the DVC architecture is based on underpins the model based, simulation methodology. The theories make it likely that subsequent investigation and testing of the *SI Generation and Improvement Framework* comprising a suite of four algorithms in Figure 1.3, when implementing in the proposed DVC model could improve DVC output to the same quality as that from conventional codecs.

The next decision is the platform to be employed for modeling the DVC codec and testing the ideas. Three options are available:

(i) Hardware based: this includes dedicated and off-the-shelf prototyping hardware solutions such as Digital Signal Processors (DSP) and Graphics Processing Units (GPU) for real time video processing applications. While DSP and GPU platforms can achieve real-time image processing (Dong and Thinh, 2014; Pieters et al., 2007), they are not so flexible in term of experimenting with different algorithms and parameters which are essential in developing the understanding of the inter-relations and cause and effect the changes may have on the performance of the system. Despite the fact that system changes are possible, it often requires a time consuming process of replacing hardware modules and uploading of new system firmware. As the overarching objective of this research is improving the DVC efficiency by trying new algorithms, modes and parameter settings, requiring flexibility, though, the hardware based platform will afford real time simulation scenarios, it is inflexible, restricts the possible algorithm changes and the development time is long and costly.

(ii) Hybrid Hardware-Software based: popularly referred to as “co-design” where computationally intensive parts of the system are undertaken by specially designed hardware, while other parts of the system are software based and realized on a general-purpose processor (Nath and Datta, 2014; Wolf, 2003). The benefit of this approach is that real-time processing can be achieved through the hardware module while a degree of flexibility in altering the system design and parameters is achievable through the software module. For this research project, it is possible to implement the *SI Generation and Improvement Framework*, in a software module with the other parts of the DVC architecture such as the SW, LDPC channel codes, and H.264 intra codec being implemented in hardware. However, this approach is still not suitable for this project for three reasons: a) an interface module is needed which seamlessly connects the *SI Generation and Improvement Framework* to other DVC parts. b) When modifying various parameters in software, the other DVC parts implemented in hardware may also require modification which cannot be easily achieved. c) Dedicated video processing hardware can be expensive.

(iii) Software based: these build upon a comprehensive set of image processing libraries, which make prototyping easier and faster so shortening the development time and makes the process more efficient and less error prone. Furthermore, the software option provides maximum flexibility in designing and experimenting with the DVC model. This means the algorithms and parameters can be easily changed and improved in such a way that the behavior of the model can be studied more effectively.

In considering the thesis objectives defined in Section 1.2, and the key design requirements of flexibility, time and cost, a software-based simulation approach was deemed the most viable option and was thus used in developing the new *SI Generation and Improvement Framework*. As for the image processing libraries, the most commonly used ones are the Matlab Image processing toolbox (Brites et al 2013;

Petrazzuoli et al 2013; Akinola, Dooley and Wong 2011; Li 2008) and the OpenCV library (Bradski 2000). The next section will analyze the benefits and drawbacks of these two libraries and explain the choice made.

a) Choice of Image Processing Library:

Matlab is a fourth-generation software computing environment, made up of functional tool-boxes that allows matrix manipulations, plotting of functions and data, implementation of algorithms, creation of user interfaces and easily interfaces with programs written in other languages. OpenCV on the other hand is a library of programming functions mainly aimed at real-time computing (especially, hybrid hardware-software based described in Subsection (ii) of Section 4.1) which is also cross- platform, which means it can interface with programs written in other languages.

Matlab and OpenCV are competing software development platforms for image and video processing, the choice of either of which could have been justified by the usual criteria of flexibility, functionality, usability, popularity, support and cost as they are quite similar. However, the acceptance of Matlab by the DVC community makes it easier to build on the functionality that is already available. Matlab has various DVC codecs and algorithms such as the Li DVC codec (Li, 2008) available. OpenCV is also more widely used by the general-purpose image processing community.

Furthermore, Matlab supplies well documented manuals with detail of all the commands, their functionalities and examples of how they are employed which makes Matlab easier to learn and use. Thus, Matlab was chosen as the development platform for this DVC research work.

b) Software Computing Platform

The DVC model was developed upon the Stanford DVC architecture (Detailed in Section 2.2.3) using Matlab version R2009b on a PC running the Microsoft Windows XP operating system with Intel Dual Core CPU at 2.20 GHz and 3.5 GB of RAM.

4.2 Performance Analysis Techniques

To assess the performance of the algorithms in the *SI Generation and Improvement Framework* of Figure 1.2, two quantitative metrics used in the DVC literature are employed, namely; PSNR and RD curves. Furthermore, for qualitatative inspection, selected frames which highlight particular perceptual observations are provided for comparison.

a) PSNR Metric

PSNR metric is based on MSE which is defined in (3.2) with the PSNR itself given in (3.3). PSNR is inversely proportional to the distortion (D) i.e., the higher the MSE between the output and original frames, the lower the PSNR value.

b) RD Curves

The bit rates *R* to achieve a given distortion for two statistically dependent sequences *X* and *Y*, where *X* is the original sequence to be encoded and *Y* is the output of the WZ codec can be represented by:

$$R = \frac{\sum_{n=1}^{m} B_n}{f_r} \qquad (4.1)$$

Where $B_n = f(\frac{1}{\text{PSNR}})$ is the number of bits required to reconstruct a frame, which means *R* is inversely proportional to PSNR, m is the number of frames in the sequence and f_r is the frame rate. Figure 4.1 shows an example RD curve for the DISCOVER (2007) codec for the *Hall* sequence. It shows the distortion (PSNR) against a range of of codec bit-rates used to reconstruct the sequence. For example, a bit-rate of 150 Kb/s is required to reconstruct *Hall* with an average PSNR of 35dB.

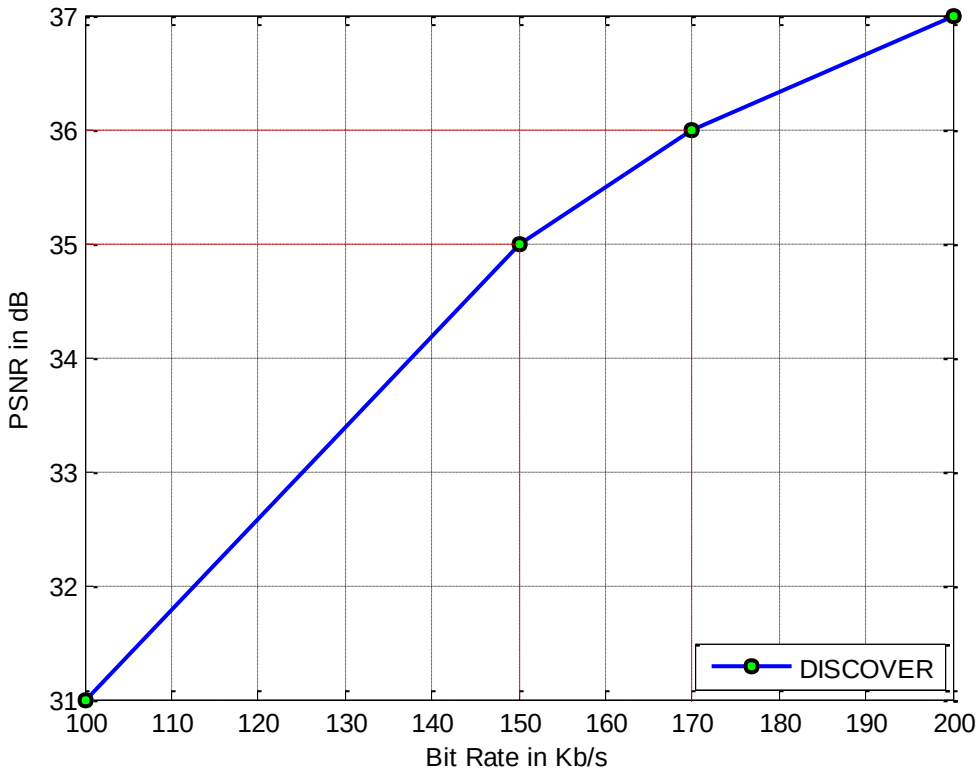


Figure 4.1 Example RD Curve showing the DISCOVER codec performance for *Hall* sequence @ 15f/s

c) PSNR and Perceptual Quality

While PSNR mathematically measures the mean square error between the processed video and the original, it does not always show the same perception in the human vision sytem because the difference in the pixels that gives the same PSNR value could come from different spatial-temporal locations in the video. Thus a video with error in the corner of a video frame can have the same PSNR with another with an error at the centre of its frame, perceptually appearing different. Other reasons why PSNR does not always match up with perception are (Girod, 1993; Yang et al. 2005):

a) not every error between the original and processed video can be noticed by the human eye.

b) not every region of the video frame receives the same attention from the human observer.

Generally, for a change in PSNR to be perceptible to the human vision system, it must be at least 0.5 dB change in PSNR (Girod, 1993; Yang et al. 2005). This is a rule of thumb for the just noticeable difference threshold.

d) Evaluation of Computational Complexity

Since the *SI Generation and Improvement Framework* is at the decoder where high computational power is available, a lower priority is given to the computational complexity of the algorithms and more given to improving SI quality. For completeness, a computational time analysis is provided which considers the overheads incurred by the various algorithms which comprise the *SI Generation and Improvement Framework* and the simulation time measurements obtained by time stamps during experimentation as in DISCOVER (2007); Ascenso and Pereira (2009); HoangVan and Jeon (2012).

To enable comparison between the various algorithms in the *SI Generation and Improvement Framework*, each is broken down into a series of components and the average time for each to process one frame is determined. The defined variables are shown in Table 4.1.

Table 4.1 TIME VARIABLES IN SECONDS

T_{Const} : Time for loading algorithm parameters and initialization per frame.
T_{Vel} : Time for computing velocity term in trajectory per frame.
T_{Accl} : Time for computing acceleration term in trajectory per frame

T_{Jolt} : Time for computing Jolt term in trajectory per frame

T_{BMA} : Time for MV based Block search and BMA per frame

T_{AOBMC} : Time for MB surrounding block enlargement and matching per frame

$T_{Off-Line}$: A lumped parameter reflecting off-line manual input and algorithms of external libraries per frame

T_{MS} : Time for empirical MS per frame

T_{IMS} : Time for intelligent MS per frame

A summary of the complexity analysis for different algorithms in the *SI Generation and Improvement Framework* is shown in Table 4.2. It shows the time variables for the various algorithms, where the differences in time variables comes from and which aspect contributes to additional complexity. For instance the difference in time variables between the empirical MS and the intelligence based MS (IMS) algorithms emanates from the difference between the T_{MS} and T_{IMS} terms plus the $T_{off-line}$ term.

Table 4.2 SUMMARY OF TIME VARIABLE ANALYSIS

Algorithm	***Time Complexity***	***Comment***
HOPTTI_Linear	$T_{Const} + T_{Vel} + T_{BMA}$	Minimum of two frames required and a velocity term calculated.
HOPTTI_Quadratic	$T_{Const} + 2T_{Vel} + T_{Accl} + T_{BMA}$	Minimum of three frames required and 2 velocity terms and 1 acceleration term calculated.

Algorithm	*Time Complexity*	*Comment*
HOPTTI_cubic	$T_{Const} + 4T_{Vel} + 2T_{Accl} + T_{Jolt} + T_{BMA}$	Minimum of for frames required and 4 velocity terms, 2 acceleration terms and 1 jolt term calculated.
HOPTII-AOBMC	$T_{Const} + 4T_{Vel} + 2T_{Accl} + T_{Jolt} + T_{BMA} + T_{AOBMC}$	HOPTTI cubic employed plus AOBMC overhead.
MS HOPTTI and HOPTTI-AOBMC	$T_{Const} + 4T_{Vel} + 2T_{Accl} + T_{Jolt} + T_{BMA} + T_{HOPTTI-AOBMC} + T_{MS}$	HOPTTI cubic employed plus HOPTTI reuse for HOPTTI-AOBMC.
IMS HOPTTI and HOPTTI-AOBMC	$T_{Const} + 4T_{Vel} + 2T_{Accl} + T_{Jolt} + T_{BMA} + T_{HOPTTI-AOBMC} + T_{IMS} + T_{off-line}$	HOPTTI cubic employed plus HOPTTI reuse for HOPTTI-AOBMC.

To complement the expressions in Tables 4.1 and 4.2, actual time measurements are also presented, obtained by taking a time stamp immediately before an algorithm is executed and another time stamp immediately after the execution of the algorithm is completed. The time difference between the two time stamps gives the time taken for the algorithm to be executed. To ensure a fair comparison, all the algorithms were run on the same computer system (refer to Section 4.1), with no other programs running at the same time.

The time complexity variables of Tables 4.1 and 4.2 are employed in the subsequent Chapters to discuss the complexity of the proposed new algorithms.

4.3 Video Dataset

In order to test the algorithms developed, a video dataset was used, consisting of sequences with various spatial-temporal characteristics ranging from single object to multiple objects

and slow to fast object motion. They include: *Container, Foreman, Mother, Hall, Carphone, Salesman, Coastguard, American Football, Stefan and Soccer* (Acticom GmbH 2002, Xiph.org 2013, Arizona State University 2014).

These sequences are standard *Quarter Common Intermediate Format* (QCIF) format, with resolution of 176 x 144 pixels at 15 Hz. All quantitative measurements refer to the luminance values only and for a GOP=2 unless otherwise stated. This format is adopted by the DVC community for the testing and validation of DVC codecs and is used in this thesis for comparison purposes.

4.4 Validation of Software Implementation

In order to verify that the proposed algorithms in the *SI Generation and Improvement Framework* were properly implemented and that the results presented were valid, a number of strategies were put in place. The following discuss the validation strategies for the algorithms implemented in the test bed:

a) Validation of Basic DVC Software Implementation

The ground truth for the validation of all results is the original video sequences. These are made available for comparison and referencing during the quantitative evaluation phase, though they would generally be not available at the decoder in an actual DVC implementation. The original sequences thus form an upper bound for any performance improvements in SI.

The basis of this software implementation is the DVC codec of Li, (2008). The codes tested using the same sequences (Li, 2008) and show agreement with published results. Subsequently, the codec formed the basis of the development of the testbed. Therefore in Chapter 5, the results from (Li, 2008) are compared with those from the higher order SI generation test bed, using exactly the same settings to validate the results.

b) Validation of Higher Order Algorithm Implementation

The higher order algorithm is introduced to replace the linear SI module employed in the codec. This was rigorously tested with squences which include, *Foreman, Mother, Hall, Carphone, Salesman, Coastguard, American Football, Stefan and Soccer* sequences with the expectation that sequences with objects incorporating more complex motion will give improved SI quality both qualitatively and quantitatively. The quantitative analysis involved a frame-by-frame inspection of the results and a comparison made with those generated by Li's codec, as will be discussed in Chapter 5.

Furthermore, a stepwise approach of progressively changing the model from linear to quadratic and then cubic with results validating incremental improvements alongside increased complexity and time delay from the theoretical consideration was also used to validate the results.

c) Validation of BMA Mitigtion Algorithm

Likewise, the BMA mitigation algorithm which was introduced to tackle artifacts in other parts of the resulting decoded video sequences is taken from reproducible codes of various authors and validated against their results (Bosc et al. 2011; Liu et al. 2010; Ye at al. 2009). After rigorously testing the codes of the original BMA mitigation algorithm and confirming the reproduced results corroborate the published results, the algorithm module was incorporated into the test bed. The results produced by the algorithm as shown in Chapter 6, agree with the expected qualitative and quantitative improvements, especially in areas where visible artifacts previously appeared on the SI produced by the higher order algorithm. This improvement is tracked by using the frame by frame analysis.

d) Validation of Artificial Intelligence Based Classifier Implementation

The role of an artificial intelligence based classifier is to choose between macroblocks produced from the higher order and BMA mitigtion algorithms in order to generate a more accurate SI. To validate the classification performance of the intelligence based algorithm,

a ground truth, which is a list of correct decisions of choosing between the higher order and BMA mitigation algorithms for each MB of a sequence, is employed. A decision is made by comparing the PSNR values of the SI for a MB produced by the two algorithms and the one with higher PSNR is chosen. This ground truth and the classification results produced by intelligence-based MS algorithm are critically evaluated in terms of their classification performance. Furthermore, overall codec performance in terms of of measurng the SI quality is evaluated using RD curves.

4.5 Summary

The research methodology framework presented forms the basis of the rigorous analyses and validation of the major contributions in the *SI Generation and Improvement Framework*, the first of which is the higher order HOPTTI algorithm which is presented in the following Chapter.

Chapter 5

SI Generation using Higher Order Piecewise Trajectory Temporal Interpolation (HOPTTI).

5.1 Introduction

As stated earlier in Chapter 3, the quality of SI impacts upon DVC performance significantly way and it is one of the most important bottlenecks in DVC performance, which have been acknowledged in the earliest practical implementations of DVC theory (Aaron, Zhang and Girod 2002; Jagnohan, Sehgal and Ahuja 2002). The quality of SI impacts DVC in two ways:

(i) RD – this reflects bit rates versus PSNR and it shows the number of decoder bits required to provide a prescribed output quality. The better quality codec would be the one that employs lower bits to achieve higher PSNR output. The decoders that have high SI quality would request lower bits to improve SI to the prescribed quality (same as, WZ fidelity criterion as stated in the theoretical aspects of DVC in Chapter 3) and they will therefore exhibit superior performance.

(ii) Robustness (error resilience) – in DVC, SI frames are constructed in most cases independent of channel fidelity at the decoder, so the better the SI quality, the more resilient the codec becomes compared to the situation where more bits are needed via error prone channels. SI is improved at the decoder using different image processing and intelligent methods such as those highlighted in the SI generation and improvement framework of Figures 1.2 and 1.3. Conventional codecs on the other hand rely on the

transmitted bits stream of both the transformed images based on prediction and its residue (Ostermann et al., 2004), discussed earlier in Chapter 1, which propagates the prediction error making it difficult to guarantee lossless receipt at the decoder.

Generating an SI with very high quality is therefore of paramount importance if DVC performance is to be improved to give similar performance as conventional codecs and this forms the major objective of this thesis. The SI frames can be improved upon to obtain better quality than the originally received frame by noise reduction (de-noising), super resolution and in-painting algorithms.

Also, from the survey of SI in Chapter 3, LMCTI has been predominantly used in SI generation (Aaron, Setton and Girod 2003; Aaron, Zhang and Girod 2002; Pereira et al. 2008; Artigas et al. 2007; Li 2008; Liu, Yue and Chan 2009; Ye et al. 2009) where the motion of objects in the video sequences are assumed to be linear. While LMCTI provides reasonable SI quality for sequences with contrived, slow-to-medium object motion, it tends not to be generally so successful for sequences exhibiting natural motion where non-linear motion (Ye et al. 2009) including acceleration, deceleration, turns, twists and jerks is quite common, most especially where fast object motion and multiple objects predominates. In order to achieve more accurate trajectories for natural video sequences higher-order trajectories (Chahine and Konrad 1995; Chahine 1995; Akinola, Dooley and Wong 2010; Petrazzuoli, Cagnazzo and Pesquet-Popescu 2010) for temporal variations have been modeled, leading to reported more accurate sequence reconstructions and greater compression efficiencies.

While most of the solutions for generation of SI in literature make use of LMCTI, there have been various suggestions as to how to generate SI as reviewed earlier in Chapter 3 and a summary of the various methods which was presented in detail in Chapter 3 is:

(i) Use of previous or future key frames as SI

(ii) Simple pixel based averaging of previous and future key frames

(iii) Pixel based edge directed linear interpolation

(iv) LMCTI

(v) Bi-directional LMCTI

(vi) Higher order motion compensated interpolation

The first case though quite straightforward and simple to implement, presents a challenge in terms of number of bits required from the encoder to improve the SI as the missing frame is often quite significantly different from both the previous and future frames, except for stationary head and shoulder sequences acquired with stationary cameras that exhibit almost no global motion. Even after the request of several bits from the encoder, the final decoder output is significantly poorer than the original since the starting point for reconstruction, which is the SI, is so poor.

The second case, though computationally more intensive also does not give a more improved SI compared to the first case as it merely averages pixel values and blurs out some parts of objects if they have moved away in the future key frame while it places a pale (ghosting) resemblance of object parts where they have moved to.

The third case gives relatively similar quality of SI with LMCTI as it is a predictive interpolation method, but the intensive pixel by pixel iteration of the algorithm is time consuming.

The fourth case, LMCTI, as discussed earlier and buttressed by literature in Chapter 3, gives a reasonably high SI quality for sequences with slow to medium object motion. The employment of fast block based motion search algorithms similar to predictive H.264 (Ostermann et al., 2004) gives it an edge and this is why it is predominantly employed by the community while its shortcoming with sequences having asymmetric, medium to high object motion is overlooked.

The fifth case employing Bi-directional LMCTI derives from an acknowledgment of the fact that the linear motion model cannot deliver further improvement in SI as the

estimation of motion from the previous to the future frames usually results in holes and artifacts on the intermediate frame while the estimation done from future frame to previous frame also results in different set of holes and artifacts, thus the combination of intermediate frames obtained from both directions tend to cover the holes.

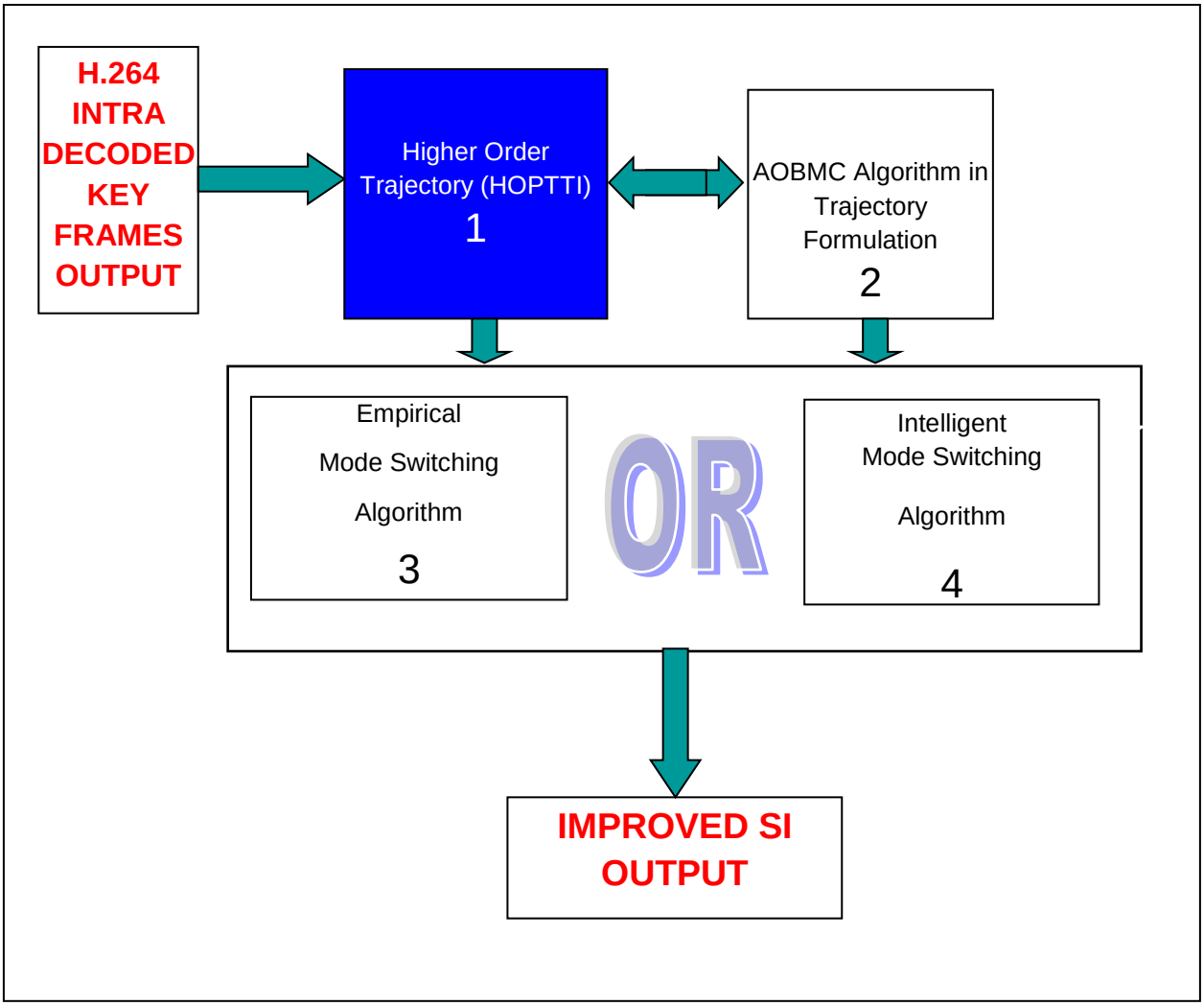


Figure 5.1 Block Diagram of SI Generation and Improvement Framework with BLOCK 1 Highlighted.

The sixth case is thus a more realistic approach to modeling of object motion. This is even more compelling since the same fast block based motion search algorithms can be engaged. Thus the concept of BLOCK 1 of Figure 1.3 in Chapter 1 is pursued here to show that a more efficient exploitation of the temporal redundancies in video sequences can be achieved by incorporating a more accurate trajectory model into the SI Generation and

Improvement Framework of DVC introduced in Figure 1.2. The block diagram of the motivated research formulation is reproduced with BLOCK 1 highlighted in Figure 5.1.

This Chapter presents the incorporation of higher order object motion trajectories in DVC *SI Generation and Improvement Framework* rather than the hitherto linear models, leading to a block-based, *higher-order piecewise* trajectory *temporal interpolation* (HOPTTI) algorithm for SI generation which is based upon the models in (Chahine and Konrad 1995; Chahine 1995; Akinola, Dooley and Wong 2010; Petrazzuoli, Cagnazzo and Pesquet-Popescu 2010). The HOPTTI model, instead of tracking every pixel, employs MB (16 x 16 pixel blocks) for tracking the motion field, with the flexibility to employ sub-blocks of 4 x 4 pixels and 8 x 8 pixels whenever necessary, though at the expense of increasing computational overhead. Furthermore, as will be evidenced, due to the use of higher-order motion trajectory models, HOPTTI proves superior SI quality both in terms of the average *peak signal-to-noise ratio* (PSNR) and qualitative visual appearance when compared with existing LMCTI techniques, with certain test sequences providing an improvement of up to 8dB when piecewise cubic polynomials is employed instead of a linear model, while an average of about 5dB improvement is achieved over a range of slow to fast object motioned sequences with single and multiple objects.

5.2 HOPTTI SI Module and DVC Architecture

The basic DVC architecture adopted for this thesis is shown in Figure 5.2, which was generally described in Chapter 2. The HOPTTI SI Generation Module (enclosed in the dashed-line box, labeled SI in Figure 5.2) and its key components are hereby described in more detail. The key components and how they interact are shown in Figure 5.3.

From the discussions and literature review in Chapters 2 and 3, it is noted that the quality of SI generation in DVC is a major bottleneck and that the quality of SI is directly linked

with the inadequacies in temporal exploitation techniques presently in use in DVC. The HOPTTI algorithms is proposed in order to ameliorate the problems identified and generate a higher quality of SI by the provision of a more accurate prediction of object position in the intervening WZ frames in the DVC architecture.

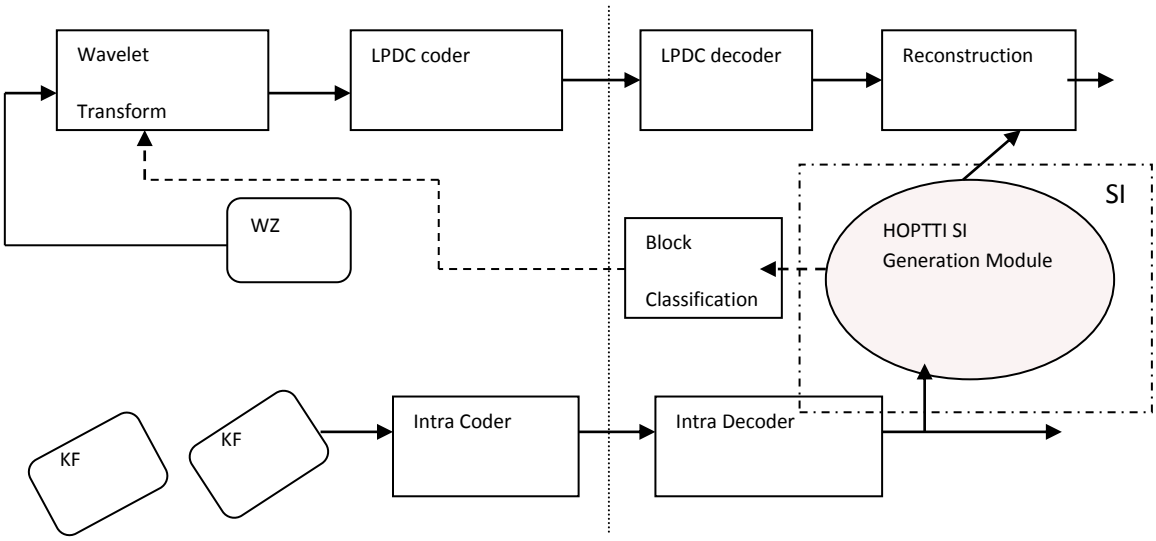


Figure 5.2 Architecture of Codec highlighting SI module

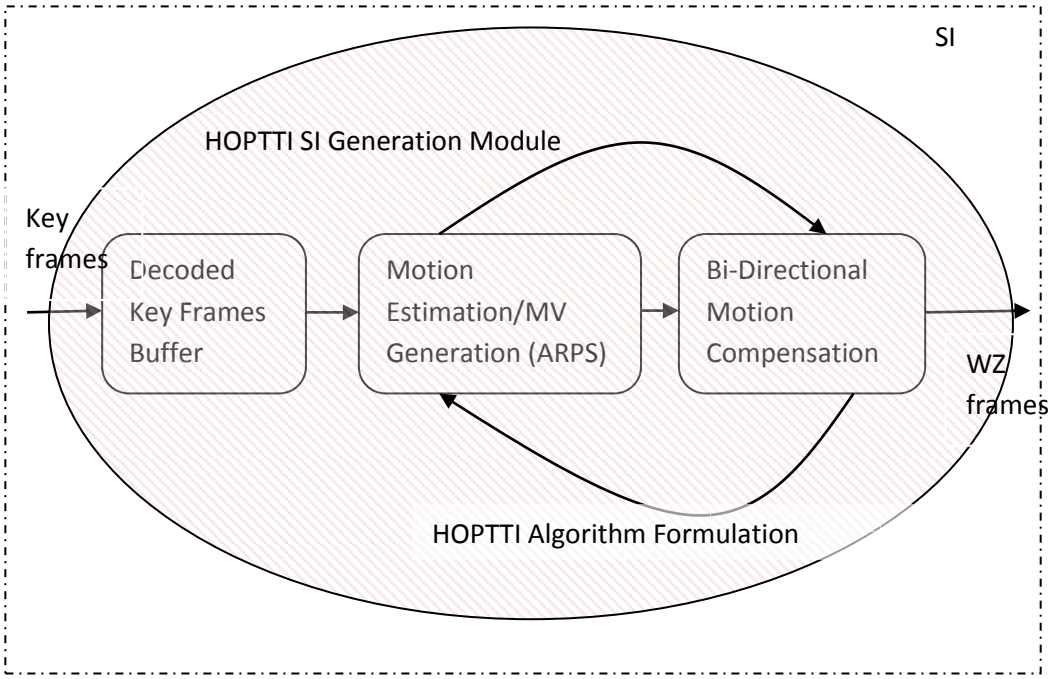


Figure 5.3 Detailed Blocks of SI Generation Module

The proposed SI generation module, the key components and their interaction in the HOPTTI algorithm formulation are discussed in the following sub-sections.

5.2.1 Decoded Key Frame Buffer and HOPTTI

Compared to conventional coding as discussed in Chapter 2, DVC cannot employ predictive coding using the original as reference frame referred to as residual coding because the original frame is not present at the decoder where the exploitation of the redundancies is taking place. DVC therefore employs motion compensated interpolation with intra-coded or lower complexity inter-coded key frames from conventional codecs with H.264 being employed in this thesis. Figure 5.4 (a), shows the conceptual linear interpolation process which requires the arrival at the decoder of two key frames in order to interpolate the missing WZ frame in a K-W-K-W-K-W arrangement. This arrangement is referred to as GOP of 2, (Petrazzuoli, Cagnazzo and Pesquet-Popescu 2010).

In order to employ higher order trajectory interpolation as being proposed, additional key frames are required and this is shown conceptually for quadratic and cubic interpolation in Figure 5.4 (b) and (c). This is straightforward as higher order means further determination of higher differentials as will be further illustrated in subsequent sections.

The key frames buffer therefore becomes more important as this is required to hold more key frames for motion estimation (ME) and furthermore to hold MB based estimated MVs for the piecewise construction of the motion trajectories which in turn allow for the formulation of a more accurate framework for the evaluation of missing intermediate WZ frames from motion compensated interpolation.

While MB, block based, MV generation is adopted for the purpose of formulating the more accurate higher order trajectories due to the advantages highlighted both in Chapter 3 and the introductory part of this chapter, two underlying assumptions which weaken the HOPTTI formulation have been implicitly made which are the following:

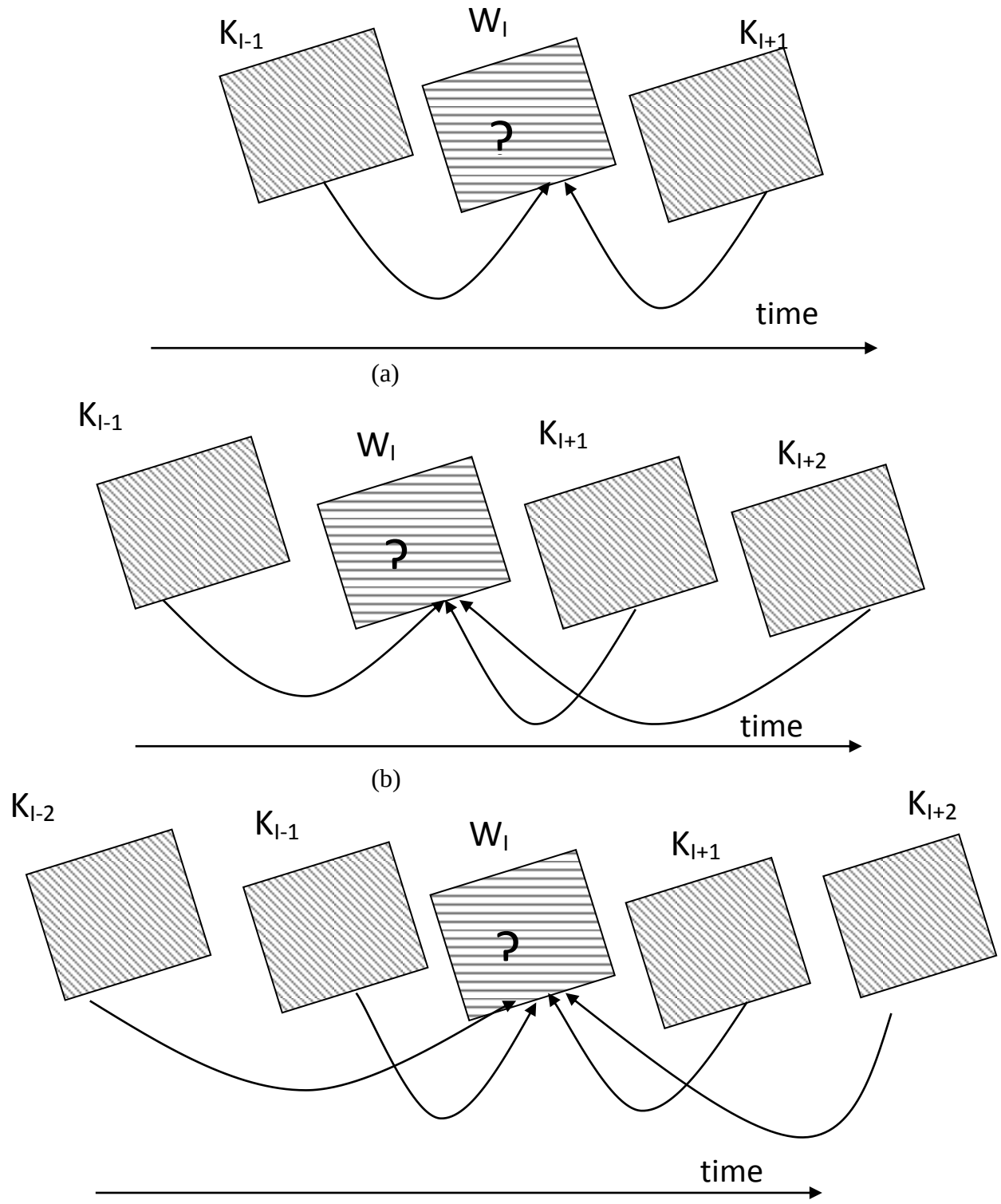


Figure 5.4 Conceptual illustration of additional frames required for higher order trajectory formulation (a) Linear Interpolation requires 2 No. Key frames (b) Quadratic (2nd Orider) Interpolation requires at least 3 No. key frames (c) Cubic (3rd order) Interpolation requires at least 4 key frames

The first is the fact that the motion to be exploited is translational which means that global and rotational motion is assumed not to be present. The major weakness in this assumption is the non-inclusion of rotational motion as global motion that leads to translational motion can be said to be taken care of. The second assumption is that MBs are assumed to contain only one singular object type which implies that the object must be larger than MB sizes.

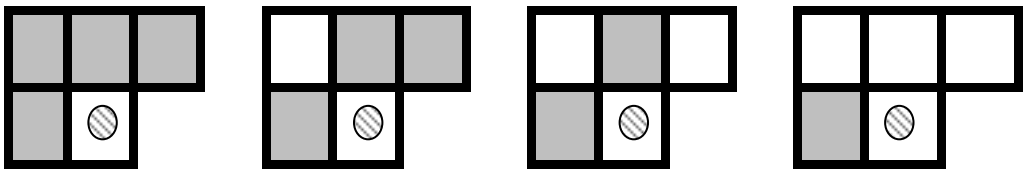
The weaknesses highlighted earlier have been addressed to some extent by the flexibility in sub-MB sizes adopted in the DVC architecture proposed for HOPTTI by allowing 8 X 8 and 4 X 4 sub-blocks as discussed both in Chapter 2 and earlier in this Chapter. Other aspects of the weaknesses will be further addressed in Chapter 6.

5.2.2 Adaptive Rood Pattern Search (ARPS) Motion Estimation

In formulating a more accurate motion trajectory model, we try to emulate true motion which, as has been noted in literature, is necessary for better exploitation of temporal redundancies (Chahine and Konrad 1995; Chahine 1995; Akinola, Dooley and Wong 2010; Petrazzuoli, Cagnazzo and Pesquet-Popescu 2010). Therefore, we introduce the fact that objects actually possess inertia and thus exhibit acceleration, deceleration and surges. Furthermore, objects thereby exhibit smooth translational trajectory motion. This implies that MVs must follow smoothly from preceding ones and outliers are not common (Choi et al., 2007). The adaptive rood pattern search (ARPS) (Zhao et al. 2008) introduced in Chapter 3 of this thesis has been adopted in our test bed and experimentation as it possesses the characteristic of producing such a set of true motion MVs with little or no outliers due to its use of the adaptive rood pattern search algorithm. ARPS makes use of neighboring blocks as region of support and is hereby described to give depth to the motion estimation algorithm as a temporal exploitation tool for video compression, while

at the same time explaining one of the crucial parts of the proposed, more accurate higher order motion trajectory formulation.

The ARPS algorithm is one of the versatile fast and accurate true motion estimation algorithms that have been shown to increase computational gain with little loss in PSNR compared to the Full search (FS) algorithm (Nie Y. and Ma K-K., 2002; Zhao et al. 2008). While ARPS avoids the use of the sequential whole frame as a search window, which makes searching cumbersome and time consuming, it at the same time remedies the inadequacies inherent in a predetermined small window search breakdown being stuck in a local maxima. In instances where widely varying motion can cause objects to move outside a pre-determined window as its zero motion pre-judgment has been shown to give higher PSNR in large motion video sequences such as *Foreman* and *Coastguard* (Zhao et al. 2008). ARPS uses a rood (cross-like) pattern that changes in line with the predicted size of MV, employing the surrounding blocks as support, which results in placing the search in the location where there is the highest probability of finding the best matching block. Employing a two-step search algorithm of 1) initial search and 2) refined search (Nie Y. and Ma K-K., 2002) dynamically changes the search window size based on a selection of region of support as shown in Figure 5.5 where the shaded blocks are the supporting blocks and the block marked "O" is the block under consideration. The region of support with maximum of two supporting blocks (Type C) is employed in this implementation as there is no visible improvement in quality beyond this (Nie Y. and Ma K-K., 2002). This implies that the ARPS implementation in HOPTTI dynamically changes the region of support between 1 and 2 in order to refine the MV search.



TYPE A TYPE B TYPE C TYPE D

Figure 5.5 Different types of regions of support based on surrounding blocks

5.3 The Higher Order Piecewise Temporal Trajectory Interpolation (HOPTTI)

From the discussions in section 5.2, the assumption of no motion by the earliest DVC models or even linear trajectories also referred to as LMCTI may not always generate the requisite high quality SI when objects in the sequence exhibit non-linear and global motion, we therefore investigate the use of higher order trajectories in motion compensated temporal interpolation. Also, in Chahine and Konrad (1995) and Chahine (1995) for instance, asymmetric object motion in video sequences was addressed by a quadratic trajectory model allied with MV sampling of one MV per pixel for temporal interpolation, which was reported to have given an overall improvement of up to 4dB over conventional linear-based models.

To investigate a higher order trajectory approach for SI generation in a DVC context, cubic higher-order polynomial trajectory models for temporal interpolation are hereby formulated in (5.1) – (5.6) and implemented in the HOPTTI algorithm, thereby replacing the SI generation module of the DVC architecture described for the use of this thesis in Chapter 3 by the HOPTTI algorithm block diagram shown in Figure 5.3. The cubic model has been chosen as it enables the formulation of a natural trajectory model that gives the most favourable PSNR results compared to overhead cost as will be evidenced later.

5.3.1 The Piecewise Trajectory Formulation and Parameterization

The cubic higher order piecewise trajectory formulation and parameterization enables objects exhibiting non-uniform, sudden accelerated or decelerated motion, such as a surge (also popularly referred to as *jolt*) which is the rate of change of acceleration, to be more accurately represented.

To illustrate the idea, example segments of the motion trajectory of an object in 3-D x, y, t space between time t_1 and t_4 are shown in Figure 5.6(a) while a 2-D view of the same is shown in Figure 5.6 (b). It is assumed the displacements (MV) of the blocks relating to the object at key frames K_1, K_2, K_3 and K_4 between t_1 and t_4 are respectively A_1, B_1, C_1 and D_1. In HOPTTI, the MV of a block is evaluated by finding the position of best match in the next key frame. All key frames are available at the decoder, while WZ frames (denoted as *W* in Figure 5.6) are produced using motion compensated temporal interpolation.

The motion trajectory $C(t)$ of an object can be represented by a set of piecewise cubic polynomials:

$$C(t) = \begin{Bmatrix} p_1(t) & for & t_1 \le t \le t_4 \\ p_2(t) & for & t_4 \le t \le t_8 \\ & \vdots & \\ p_n(t) & for & t_n \le t \le t_{n+3} \end{Bmatrix} \tag{5.1}$$

where each segment of the trajectory $p_i(t)$ is represented by an equation of motion similar to Chahine and Konrad, (1995) and considering a constant jolt given by:

$$p_i(t) = \frac{1}{6} j_i (t - t_i)^3 + \frac{1}{2} a_i (t - t_i)^2 + v_i (t - t_i) + d_i \tag{5.2}$$

For $i = 1, 2 .....n$. In (5.1), n is the number of available key frames, while in (5.2), j_i is the average *jolt* (the rate of change of acceleration), a_i the average acceleration, v_i the average velocity between t_i and t_{i+1} and d_i the initial displacement at t_i.

To calculate the four parameters j_i, a_i, v_i and d_i, a minimum of 4 key frames are required, and if it is assumed the respective displacements of the blocks at these key frames are A_i, B_i, C_i and D_i, then the following holds:

$$d_i = A_i \tag{5.3}$$

$$v_i = \frac{B_i - A_i}{T} \tag{5.4}$$

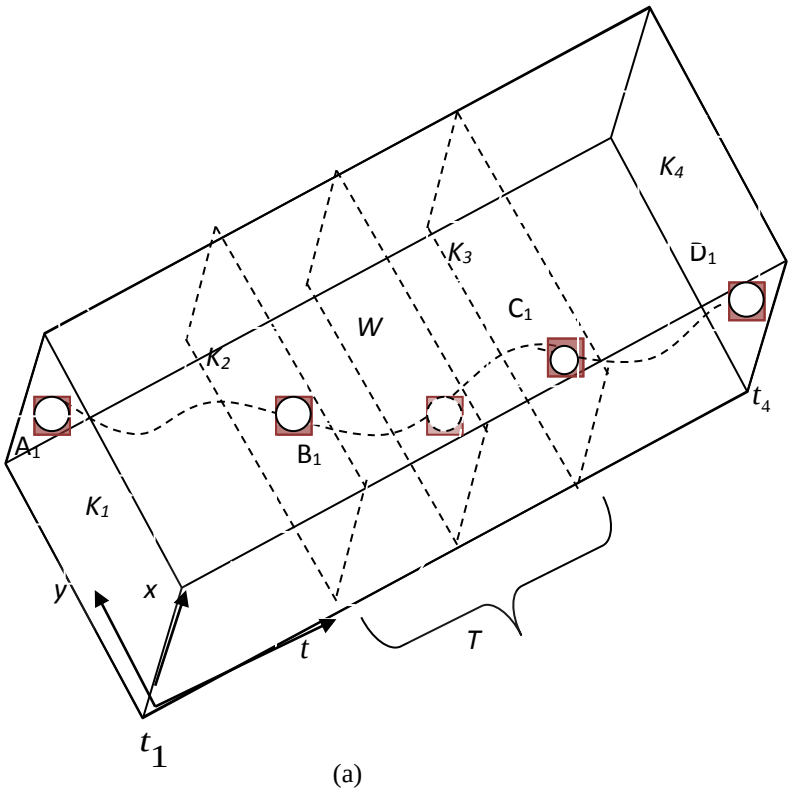
$$a_i = \frac{v_{i+1} - v_i}{2T} = \frac{C_i - 2B_i + A_i}{2T^2} \tag{5.5}$$

$$j_i = \frac{a_{i+1} - a_i}{3T} = \frac{D_i - 3C_i + 3B_i + A_i}{6T^3} \tag{5.6}$$

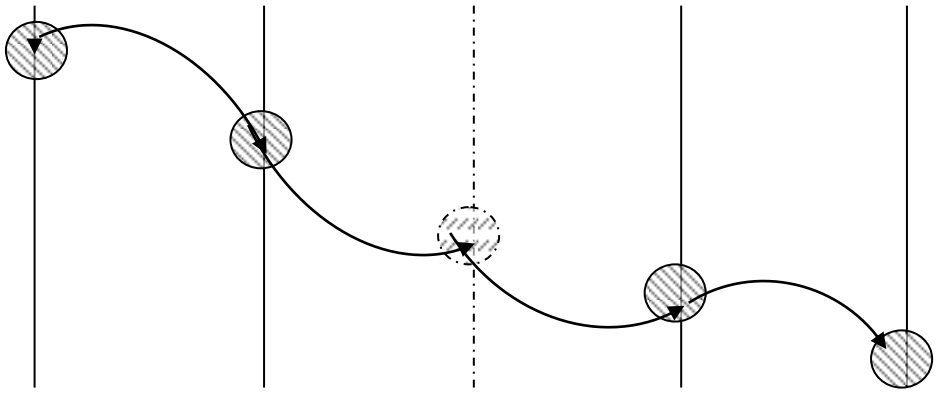
where T is the time between two consecutive key- frames, $A_{i+1} = B_i, B_{i+1} = C_i, C_{i+1} = D_i$.

The forward motion trajectory of an object can be evaluated using (5.1) – (5.6), thus enabling the MV of the object at any time between t_1 and t_{n+1} to be accurately interpolated. The backward motion trajectory is evaluated the same way as the forward one using (5.1) – (5.6) as described but in reverse direction i.e. D_i, C_i, B_i *and* A_i.

The use of additional future and past frames and the additional overhead this imposes on the codec can be better appreciated from Figure 5.6 that shows how the number of frames needed to complete a piecewise trajectory increases for GOP of 2 as the trajectory order increases from linear (first order, used in LMCTI), to quadratic (second order) and to cubic (third order) trajectory used in the formulation of HOPTTI.



(a)



(b)

Figure 5.6 Example segments of the higher order motion trajectory of an object in (a) 3-D space between time t_1 and t_4, where K are the key frames (Akinola, Dooley and Wong 2010) and (b) 2-D slice of same object and SI is the object of the WZ frame.

5.3.2 The HOPTTI Algorithm

The structure for HOPTTI framework is shown in Figure 5.7 with the individual blocks explained.

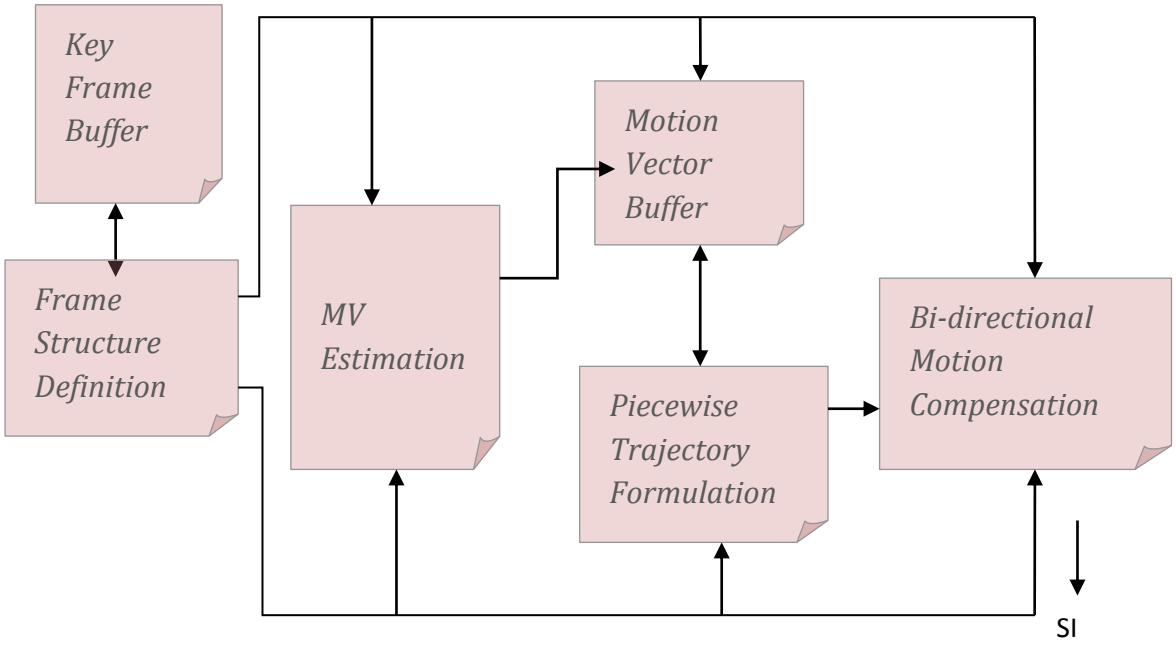


Figure 5.7 Block diagram of the HOPTTI algorithm

(i) *Frame structure definition:*

The structure is as follows:

Estimate the MVs (*A*, *B*, *C* and *D* as shown in Figure 5.6) for both forward and backward path using ARPS (Zhao et al. 2008).

A fractional weight ζ is introduced to combine the temporal SI (the missing WZ frames) in both forward and backward directions, shown in Figure 5.8, which gives the highest PSNR. For LMCTI ζ=0.5 (Artigas et al. 2007), averaging both forward and backward MC frames, while for higher-order models the value of ζ must take cognizance of the fact a MV may not necessarily intersect at the centre of a block.

Using the estimated parameters from Step 1, calculate the cubic polynomial motion trajectory; the SI using bi-directional motion compensation and the weight ζ to generate the final interpolated frame (see Figure 5.8).

Repeat Steps 1 to 3 for all SI frames.

(ii) *Piecewise trajectory formulation and bi-directional motion compensation:*

The HOPTTI formulation uses a trajectory formulation analogous to Chahine and Konrad (1995), Chahine (1995) and the same as (Akinola, Dooley and Wong 2010). The block-based motion estimation scheme does not capture all aspects of the motion field, therefore the higher order piecewise trajectory and bi-directional motion estimation and compensation reduces the impact of this by refining the MV in a way similar to B-frames in conventional video coding. In DVC however, the original block is not known at the decoder and the corresponding residue is not available so a different refinement strategy following the higher order trajectory is applied. In addition, MV estimation for uncovered areas (corresponding to holes) is estimated from previous frames in both the forward and backward directions resulting in forward and backward interpolated frames. In bi-directional motion estimation and compensation in LMCTI schemes, there are two methods used to generate the final interpolated frame, namely: i) spatial hole tracking and filling as in Li (2008), which involves pixel-wise searching of both forward and backward frames which is computationally very expensive, and ii) a temporal MV scheme (Aaron, Setton and Girod 2003; Aaron, Zhang and Girod 2002; Pereira et al. 2008; Artigas et al. 2007; Liu, Yue and Chan 2009; Ye et al. 2009) where the forward and backward MV interpolated frames are averaged together. The second approach has been adopted most often due to its simplicity and lower computational cost. However, due to irregular object motion in real world video, the scheme which assumes linearity and regular object motion is not sufficient. Figure 5.8 shows the approach adopted in this work with MC_F and MC_B representing forward and backward motion compensation respectively, ζ representing the weighting factor and K_A K_B K_C K_D representing the ME of MBs, *A, B, C and D* rerespectively.

HOPTTI introduces a weighting factor ζ to adjust the respective contributions of the forward and backward MVs to generate the interpolated frame, as illustrated in Figure 5.8. The best ζ is empirically determined to provide the highest average PSNR for each sequence. The empirical analysis of best ζ and how it is determined is provided in the results section (section 5.5) of this Chapter. Figure 5.8 shows the use of the weight ζ to obtain the final interpolation frame from the forward and backward frames, while at the same time illustrating how the MV of a block from 4 key frames is used to obtain the piecewise cubic motion trajectory, which is shown more clearly by the illustration in Figure 5.9 (a) and (b).

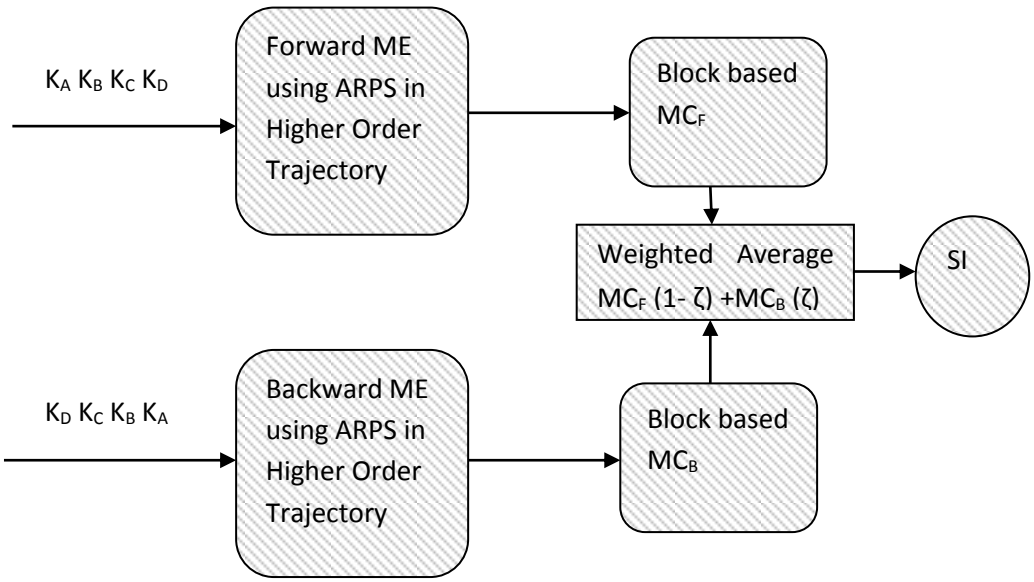


Figure 5.8 Bidirectional motion estimation and compensation with cubic trajectory and MV sampling estimation at decoder. Fractional weight ζ combines for final SI frame.

5.4 HOPTTI AND DIFFERENT GOPs

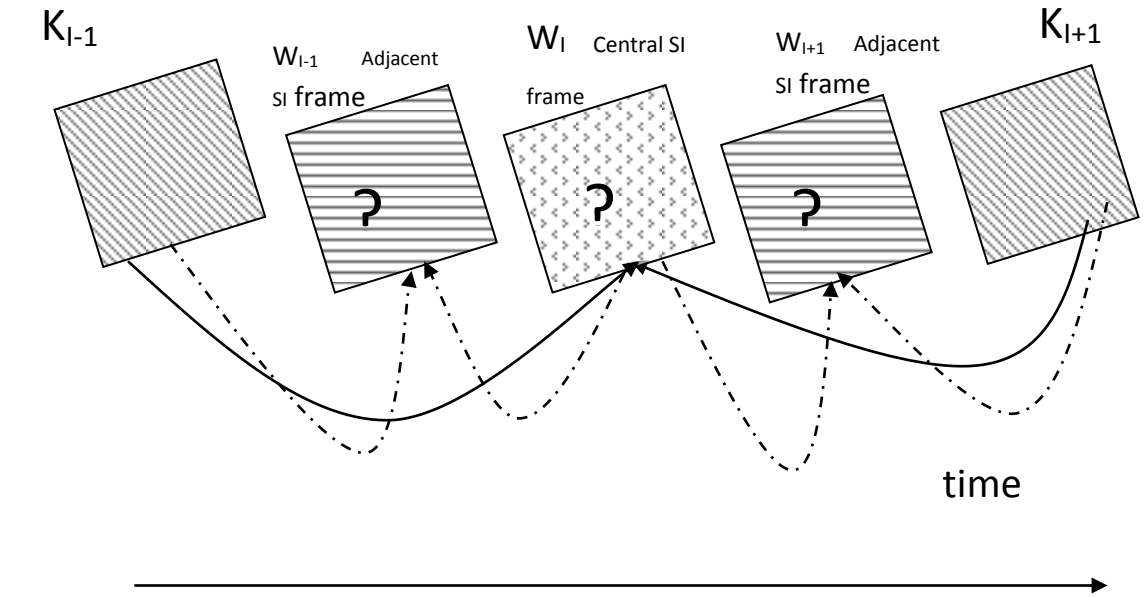
Until now, the GOP of 2 i.e. the case where the sequence is split evenly between key frames and WZ frames (KWKWKW…), was considered. It is of interest to observe the effect of the higher order trajectory on different GOPs as this will imply more WZ frames

and fewer key frames. Higher GOPs are employed in DVC in order to reduce the encoder overhead in DVC as this reduces the number of key frames that have to be encoded via the conventional codec at the encoder, while transferring the work load to the decoder where the exploitation of the WZ frames takes place. As will be evidenced in the results section, the higher order trajectory formulation gives a more accurate model also for higher GOP values.

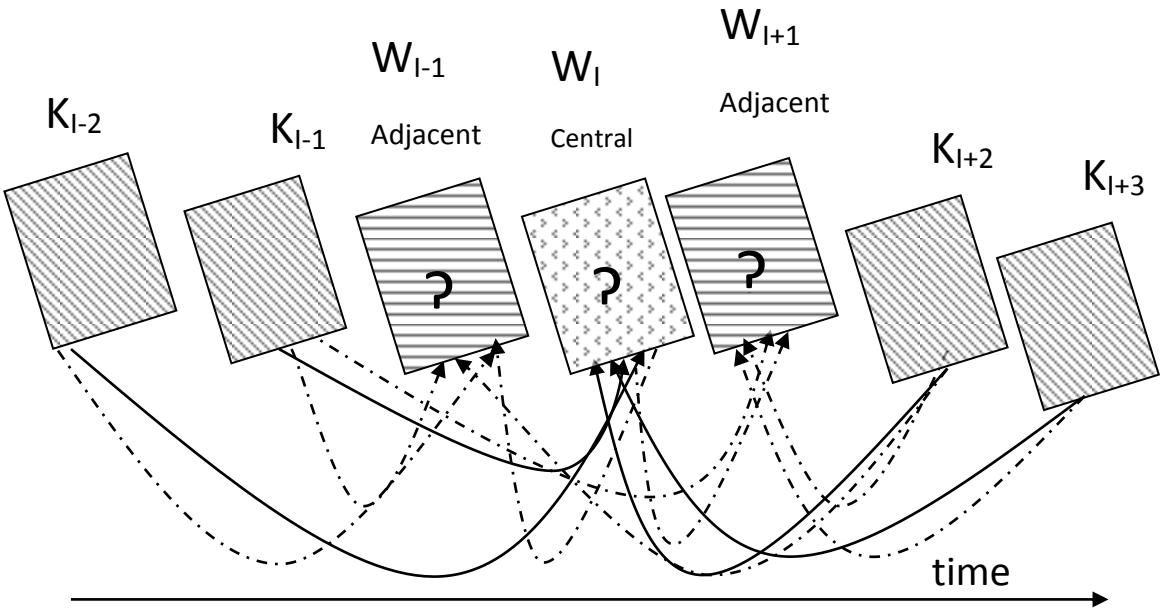
The structure adopted is similar to Aaron, Setton and Girod (2003), Ascenso, Brites and Pereira (2006) and Petrazouli et al. (2010). Figure 5.9 (b) follows from Figure 5.4 (a) and (b) where the number of frames for higher order interpolation was illustrated for the GOP of 2. While it is straight forward in higher order interpolation for a GOP of 2, where one WZ frame is missing between key frames, to use the adjacent key frames (previous and future frames), it becomes more complicated when higher order trajectories are considered for higher GOP values.

In Figure 5.9 (a), the linear interpolation of a GOP of 4 (KWWWKWWWK…) is depicted with three WZ frames and SI for these three frames have to be interpolated. The key frames are used to generate the central SI frame first and after generating the central SI frame, this is then used alongside the future frame to generate the adjacent SI frame between the future frame and the central SI frame. Likewise, the previous key frame is employed alongside the central SI frame to generate the adjacent SI frame between the previous frame and the central SI frame.

Figure 5.4 (b), illustrated the case where two past frames and two future frames were employed for the piecewise cubic trajectory formulation for a GOP of 2 (KKWKK…). This is further extended in Figure 5.9 (b) to illustrate how the cubic trajectory formulation works for the GOP of 4. While the linear GOP of 4 case is (KWWWK…), the cubic trajectory formulation for GOP of 4 case is (KKWWWKK…). Two past frame and two future frames are employed to generate the central SI (W_I), then the central SI is employed



(a)



(b)

Figure 5.9 Illustration of GOP of 4 using (a) Linear and (b) Cubic SI frame generation.

alongside two future frames and the past key frame to generate the adjacent SI frame (W_{I+1}) between the central SI and the past frame. Likewise, the central SI is employed

alongside the two past frames and one future frame to generate the adjacent SI frame between the central SI and the past frames (W_{I-1}).

The summary of the different GOPs using HOPTTI is as follows:

(i) The generation of the central SI (W_I) is done using at least four frames corresponding to *A, B, C and D* in the HOPTTI formulation.

(ii) An implementation of (a.) is illustrated in Figure 5.9 by using one previous key frame and three future key frames.

(iii) Lateral SIs {(W_{I+1}), (W_{I-1})}, are then generated using the central SI in combination with the least number of key frames that is needed to complete HOPTTI trajectory that encloses the lateral SI in question.

5.5 Simulation and Results

The HOPTTI algorithm and test bed as described in Section 2.2.5 was implemented in Matlab version 7.5.0 (R2007b) running under Microsoft Windows XP on a PC with an Intel Duo Core CPU at 2.20 GHz. A GOP size of 2 was chosen for the initial proof of concept experiments reported. Where higher GOP is utilized this will be explicitly stated in this section.

The cubic trajectory is employed in HOPTTI which is one order higher than the implementation in Chahine and Konrad (1995) and Chahine (1995) and uses polynomial trajectory parameterization as described in Section 4.3. The reason for choosing cubic as will be evidenced shortly is that the complexity increases as the order increases while the gain in terms of improved SI quality reduces.

Furthermore, the simulation and results are employed to show rigorous proof of the concept that the HOPTTI higher order algorithm gives a more accurate temporal

exploitation of video sequences in DVC as espoused in BLOCK 1 of Figure 1.1 of Chapter 1 and introductory section of this Chapter.

To numerically and qualitatively evaluate HOPTTI, the simulation to analyze and determine that the cubic higher order trajectory gives the best trade off as proposed in the theory sections is shown, then the direct comparison of HOPTTI with the results from Li's SI generation algorithm (Li, 2008) that was replaced with HOPTTI, furthermore, various QCIF (Quarter Common Intermediate Format) test sequences were applied including *Carphone, Mother, Coastguard, Silent, Hall,* and *Foreman,* which provided a range of different types of motion and objects. Three LMCTI interpolation-based SI generation approaches widely referenced in literature and that also employed various QCIF sequences were used for comparison, namely the 3-D content *adaptive recursive search* (3DCARS) (Borchert et al. 2007) which employs a quarter-pixel MV search implying 16 more MV searches than HOPTTI, the *pixel-domain WZ* (PD-WZ) codec (Tagliasacchi et al 2008) and the publication introducing global and feature based points in motion estimation (MPBTI), (Hansel, Richter and Muller 2011). All these SI generation techniques use linear interpolation allied with various temporal and spatial refinements respectively.

5.5.1 The Higher Order SI Algorithm Computational Complexity Evaluation

Experimental analysis corroborates the theoretical choice of the cubic trajectory in the HOPTTI formulation as PSNR quality is employed with overhead analysis. Table 5.1 summarizes the HOPTTI algorithm SI average PSNR results for linear, quadratic and cubic-order trajectories for various sequences. The results confirm consistently superior SI quality is achieved when a cubic polynomial trajectory model is applied to the various sequences, with for instance *Foreman* providing an average improvement of up to 5dB and *Coastguard* an 8dB improvement compared with the linear HOPTTI model. Furthermore, Table 5.1 reveals the HOPTTI algorithm exhibits progressive SI quality improvement for

increasing polynomial trajectory order, i.e., quadratic over linear and cubic over quadratic. However, the percentage change in PSNR reveals that while there is improvement from linear to quadratic and quadratic to cubic, the increase from quadratic to cubic is less than half of the increment from linear to quadratic. Table 5.2 shows the increase in overhead in terms of computational time for SI generation from linear to quadratic and cubic trajectories. The Table 5.2 shows that the overhead increases dramatically with video sequences exhibiting up to 600% increase in complexity as we go from linear to quadratic and cubic trajectories even as the PSNR improvement change reduces as seen in Table 5.1. Pragmatically however, when the SI improvements are juxtaposed by a higher complexity overhead, it will be seen that consideration of even higher-order polynomial trajectories above the cubic cannot be justified. Therefore, the cubic trajectory is chosen for the HOPTTI implementation.

Table 5.1 SI AVERAGE PSNR PERFORMANCE COMPARISON IN dB FOR VARIOUS TRAJECTORY ORDERS FOR HOPTTI

Sequences	***HOPTTI Linear(L)***	***HOPTTI Quadratic(Q)***	***%change Q-L/L***	***HOPTTI Cubic(C)***	***%change C-L/L***
Carphone	30.9	34.0	10.0	35.3	14.2
Coastguard	28.4	34.3	20.8	36.4	28.2
Foreman	29.9	33.0	10.4	35.1	17.4
Mother	36.2	44.4	22.7	47.3	30.7
Hall	30.5	36.2	18.7	38.5	26.2
Silent	31.7	37.2	17.4	38.9	22.7

Table 5.2 AVERAGE SI GENERATION TIME PER FRAME IN SECONDS

Sequences	***HOPTTI Linear(L)***	***HOPTTI Quadratic(Q)***	***%change Q-L/L***	***HOPTTI Cubic(C)***	***%change C-L/L***
Carphone	0.01	0.03	200	0.07	600
Coastguard	0.02	0.06	200	0.10	400
Foreman	0.03	0.08	166.7	0.14	366.7
Mother	0.02	0.04	100	0.08	300
Hall	0.03	0.06	100	0.11	266.7
Silent	0.01	0.03	200	0.06	500

Finally, from the computational complexity variables defined in Table 4.1 of Chapter 4, the complexities for HOPTTI_Linear is $T_{Const} + T_{Vel} + T_{BMA}$, HOPTTI_Quadratic is $T_{Const} + 2T_{Vel} + T_{Accl} + T_{BMA}$, while overhead for HOPTTI_Cubic is $T_{Const} + 4T_{Vel} + 2T_{Accl} + T_{Jolt} + T_{BMA}$. Thus, it is possible to observe that there is correlation with the time recorded in Table 4.4. In terms of complexity, HOPTTI cubic is twice that of the quadratic and 5 times greater than HOPTTI linear. The variations in time between video sequences and order can be accounted for from the fact that there are slight variations in the BMA complexity of the sequences with video sequences and further variation between complexity for calculating of velocity, acceleration and jolt terms.

5.5.2 Comparison of HOPTTI with other SI generation schemes

In order to compare HOPTTI with other SI generation schemes, we have employed the same sequences used by the authors and taken the results published. Tables 5.3, 5.4 and 5.5 show the comparative results for HOPTTI with 3DCARS (Borchert et al. 2007), PD-WZ (Tagliasacchi et al 2008) and MPBTI (Hansel, Richter and Muller 2011). This reveals for example, that for *Mother,* HOPTTI gave better SI quality with an improvement on average of 2.9dB and 9.0dB respectively compared with 3DCARS, PD-WZ and MPBTI. This is

particularly noteworthy when noted the enhancements introduced in both (Borchert et al. 2007) and (Tagliasacchi et al 2008). While MPBTI introduced feature points in ME and global ME alongside LMCTI, 3DCARS for example used quarter-pixel interpolation accuracy compared with the integer accuracy for HOPTTI alongside LMCTI. The HOPTTI algorithm also provided better SI quality of more than 2dB over 3DCARS for *Coastguard*, though this was counterbalanced by its performance being less satisfactory compared to PD-WZ (Borchert et al. 2007; Tagliasacchi et al 2008). This was due to the influence of the significant temporal noise components produced by the water in this sequence. While it should be noted that the PSNR values we are comparing HOPTTI with are not the results of LMCTI implementation alone, but combined with other algorithms, which in itself is an indication that the authors realized the fact that LMCTI is not quite adequate as alluded to earlier. The fact that HOPTTI without any additional improvements gives better performance compared to these improved LMCTI implementations further strengthens the higher order approach.

Table 5.3 COMPARISON OF AVERAGE PSNR IN dB PERFORMANCE FOR 3DCARS SI INTERPOLATION TECHNIQUE FOR VARIOUS VIDEO SEQUENCES

Sequences	***3DCARS (Borchert et al. 2007)***	***HOPTTI***
Carphone	34.9	35.3
Mother	44.4	47.3
Foreman	34.9	35.1
Silent	36.4	38.9
Coastguard	37.5	36.4
Hall	37.4	38.5

Furthermore, the argument of lower overhead for LMCTI cannot be sustained as the additional algorithms required to improve LMCTI based SI increase the overhead considerably (Borchert et al. 2007; Tagliasacchi et al 2008; Hansel, Richter and Muller 2011).

Table 5.4 COMPARISON OF AVERAGE PSNR IN dB PERFORMANCE FOR PD-WZ SI INTERPOLATION TECHNIQUE FOR VARIOUS VIDEO SEQUENCES

Sequences	***PD-WZ (Tagliasacchi et al. 2008)***	***HOPTTI***
Mother	38.3	47.3
Foreman	33.0	35.1
Coastguard	34.2	36.4
Hall	36.8	38.5

Table 5.5 COMPARISON OF AVERAGE PSNR IN dB PERFORMANCE FOR MPBTI SI INTERPOLATION TECHNIQUE FOR VARIOUS VIDEO SEQUENCES

Sequences	***MPBTI (Hansel et al. 2011)***	***HOPTTI***
Foreman	30.2	35.1
Coastguard	32.8	36.4
Soccer	22.9	26.4
Stefan	24.2	28.5

Figures 5.10, 5.11 show the perceptual SI quality of sample frames and the frame by frame quantitative quality for *Hall*, where HOPTTI performed better in comparison with the

other SI generation schemes, with Tables 5.3 and 5.4 confirming the average numerical PSNR improvements are 1.1dB and 1.7dB respectively.

For the same *Hall* sequence, looking at the frame by frame plot in Figure 5.11, it is noted that that the plot pattern is closely related to other such frame-wise plots for the *Hall* sequence in literature (Borchert et al. 2007; Tagliasacchi et al 2008; Hansel, Richter and Muller 2011). The major differences in Figure 5.11 however are: 1) The SI PSNR values are higher than those already reported in literature and 2) the falling off in performance (dips in PSNR) while new objects emerge, while objects increase their motion and while background changes occur due to translational motion, global motion panning, sudden movements etc., still occurs in similar locations for HOPTTI, but unlike the other generation schemes that show greater reduction in quality, HOPTTI quickly stops the decline as it predicts the direction of motion more accurately by the higher order trajectory as predicted in the earlier sections of this Chapter. Furthermore, Figure 5.10 which shows qualitative sample frames of SI produced from the *Hall* sequence, proves that the quality of SI produced from the HOPTTI algorithm is quite high as it shows that there are no holes evident in the sample frames while the MB forms are continous without breaks visible to the human perception. Figure 5.10 also highlighted major areas of difficulty where the HOPTTI algorithm struggles, which include appearance of a man in the hall, turning of the man from doorway into direction of hall, acceleration of movement etc. In all these cases the qualitative SI output of HOPTTI is commendably high. The sample frames in Figures 5.10 and 5.11 also shows the places where HOPTTI fails, and as such further improvements can be contemplated. Such areas where there is panning in the background and new objects appear have been adequately filled by the bi-directional estimation employed by HOPTTI but they show overlapping of blocks and slight ghosting, especially in high motion object areas such as the legs of the human objects. These and other weaknesses observed in the HOPTTI algorithm will be further explored in Chapter 6 in

order to improve the DVC codec and make a further contribution by tackling the overlapping observed.

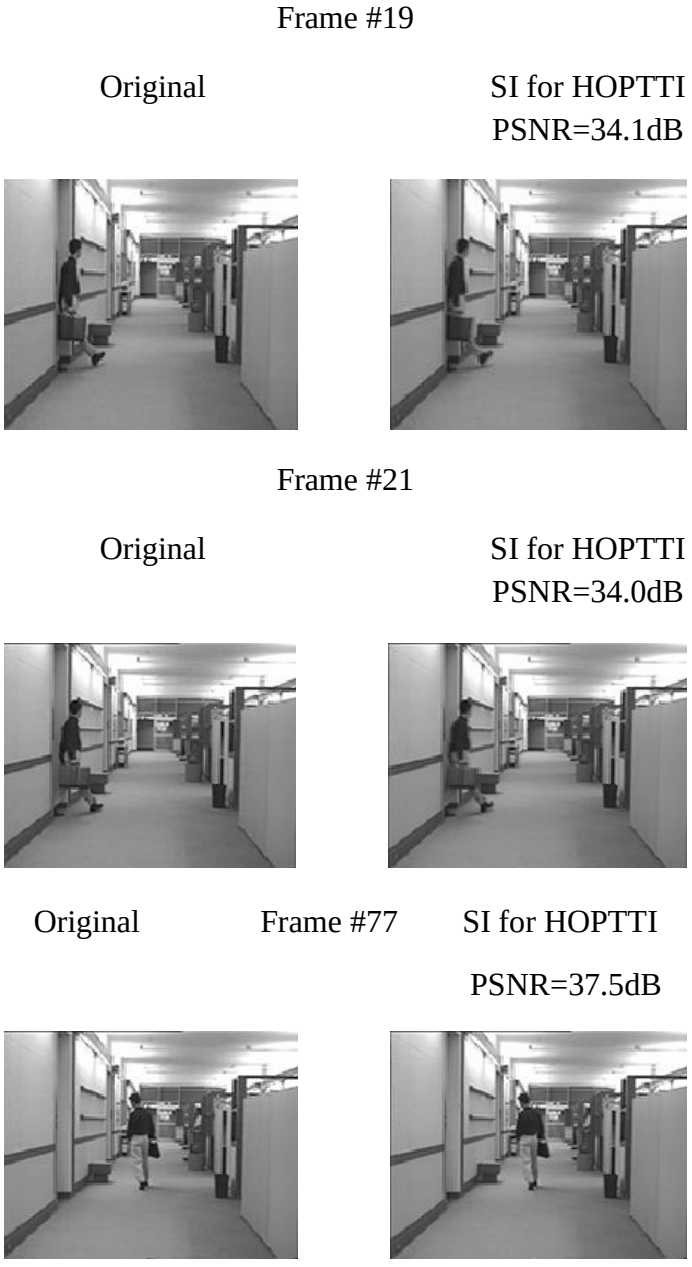


Figure 5.10 Sample frames for the *Hall* sequence showing the SI quality obtained using HOPTTI algorithm

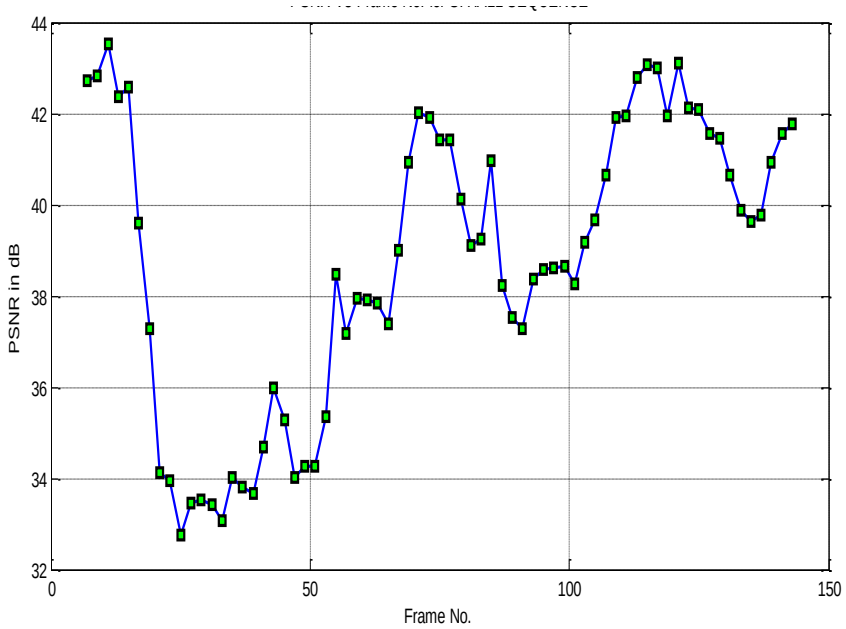


Figure 5.11 Frame-wise plot showing SI quality of HOPTTI algorithm for *Hall* sequence

The *Stefan* sequence sample frames shown in Figure 5.12 and the frame-wise PSNR plot of Figure 5.13 shows that HOPTTI copes better with sudden movements such as jerks and jumps by the player as predicted from the HOPTTI formulation. Though the frame-wise plot in Figure 5.13 follows the pattern of *Stefan* sequence framewise plots, falling at similar locations as usual (Borchert et al. 2007; Tagliasacchi et al 2008; Hansel, Richter and Muller 2011), HOPTTI gives higher PSNR performance and thus shorter dips showing that HOPTTI handles the translational movements, bounces, jumps and jerks of the object movement better than LMCTI SI generation algorithms. Looking more closely at the sample frames of Figure 5.12, the qualitative output of the HOPTTI generation scheme gives good perceptual quality making it possible to distinctly recognise all the objects. While there are no visible holes, the multiple objects in the background (spectators) sometimes are blurry showing multiple overlapping of MV blocks. This further reinforces the fact that HOPTTI suffers from overlapping of multiple MVs on the same MB.

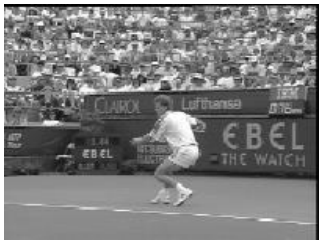
Frame #25

Original

SI for HOPTTI
PSNR=32.4dB

Frame #89

Original



SI for HOPTTI
PSNR=23.9dB

Frame #93

Original

SI for HOPTTI
PSNR=23.1dB

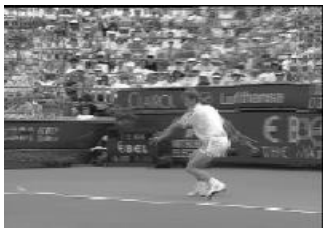


Figure 5.12 Sample frames for the *Stefan* sequence showing the SI quality obtained using HOPTTI algorithm.

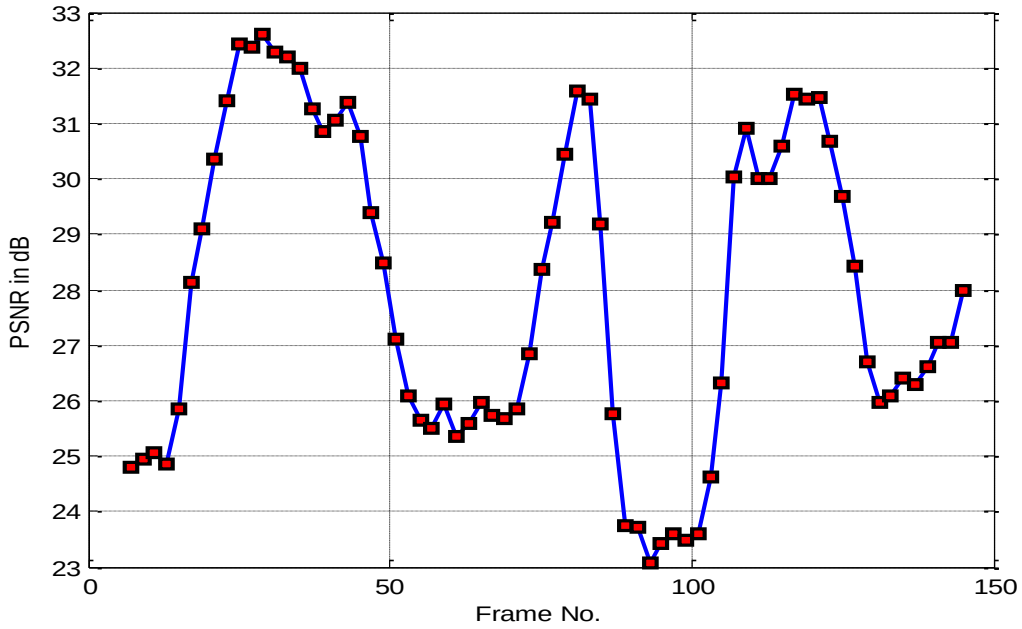


Figure 5.13 Frame-wise plot showing SI quality of HOPTTI algorithm for *Stefan* sequence

The *Foreman* sequence shows another set of characteristics which is difficult for the HOPTTI SI generation formulation to deal with, as the objects occupy large swathes of the sequence frame making it traverse many blocks. Also, translational motion is not very frequently prevalent though the *Foreman* sequence is regarded as a fast motion sequence. The fast motion comprises of the deformation of parts of the face and movement of the head which results in sudden jerks and rotational motion. Figures 5.14 and 5.15 shows sample frames of the qualitative SI output of the HOPTTI SI generation module and the frame-wise PSNR results which also shows that HOPTTI performs quite remarkably well considering that the formulation captures mostly translational motion. Tables 5.3, 5.4 and 5.5 shows up to 3.9dB gain over published LMCTI SI generation results, as LMCTI finds it really difficult to cope with the complex motion component of the *Foreman* sequence. The qualitative results of HOPTTI shows serious overlapping and in some extreme cases the MBs in some frames can be distinguished, which is evidence that there are some holes at the MB edge or that the transition between MBs are not smooth. For example in frame

81, there appeared two ears and the MBs around the cheek area are clearly visible. One immediate area where HOPTTI needs improvement is the inclusion of an algorithm that will tackle overlapping while at the same time improving the smoothing between MBs so that MB borders are no longer visible in all sequences.

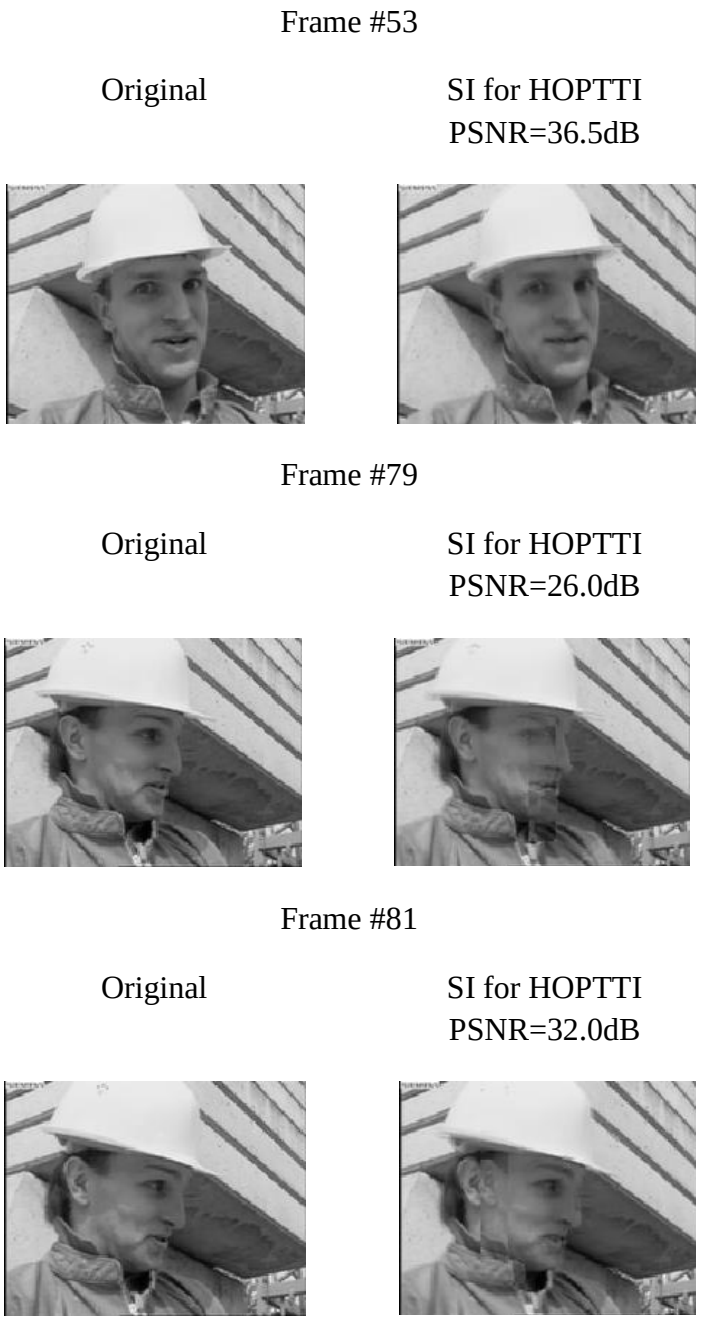


Figure 5.14 Sample frames for the *Foreman* sequence showing the SI quality obtained using HOPTTI algorithm

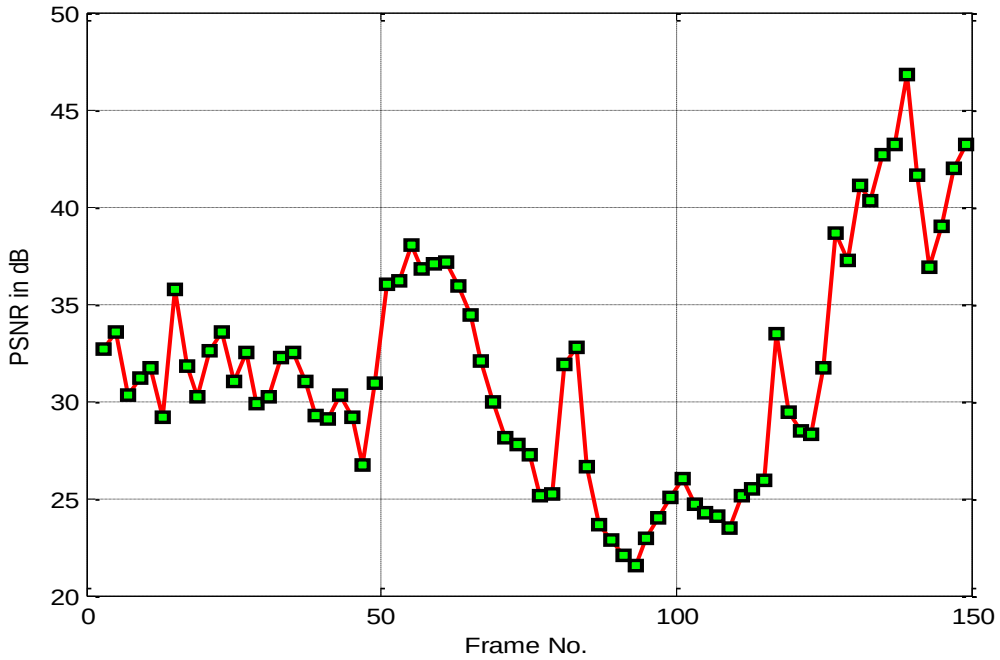


Figure 5.15 Frame-wise plot showing SI quality of HOPTTI algorithm for *Foreman* sequence

5.5.3 Overall DVC RD Performance using SI generated by HOPTTI

Conventionally, the accepted method in coding literature to access the performance of a codec is to plot its RD curves. Therfore, in order to effectively evaluate the overall effect of the HOPTTI SI generation scheme it is necessary to insert the HOPTTI SI algorithm in a complete DVC codec, and as explained in Chapter 3 and mentioned in earlier sections of this chapter, it is possible to employ any modular codec that enables us to replace the SI generation module.

For this purpose as stated earlier, the DVC codec by Li *(2008)* has been employed. The first set of evaluation performed would be to compare HOPTTI results with the results published from Li using the same testing scenarios. In Li the first 30 frames of *Foreman* and *Hall* sequences were tested making use of QCIF test sequences, additionally the originals were made available and the published results are shown alongside other RD curves in Figures 5.16 and 5.17.

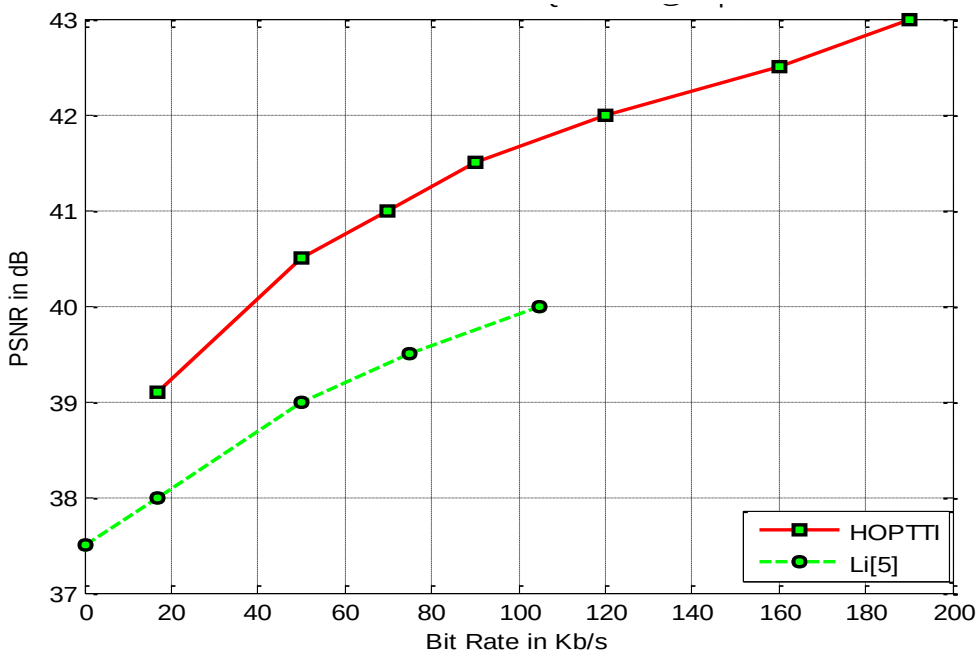


Figure 5.16 RD Curves showing HOPTTI PSNR performance for *Foreman* sequence @ 15f/s using original as key frames for first 30 frames.

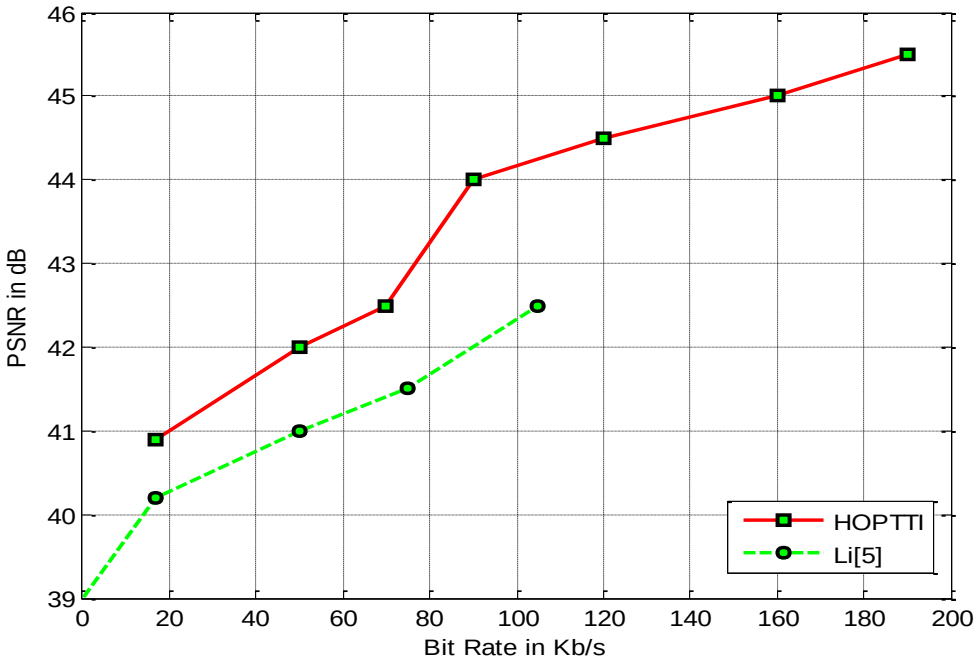


Figure 5.17 RD Curves showing HOPTTI PSNR performance for *Hall* sequence @ 15f/s using original as key frames for first 30 frames.

The results for the *Foreman* sequence, in Figure 5.16, shows that the HOPTTI algorithm has improved the codec with a PSNR gain of at least 1.8 dB for the same scenario. Thus,

showing that the HOPTTI based codec outperforms the Li codec under the same conditions, due primarily to the gains from the SI generation scheme employed in HOPTTI.

Likewise, the RD curves for the *Hall* sequence is plotted for rigorous comparison with the Li (2008) codec. Figure 5.17 shows the RD plots for the scenario similar to that employed for the published Li results, which implies that the first 30 frames of the *Hall* sequence is considered and key frames are original frames. Figure 5.17 shows that the HOPTTI based codec outperforms the Li codec by up to 2.6dB.

In order to compare the HOPTTI based codec with the DISCOVER codec (Artigas et al. 2007) which is discussed in detail in the introductory Chapter 1 of this thesis and is a popular DVC codec that is well referenced and highly regarded in DVC literature, RD curves of the HOPTTI based codec employing similar scenarios to the ones under which the DISCOVER codec operates are plotted. The most important fact in the DISCOVER codec is that it employs the H.264 Intra codec as key frames. This implies lower quality key frames than that employed by Li *(2008)* and this in turn affects the codec RD performance.

The RD curve of HOPTTI based codec employing H.264 intra codec as key frames as shown in Figure 5.18 outperforms the DISCOVER RD curves and the H.264 intra RD curve by up to 1.4 dB while it also gives a performance very close to the H.264 No Motion codec at low bit rates.

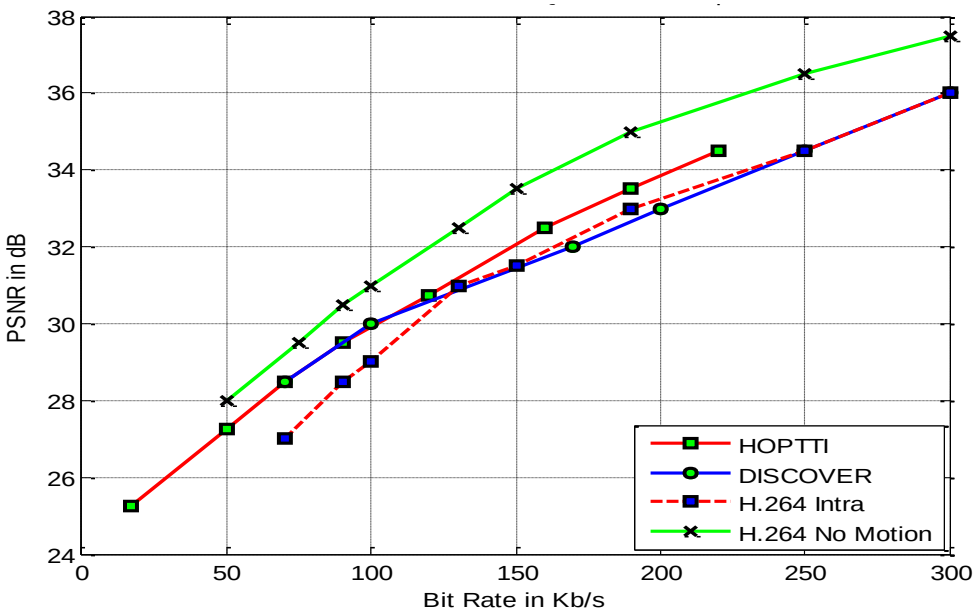


Figure 5.18 RD Curves showing HOPTTI PSNR performance for *Foreman* sequence @ 15f/s using H.264 Intra as key frames and the whole *Foreman* sequence WZ frames only.

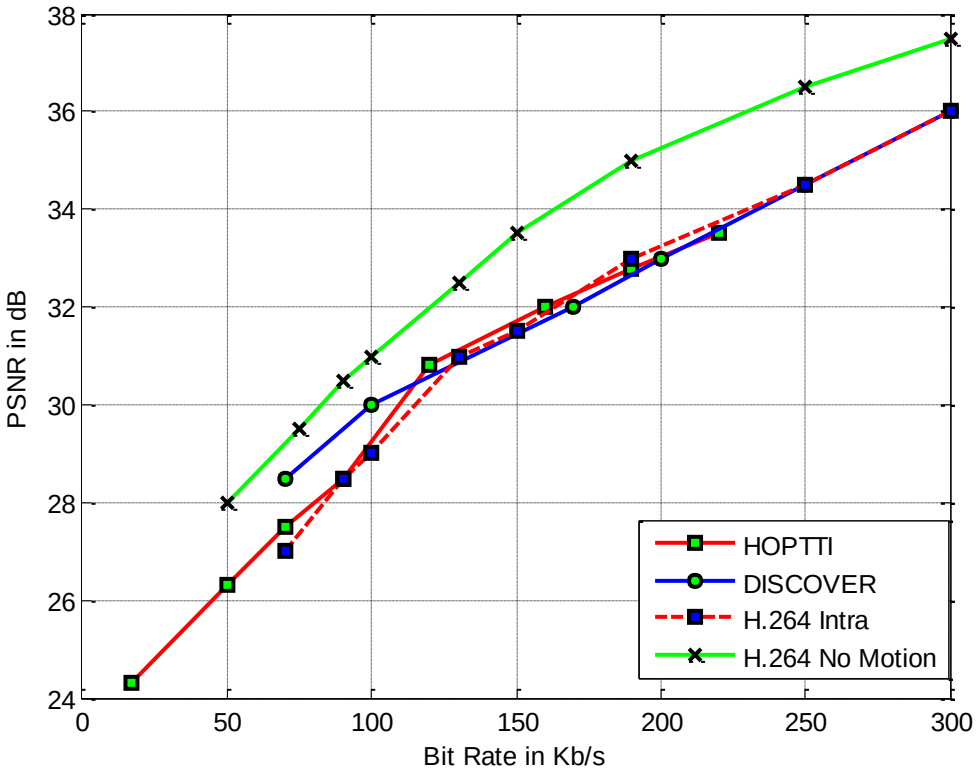


Figure 5.19 RD Curves showing HOPTTI PSNR performance for *Foreman* sequence @ 15f/s using H.264 Intra as key frames and the whole *Foreman* sequence with key frame rates added.

Figure 5.18 shows the RD curve of the HOPTTI based codec with the whole of the *Foreman* sequence being employed but considering only the reconstructed WZ frames and

H.264 Intra as key frames. This shows that the HOPTTI based codec outperforms the DISCOVER codec and that the H.264 No Motion conventional codec acts as an upper bound. Finally, Figure 5.19 shows the RD curve of the HOPTTI based codec with the most restrictive set of conditions which includes the fact that H.264 Intra is a key frame and the bit rates for the intra key frames being included in the RD plot. Under this condition, the HOPTTI based codec still slightly outperforms the DISCOVER codec.

In order to complete the RD analysis for the *Hall* sequence and compare with the DISCOVER codec as well, the HOPTTI based codec is tested with the H.264 intra conventional codec as key frames is shown in Figure 5.20, where the HOPTTI based codec outperforms the DISCOVER codec and performs very close to the H.264 No Motion conventional codec at low bit rates.

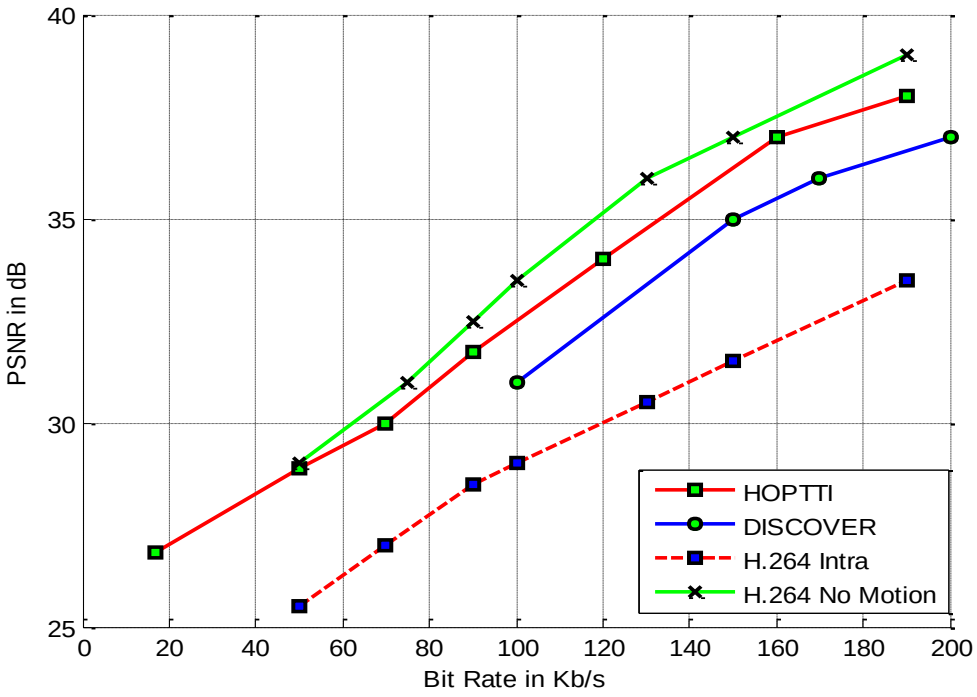


Figure 5.20 RD Curves showing HOPTTI PSNR performance for *Hall* sequence @ 15f/s using H.264 Intra as key frames and the whole *Hall* sequence WZ frames only.

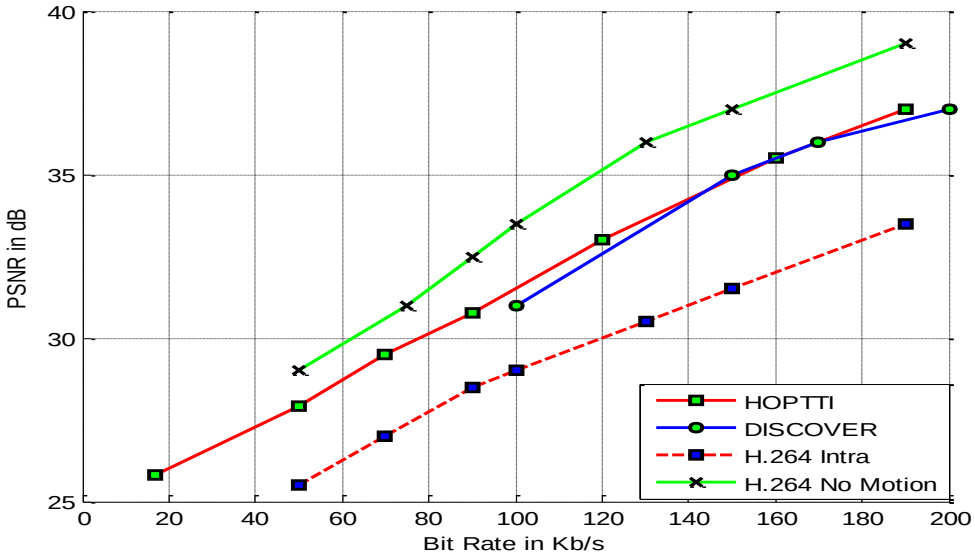


Figure 5.21 RD Curves showing HOPTTI PSNR performance for *Hall* sequence @ 15f/s using H.264 Intra as key frames and the whole *Hall* sequence with key frames rates added.

Furthermore, Figure 5.21 further relaxes the scenario by including the whole *Hall* sequence and including the rates from the key frames and the HOPTTI based codec outperforms the DISCOVER codec by up to 0.6dB which is a qualitatively perceptible difference.

5.5.4 Empirical Results, Analysis and Evaluation of Weighted (ζ) Combination of Forward and Backward MC frames in HOPTTI

The theoretical underpinnings of the HOPTTI formulation in Section 5.3 suggest that depending on the motion and other characteristics of the objects in the video sequences, the weighting of the forward trajectory MC frame and backward trajectory MC frame combination for maximizing PSNR performance will differ. In order to empirically determine the best weight therefore, simple experimentation for different values of weights for different sequences was performed and plotted in Figure 5.22. The results show that the weights have an impact on the PSNR performance and that the maximizing weight differs

from sequence to sequence. The effect of the video object content on PSNR performance will be further exploited in Chapter 7 of this thesis.

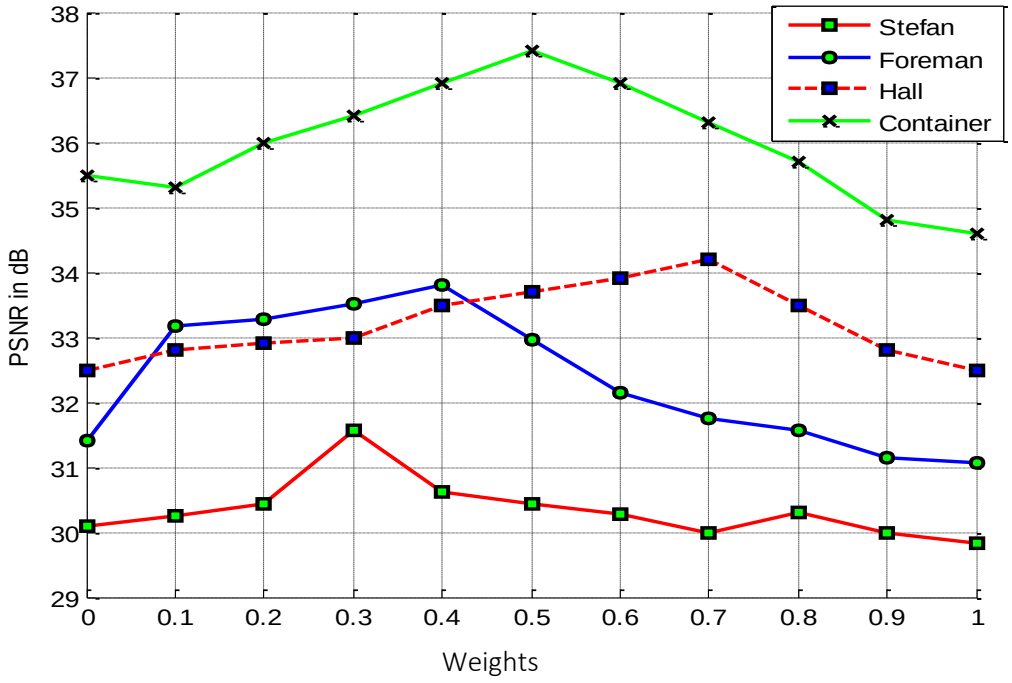


Figure 5.22 Empirical results for the determination of the Weights to maximize the PSNR when combining forward and backward MC in HOPTTI for various sequences.

5.5.5 Results for Various GOPs in HOPTTI

Using the same HOPTTI testbed described earlier and modifying the number of WZ frames dropped at the encoder, the various central and lateral SI frames were generated as discussed earlier and the results are shown in Tables 5.6 and 5.7. For effective comparison, the results for GOP 4 using linear interpolation with enhancement in *3DCARS* (Borchert et al. 2007) and GOP of 4 in *CA-WZ* (Ascenso et al. 2008) which is the *DISCOVER* implementation of longer temporal distance was employed. Furthermore HOPTTI was used to extensively test the various sequences and up to a GOP of 8 was tested.

Table 5.6 shows that *HOPTTI* provides improved quantitative results for the sequences tested in comparison with *3DCARS* and *CA-WZ* with up to 1.8 dB improvement in the *Stefan* sequence.

Table 5.7 shows that HOPTTI improves as the GOP increases when compared to 3DCARS (Borchert et al. 2007) as the reduction in quality is less than that reported for GOP of 4, even when there is enhancement of the results. As the GOP increases to 8 the decrease in quality is not significant as it is less than 0.5 dB for most of the sequences tested.

Table 5.6 COMPARISON OF AVERAGE PSNR IN dB PERFORMANCE FOR VARIOUS SI INTERPOLATION TECHNIQUES FOR GOP SIZE 4

Sequences	***3DCARS (Borchert et al. 2007)***	***CA-WZ (Ascenso et al. 2008)***	***HOPTTI***
Foreman	29.2	28.85	30.1
Coastguard	31.9	30.59	32.4
Stefan	23.2	21.53	25.0
News	32.7	32.80	33.6

Table 5.7 COMPARISON OF AVERAGE PSNR IN dB PERFORMANCE FOR VARIOUS SI HOPTTI INTERPOLATION FOR DIFFERENT GOP SIZES

Sequences	***HOPTTI Linear GOP 2***	***HOPTTI Linear GOP 4***	***HOPTTI Linear GOP 8***	***HOPTTI GOP 2***	***HOPTTI GOP 4***	***HOPTTI GOP 8***
Carphone	30.9	29.0	27.4	35.3	34.0	33.7
Mother	36.2	34.9	33.6	47.3	46.6	46.1
Foreman	29.9	26.4	23.2	35.1	33.1	30.9
Silent	31.7	30.2	28.7	38.9	38.1	37.8
Coastguard	28.4	23.5	19.5	36.4	32.4	31.1
Hall	30.5	28.5	26.4	38.5	37.6	37.2
Stefan	20.1	18.3	16.5	28.2	25.0	22.2
News	32.8	29.5	26.4	39.0	33.6	33.1

Furthermore, from Table 5.7, the HOPTTI linear (LMCTI) model using the HOPTTI codec (Akinola, Dooley and Wong 2010) to simulate various GOPs, it can be seen that the

degradation of LMCTI is progressive. As the GOP doubles from 2 to 4 the degradation in quality of SI is the same as when the GOP is doubled from 4 to 8.

5.5.6 Qualitative Results Showing Challenges to HOPTTI Algorithm

Though it has been shown conclusively from the qualitative and quantitative results presented in Sections 5.5.2 to 5.5.5 that a more accurate HOPTTI formulation gives an improved SI compared to LMCTI that is predominantly employed in literature, in this section we present some qualitative results showing the challenges that the HOPTTI algorithm has yet to overcome, which forms the bedrock of further algorithms and techniques introduced in the subsequent Chapters of these thesis to further tackle these challenges and overcome the bottleneck of poor SI in DVC. In particular, sample frames from video sequences that possess complex and challenging spatial-temporal characteristics that HOPTTI cannot overcome are presented.

First, frames from the multiple object, fast motion sequence *Hall* is presented. The characteristics are exemplified by frames #51, #85 and #99. In frame #51 there is ghosting in the left foot which is caused by overlapping of the higher order trajectories.

In frame #85 we have a different artifact on the left leg which presents itself as multiple lower part of the left leg which is due to the rotational movement that is not adequately captured by the HOPTTI formulation. Lastly, in frame #99, there is a slight protrusion around the neck which is caused by the deformable neck part of the object that is represented by different MVs.

Next, frame #143 from the multiple object, complex spatial characteristics sequence *Coastguard* is also presented which exemplifies another challenge that the HOPTTI algorithm was unable to deal with.

Frame #51

Original

SI for HOPTTI

PSNR=37.6dB

Frame #85

Original

SI for HOPTTI

PSNR=37.6dB

Frame #99

Original

SI for HOPTTI

PSNR=37.6dB

Figure 5.23 Sample Illustrations of Artifacts causing challenges for Qualitative performance of HOPTTI in *Hall* Sequence.

In the video there was fast panning of the background which results in the ghosting of the shrubs and the placing of shrubs where there was none in the original. This is due mainly to the spatial similarity between the background objects and the global motion that is not adequately captured in the HOPTTI formulation.

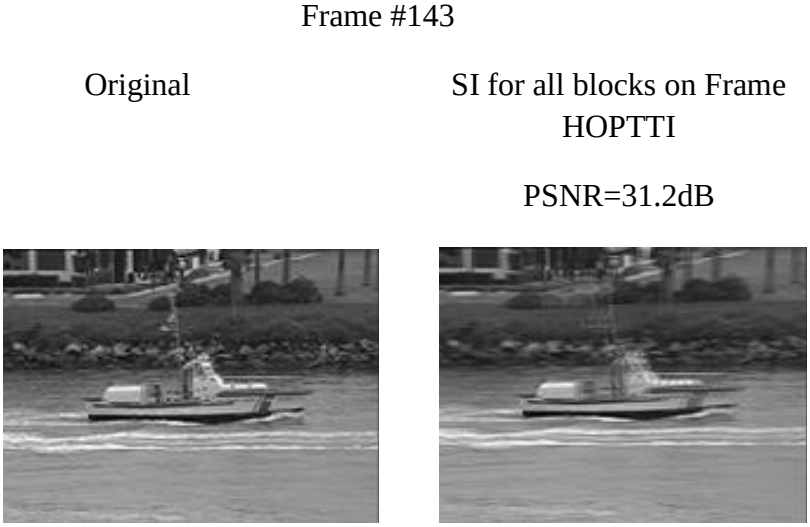


Figure 5.24 Sample Illustration of Artifacts causing challenges for Qualitative performance of HOPTTI in *Coastguard* Sequence.

Lastly, frames from the multiple image fast motion and complex spatial-temporal characteristics video sequence of American Football is presented. This is a very challenging sequence and HOPTTI gives a much more improved SI compared to LMCTI but frame #61 exemplifies the problems of ghosting which emanates from overlapping of MV trajectories of HOPTTI formulation.

Added to this is double object representation which emanates from the representation of deformable objects by the same MV. Frames 93 and 99 also shows serious artefacts that emanate from the same sources coupled with rotational and global motion that are not adequately captured in the HOPTTI formulation.

Frame #61

Original

SI for all blocks on Frame HOPTTI

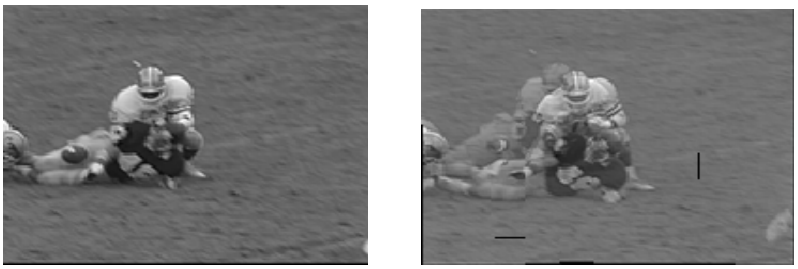
PSNR=24.6dB

Frame #93

Original

SI for all blocks on Frame HOPTTI

PSNR=24.5dB



Frame #99

Original

SI for all blocks on Frame HOPTTI

PSNR=24.8dB

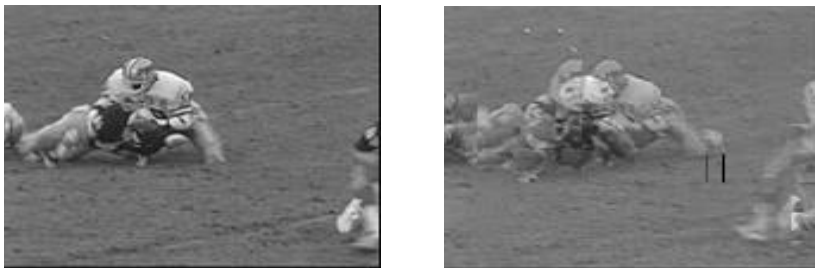


Figure 5.25 Sample Illustration of Artefacts causing challenges for Qualitative performance of HOPTTI in *American Football* Sequence.

5.6 Summary

This Chapter presents a novel method of generating SI in WZ coding using a *higher-order piecewise trajectory temporal interpolation* (HOPTTI) algorithm. Both numerical and qualitative results confirm that HOPTTI consistently provided superior SI quality compared to a number of existing interpolation techniques, especially for sequences which exhibited non-linear object motion.

Furthermore, the desirability of reducing the complexity of the encoder by dropping more frames and the resultant lengthening of the temporal distance by the use of higher GOP values is investigated showing that the use of higher order interpolation techniques brings the same benefits as that obtained from increased object motion and multiple objects in video sequences.

Finally, the lapses in the HOPTTI algorithm and inability to eradicate completely visual artifacts are highlighted. Though very high quantitative improvements are recorded for some high motion video sequences, it does not result in artifact free qualitative improvement and this is rooted in the formulation of the HOPTTI algorithm that employs fast block based ME techniques. The next Chapter is therefore devoted to improving these qualitative visual defects.

Chapter 6

Improved SI Generation Using Adaptive Overlapped Block Motion Compensation and Higher Order Interpolation

6.1 Introduction

The exploitation of correlation at the decoder where original video frames are not available is non-trivial and one of the critical factors impacting upon DVC performance is SI quality (Wyner and Ziv 1973; Akinola, Dooley and Wong 2010). As discussed and shown in Chapter 5, LMCTI has been widely adopted for SI generation, though findings (Petrazzuoli, Cagnazzo and Pesquet-Popescu 2010) show that better PSNR can be achieved when a more accurate object motion trajectory is adopted to produce SI. As natural motions are not always linear, HOPTTI as shown in Chapter 5 is able to better model these types of motion, by using 3 or more MVs from previous and future frames to predict the MV for a MB in the current frame which better exploits the temporal redundancies in the video sequences.

It is also clear from the results in Chapter 5 that visual artifacts still remain though HOPTTI is able to more accurately model object motion. One of the root causes of artifacts is the fact that BMA is employed in HOPTTI. Though HOPTTI comes with distinct advantages shown in Chapter 5, it has its own disadvantages including overlapping which lowers PSNR where abrupt changes in trajectory and global motion occur due to overlapped multiple trajectories created from the MVs of previous and future frames. When such motions occur,

this leads to ghosting in HOPTTI as well as other algorithms that employ variants of averaging the forward and backward MCI as explained in Chapter 3.

Another disadvantage that the use of BMA in HOPTTI is also blocking artifacts (sometimes called macro-blocking, quilting or checker-boarding, see sources of artifacts in Chapter 3) which makes the PSNR lower when deformable objects or more than one object is present in some MBs (Bosc et al. 2011; Liu et al. 2010), which are represented by a single MV per MB in the HOPTTI formulation though these may contain differing motions and would therefore require more than one MV to adequately characterize the motions in one MB. The concepts of overlapping and blocking artifacts are explored in more detail in Section 6.2.

In Chapter 3, presentation was made of various ways employed in literature of mitigating the introduction of artefacts in video sequences, reinforcing the fact that the analysis of artifacts to ascertain the root causes and the source of such artifacts is an all-important prelude to proffering effective solutions. Furthermore, in the literature survey in Chapter 3, several solutions have been proposed to mitigate the root problems of BMA that leads to artifacts in video outputs with BMA based processing. For example in Choi et al. (2007), the problem of discontinuities in BMA based MV estimation was highlighted, and the need to address this root cause of artifacts from BMA based processing by making sure that the discontinuities were eliminated and making use of MV estimation techniques that follow the true motion trajectory, eliminating outliers, and thus making sure that MVs are not distinctively different from their neighbours, was suggested. Another example is the proposal of Kuo and Kuo (1998) for the use of a median filtering technique that is to replace the MV at a point with the median value in the filtering neighbourhood, which requires complicated operations for the same BMA based artifacts and could produce undesirable artifacts of its own. However, from Chapter 3, the solution which effectively mitigates most

of the enumerated problems caused by BMA based processing while offering a video content adaptive approach is AOBMC (Orchard and Sullivan 1994; Choi et al. 2007).

In summary, AOBMC allows for the following:

(i) It allows for the use of a MV estimation that follows more closely the true motion trajectory and improves on it by moderating the MVs with the surrounding MBs.

(ii) It allows for the selection of the most appropriate MV when there are two or more MV candidates for the same pixel due to overlapping by grading the MVs using both the surrounding MBs and the video content parameter.

(iii) It allows for the use of bi-directional MV estimation such as employed in HOPTTI, thereby reducing drastically the possibility of holes in the output SI and thus handling blocks with deformable objects or parts of different objects.

(iv) It avoids over-smoothing at the edges by not applying the overlapping uniformly but with appropriate weighting and adaptive video content parameters taken into consideration.

Therefore, to further improve the qualitative performance of SI generated by HOPTTI while at the same time increasing PSNR and overall codec performance, tackling of root causes of artifacts as stated by Grouiller et al. (2007), is necessary, leading to the selection of AOBMC as an appropriate algorithm to explore. This explores the BLOCK 2 and BLOCK 3 in Figure 1.3 and the *SI Generation and Improvement Framework* in Figure 1.2 of Chapter 1 of this thesis which has recognized the problems associated with the use of BMA rather than pixel level or even sub-pixel level iteration would need addressing. The motivated research framework block diagram is reproduced with the BLOCK 2 and BLOCK 3 highlighted in Figure 6.1.

Thus investigation of applying the higher-order HOPTII algorithm alongside AOBMC to enhance SI quality and reduce the artifacts caused by the BMA is pursued here.

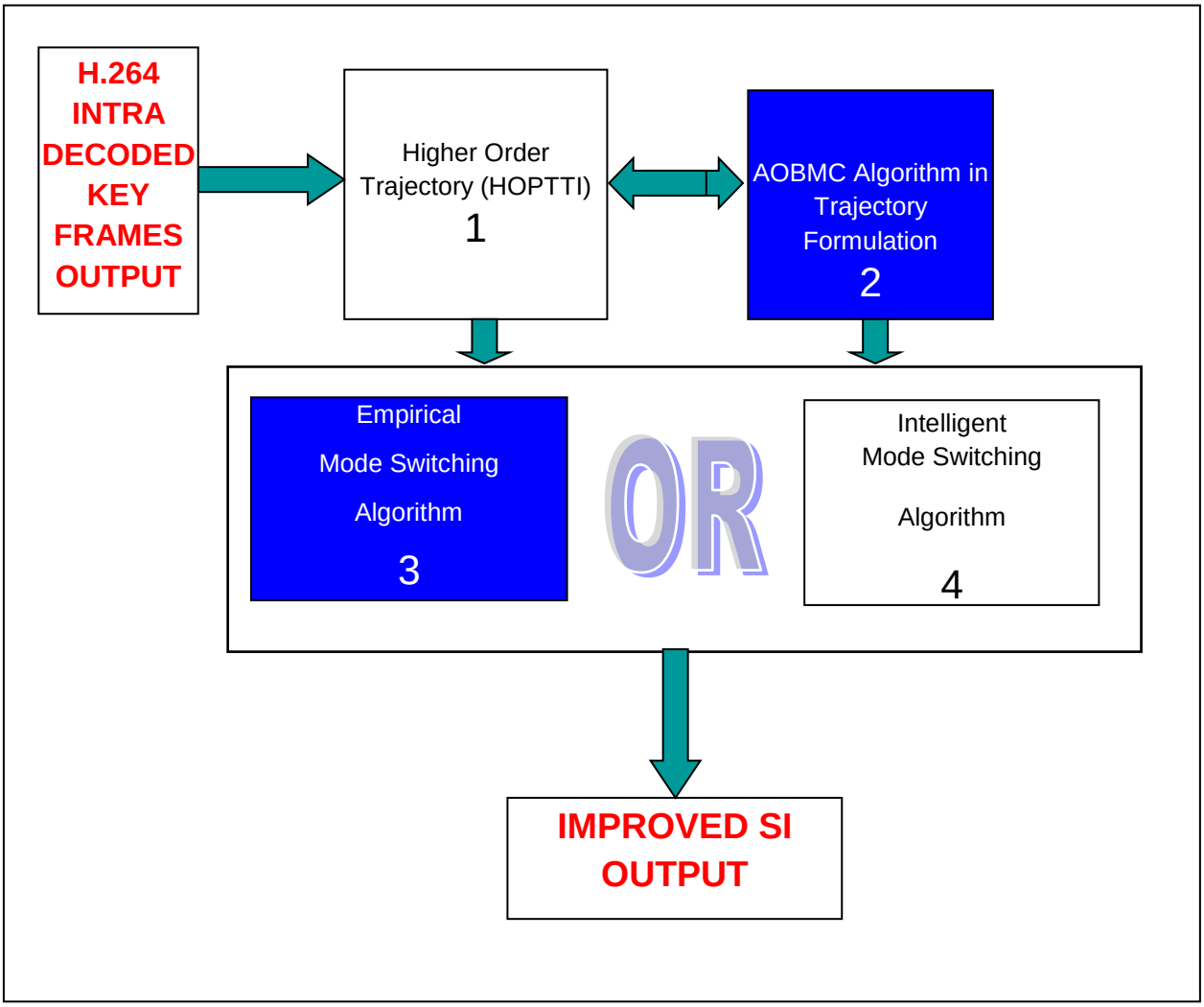


Figure 6.1 Block Diagram of *SI Generation and Improvement Framework* with BLOCK 2 and BLOCK3 Highlighted.

While an overall SI improvement is achieved, analysis reveals that for certain frames in various test sequences, HOPTII produced better SI quality than when combined with AOBMC. The reason for this is that some of the neighboring MVs are not correlated with one another and their addition to the reference MB used in the enlarged window degrades the overall SI quality in that particular frame. A *mode switching* (MS) technique based on Ye at al. (2009) is thus introduced which uses a matching criterion to switch between HOPTTI, and AOBMC combined with HOPTTI (AOBMC-H) to obtain a Switched HOPTTI-AOBMC final SI. The corresponding impact on SI quality of both the new

AOBMC-H approach and MS mechanism are also analyzed in this chapter, with numerical and perceptual results exhibiting a consistent improvement in overall SI quality.

The positioning of the AOBMC algorithm in the SI generation module of HOPTTI is shown in Figure 6.2 which shows the overall AOBMC-H block diagram. The AOBMC algorithm is the gridlocked circle on top of the Bi-directional MC module.

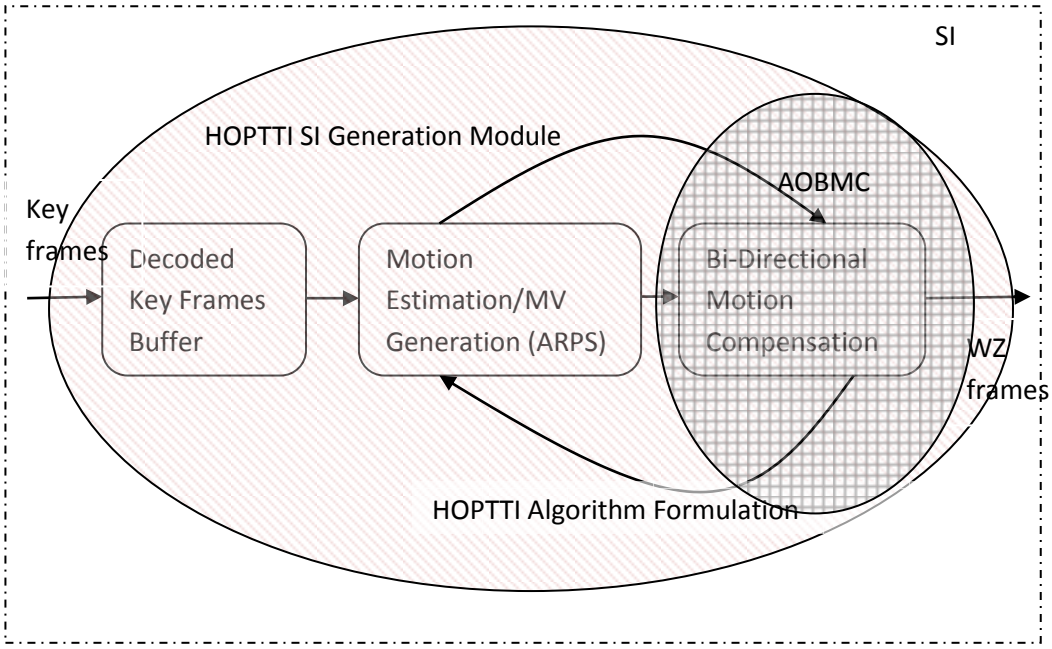


Figure 6.2 Detailed Blocks of SI Generation with AOBMC Module.

6.2 The AOBMC and Higher Order Piecewise Trajectory Interpolation

To deal with the issues relating to the use of BMA that introduces blocking artifacts and overlapping MVs, a higher order (cubic) trajectory model allied with AOBMC algorithm with a MS mechanism was used. Numbered bullet points (I. – V.) describing how the AOBMC algorithm fits into the existing HOPTTI algorithm (See Chapter 5 for HOPTTI details) and how AOBMC actually handles the issues of overlapping and blocking artifacts discussed in Section 6.1 is shown in Figure 6.6 and Figure 6.7.

6.2.1 The Artifacts From Block Matching Algorithm

While higher-order interpolation with BMA for MV estimation has been shown promising results (Petrazzuoli, Cagnazzo and Pesquet-Popescu 2010; Akinola, Dooley and Wong 2010), there are two issues to be resolved: i) MB overlapping caused by inaccurate MV estimations from the forward and backward trajectories; and ii) blocking artifacts caused by multiple or deformable objects having different motions in the same MB (Bosc et al. 2011; Liu et al. 2010).

These two scenarios are respectively illustrated in Figures 6.3 and 6.4. The overlapping scenario is illustrated in Figure 6.3, the solid blocks represents the actual MBs where the points that intercept in the intermediate frame emanate from different MBs and trajectories (blue trajectory, forward and red trajectory backward) in the previous frame and the next frame but intersect in the middle of the same block in the intermediate frame (green coloured solid block), thus showing how multiple trajectories passing through the intermediate frame can cause overlapping in the intermediate frame.

The blocking artifact scenario is illustrated in Figure 6.4 which shows a four pixel MB that has only one trajectory connecting the pixel, on the upper right hand pixel of the MB, both in the previous and future frames (marked as pixel 1). The dashed lines and dashed MB however shows an ideal case where the single dashed trajectory adequately representing all four pixels in the dashed MB as all the pixels belong to the same solid object (probably a background). The other case with the solid trajectory in Figure 6.4 with four differently coloured and numbered pixels presents another extreme case where all the four-pixel MB that actually contain the pixels belongs to four different objects and is traversed by only one trajectory so the intermediate frame can only correctly locate pixel 1 and all other three pixels are replaced by the closest match that can be located in the next frame, resulting in an artifact in the MB in the intermediate frame. In HOPTTI, because forward and backward trajectory MV are employed and compensation is usually the weighted

averaging of both forward and backward pixels, artifacts are more likely to be lighter patches and ghosting.

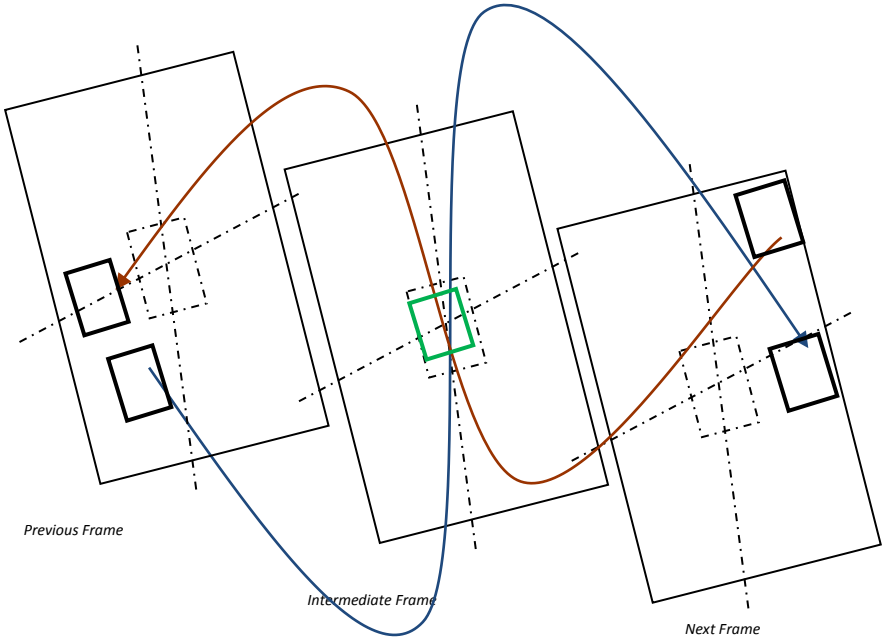


Figure 6.3 Example of multiple motion trajectories of a block passing through the intermediate frame which leads to overlapping.

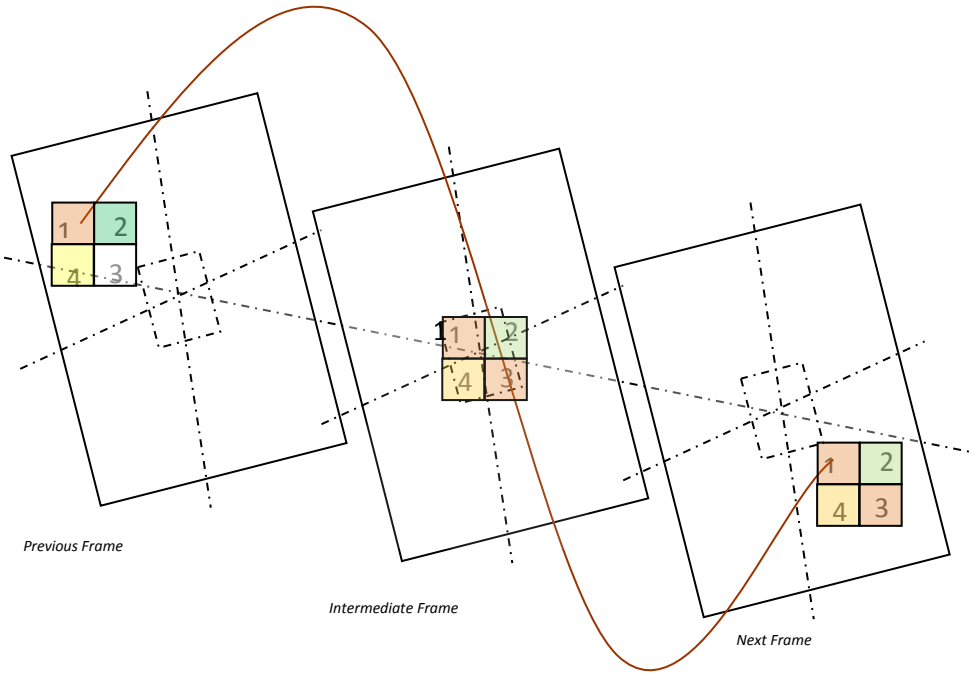


Figure 6.4 Example of a 4-pixel block with each having different motions but being represented by one trajectory and MV.

6.2.2 The AOBMC Algorithm

AOBMC is applied to each MB in the interpolated frame by modulating its MV with a set of surrounding pixels using a raised cosine weighting window which determines adaptively the amount by which the surrounding pixels modifies the boundary pixels in the MB under consideration and is illustrated in Figure 6.6 and Figure 6.7.

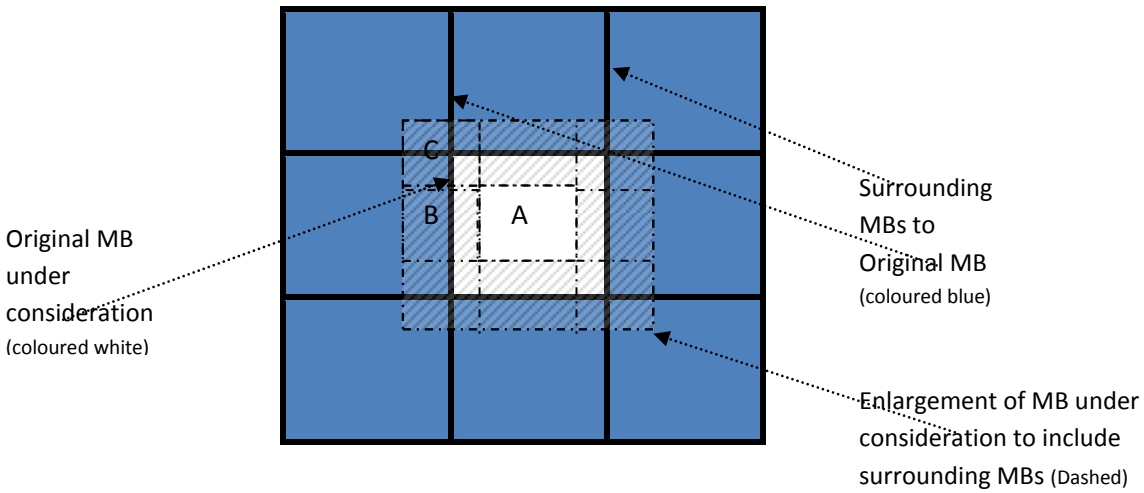


Figure 6.5 Illustration of sample overlapped blocks for AOBMC.

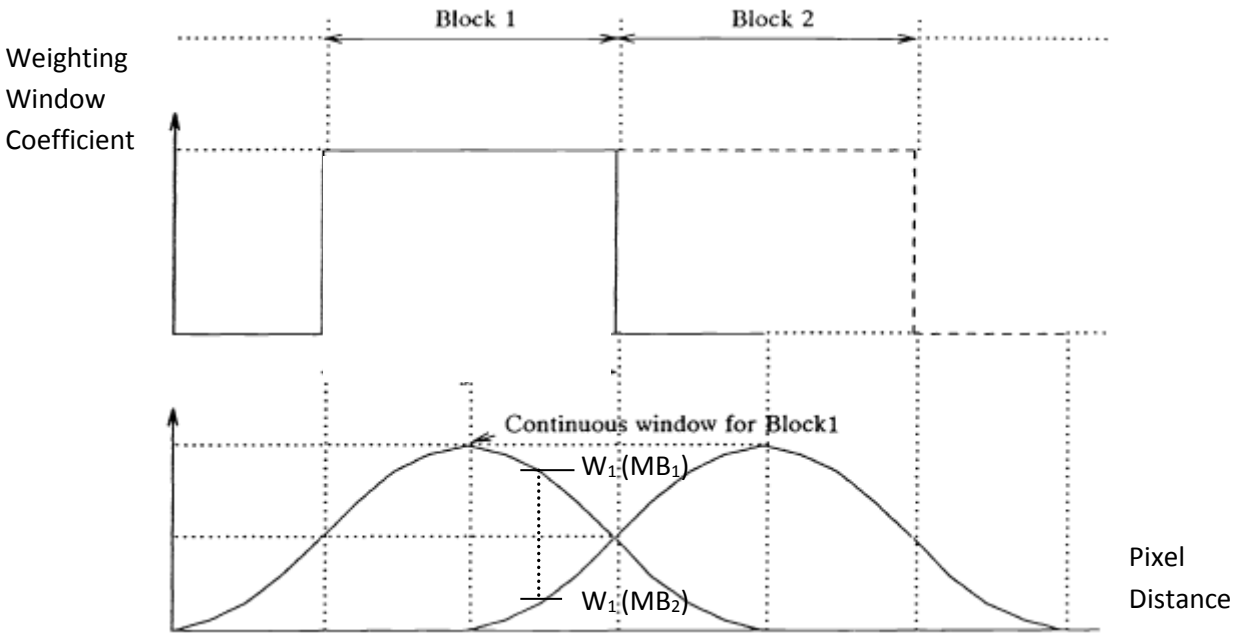


Figure 6.6 Illustration of the how raised cosine window is drawn for one block overlapping the other for OBMC.

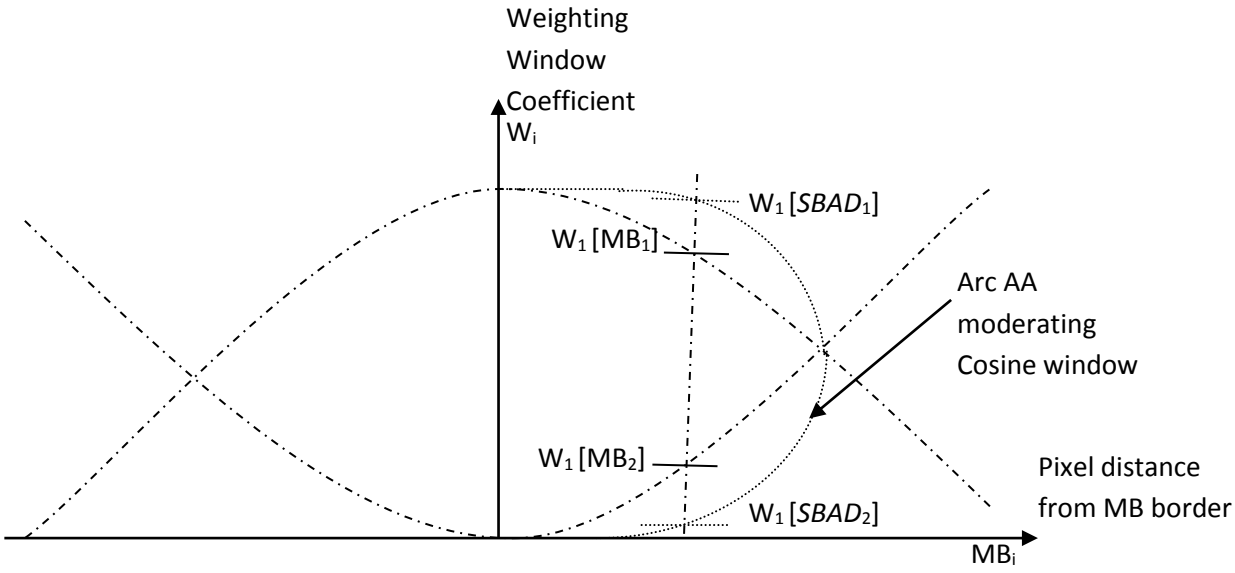


Figure 6.7 Illustration of the raised cosine window for one hypothetical supporting block in AOBMC.

The way that AOBMC works is as follows:

(i) As opposed to the conventional MC where the MV is applied to each block, in OBMC, the MV of a block under consideration is applied to a larger set of pixels using the raised cosine weighting window (Choi et al. 2007).

(ii) The MV of the block under consideration is used to predict the MV of every pixel in the block by moderating the MV of the block with the MV of the supporting blocks and the distance of the pixel from the supporting block.

(iii) The pixels and the region which they fall into depend on the distance from the border of the block and are usually determined by a smooth function which is the raised cosine weighting window, in this thesis (Orchard and Sullivan 1994).

(iv) The predicted MV is then further moderated by adapting it to the *sum of boundary absolute difference* (*SBAD)*, (defined in (6.1)), since in regular overlapped block motion compensation (OBMC) where the proximity of pixels only determine the weights, if neighbouring blocks belong to different fast moving objects or one belonging to

background and the other a moving object weights due to proximity will be high which can have an erroneous over bearing influence to the pixel MV prediction. Therefore, *SBAD* between neighbouring blocks is further employed to adapt weights and neighbouring blocks with high disparity in *SBAD* from the block under consideration would have their weight reduced. The prediction weight is therefore moderated adaptively by the *SBAD* of the surrounding blocks as well as proximity from block under consideration (Choi et al. 2007).

(v) The predicted MV for each pixel in the MB under consideration is then employed for MC for the entire block on a pixel by pixel basis.

The process of enlarging the blocks can be better illustrated with Figure 6.5 where solid lines represent the original blocks and dashed lines represent elastic lines covering enlarged blocks as well as areas inside the original block that benefits from overlapping, forming a variable border region (shaded) which enable the pixels in the border area inside the original blocks to overlap with neighboring blocks. The variable region in this illustration has been taken to be square for simplicity but can be any smooth function which in this thesis is the raised cosine function (Orchard and Sullivan 1994; Choi et al. 2007). Thus, the enlargement of the borders gives rise to three typical regions *A, B and C* that can be formed in and around the MB under consideration with the region A not overlapping with any neighboring blocks which implies the neighboring blocks are not likely to have any significant spatial relationship with the pixels outside the block, thus no action will be required to be undertaken. Typical region *B* overlaps with one neighboring block, thus the action that is taken is that part of the MB under consideration that falls in region *B* is supported by one overlapping supporting block, while a typical region *C* overlaps with three neighboring blocks, thus the action taken in this case is to support the part of the MB under consideration that falls under region *C* with three overlapping neighbouring blocks.

The size of the enlarged window (marked by the outer dashed line and labeled in Figure 6.5) is the raised cosine function weighting window in the implementation in this thesis as explained earlier. The corresponding weighting coefficient for predicting the MV for each pixel in the MB under consideration is proportional to the pixel distance from the boundary of the block under consideration which is moderated by the reliability of the neighboring MV determined by adapting the weights to a grading of the video content parameter illustrated in Figure 6.5. The video content parameter employed is the *sum of boundary absolute difference* (*SBAD*) (Choi et al. 2007). *SBAD* is defined as:

$$\begin{aligned} SBAD &= \frac{1}{M}\sum_{i=0}^{M-1}|F_1(i+x_0, y_0) - F_2(i+x_0+MV_x, y_0+MV_y-1)| \\ &+ \frac{1}{M}\sum_{i=0}^{M-1}|F_1(i+x_0, y_0+N-1) - F_2(i+x_0+MV_x, y_0+MV_y+N)| \\ &+ \frac{1}{N}\sum_{j=0}^{N-1}|F_1(x_0, j+y_0) - F_2(x_0+MV_x-1, j+y_0+MV_y)| \\ &+ \frac{1}{N}\sum_{j=0}^{N-1}|F_1(x_0+M-1, j+y_0) - F_2(x_0+MV_x+M, j+y_0+MV_y)| \end{aligned} \quad (6.1)$$

Where (x_0, y_0) is the coordinate of the top left corner of the MB under consideration in the interpolated frame F_1, F_2 is the HOPTTI SI frame used as the reference since the original frame is unavailable at the decoder (MV_x, MV_y) is candidate MV and (M, N) are the row and column dimensions of the MB.

From the foregoing, continuing with the illustration of Figure 6.5, a typical region *C* from Figure 6.5 with three neighboring blocks adjoining the original block under consideration will not have all three blocks weighted the same in their support of the predicted MV of the pixels in the block under consideration, but will be weighted by W_i, a sum of weights comprising the weighted coefficients of the cosine window proportional to the distance from the border of the block for each supporting overlapped block, then adapted by the weight of grading the video content parameter *SBAD*, an iterative process that uses (6.1) for grading the reliability of the neighboring blocks. Likewise for the region *B* in Figure

6.5, with one adjoining (overlapped) block to the block under consideration, the MV will be moderated by that one block but the weight W_i would also be determined by the cosine window and the reliability adapted by the *SBAD* grading. Finally, for region A, the predicted MV will be employed without moderation as it has no adjoining (overlapped) neighbouring block.

Figure 6.6 shows how the raised cosine window is drawn for two blocks adjacent to each other. For Block 1, the continuous cosine window starts at zero reaching a maximum at the middle of the block, while the window for Block 2, is a minimum at the middle of Block1 and a maximum at the middle of Block 2. Thus each pixel point in Block 1 is associated with two weights moderating its MV, W_1 (MB_1) and W_1 (MB_2). Each pixel is predicted by the weighted sum of the window coefficients $W_i = W_1 (MB_1) + W_1 (MB_2)$ where $W_1 (MB_1)$ is the weight associated with the distance of the pixel in the window of Block 1 which is the MB under consideration and $W_1 (MB_2)$ weight is for the overlapping block.

Similarly, Figure 6.7 illustrates the raised cosine weighting window for a region in a block under consideration that has one surrounding block to support the prediction of the MV for each pixel in that region of block under consideration but introducing weights for adaptability by applying *SBAD*. This gives a disparity measure between blocks such that if, for example, the neighbouring block 2 in Figure 6.6 is has a high *SBAD* measure, the arc AA in Figure 6.7 moderates the cosine window, increasing $W_1(MB_1)$ due to the block under consideration by $W_1(SBAD_1)$ while reducing $W_1(MB_2)$ by $W_1(SBAD_2)$. The weight W_1 ($SBAD_1$) is due to the video content parameter *SBAD* on the cosine window of block under consideration and W_1 ($SBAD_2$) is due to the surrounding block. They moderate the final weights W_i, thus giving the final $W_i = [W_1 (MB_1) \pm W_1 (SBAD_1)] + [W_1 (MB_2) \pm W_1 (SBAD_2)]$. The illustration is equivalent to that for the region *B* from Figure 6.5 having one supporting block.

The illustration can be extended for the region *C* from Figure 6.5 also, the amount of weight $W_{i,}$ attributed to each of the three surrounding blocks would be $W_i = [W_1 (MB_1) \pm W_1 (SBAD_1)] + [W_1 (MB_2) \pm W_1 (SBAD_2)] + [W_1 (MB_3) \pm W_1 (SBAD_3)] + [W_1 (MB_4) \pm W_1 (SBAD_4)]$ where, $W_1 (MB_1)$, $W_1 (MB_2)$, $W_1 (MB_3)$, $W_1 (MB_4)$ are the weights due to the coefficients cosine window (proportional to distance) for the block under consideration and for each of the overlapping blocks, and $W_1 [SBAD_1]$, $W_1 [SBAD_2]$, $W_1 [SBAD_3]$ $W_1 [SBAD_4]$ are the weight adapted by the grading of the spatial video content parameter, *SBAD* for the block under consideration and each of the overlapping blocks.

The AOBMC algorithm described is incorporated into the HOPTTI based DVC codec as illustrated in Figure 6.2. The details of the integration of AOBMC into the HOPTTI algorithm are as follows:

(i) ARPS in the HOPTTI algorithm is employed for block based MV estimation (See the ARPS algorithm and advantages in Chapter 5) for forward path and backward path following closely the higher order trajectory.

(ii) AOBMC utilizes the MV of each block to predict MVs for each pixel of the block under consideration, moderating the MV by the raised cosine window and the video content parameter adaptively.

(iii) AOBMC based forward and backward intermediate frames are thus generated.

(iv) The AOBMC based forward and backward frames are combined together as in HOPTTI using the weighted average, $MC_F (1-\zeta) + MC_B (\zeta)$ as shown in Figure 5.8 in Chapter 5.

(v) The output MCI frame is then further utilized in the Mode Switching algorithm described in the next section 6.2.3 such that the most improved frame generated by both HOPTTI and AOBMC based algorithms is selected as SI.

6.2.3 The Mode Switching Algorithm

While AOBMC reduces errors caused by the scenarios discussed in Section 6.2.1, the results produced by HOPTTI with AOBMC reveal that the aggregate sum of using the spatial correlation of pixels around a MB in AOBMC results in specific frames becoming degraded. The reason for degradation was further investigated in Akinola, Dooley and Wong (2010) showing that not all frames of the sequence nor all blocks in the frame benefit equally from the addition of AOBMC, as the neighboring pixels to the MB under consideration used in AOBMC algorithm due to different motion and object content give conflicting information so the modification of the MV using them produces erroneous results giving rise to situations where original HOPTTI outperforms HOPTTI-AOBMC output. The spatial-temporal pixel correlations surrounding a MB are exploited to determine those frames most likely to exhibit this tendency and this formed the basis of the Mode Switching (MS) facility.

The aim is to define a *matching criterion* (*M*) that determines the level of the spatio-temporal correlation and a threshold *T* that separates aggregate contributions to all the MBs in a frame. When *M* is below *T*, it gives negative aggregate contribution and as a result AOBMC-H is disallowed, and the MB from HOPTTI is switched in to replace it. In contrast, when *M* is above *T*, it gives positive aggregate contribution and AOBMC-H is allowed. Therefore, in addition to the spatial measure *SBAD* that measures spatial continuity of MVs from the MB under investigation, the *sum of mean absolute difference* (SMAD) which measures temporal continuity by employing the future frame F_3 is included in the matching criterion following Ye at al. (2009). SMAD is defined as:

$$SMAD = \left[\frac{1}{MN} \sum_{i=0}^{M-1} \sum_{j}^{N-1} |F_1(i+x_0, j+y_0) - F_2(i+x_0+MV_x, j+y_0+MV_y)| \right] + \left[-\frac{1}{MN} \sum_{i=0}^{M-1} \sum_{j}^{N-1} |F_3(i+x_0, j+y_0) - F_2(i+x_0+MV_x, j+y_0+MV_y)| \right] \quad (6.2)$$

where the various parameters are defined as in (5.1), F_3 is the future frame and F_2 the reference frame as earlier. A weighted sum of *SMAD* and *SBAD* is then used to form the matching criterion such that:

$$M = \lambda * SBAD + (1-\lambda) * SMAD \qquad (6.3)$$

where λ is a predefined weighting factor. In (6.3) the spatial-temporal continuity of *SBAD* and *SMAD* are exploited as the measure to match the surrounding blocks with the reference MB. The MS mechanism then applies a threshold *T* so:

Calculate *M* in (6.3)

If $M \leq T$ THEN apply HOPTTI with AOBMC

ELSE use HOPTTI only

END

The performance of this Switched HOPTTI-AOBMC DVC codec in improving SI quality will be shown and analyzed in section 6.3.

6.3 Simulation and Results

The same HOPTTI algorithm implemented in Matlab version 7.5.0 (R2007b) running under Microsoft Windows XP on a PC with an Intel Duo Core CPU at 2.20 GHz is employed and modified to include the AOBMC module shown in Figure 6.2. Similarly, a GOP size of 2 was chosen for all experiments i.e. *KWKW,* where *K* and *W* denote key and WZ frames respectively. HOPTTI (Akinola, Dooley and Wong 2010) used a cubic trajectory and parameterization as outlined in Section 5.31 of Chapter 5. To evaluate both quantitatively and qualitatively, HOPTTI with and without AOBMC, various QCIF (Quarter Common Intermediate Format) test sequences were applied including *Carphone*, *Table Tennis*,

Mother, Coastguard, Silent, Hall, Mobile, Soccer, Foreman and *American Football,* which provided a wide range of different types of motion and objects features. Where average results are presented, they are the result of averaging the result for the entire frames in the various video sequences. Both λ in (6.3) and the threshold *T* in the MS mechanism were determined empirically and set to $\lambda = 0.4$ and *T*=10 which provided the best results for a number of slow, medium and high object motion sequences tested with Table 6.1 showing the experimental results.

Table 6.1 EMPIRICAL EXPERIMENT SHOWING AVERAGE PSNR (dB) VERSUS WEIGHT(λ) OF *SWITCHED HOPTTI-AOBMC* FOR SELECTED TEST SEQUENCES

T=10 (Fixed)						
American Football						
Weight (λ)	0.0	0.2	0.4	0.6	0.8	1.0
PSNR (dB)	25.65	25.71	25.80	25.68	25.63	25.59
Coastguard						
Weight (λ)	0.0	0.2	0.4	0.6	0.8	1.0
PSNR (dB)	37.20	37.60	37.90	37.55	37.10	35.65
Hall						
Weight (λ)	0.0	0.2	0.4	0.6	0.8	1.0
PSNR (dB)	39.00	39.50	39.90	39.35	38.90	38.70

In Table 6.1, while the threshold and weight provides a good range for empirical experimentation for *Coastguard* (2.25 dB) and *Hall* (1.20 dB). This is not the same for *American Football* where the range of PSNR values for the weights only gives a range of

0.31 dB. This is due to the fact that the spatial-temporal characteristics of the *American Football* is significantly different from the *Coastguard* and *Hall* Sequences which shows that empirically fixing thresholds and weights may provide significant improvements for sequences whose characteristics are in the middle of the range (single object, medium motion), it is inadequate for outlier sequences (multiple objects, high motion) which may require intelligent setting of the threshold and weights.

Furthermore, careful observation of the *American Football* sequence range of PSNR results over the weighted range for T=10 shows that the PSNR is only increasing whereas that of the *Coastguard* and *Hall* sequences went up and then down showing a point of maxima which the averaging out of spatial-temporal characteristics does not allow for the *American Football* sequence. This again should be mitigated if the thresholds and weights are fixed adaptively by intelligently applying the spatial temporal characteristics which is explored in Chapter 7 of this thesis.

6.3.1 Experimentation to Determine Weight (λ) Empirically

First we perform experiments to determine empirically the best value for the weight (λ). Thus, Table 6.1 shows that for fast to medium object motion sequences with single and multiple objects, the empirical setting applied in this thesis, using T=10 and λ=0.4 gives the best possible setting as is shown for *Hall*, *Coastguard* and *American Football* sequences.

Three other AOBMC-based implementations which all attempt to address the restrictions caused by BMA by using LMCTI in SI generation were used as SI quality performance comparators. They are:

(i) *motion compensated frame interpolation and adaptive object block motion compensation* (MCFIAOBMC), (Choi et al. 2007) where bilateral LMCTI is applied to

overcome both hole and overlapping problems by coupling AOBMC with an object segmentation and MV clustering technique.

(ii) ISIG-DVC, (Huang and Forchhammer 2008) where AOBMC is combined with a variable block-size refinement algorithm to produce improved SI.

(iii) *Low complexity motion compensated frame interpolation* (ALCFI), (Zhai, Yu and Li 2005) that utilizes AOBMC together with MV smoothing.

6.3.2 SI Generation Simulation Results

Tables 6.2, 6.3, 6.4 and 6.5 show the corresponding PSNR values for various test sequences, with the aforementioned AOBMC based, LMCTI centered algorithms compared to the Switched HOPTTI AOBMC algorithm introduced in this Chapter.

Table 6.2 AVERAGE PSNR (dB) FOR *SWITCHED HOPTTI-AOBMC* and HOPTII FOR DIFFERENT TEST SEQUENCES

Sequences	***Switched HOPTTI-AOBMC***	***HOPTTI (Akinola, Dooley and Wong 2010)***
Carphone	36.2	35.3
Mother	48.4	47.3
Foreman	36.7	35.1
Silent	39.9	38.9
Coastguard	37.9	36.4
Hall	39.9	38.5
American Football	25.8	24.5

Table 6.3 AVERAGE PSNR (dB) FOR *SWITCHED HOPTTI-AOBMC* and MCFI-AOBMC FOR DIFFERENT TEST SEQUENCES

Sequences	***Switched HOPTTI-AOBMC***	***MCFI-AOBMC algorithm (Choi et al. 2007)***
TableTennis	48.4	32.0
Foreman	36.7	36.0
Mobile	35.5	25.2
American Football	25.8	24.0

Table 6.4 AVERAGE PSNR (dB) FOR *SWITCHED HOPTTI-AOBMC* and ISIG-DVC FOR DIFFERENT TEST SEQUENCES

Sequences	***Switched HOPTTI-AOBMC***	***ISIG-DVC algorithm (Huang and Forchhammer 2008)***
Foreman	36.7	29.3
Coastguard	37.9	31.8
Hall	39.9	36.5
Soccer	29.6	21.3

The results reveal that *Switched HOPTTI-AOBMC* consistently provided an SI improvement for each sequence analyzed, with *Foreman* for instance giving a 1.6dB PSNR improvement over both the original HOPTTI and various AOBMC-based results. Though the PSNR results from the Table seem to be high compared to other SI results, this can be directly attributed to the HOPTTI algorithm which is the basis of the experimentation that has been shown to give more accurate results in Chapter 5 of this thesis and in Akinola,

Dooley and Wong (2010, 2011). The actual average PSNR improvement due to AOBMC and empirical mode switching is about 1.6 – 2.0 dB.

Table 6.5 AVERAGE PSNR (dB) FOR *SWITCHED HOPTTI-AOBMC* and ALCFI FOR DIFFERENT TEST SEQUENCES

Sequences	***Switched HOPTTI-AOBMC***	***ALCFI algorithms (Zhai, Yu and Li 2005)***
Carphone	36.2	33.6
Mother	48.4	38.0
Foreman	36.7	34.7
Coastguard	37.9	34.0
Mobile	35.5	31.4
Clair	43.6	41.7

From a perceptual perspective, the sample frames from *Hall* and *American Football* shown in Figures 6.8 and 6.9 reveal how the inclusion of HOPTTI into the AOBMC algorithm and applying the MS mechanism qualitatively improved SI quality. These qualitative judgments are numerically confirmed in the average PSNR values in Table 6.2, with improvements of 3.4dB and 1.4dB respectively over the other AOBMC variants. For *Hall*, the improvement is readily apparent in the extended leg of the moving object (man), while

Frame #23

Original

SI for HOPTTI

PSNR=36.6dB

SI for Empirically *Switched HOPTTI-AOBMC*

PSNR=38.3dB

Frame #49

Original

SI for HOPTTI

PSNR=37.8dB

SI for Empirically *Switched HOPTTI-AOBMC*

PSNR=39.2dB

Frame #51

Original

SI for HOPTTI

PSNR=38.6dB

SI for Empirically *Switched HOPTTI-AOBMC*

PSNR=39.6dB

Frame #85

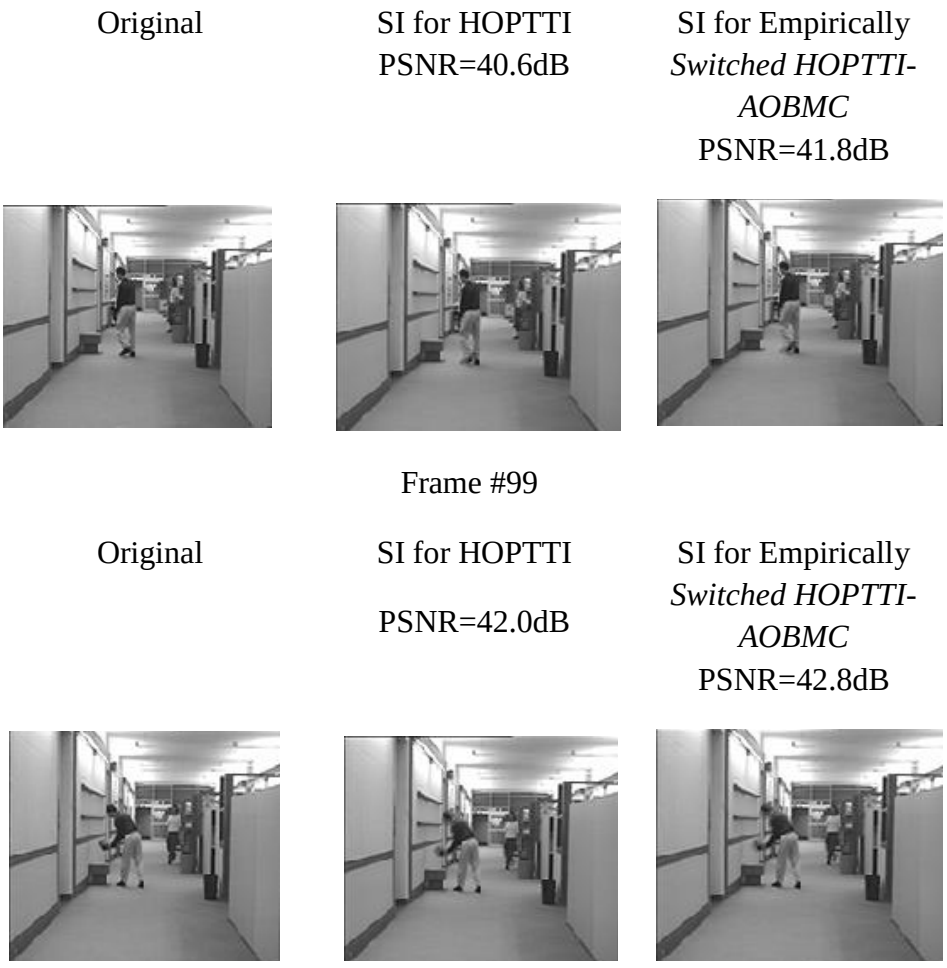


Figure 6.8 Sample frames for *Hall* showing the SI quality obtained using HOPTTI (Akinola, Dooley and Wong 2010) and *Switched HOPTTI-AOBMC*.

in *American Football*, object ghosting visible in the HOPTTI frame has been significantly attenuated in the Empirically *Switched HOPTTI-AOBMC* frame. The corresponding frame-wise plots corroborate the role of MS in ensuring that no frame for any sequence analysed had the PSNR value for *Switched HOPTTI-AOBMC* lower than HOPTTI, i.e. HOPTTI provided a lower performance bound in terms of SI quality.

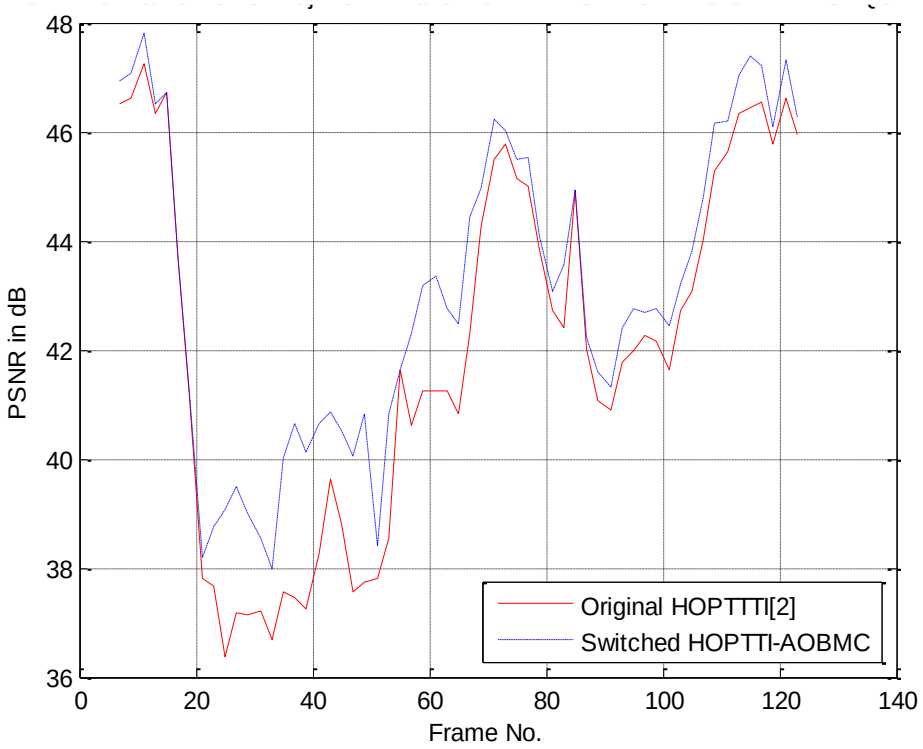


Figure 6.9 Frame-wise SI-quality of HOPTTI (Akinola, Dooley and Wong 2010) and Switched HOPTTI-AOBMC for *Hall*

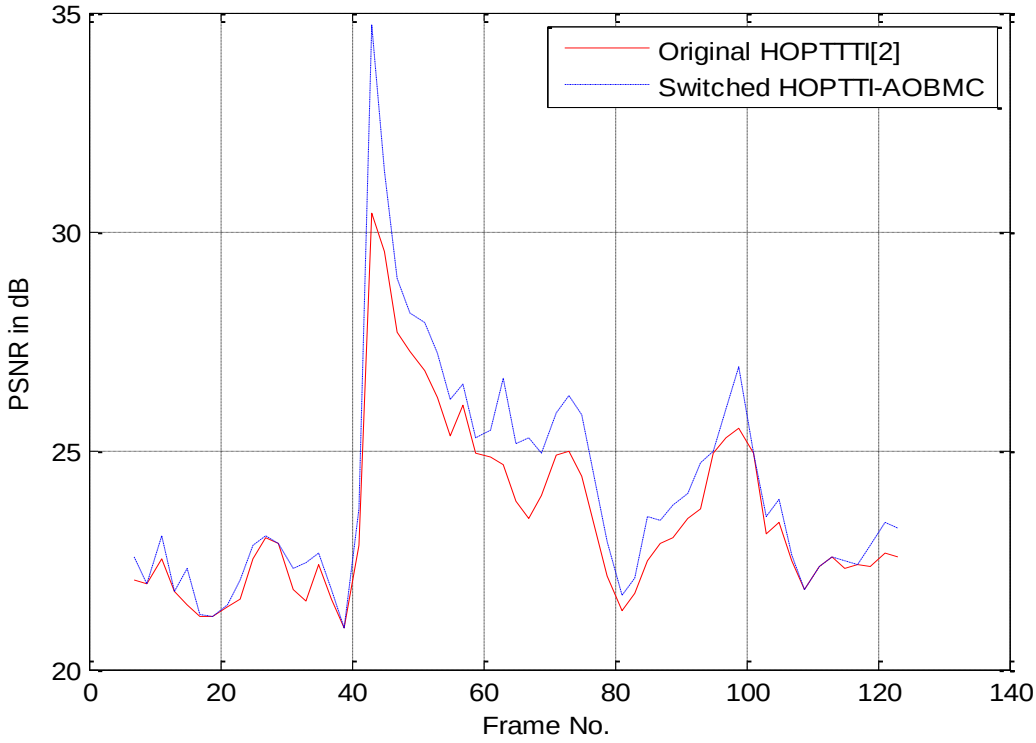


Figure 6.10 Frame-wise SI-quality of HOPTTI (Akinola, Dooley and Wong 2010) and Switched HOPTTI-AOBMC for *American Football*

Frame #61

Original	SI for Original HOPTTI PSNR=24.6dB	SI for Empirically *Switched HOPTTI-AOBMC* PSNR=26.0dB

Frame #93

Original	SI for Original HOPTTI PSNR=24.5dB	SI for Empirically *Switched HOPTTI-AOBMC* PSNR=26.0dB

Frame #99

Original	SI for Original HOPTTI PSNR=24.8dB	SI for Empirically *Switched HOPTTI-AOBMC* PSNR=26.2dB

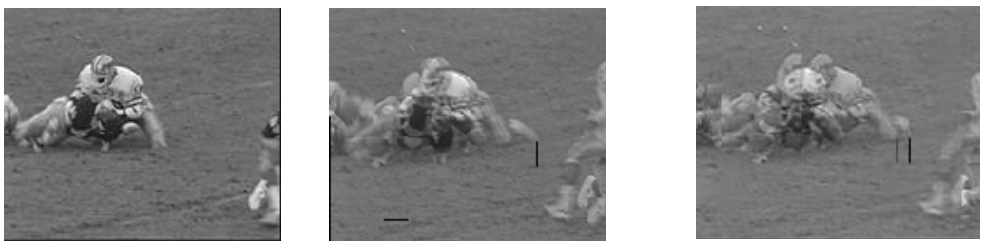


Figure 6.11 Sample frames for *American Football* showing the SI quality obtained using HOPTTI (Akinola, Dooley and Wong 2010) and *Switched HOPTTI-AOBMC*.

6.3.3 Computational Complexity and Improvement in Qualitative Results on Challenges to HOPTTI Algorithm

The qualitative and quantitative results presented in Section 6.3.2 shows that AOBMC combined with the more accurate HOPTTI formulation gives an improved SI compared to LMCTI and HOPTTI only. However, there is a computational cost that the HOPTTI algorithm adds which is shown by the additional time required for the HOPTTI-AOBMC algorithm to execute in the Table 6.6. The Table 6.6 shows that the AOBMC algorithm adds a significant amount of computational time for the performance improvement that it brings as it adds at least 18.1% time to the execution time. The time however is about the same per frame for all the video sequences tested which shows that the computational complexity will depend on the number of frames in the input sequence. Employing the defined complexity variables in Table 4.1, HOPTTI-AOBMC complexity is $T_{Const} + 4T_{Vel} + 2T_{Accl} + T_{Jolt} + T_{BMA} + T_{AOBMC}$ and it can be observed that the added complexity compared to the HOPTTI algorithm is from the AOBMC algorithm whose overhead between different video sequences only varies slightly, corroborating the direct time measurements of Table 6.6.

Table 6.6 AVERAGE SI GENERATION TIME PER FRAME IN SECONDS

Sequences	***HOPTTI***	***HOPTTI-AOBMC***	***Difference***	***%change***
Carphone	0.07	0.095	0.025	26.3
Coastguard	0.10	0.125	0.025	25.0
Foreman	0.14	0.166	0.026	18.6
Mother	0.08	0.105	0.025	31.3
Hall	0.11	0.136	0.026	23.6
Silent	0.06	0.085	0.025	41.7

Frame #51

Original

SI for all blocks on Frame HOPTTI PSNR=38.6dB

SI for empirically Switched MB in Frame PSNR=39.6dB

Frame #85

Original

SI for all blocks on Frame HOPTTI PSNR=40.6dB

SI for empirically Switched MB in Frame PSNR=41.8dB

Frame #99

Original

SI for all blocks on Frame HOPTTI PSNR=42.0dB

SI for empirically Switched MB in Frame PSNR=42.8dB

Figure 6.12 HOPTTI, Improved by Empirical MS in HOPTTI-AOBMC for *Hall*

The frames presented in Chapter 5 to show the challenges to HOPTTI algorithm are hereby represented to show the improvement that AOBMC and the MS mechanism has achieved and how far the artifacts have been mitigated.

First the frames from the *Hall* sequence are presented and show clearly the ghosting around the feet in frame #51.

Next the challenging frame from the *Coastguard* sequence is presented which shows clearly that the ghosting in the background have been removed.

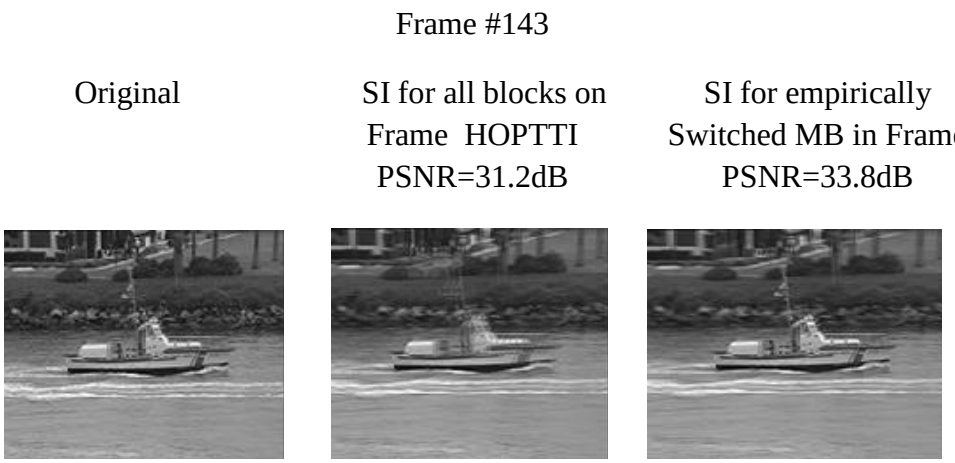


Figure 6.13 Challenging Artifacts in HOPTTI, Improved by Empirical MS in HOPTTI-AOBMC for *Coastguard*

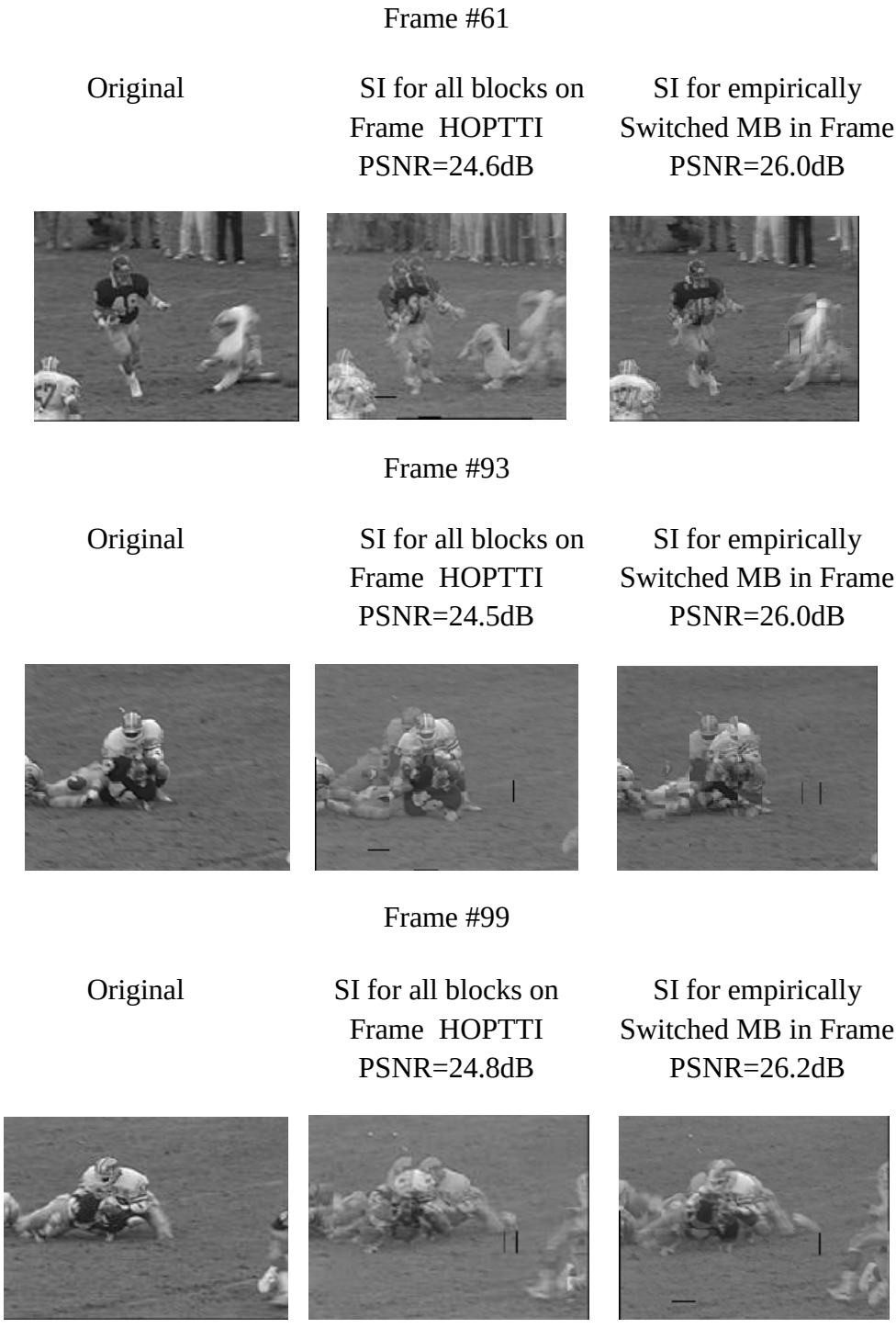


Figure 6.14 Challenging Artifacts in HOPTTI, Improved by Empirical MS in HOPTTI-AOBMC for *American Football*

Lastly, the challenging frames for the HOPTTI algorithm for America Football sequence are presented. This shows significant improvement in the frames after the AOBMC algorithm is introduced.

Rate Distortion Performance of Empirical MS results:

The overall RD results for *Foreman* and *Hall* sequence presented to accommodate both a complex single object sequence and a multiple object sequence are shown in Figure 6.15 and Figure 6.16 respectively. The result is that empirical based MS outperforms HOPTTI, H.264 No Motion and H.264 intra while the H.264 inter remains the upper limit that outperforms Empirical MS in *Hall*, while H.264 inter outperforms Switched RST by up to 2 dB in the *Foreman* sequence.

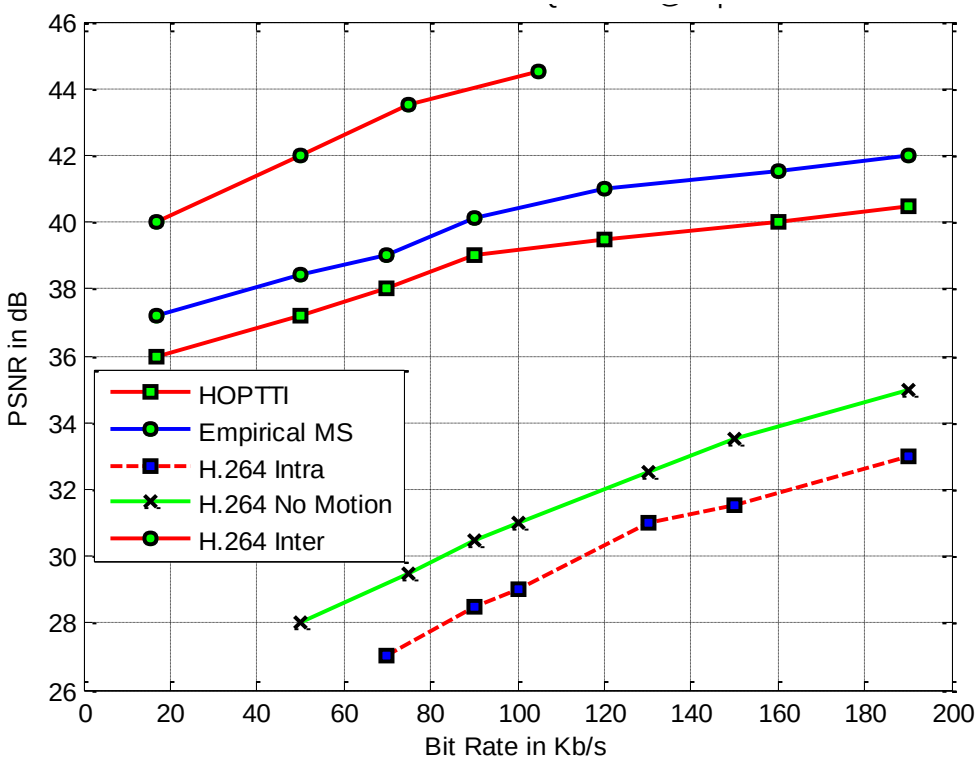


Figure 6.15 RD Curves showing HOPTTI PSNR performance in codec based on (Li, 2008) for *Foreman* sequence @ 15f/s

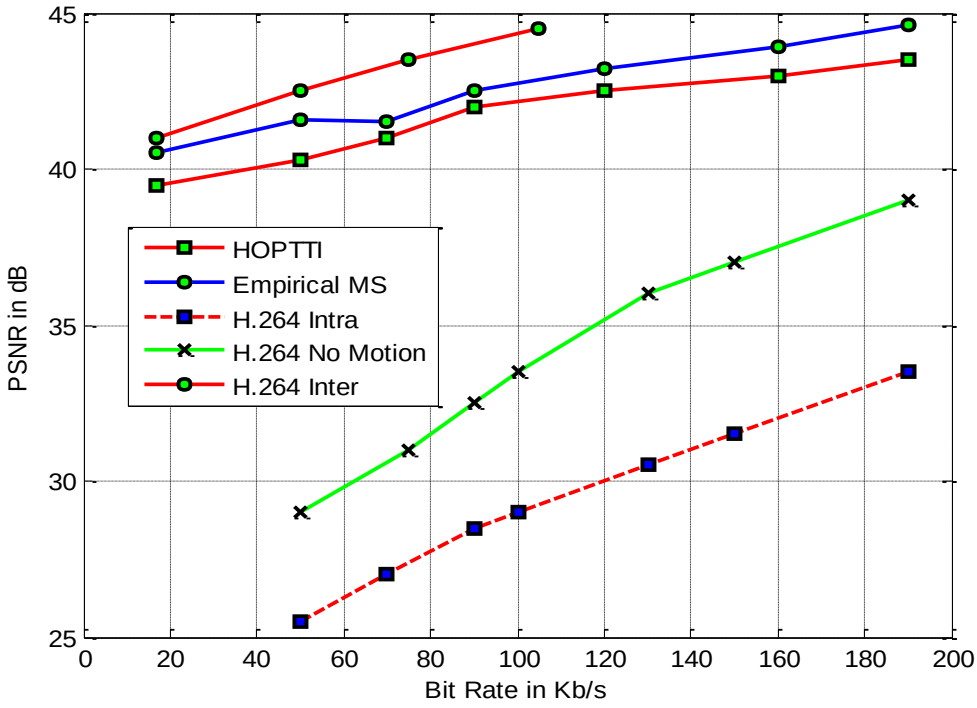


Figure 6.16 RD Curves showing HOPTTI PSNR performance in codec based on (Li 2008) for *Hall* sequence @ 15f/s

6.4 Summary

This Chapter tackled the twin problems of overlapping and blocking artifacts in *higher order piecewise temporal trajectory interpolation* (HOPTTI) due to the use of BMA by selectively incorporating it into the AOBMC algorithm, and using a mode switching mechanism to generate the *Switched HOPTT-OABMC* SI. Both numerical and perceptual results confirm the SI quality improvement in applying the HOPTTI and AOBMC combination, with up to 3.6dB improvement in PSNR achieved. From frame based empirical results (Akinola, Dooley and Wong 2010) presented in this chapter, it is evident that improved SI quality can be achieved if the more appropriate algorithm is selected on a MB basis aided by video content parameters. This forms the basis of further improvement of the SI in the next Chapter to intelligently maximize the threshold and parameter settings, and to do this automatically at MB level as it is cumbersome to undertake empirically.

Chapter 7

Improved SI Generation Using Rough Set Theory

7.1 Introduction

The main aim of this thesis is to tackle a major DVC bottleneck that impacts on its output quality which is a coarse approximation of the original video frames that are not available at the decoder. This Chapter concentrates on BLOCK 4 of the framework from Figure 1.3 which is reproduced in Figure 7.1 for convenience with BLOCK 4 highlighted. It addresses the need for RST based intelligent mode switching (IMS) at MB level and the use of video content to maximize the parameters by employing RST. In Chapter 6, it was shown that as the spatial-temporal characteristics of the video changes, it is necessary to switch the threshold settings and the mode of the algorithm being used to generate the SI. Furthermore, empirical experiments in Section 6.3 showed that MS yields improvement to the SI thus generated. This Chapter tackles the aspect of *SI Generation and Improvement Framework* of Figure 1.2, that has been identified in literature in Chapter 3, Section 3.3.6 that the changing spatial-temporal characteristics in video sequences and spatial-temporal differences between different video sequences is the reason why different algorithms perform differently or even fail (Ascenso et al. 2005; Weerakkody et al. 2007; Martin et al 2009; Martin et al 2010). Thus, RST an artificial intelligence (AI) algorithm first discussed

in Chapter 3 is introduced into MS, employing the spatial-temporal characteristics of video sequences.

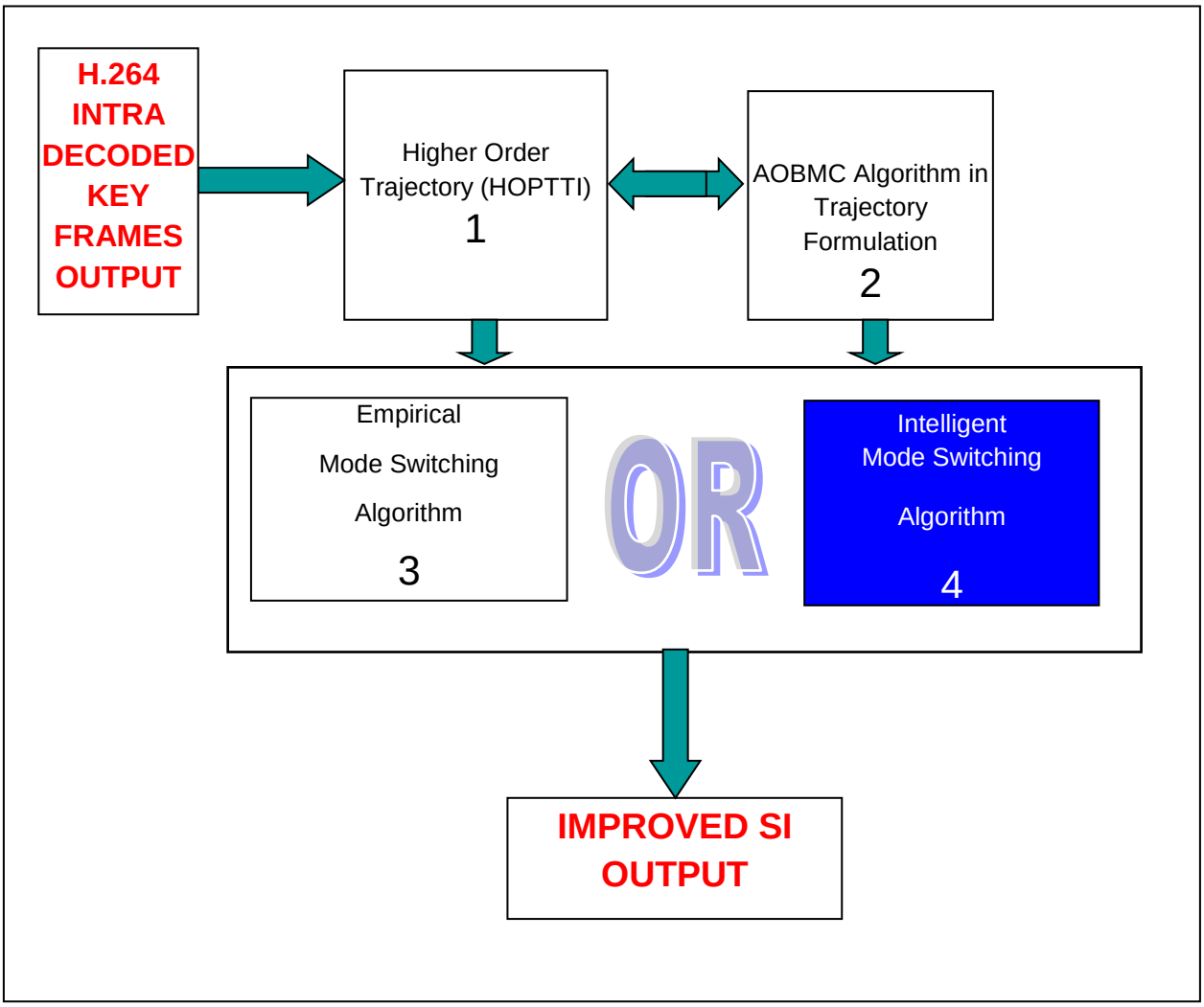


Figure 7.1 Block Diagram of *SI Generation and Improvement Framework* with BLOCK 4 Highlighted.

In Chapter 6, AOBMC was employed in HOPTTI and an empirical approach to setting thresholds was used to show that Switched HOPTTI-AOBMC performed better than a number of variants where LMCTI was combined with AOBMC to improve SI quality. While the addition of AOBMC to HOPTTI and empirical MS showed improved SI, investigations reveal that algorithm performance fluctuate within the same video sequence and from one video sequence to the other resulting in subsequent analysis in Section 6.2.3 of Chapter 6 that for certain frames in the various test sequences investigated HOPTTI produced better SI quality than when combined with AOBMC. The introduction of AI will

determine when to switch between HOPTTI algorithm and the AOBMC-H algorithm which results from the following summaries:

(i) The spatial and temporal characteristics of the sequences, for example how spatial contrast between different objects and that between objects and the background both intra-frame and inter-frames change in video sequences.

(ii) The thresholds are constant for every part of a given sequence without due regard to spatial-temporal variations. For example, Table 6.1 T=10 while the weight is varied and eventually set at λ=0.4.

(iii) The complexity of the fluctuations of the spatial-temporal characteristics makes it difficult to make one rule fits all setting of thresholds as the setting of one rule dynamically affects others already set.

(iv) Thresholds are subsequently approximated and depend on other approximated thresholds earlier set leading to accumulation of errors. Thus, the knowledge of how the setting of one threshold affects others is not explicitly known, learnt or fed back to adjust previously set ones.

To evaluate the use of spatial-temporal characteristics of video sequences in Chapter 6, a *mode switching* (MS) technique based on (Ye et al. 2009) was introduced, showing that an improved SI can be generated using the spatial-temporal characteristics SMAD and *SBAD*. MS uses a matching criterion to switch between SI generation using HOPTTI only, and that using the AOBMC algorithm combined with HOPTTI (AOBMC-H) to obtain a final SI named the *Switched HOPTTI-AOBMC*. The corresponding impact on SI quality of both the new AOBMC-H approach and MS mechanism which introduces an explicit rule (if, else) and empirically determines a cut off threshold for the rule was shown to exhibit a consistent improvement in overall SI quality with numerical and perceptual results. This Chapter takes a further step to improve SI by introducing RST based IMS as shown in Figure 7.2 which

gives the detailed block diagram of the SI generation scheme using RST, with the RST mode switching block highlighted.

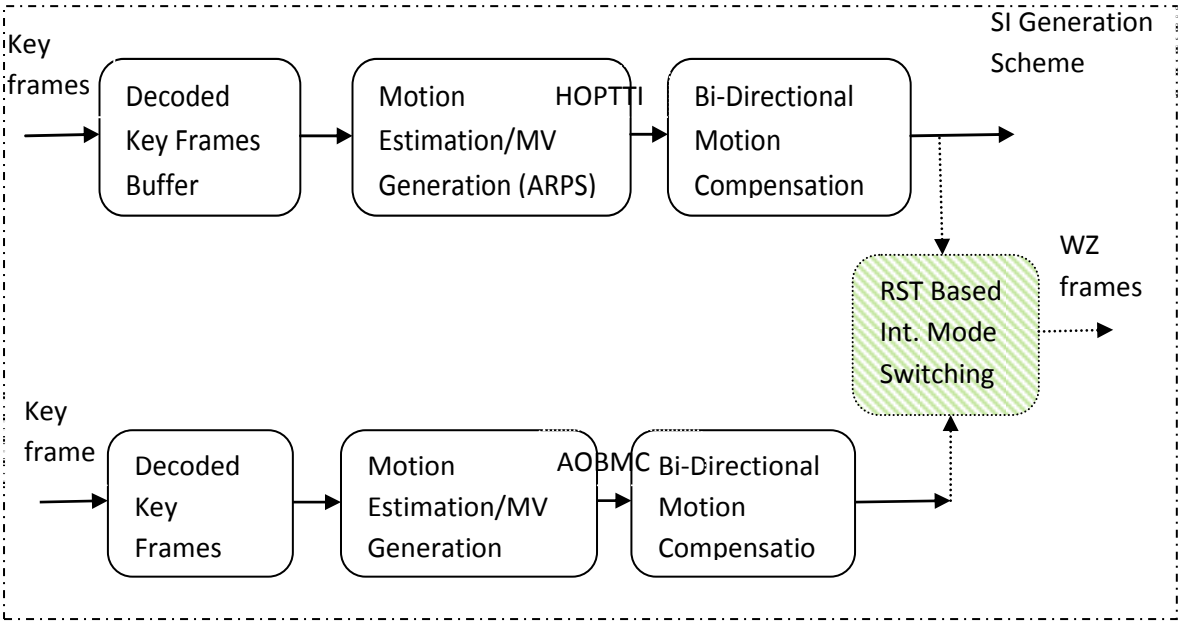


Figure 7.2 Detailed Blocks of SI Generation with RST Based IMS Module

As emphasized in Chapter 6, coupled with the four points on the nature and spatial-temporal characteristics of video sequences raised above, a more intelligent way of setting weights and thresholds is likely to further improve SI since AI methods are knowledge based and non-linear, taking into consideration various complex features and attributes of the video sequences to make essential rules to handle efficiently the fluctuations and as such can improve quantitative and qualitative DVC performance.

In Chapter 3, the various AI based methods of handling the setting of weights and thresholds were discussed including SVM, neural networks, FL, genetic algorithm, and RST. An intelligent method that classifies data features for correct decision making is the Gaussian Mixture Model but the attributes and features in videos may not necessarily be based on Gaussian distributions when the data is non-Gaussian, as stated in (Verbeek, Vlassis and Krose 2003). Since an intelligent method is required to select between HOPTTI and AOBMC-H to give a better result considering the spatial-temporal features of video

sequences with a wide variety of sequences, a method that assumes a specific probability model is not suitable.

Another versatile AI classifier is the Support Vector Machines (SVM) which works particularly well with small data size, able to deal with non-linear problems with high dimensions. However, SVM requires the construction of a classification function which matches the input space to the non-linear transform, referred to as the kernel function which includes a polynomial function, a radial basis function and a multi-layered perception function. That is usually a complex exercise whose validity or otherwise affects the accuracy of the classification.

While Neural Networks (NN) can be utilized for classification of data and have been widely used, they have a lot of drawbacks which include slow learning rate, difficult convergence and complex network structure especially when there is limited and incomplete information in the data set. These problems lead to under-learning or over-learning as the NN falls into a local minimum or local maximum and results in high computational complexity which is not desirable where algorithms with higher order trajectories are already in use.

FL can also be used to develop functional rules for classification of data. FL however, involves an elaborate initialization process which involves the definition of linguistic variables and terms, construction of membership functions and construction of the rule base, processes that have to be empirically determined. Furthermore, FL also involves a fuzzification process with the use of a membership function whose appropriateness is empirically determined, which makes it similar to empirically switched MS introduced in Chapter 6.

Genetic algorithms (GA) are another intelligent optimization method for decision making which is based on the principles of biological evolution but in order to utilize GA, *a priori* knowledge of the data is required to design the objective function. Also, the number of

chromosomes, crossover probability and mutation probability are set empirically with high computational complexity.

Intelligent methods that more closely fit the problem domain where complex attributes and rule system is to be used to set thresholds and classify data are SVM, FL and RST. FL has gained popularity in the image processing research community due to its ability to offer a structure for the development of functionality and rules using FL reasoning. Examples in interpolation include Taguchi and Kimura (1997) and Ting and Hang (1997) where FL was applied to edge preserving interpolation. RST on the other hand have also been employed in de-interlacing interpolation which include Yeo and Jeong (2002) and Jeon et al. (2008). SVM has been applied extensively in pattern recognition such as text categorization (Joachims 1997) and face detection in images (Osuna, Freund and Girosi 1997) successfully and can handle incomplete data gracefully provided the right support vectors, kernel and prior knowledge are employed. FL utilizes a similar (if, else) rule system to the one already introduced in MS mechanism in Chapter 6 where fixed thresholds still have to be set which might be difficult where the attributes information system has limited information with indistinguishable rules. While FL requires fuzzy rules to set the thresholds RST on the other hand utilizes the data directly to induce the rules based on the spatial-temporal attributes system with rough thresholds that can handle complex systems with a high degree of uncertainty and limited information. While SVM can handle limited data information, it has no capacity to handle uncertainty, also the complexity of the kernel and the identification of the ideal kernel are disadvantages. RST can robustly handle the roughness in information systems and most importantly, where it has been applied in de-interlacing interpolation alongside FL, (Jeon et al. 2008) RST has been reported to give better quantitative and qualitative results. This Chapter introduces RST, a decision making algorithm into DVC to increase performance and intelligently provide a switching mechanism, referred to as RST based *intelligent mode switching* (IMS).

RST provides a method to decide which MB in the frame will benefit from the addition of AOBMC by characterizing a video sequence as a decision Table of attributes and objects. The implicit complex interrelationships within the attributes of the sequence can be known and thresholds to switch between the HOPTTI and AOBMC-H can thus be intelligently and more accurately determined, leading to the improved performance of the MS mechanism thereby eliminating the need to empirically determine thresholds for each sequence that is to be improved as is done in Chapter 6.

Experimental results using the RST for improving SI generation show that after training, the intelligently switched HOPTTI (HOPTTI-RST) can produce up to 1.6dB improvement in PSNR over the empirically switched HOPTTI-AOBMC when RST predictions are based on the conventional cross-validation method used for the training sequences during training sessions. When the video sequence is tested using RST based IMS rules, a 3dB improvement over HOPTTI is achieved. Lastly, employing the original as "Target" which gives the ground truth of 100% of MBs that could be correctly switched, we find that the RST based switching could give up to 98.5% of correctly switched MBs.

7.2 RST based Intelligent Mode Switching (IMS)

In introducing some intelligence into the MS algorithm, RST is investigated and developed here for IMS. RST which is a method that has been used to address the processing of redundant video data in an efficient manner such as for example, in video shot segmentation (Cooper, Liu and Rieffel 2007; De-Bruyne, Van-Deursen and De-Cock 2008), key frame extraction (Lee, Yoo and Jang 2006), video summarization (Shuping and Xinggang 2005; Lie and Lai 2004) and as a technique for selecting the best methods of de-interlacing various sequences (Jeon et al. 2005).

RST is a tool which systematically extracts knowledge from various types of data to produce a reduced (core) decision database containing discrete essential information for increasing the efficiency of any decision making that involves the data under consideration, used extensively in applications where video content determines parameter setting (Yuan et al. 2006; Jeon et al. 2005; Qin, Wu and Wang 2006; Burges 1998; Hur et al. 2001). In Jeon et al. (2005), RST was successfully applied to selection of and switching between various re-sampling methods, depending on video content parameters used for de-interlacing which is a similar problem as switching between the algorithms for SI generation in DVC, and the RST solution to the DVC switching problem is explored in this Chapter. The key advantages that RST has over AI solutions outlined in Chapter 3 can be summarized as:

1) RST does not need *a priori* information about the data. For example FL needs prior knowledge of the data in order to choose the best membership function to apply that works effectively with provided data. The spatial-temporal characteristics, typically the *SBAD* and *SMAD* measures defined in Chapter 5 can be used as data inputs to RST to switch between either HOPTTI or AOBMC-H in order to secure the best SI quality.

2) RST provides tools that efficiently find hidden patterns in the spatial-temporal characteristics of the video by systematically applying the RST tools. These have been deployed in this thesis to find the hidden patterns in the spatial-temporal characteristics of MBs that will benefit from AOBMC being selected.

3) RST allows for an automatic way to generate the decision making rules, by evaluating the objects, attributes and decisions in the information Table and deducing the minimal set that is important in the decision making process which is used to compile the decision making rules. RST thus effectively removes attributes and objects that does not aid the decision making process, while compiling the decision making rules.

4) RST provides straightforward results that are easy to interpret as they are given as a decision. For example, in the use of RST for selecting the best methods of de-interlacing various sequences in (Jeon et al. 2005) RST gives a decision which method will give better de-interlacing given the data under investigation. The results in the context of DVC are presented in the form of which method, AOBMC or HOPTTI, gives a better quality MB in the frame of the video sequence.

A drawback of RST is the assumption that the spatial-temporal characteristics of each MB contains all the required information to induce the rules needed to decide which algorithm will give a better output. Whenever the Table of spatial-temporal characteristics does not contain all the characteristics to form rules for certain video types, erroneous decisions can occur when predicting decisions. Furthermore, there is an underlying assumption that all the possible relationships within the spatial-temporal Table can be discerned and its importance correctly induced. While efficient algorithms and RST tools have been developed, some relationships might still not be discerned leading to erroneous decisions. Lastly, there is a learning phase (rules discovering phase) after which the rules are applied to new information not previously encountered which presents a weakness, as hitherto unknown relationships will definitely lead to erroneous decisions. However, given the fact that a large enough database of video sequences with various spatial-temporal characteristics are available and that RST can induce rules based on limited and incomplete data, the drawbacks highlighted above does not preclude the use of RST to great effect as shown by the improvements in SI resulting from RST based IMS presented in Section 7.2.6.

7.2.1 Rough Set Information Table for DVC SI generation

In RST based IMS, the information Table is used to describe MBs. It consists of a Table where each column contains attributes derived from spatial-temporal characteristics of the MBs, and the attributes of each row is describing a particular MB (object in RST term).

More specifically, a full set of MBs that are generated from both the HOPTTI and AOBMC-H algorithms applied on a video sequence. Each MB is described by five attributes which are derived from the spatial-temporal characteristics of the video sequence namely; Mean Pixel which gives the mean pixel value of each MB, *SMAD* defined in (6.1) in Chapter 6, *SBAD* defined in (6.2) in Chapter 6, Condition MAD which is obtained by digitizing SMAD using the RST classification tool and Condition BAD also obtained by digitizing *SBAD*. The attributes have an association with the decision to switch to and employ one of the two algorithms of HOPTTI or AOBMC-H. In the training phase, known outcomes are put in the place of the decisions such that rules are induced from the attributes and outcome. In contrast, outcomes are deduced from attributes and rules in the test phase. Conditional BAD and Conditional MAD are used to deepen the relationship between *SBAD*, *SMAD* and the MBs in the Table looking from different perspectives. For example, the difference between pixels might require more than association with being from the object or the background, for it to be useful in predicting spatial-temporal behaviour, as realizing that the number of different objects in the frame is more than one, thereby dividing (cutting) the object into object 1, object 2 etc would increase the usability of the pixel difference data.

Employing the Table of MBs and their spatial temporal characteristics described earlier as training data, RST tools derive the rules that enable decisions as to which algorithm (HOPTTI or AOBMC-H) will generate MBs that will improve the SI from a new video sequence different from the one utilized for training.

7.2.2 RST basics and Mode Switching

RST was introduced by Z. Pawlak (Pawlak 1982) for reasoning about data. It provides a formal method for manipulating the various features and attributes in data sets which leads to the determination of the nature of the data. The features and attributes in the DVC context are the spatial-temporal characteristics of the various video sequences that are used

to switch intelligently and efficiently between the AOBMC-H algorithm and the HOPTTI algorithm. Also, the objects are the individual MBs whose characteristic attributes are being utilized to deduce the algorithm that will produce the most improved SI. These spatial-temporal characteristics have been applied empirically and shown to produce SI generation improvements in Chapter 5, which can be further increased by employing RST tools.

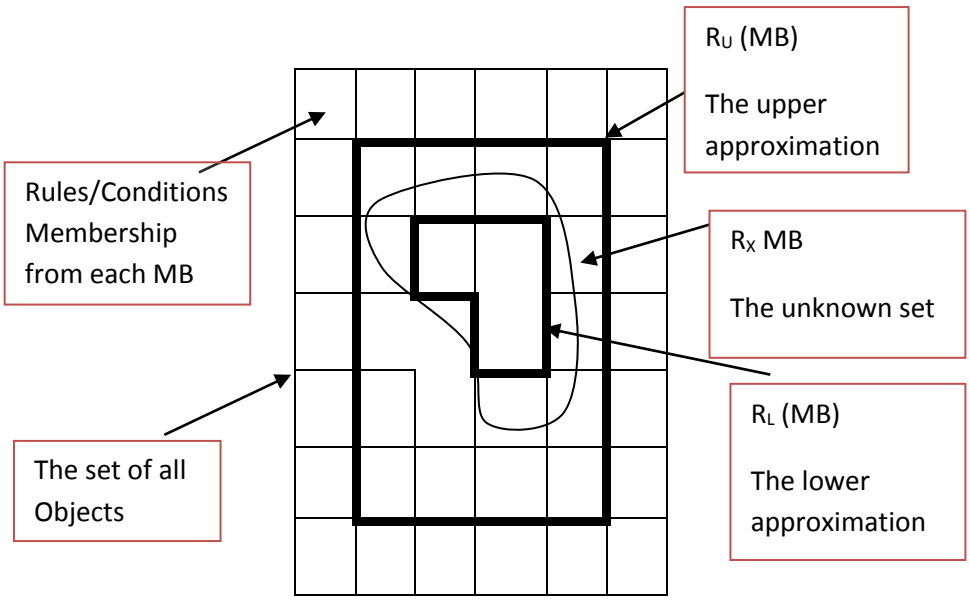


Figure 7.3 Schematic Illustration of RST.

This is illustrated in Figure 7.3, where $R_X(MB)$ is a set where the rough imprecise (unknown) boundary lies which in terms of IMS, and DVC is the set of MBs (these are usually MBs from a new video sequence whose SI we wish to improve) where we are not sure that generating the MBs with either HOPTTI or AOBMC-H algorithm will improve the SI quality of the video, thus a decision has to be made concerning them. $MB \subseteq X$ - the universal set of all the MBs generated either by HOPTTI or AOBMC-H. $R_L(MB)$ is the lower approximation, which is the known set of MBs that will be generated by HOPTTI and included in the improved SI. $R_U(MB)$ is the upper approximation, which is the known

set of MBs that will be generated by AOBMC-H and are included in the improved SI. The decision as to which algorithm the MB of the set $R_X(MB)$ will be generated with, for improved SI, is made by employing the spatial-temporal characteristics of the MBs and RST tools in the same way as the task undertaken in Chapter 5 using empirical thresholds. The basic description of the RST algorithm for implementation in the decision making in IMS for DVC SI generation is defined in the pseudo-code in Table 7.1.

Table 7.1 PSEUDOCODE FOR RST-BASED MODE IMS

Input: Table of spatial-temporal characteristics of Video for each MB

Output: Decisions between AOBMC-H or HOPTTI for SI generation

Processing:

Algorithm 1 RST based IMS Algorithm

STEP 1

1. Initialize variables: *SMAD*, *SBAD*, Mean Pixel, Conditional MAD, Conditional BAD.

2. FOR *i* = 1 to n where *n* is number of MB entries for particular video sequence.

3. Read in Variables SMAD, *SBAD*, Mean Pixel, Conditional MAD and Conditional BAD MB entries

4. Determine which algorithm HOPTTI or AOBMC-H is improved by each variable.

5. IF attribute improves HOPTTI

Variable is essential Rule for decision HOPTTI

Save Variable and it's characteristics

6. IF attribute improves AOBMC-H

Variable is essential Rule for decision AOBMC-H

Save Variable and it's characteristics

7. ELSE

Variable is not essential

Discard non-essential Variable

ENDIF

ENDIF

ENDFOR

STEP II

8. FOR i = 1 to k where k is number of MB entries for new Video sequence

9. Read in Variables *SMAD*, *SBAD*, Mean Pixel, Conditional MAD and Conditional BAD for new Video MB entries

10. Compare attributes with Saved Variables and their characteristics in **STEP I**.

IF Attributes Match HOPTTI

Output HOPTTI as decision

ELSE

Output AOBMC-H

ENDIF

ENDFOR

END

The Table 7.1 contains the pseudo code for the RST algorithm which outputs the decision as to which algorithm between HOPTTI and AOBMC-H will generate an improved SI MB based on the spatial-temporal characteristics of the video. STEP I, with pseudo code numbers 1-7 generates the rules from the object MBs and attributes *SMAD*, *SBAD*, Mean Pixel, Conditional MAD and Conditional BAD utilized to decide if HOPTTI or AOBMC-

H will produce improved SI using training video sequences. STEP II, with pseudo code numbers 8-10, compares the established rules with the spatial-temporal variables from a new video sequence and decides if it matches the HOPTTI algorithm or the AOBMC algorithm and outputs the decision.

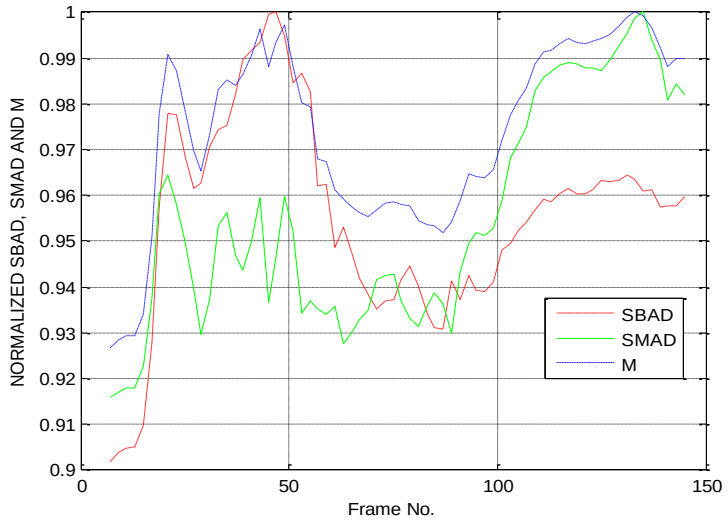
7.2.3 Video Sequence Content (spatial-temporal) Analysis and Composition of the Information Table

An important factor in RST processing is the Table of MBs and its spatial-temporal characteristics, and the choice of features to populate the Table is of critical importance. In video processing algorithms using RST, the features that have been employed in literature are luminance values, spatial characteristics of the luminance values such as luminance difference values within MBs or frames, *SBAD* and the temporal characteristics of the luminance values between MBs or frames, *SMAD* (Shuping and Xinggang 2005; Lie and Lai 2004; Jeon et al. 2005). In Chapter 5, for HOPTTI and AOBMC MS, classification tools operate on MBs generated by different algorithms thus using the empirical MS for switching between the two algorithms employing *SMAD* and *SBAD* (defined in (6.1) and (6.2) in Chapter 6) similar to (Jeon et al. 2005), we confirmed that classifying the MBs into the two modes using the spatial-temporal parameters *SBAD* and *SMAD* generates an improved SI.

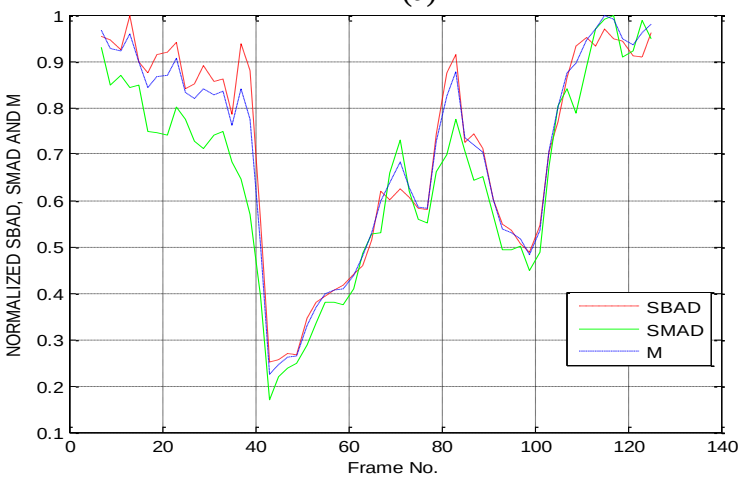
The video content attributes *SBAD* is a measure of the spatial entropy in the video sequences that measures the pixel intensity difference between pixels along the border of a MB and the pixels outside the MB. This implies that *SBAD* enforces spatial smoothness property between internal and external borders of the MB in the interpolated frame, while SMAD is a measure of the temporal entropy in the video sequence that minimizes the difference between the mean of pixels in a MB and the mean of pixels in MBs in the

intermediate frame indicating the motion of the MB and pixels under consideration (Ye et al. 2009). M is the mean of *SBAD* and *SMAD*. Examining the variation of the spatial-temporal characteristics in a systematic manner gives an indication of when thresholds need to change and modes of algorithms needed to be switched. Thus effectively, an intelligent system that could be interpreted in its simpler terms by both humans and machine can be developed (Shirahama and Uehara 2010).In order to improve SI generation using RST, the available attributes and objects must be properly classified. Employing the same strategy used in RST literature (Shuping and Xinggang 2005; Lie and Lai 2004; Jeon et al. 2005; Shirahama and Uehara 2010) where RST have been employed for video processing by relying on heuristics, familiar cuts already observed from the data enable us track what is going on in the RST based system.

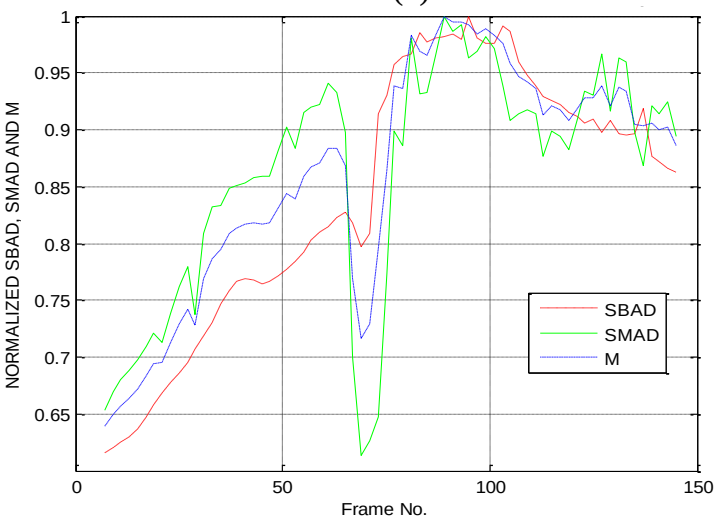
Using the properties of the various video sequences two spatial-temporal properties of *SBAD* and *SMAD* (defined in (6.1) and (6.2) in Chapter 6) and further defining other spatial-temporal characteristics in the video sequences employed in the SI generation system which include high motion region, complex motion region, low motion region, low spatial detail region, complex spatial detail region and high spatial detail region, similar to heuristics employed in literature (Jeon et al. 2005), which are observable characteristics referred to as regions for decision making and equivalent classes in RST terminology. Table 6.2 shows a sample spatial-temporal characterization of three video sequences that can be made from video sequence data such as *SMAD* and *SBAD* used in this thesis. The regions for decision making from heuristics are included to enable us to understand what is going on as they are observable by human experts. However, these classes can be generated automatically using RST tools and their generation is further discussed in 7.2.5 where the additional use of the heuristics enable us to explain the classification statistics easily.



(a)



(b)



(c)

Figure 7.4 Frame by frame normalized *SBAD* and *SMAD* for (a) *Hall*, (b) *American Football*, and (c) *Coastguard* sequences.

Figure 7.4 plots the spatial-temporal characteristics of *Hall*, *American Football* and *Coastguard* sequences and depicts the scope of spatial-temporal fluctuations that various algorithms try to cope with, and act as a pointer to the reasons why thresholds being kept constant lead to marginal improvements or failure of the algorithms. Analysis of spatial-temporal content in *Coastguard* for example, using the normalized *SBAD* and *SMAD* in Figure 7.4 (c), shows the fluctuations in the spatial-temporal characteristics, with *SMAD*, a temporal measure, assuming predominance in the early part of the sequence while *SBAD*, a spatial measure, assumes dominance between frames 65 and 120. With the above fluctuations, it is evident that keeping a single threshold or even applying the same algorithm for the whole sequence cannot give the same improvement throughout the sequence.

Based on spatial-temporal analysis six video sequences were chosen for the experimentation and rigorous testing of the RST based algorithm namely, *American Football*, *Hall*, *Coastguard*, *Carphone*, *Mother* and *Foreman* because they have a range of spatial-temporal characteristics as stated in the Table 6.1 in Chapter 6, supported by their wide use in literature, thus this thesis relies on these video sequences for training, rule extraction and testing as they cover most of the spatial-temporal characteristics required for video classification.

7.2.4 Practical Illustration of Feature Attribute Extraction

The RST for the AOBMC and HOPTTI switching is formulated into a system of attributes, objects and decisions in an information Table. Table 7.2 shows a typical characterization of the spatial-temporal characteristics of some sequences of the video dataset employed. A sample illustration of which feature attributes cuts and digitization are employed and how the labels are extracted is shown in Table 7.3 where *SBAD* and *SMAD* are divided into three categories with the labels; Small (*Sm*), Medium (*Me*) and Large (*La*). *A* and *B* in the Table are the variables denoting the values of *SBAD* and *SMAD*.

Table 7.2 SAMPLE SPATIAL-TEMPORAL CHARACTERIZATION

American Football	Complex spatial detail, high motion, complex motion
Hall	Simple spatial detail, low motion, complex spatial detail
Coastguard	Complex spatial detail, low motion
Carphone	Simple spatial detail, low motion, complex motion
Mother	Complex spatial detail, low motion
Foreman	Complex spatial detail, high motion, complex motion.

Table 7.3 ILLUSTRATION OF POSSIBLE DIGITIZATION OF ATTRIBUTES

IF	$SBAD < 8$	THEN	A is Sm
IF	$8 \leq SBAD \leq 16$	THEN	A is Me
IF	$SBAD > 16$	THEN	A is La
IF	$SMAD < 8$	THEN	B is Sm
IF	$8 \leq SMAD \leq 16$	THEN	B is Me
IF	$SMAD > 16$	THEN	B is La

The cuts here have been arbitrarily chosen to divide the numbers into three parts following a pattern that will recognize what is happening in the RST system by forming all possible classes in the information Table, and this does not affect the output SI as in practice as many distinct categories as necessary will be made by the chosen classifier (Shuping and Xinggang 2005; Lie and Lai 2004; Jeon et al. 2005).

Table 7.4 shows a sample illustration of the Table generated from spatial-temporal characteristics of MBs for *American football* as extracted by the digitization of Table 7.3. *SBAD, SMAD* are thus the basic information from the video sequences which is the same from the fixed parameter based model that uses the weighted sum of *SMAD* and *SBAD* to form a matching criterion in Chapter 6. The information Table illustrated is used for

training and the decisions are taken from the ground truth which compares the PSNR of MBs generated from AOBMC-H with HOPTTI: whichever is higher should be the right decision. This is further discussed in section 7.2.6. A similar Table without the DECISION class is used for testing.

Table 7.4 SAMPLE ILLUSTRATION OF INFORMATION TABLE SHOWING OBJECTS, ATTRIBUTES AND DECISION FOR *AMERICAN FOOTBALL*

MB No.	**mean pixel (px)**	***SMAD* (px)**	***SBAD* (px)**	**Conditional MAD**	**Conditional BAD**	**DECISION**
7	110.31	19.78	38.02	La	La	AOBMC
9	110.64	18.04	37.66	La	La	AOBMC
51	96.91	6.12	13.82	Sm	Me	AOBMC
53	96.56	7.12	15.19	Sm	Me	AOBMC
97	107.00	9.57	19.42	Me	La	AOBMC
99	107.78	10.36	21.81	Me	La	HOPTTI

7.2.5 Classification and Training Using ZeroR K-Nearest Neighbour Algorithm and Matlab Test Bed

To employ RST for the AOBMC and HOPTTI switching, a classification module is necessary to generate equivalent classes, instead of relying solely on heuristics as discussed in Section 7.2.4, and subsequently discernible matrix rules under training, which will be used to reach decisions for other totally different video sequences, in line with the RST rules formulation steps of Section 7.2.2. The ZeroR k-*nearest-neighbour* (KNN) algorithm (Clarkson K. 2005; Liu, Moore and Gray 2006; McNames 2001; Kim and Park 1986) is employed in order to avoid the use of heuristics, and it is a simple linear classifier that fits

well the premise of using the spatial-temporal characteristics of neighbouring MBs both intra-frame, inter-frame and between video sequences to choose the algorithm that will best improve the MB under consideration, and subsequently produce an improved SI.

The ZeroR KNN algorithm works by constructing a frequency Table for the target class and is usually employed as a baseline classifier, which is exactly what it is employed for in RST. The ZeroR KNN minimizes zero order error and is part of a group of fast learning algorithms that are called lazy learners, because most of its calculations are done at test time, making it useful for online, real time training data which have been found to be a better trade-off between speed and accuracy (Clarkson K. 2005; Liu, Moore and Gray 2006; McNames 2001; Kim and Park 1986).

Also, the ZeroR linear classifier is used instead of the more complex SVM which, though it can have a linear kernel, support vectors have to be calculated and the optimal hyper plane determined as discussed in section 7.1.

Furthermore, the ZeroR classifier is chosen over Neural Networks which require a higher number of training samples (Verbeek, Vlassis and Krose 2003) in order to achieve similar accuracies as the ZeroR KNN classifier. These advantages are shown in the real time training time and accuracies illustrated in Table 7.5 of this Section.

The role of ZeroR KNN here is acting as a tool in RST to obtain equivalent classes instead of relying on heuristics, which is then used to induce the RST rules during training where the discernible matrix rules are generated in line with the pseudo code of Table 7.1.

The information Table of MBs, their spatial-temporal characteristics and decisions on algorithms was used to generate the MBs in Table 7.4 which contains data from the *American Football* in combination with *Coastguard* and *Hall* sequence.

Tables 7.5, 7.6, 7.7 and 7.8 illustrate the classification statistics for the evaluation on the training set of MBs from the *American Football* sequence. 202 MBs using the training set of spatial-temporal characteristics from the video were used to make decisions for AOBMC MBs.

The illustration statistics from the equivalent classes in Tables 7.5 and 7.6 shows the classification for class ConditionMAD, predicting AOBMC and HOPTTI respectively and giving the statistics of classification based on 202 instances of which 155 predict AOBMC and 47 predict HOPTTI during training.

The Tables illustrate how the relationship between the charateristic attribtes and the decision for correctly classified instances during training becomes RST rules and uncertainties are eliminated. Also, ConditionMAD is chosen for illustration because it follows known heuristics, whereas the relationship between the characteristic attributes in *SBAD*, for instance, is internal to ZeroR KNN.

Likewise, Tables 7.7 and 7.8 show the classification for class DECISION, predicting AOBMC and HOPTTI respectively and giving the statistics based on the 202 instances. Thus the ZeroR KNN classifier acts as a baseline classifier for equivalent classes from which the RST rules will be extracted using the values of the characteristic attributes that are correctly classified.

Table 7.5 SAMPLE ILLUSTRATION OF EQUIVALENT CLASSIFICATION STATISTICS FROM TRAINING USING *HALL, AMERICAN FOOTBALL* AND *COASTGUARD* SEQUENCE PREDICTING AOBMC, CLASS CONDITIONMAD

Test Mode: Evaluation on training data; ZeroR classifier; Predicting AOBMC									
Total Number of training instances: 202									
		Predicted				True Positives %	False Positives %	Time Taken (ms)	RMS Error
		AOBMC	Correct %	HOPTTI	Incorrect %				
Actual Class	La	58	96.7%	2	3.3%	98%	2%	5	0.022
	Me	50	100%	0	0%				
	Sm	44	97.8	1	2.2%				

Table 7.6 SAMPLE ILLUSTRATION OF EQUIVALENT CLASSIFICATION STATISTICS FROM TRAINING USING *HALL, AMERICAN FOOTBALL* AND *COASTGUARD* SEQUENCE PREDICTING HOPTTI, CLASS CONDITIONMAD

Test Mode: Evaluation on training data; ZeroR classifier; Predicting HOPTTI									
Total Number of training instances: 202									
		Predicted				True Positives %	False Positives %	Time Taken (ms)	RMS Error
		AOBMC	Incorrect %	HOPTTI	Correct %				
Actual Class	La	0	0%	14	100%	96%	4%	2	0.025
	Me	1	5.9%	16	94.1%				
	Sm	1	6.2%	15	93.8%				

Table 7.7 SAMPLE ILLUSTRATION OF EQUIVALENT CLASSIFICATION STATISTICS FROM TRAINING USING *HALL*, *AMERICAN FOOTBALL* AND *COASTGUARD* SEQUENCE PREDICTING AOBMC, CLASS DECISION

Test Mode: Evaluation on training data; ZeroR classifier; Predicting AOBMC									
Total Number of training instances: 202									
		Predicted				True Positives %	False Positives %	Time Taken (ms)	RMS Error
		AOBMC	Correct %	HOPTTI	Incorrect %				
Actual Class	AOBMC	155	100%	0	0%	100%	0%	5	0.02
	HOPTTI	0	0%	0	0%				

Table 7.8 SAMPLE ILLUSTRATION OF EQUIVALENT CLASSIFICATION STATISTICS FROM TRAINING USING *HALL*, *AMERICAN FOOTBALL* AND *COASTGUARD* SEQUENCE PREDICTING HOPTTI, CLASS DECISION

Test Mode: Evaluation on training data; ZeroR classifier; Predicting HOPTTI									
Total Number of training instances: 202									
		Predicted				True Positives %	False Positives %	Time Taken (ms)	RMS Error
		AOBMC	Incorrect %	HOPTTI	Correct %				
Actual Class	AOBMC	0	0%	0	0%	100%	0%	2	0.02
	HOPTTI	0	0%	47	100%				

Closer scrutiny of the classification statistics shows that True Positives (TP) and False positives (FP) rates are very good, especially for the DECISION class. As stated earlier in Section 7.2.4 the decisions are known and are taken from the ground truth such that TP is 1

and FP is 0 both for predicting AOBMC and HOPTTI, which validates the ZeroR KNN classifier as a good baseline classifier.

Also for the class ConditionMAD, the TP is 0.98 for AOBMC prediction while it is 0.96 for HOPTTI prediction. FP for the ConditionMAD class for AOBMC prediction is 0.02 and 0.04 for HOPTTI prediction.

The correctly classified instances which is another important statistic is up to 100% in this training phase for DECISION class. The error values are insignificant and the root mean squared error of 0.02 - 0.025, though not very high, is an indication that there are still errors in the training.

The *discernible matrix*, an illustration of which is shown in Table 7.9, depends on the number of accurate decisions that are made. In this case, there are two decisions and therefore, there is a 2x2 matrix that shows correctly classified MBs. The illustration is for the ConditionMAD class based on 202 instances.

Table 7.9 SAMPLE ILLUSTRATION OF DISCERNIBLE MATRIX FROM TRAINING USING *HALL*, *AMERICAN FOOTBALL* AND *COASTGUARD* SEQUENCE FOR CONDITIONMAD CLASS.

Classification	AOBMC	HOPTTI
correct	152	45
incorrect	3	2

After the training and induction of the rules, they are then employed to predict the MBs for video sequences starting with the ones that were employed for training.

a) Modification of Software Implementation of HOPTTI Test Bed for Simulation and Results of RST based SI generation

A prediction of the decision is made at each MB based on the attributes using the *cross-validation* method because of its advantage of using the maximum possible number of training MBs. In the training statistics of Table 7.5, the training time of 0.005 seconds for ZeroR algorithm running under Microsoft Windows XP on a PC with an Intel Duo Core CPU at 2.20 GHz is negligibly small showing that ZeroR K-NN algorithm is a fast learner. The time taken is not significant compared to the overall time taken to generate SI. Thus the rules for accurate decision making is built very fast by the ZeroR K-NN algorithm from the number of accurate predictions in this training phase.

b) Computational Complexity Analysis

Generally, the DVC paradigm recognizes the fact that the decoder can be complex at the expense of the encoder complexity being reduced compared to conventional codecs like the H.264 codec.

However, the complexity and overhead that the decoder can bear is not unlimited and the computational complexity of algorithms being utilized need to be analyzed. In RST, by far, the most computationally complex part is the generation of the information Table that has adequate characteristic attributes to forms the bedrock on which the RST tools can be applied and subsequent training classification phase.

Using the SI generation time per frame presented in Table 5.1 of Chapter 5, we get the average time taken to generate an MB HOPTTI SI based on the fact that QCIF frames (176X144 pixels) employed in the experimentation results contain 99 MBs (16X16 pixels) and compare with the average learning time for ZeroR K-NN algorithm per MB. Furthermore, the time taken to generate a typical Table for various test sequences for 99 MBs

are included in the Table 7.10, with the generation of the DECISION for each MB accounting for more than 50% of the time as SI has to be generated using both HOPTTI and AOBMC-H in order for the best decision between the two algorithms to be known.

Table 7.10 AVERAGE SI GENERATION TIME PER FRAME IN MILISECONDS

Sequences	***HOPTTI Cubic(C)***	***ZeroR K-NN Algorithm***	***Information Table Generation***
Carphone	70	2.5	144
Coastguard	100	2.7	205
Foreman	140	3.0	288
Mother	80	2.5	165
Hall	110	2.8	226

Table 7.10 coupled with the learning time shown in the statistics Table 7.5 - Table 7.8, shows that the additional computational overhead for the ZeroR K-NN algorithm is not significantly high. Generation of the information Table however takes about 2 times the time taken for HOPTTI algorithm.

Employing the defined complexity variables from Table 4.1, the complexity is $T_{Const} + 4T_{Vel} + 2T_{Accl} + T_{Jolt} + T_{BMA} + T_{HOPTTI-AOBMC} + T_{IMS} + T_{off-line}$ which shows that the additional complexity is from the IMS (includes information Table generation and classification) and off-line input during discretization and learning phase. While the information Table

generation and classification can be timed, the offline discretization and learning phase cannot be timed, showing the limitation of the time stamping method of complexity evaluation.

7.2.6 Simulation and Results

a) Intelligently Switched RST SI Generation Results:

The simulations employing RST use the rules deduced from predictive performance of the ZeroR classifier algorithm i.e. the correctly classified instances. The spatial-temporal characteristic attributes of *American Football, Hall* and *Coastguard* sequences, are employed to generate the rules during training for a generic RST based IMS classifier.

The three sequences are chosen because of the range from low to complex spatial-temporal characteristics that they possess. Furthermore, they are all multiple object sequences with a variety of object types.

The generic RST classifier is used to generate SI output by switching between HOPTTI and AOBMC-H as predicted. Figures 7.5, 7.6 and 7.7 show the frame by frame generated RST based IMS SI curves compared to the empirically generated switched HOPTII-AOBMC SI and the original HOPTTI algorithm for the selected training sequences of *American Football, Coastguard,* and *Hall* sequences.

These reveal an improvement in the PSNR by intelligently switching in the correct MBs in the frames. RST based IMS outperformed original HOPTTI by up to 4 dB improvement on some frames and it is also shown to perform better than empirically switched MS.

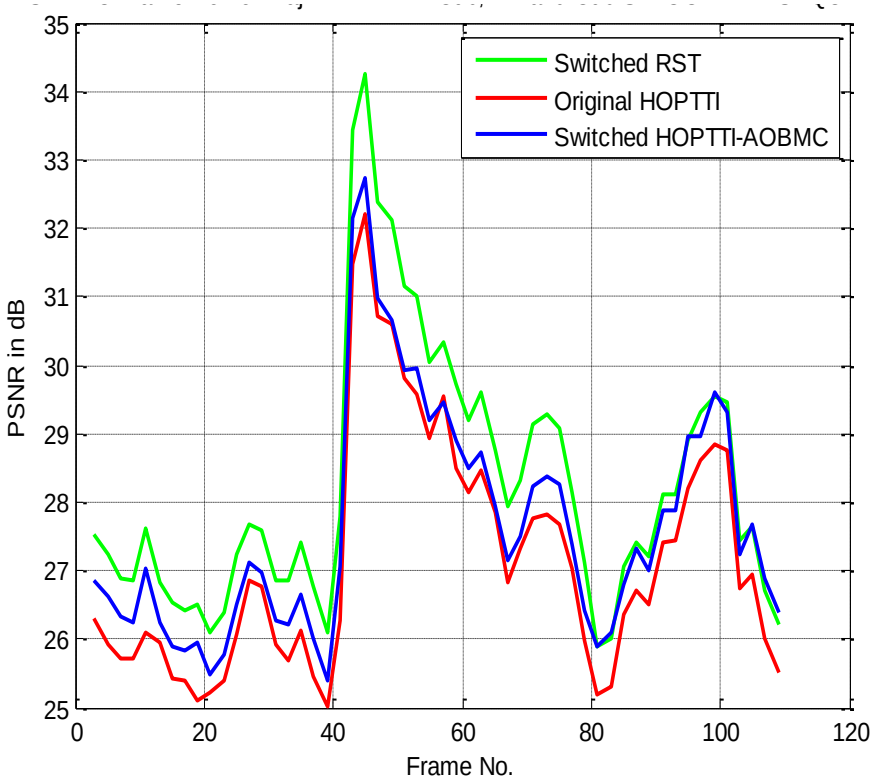


Figure 7.5 Frame-wise SI-quality of Original HOPTTI, *Switched HOPTTI-AOBMC and Swiched RST* for the *American Football* sequence

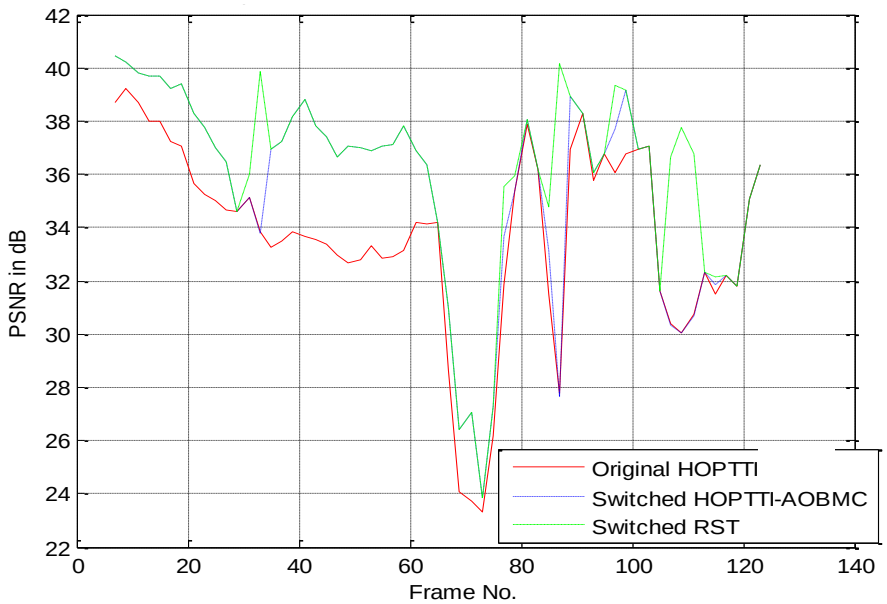


Figure 7.6 Frame-wise SI-quality of Original HOPTTI, *Switched HOPTTI-AOBMC and Switched RST* for the *Coast Guard* sequence

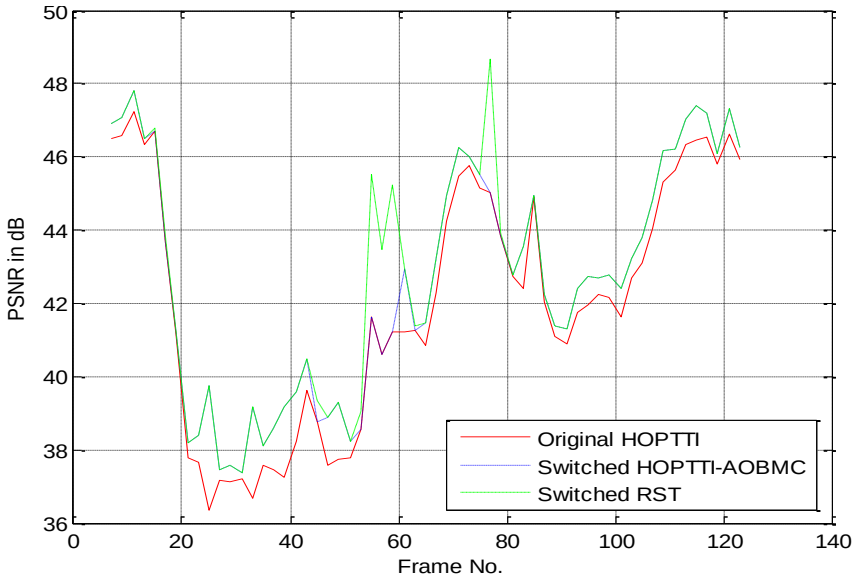


Figure 7.7 Frame-wise SI-quality of Original HOPTTI, *Switched HOPTTI-AOBMC* and Switched RST for the *Hall* sequence

Table 7.11 AVERAGE PSNR (dB) FOR *SWITCHED HOPTTI-AOBMC*, SWITCHED RST and HOPTTI FOR THE SELECTED TESTING SEQUENCES INCLUDED IN TRAINING PHASE

Sequences	***Switched HOPTTI-AOBMC (Akinola, Dooley and Wong 2011)***	***HOPTTI (Akinola, Dooley and Wong 2010)***	***Switched RST***
Coastguard	37.9	36.4	39.45
Hall	39.9	38.5	41.42
American Football	25.8	24.5	27.04
Carphone	36.2	35.3	37.34
Mother	48.4	47.3	49.12
Foreman	36.7	35.1	38.11

Table 7.11 shows the average PSNR improvement in SI generated employing the generic RST rules induced during training, giving an improvement of about 3dB for the *American Football* sequence over HOPTTI and about 1.2 dB over empirically switched SI. The *Foreman* sequence also gave above 3dB improvement over HOPTTI with a 1.4dB improvement over empirically switched SI. Likewise, the *Hall* sequence also showed about 3dB improvement over HOPTTI SI.

The results however show that when the information Table contains enough data to show adequate interrelationships for decision making, more accurate decisions can be made and higher levels of improvement achieved. Thus future works should make use of a larger pool of sequences for the rule inducing phase and training. Further analysis of the results is made using the target which is the ground truth of the algorithm that will produce better SI performance.

b) Target and Ground Truth Analysis:

A Ground truth is introduced to compare the switching performance both in terms of correct switches and PSNR performance; The target is obtained by comparing the PSNR of the SI MBs for AOBMC-H with HOPTTI, whichever is higher should be the MB that should ideally have been chosen and this is taken as the ground truth. In reality however, this is not so due to errors of prediction and the limitations of the spatial-temporal video content parameter employed in the algorithm which shows up in the information Table of attributes, objects and decisions of Table 7.6. Table 7.12 shows that intelligent switching (Switched RST) consistently outperforms the empirically switched algorithm (Switched HOPTTI-AOBMC) both in terms of percentage correctly switched frames and PSNR.

The benchmark switching performance analysis shows that intelligent switching approach improves SI generation performance both in terms of switching accuracy and PSNR performance giving up to 1.1dB over empirically switched MB based switching.

Table 7.12 BENCHMARK ANALYSIS FOR *SWITCHED HOPTTI-AOBMC(Empirical) VERSUS* SWITCHED RST (Intelligent) FOR THE SELECTED TEST SEQUENCES

Sequences		***Mode Switching***	***Switched RST***	***Target/ Ground Truth***
Coastguard	% correct switch	78.1%	98.5%	100%
	PSNR dB	37.9	39.45	40.2
Hall	% correct switch	75.6%	87.0%	100%
	PSNR dB	39.9	41.42	41.8
American Football	% correct switch	88.2%	95.5%	100%
	PSNR dB	25.8	27.04	27.7
Carphone	% correct switch	79.9%	92.3%	100%
		36.2	37.34	38.1
Mother	% correct switch	88.0%	96.4%	100%
		48.4	49.12	49.9
Foreman	% correct switch	88.0%	96.5%	100%
	PSNR dB	36.7	38.11	38.6

`

Also, qualitative (visual) results show that SI produced by RST MB based intelligent switching is qualitatively improved over that produced by HOPTTI-AOBMC and the basic mode switching algorithm. Further analysis of Table 7.12 shows that the difference between the PSNR of switched RST and the ground truth should still be within the threshold where some artefacts should still be visible as PSNR difference is in most cases above the 0.5 dB threshold (see discussion in section 4.2), the qualitative results in Figure 7.8 for example, show that there are no more visible artefacts. This further buttresses the discussion in section 4.2, showcasing the problems with PSNR as the quantitative measure of choice in the video processing community (Girod, 1993).

The challenging video sequences and frames to the HOPTTI algorithm introduced in Chapter 5, Section 5.6.3, which are further improved in Chapter 6, Section 6.3.6 are presented so that it can be concluded that improved quality of SI is achieved by RST based IMS.

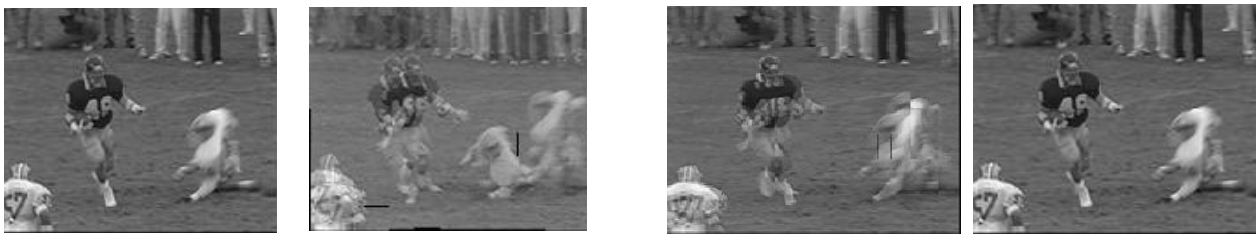
Frames of the *American Football* sequences showing perceptual improvements are illustrated in Figure 7.9, where qualitative and quantitative performance is seen to be improving as we go from all the MBs being HOPTTI to empirical MS to RST based IMS with the ghosting disappearing and PSNR increasing.

In frame #61, overlapping is overcome as we move from HOPTTI to empirical MS with the people in the background becoming as distinct as they are in the original frame. Also, in the same frame #61, as we move to the RST based IMS, the ghosting challenges in the HOPTTI frame have almost all disappeared.

The most illustrative example is frame #93 which shows the ghosting being gradually removed as the various algorithms are introduced, with the RST based IMS frame providing improved qualitative SI performance that is quite good in visual perception compared to the original frame.

Frame #61

Original	SI for all blocks on Frame HOPTTI PSNR=24.6dB	SI for empirically Switched MB in Frame PSNR=26.0dB	SI for RST Switched MB in Frame PSNR=26.8dB



Frame #93

Original	SI for all blocks on Frame HOPTTI PSNR=24.5dB	SI for empirically Switched MB in Frame PSNR=26.0dB	SI for RST Switched MB in Frame PSNR=26.8dB

Frame #99

Original	SI for all blocks on Frame HOPTTI PSNR=24.8dB	SI for empirically Switched MB in Frame PSNR=26.2dB	SI for RST Switched MB in Frame PSNR=26.8dB

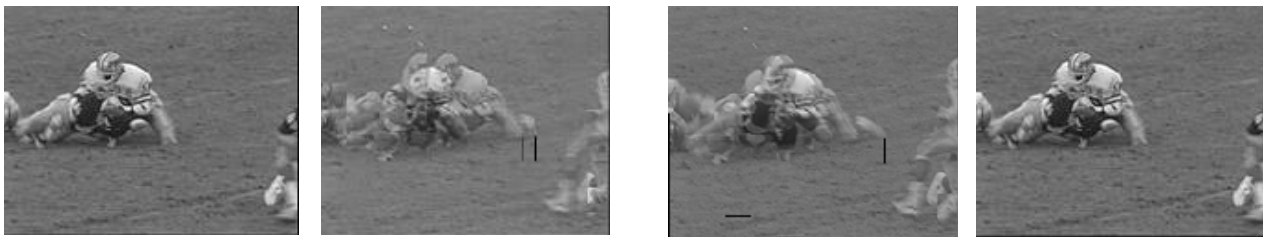


Figure 7.8 Sample frames for *American Football* showing the SI quality obtained using all blocks HOPTTI (Akinola, Dooley and Wong 2010), empirically *Switched MS and RST based IMS*.

Another of the challenging HOPTTI frames introduced in Chapter 5 and 6, in the *American Football* sequence, frame #99, is also presented showing improvement with the RST based IMS providing the best perceptual frame.

Table 7.13 shows the PSNR of SI generated using higher GOP values by employing the HOPTTI algorithm and the Switched RST algorithm. The results show that SI generation by higher GOP values which effectively increase the distances between key frames results in low degradation with Switched RST giving lower percentage PSNR reduction ($\Delta\%$) values throughout all sequences tested giving higher PSNR improvement compared to HOPTTI.

In like manner, the frames of the *Hall* sequence that pose challenges to the HOPTTI algorithm introduced in Chapters 5 and 6 are presented showing perceptual improvements are illustrated in Figure 7.10, where qualitative and quantitative performance is seen to be improving as we go from all the MBs being HOPTTI to empirical MS to RST based IMS with the ghosting disappearing and PSNR increasing. One illustrative example is the frame #51 where the left leg shows ghosting when compared to the original, which then disappeared on the introduction of AOBMC in the empirically switched MS algorithm.

The most illustrative example is frame #99 which shows the artifact on the neck and head due to the rotational motion, that HOPTTI did not accommodate, being gradually removed as the various algorithms are introduced with the RST based IMS frame providing the clearest frame with improved qualitative SI performance with the face, neck and shoulder distinctively clearer comparable to the original frame.

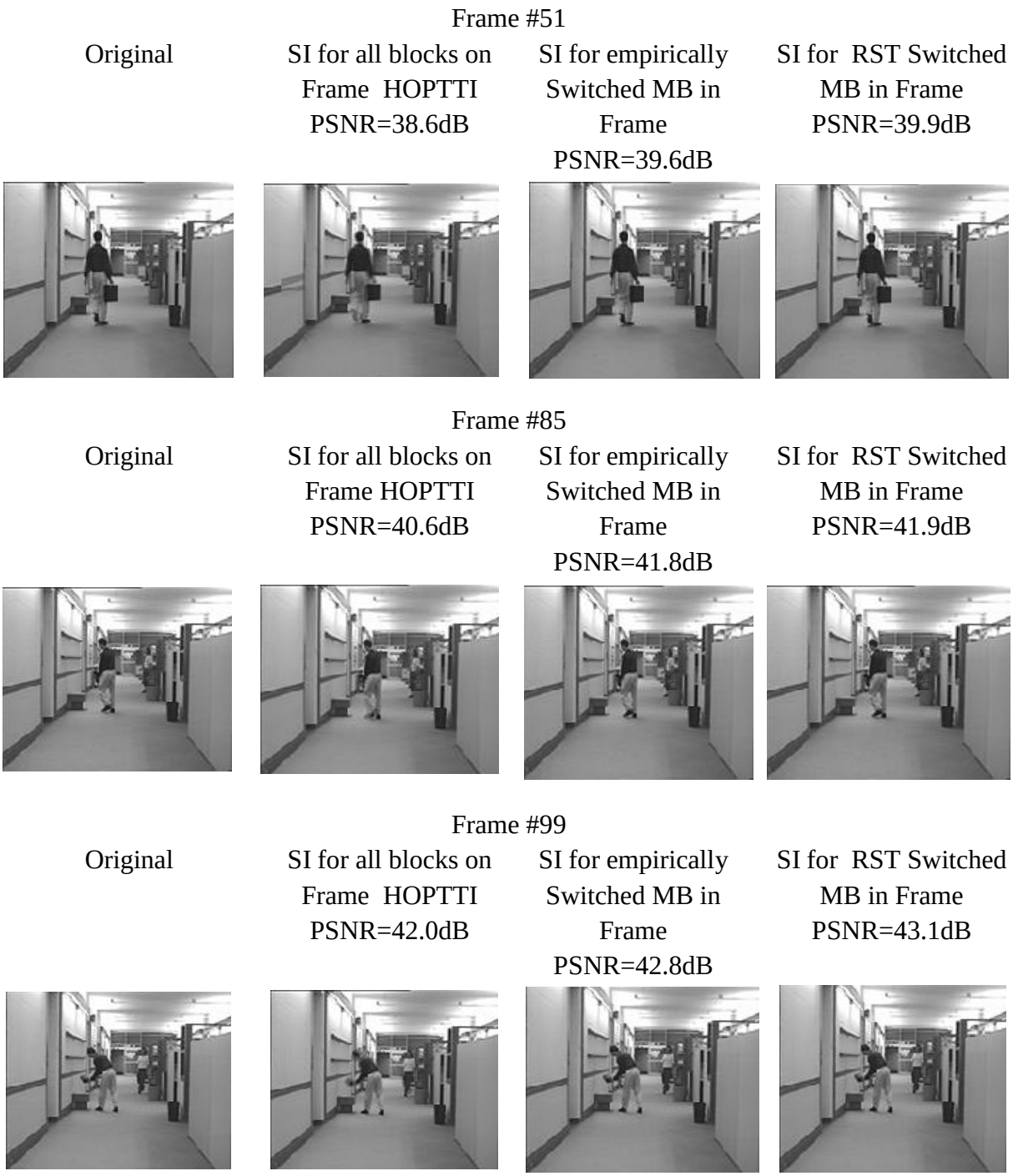


Figure 7.9 Sample frames for *Hall* showing the SI quality obtained using all blocks HOPTTI (Akinola, Dooley and Wong 2010), empirically *Switched MS and RST based IMS*.

Lastly, frames #143 of the *Coastguard* sequences showing exemplary perceptual improvements to the challenges in the HOPTTI algorithm is illustrated in Figure 7.11, where qualitative and quantitative performance is seen to be improving as we go from all the MBs being HOPTTI to empirical MS to RST based IMS with the ghosting and overlapping disappearing and PSNR increasing. This most illustrative example is a frame which shows the ghosting around additional shrub in the background of the HOPTTI based frame, including the flag that is quite faint being gradually improved as the various algorithms are introduced, with the RST based IMS being the clearest frame with improved qualitative SI perfomance.

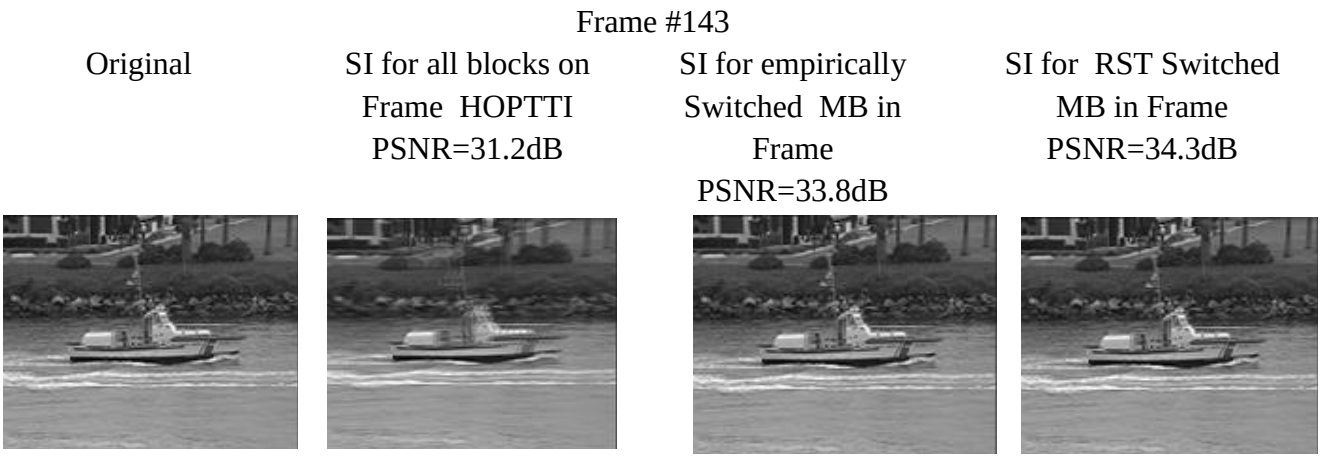


Figure 7.10 Sample frames for *Coastguard* showing the SI quality obtained using all blocks HOPTTI (Akinola, Dooley and Wong 2010), empirically *Switched MS and RST based IMS*.

c) GOP Analysis of Switched RST results:

Further testing of the algorithm is undertaken by increasing key frame temporal distances and reducing *WZ* frames to Key frames ratio employing higher GOP sizes as in Chapter 5,

Section 5.4 with the results for various GOPs for HOPTTI in Section 5.5.5. The analysis shows that Switched RST shown in Table 7.13 gives lower PSNR percentage reduction Δ% degradation for up to GOP size of 8 while giving higher PSNR performance compared to HOPTTI. This means that RST based IMS would give better SI when compared to HOPTTI when the encoder is further simplified with the dropping of more key frames such that more frames will go through the WZ path rather than the conventional codec path of the DVC architecture.

TABLE 7.13 AVERAGE PSNR OF SI OUTPUT FOR THE SELECTED SEQUENCES USING SWITCHED RST ALGORITHM FOR VARIOUS GOP SIZES.

Sequences	***Switched RST PSNR GOP 2 (in dB)***	***Switched RST PSNR GOP 4 (in dB)***	***Switched RST PSNR GOP 8 (in dB)***
American Football	27.0	25.8	25.3
	ΔPSNR(Δ%) →	1.2 (4.2)	1.7 (6.3)
Foreman	38.1	36.5	34.7
	ΔPSNR(Δ%) →	1.6 (4.2)	3.4 (8.9)
Coastguard	39.5	37.4	36.1
	ΔPSNR/Δ% →	2.1 (5.3)	3.4 (8.6)
Hall	41.4	40.1	39.9
	ΔPSNR/Δ% →	1.3 (3.1)	1.5 (3.6)

d) Rate Distortion Performance of Switched RST results:

Rate distortion results show that not all the improvement in SI from the SI generation module is carried to the final codec output. While SI improvement is up to 3dB the final codec output is only up to 2dB improvement over HOPTTI. This shows that the bottlenecks

in the other parts of the *codec* which include the channel model and rate allocation modules, also need to be improved upon in addition to SI generation. These bottlenecks are however major areas of research in themselves and are thus included in future works in Chapter 8. The overall RD results for *Foreman* and *Hall* sequence chosen to accommodate both a complex single object sequence and a multiple object sequence are shown in Figure 7.12 and Figure 7.13 respectively. The reason for choosing these two sequences is that they contain complex motions and are commonly used as comparators. The result is that RST based IMS outperforms HOPTTI, H.264 No Motion and H.264 intra while the H.264 inter remains the upper limit that outperforms Switched RST in Hall, while H.264 inter outperforms Switched RST by up to 4dB in the *Foreman* sequence. Furthermore, MC in DVC does not have access to original frames at the encoder as this is done at the decoder which makes its architecture favour the class of video sequences with slow and rigid motion, where temporal characteristics are not complex, thus this shows in the results presented in the RD results as the *Hall* sequence performs better than the *Foreman* sequence. The amount of residue that is decoded at the decoder which helps the performance of conventional codecs increases with higher bit rates and this is the additional reason why DVC outperforms H.264 Intra at low bit rates. Also overall, this shows that the improvements in the key bottleneck of SI generation effected significant RD improvement which is more visible in simpler sequences as *Hall* compared to sequences with faster and more complex spatial-temporal characteristics.

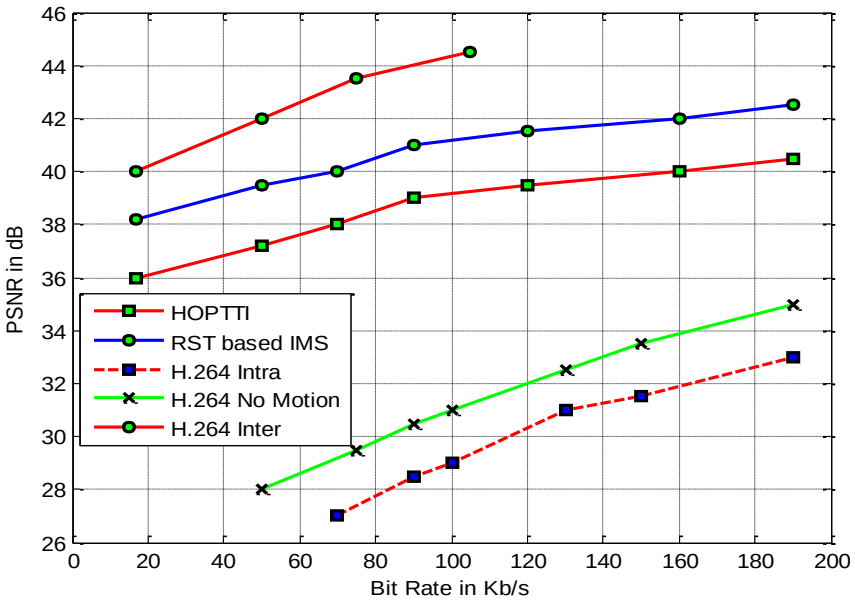


Figure 7.11 RD Curves showing HOPTTI PSNR performance in codec based on (Li, 2008) for *Foreman* sequence @ 15f/s

The RD results for the *Hall* sequence show that at low rates RST based RST outperforms H.264 inter. This is mainly due to the fact that the residue in H.264 which accounts for major performance only kicks in at medium to high bit rates and DVC is therefore more competitive at low bit rates.

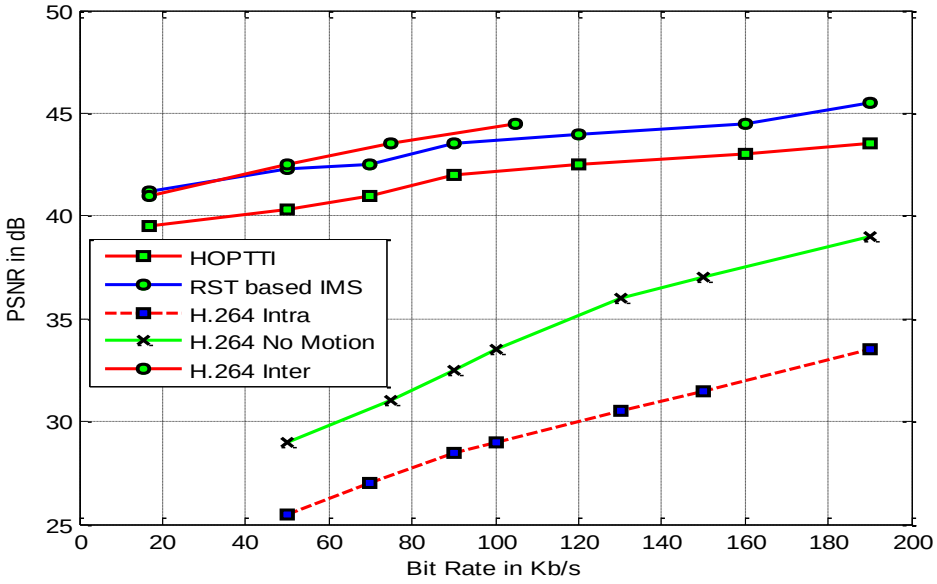


Figure 7.12 RD Curves showing HOPTTI PSNR performance in codec based on (Li 2008) for *Hall* sequence @ 15f/s

7.3 Summary

This Chapter presents the RST based IMS employed to further improve the empirical mode switching which produced up to 1.6dB improvement in PSNR over the empirically switched HOPTTI-AOBMC. Furthermore, about 3dB improvement is achieved over the HOPTTI SI generation algorithm. The ground truth switching performance analysis further shows that intelligent switching approach can correctly switch up to 98.5% of MBs to improve SI generation performance. However, it is possible that the switching performance can be improved further by incorporating more representative spatial-temporal characteristic attributes and objects in the information Table. One way to achieve this is to include a larger pool of MBs which are carefully selected to ensure fair representation from the various types of spatial-temporal stratum ranging from high motion to low motion, single object to multiple object sequences so that the RST induction rules are more accurate. Nevertheless, this is out of the scope of this proof-of-concept study. Future works including the use of multi-view images that contain highly complex sequences such as *crowd* and *break-dancers* are recommended.

Qualitative perceptual results show that frames which pose challenges for the HOPTTI algorithm that have been marginally improved in Chapter 5 with the introduction of AOBMC are further improved with the RST based IMS. This shows that RST is a good choice for IMS but future works comparing RST with other theoretically viable artificial intelligent alternatives such as SVM and Neural Network should be pursued.

Finally, overall RD curves show that a DVC codec employing Switched RST algorithm performs better than H.264 intra and H.264 no motion. Although the performance is not as good as H.264 inter, the gap is getting closer especially at low bit rates.

Chapter 8

Future Work

In this chapter, several ideas are presented in line with current literature to extend the findings from this thesis. The presented *SI Generation and Improvement Framework* in Figure 1.2 has contributed to the development of more effective DVC by tackling the pre-eminent bottleneck of SI generation and narrowing the performance gap between DVC and conventional codecs. Some possible future directions for this research which directly build on the original contributions made in this thesis are outlined in the following sections.

8.1 Extending RST Based Intelligent MS

The main conclusion from the proof-of-concept RST-based solution in Chapter 7 was that integrating intelligent MS within the *SI Generation and Improvement Framework* meant further gains in SI quality could be achieved. This outcome could be extended to optimizing parameter settings such as *SMAD, SBAD* and Mean Pixel which were the key parameters in the information Table of Table 7.4, by formulating an objective function so their contributions to RST rule formation can be adjusted to ensure MS decisions lead to lower errors and better SI output quality.

Another possibility could be to broaden the concept of intelligent MS to more than the two modes (HOPTTI and AOBMC-HOPTTI) which were investigated in Chapter 7, to include for instance, switching trajectory orders depending on the input video characteristics. This means there could be a set of MS options including HOPTTI_LINEAR, HOPTTI_QUADRATIC, HOPTTI_CUBIC, AOBMC-HOPTTI_LINEAR, AOBMC-HOPTTI_QUADRATIC and AOBMC-HOPTTI_CUBIC. A low object motion sequence like *Container* could thus switch at a MB level to HOPTTI_LINEAR and AOBMC-HOPTTI_LINEAR for example, while *American Football* which has multiple high-object motion, may instead use HOPTTI_CUBIC and AOBMC-HOPTTI_CUBIC. This would not only boost SI quality of the proposed framework, but potentially also reduce computational complexity.

It is also possible to progress the RST training classification processes by studying how to compose an optimal pool of video sequences for training purposes. The proposition is to increase the training dataset in careful and deliberate manner as it requires in-depth study (expert knowledge) in order to include sufficient number of training members with diverse characteristics to enhance learning. This enables the avoidance of the "memorization" problem that reduces the ability of AI training algorithms to truly learn. The development of an optimal pool of video training dataset will thus improve IMS, SI generated and DVC output performance.

8.2 Other Intelligent MS Strategies

As pointed out in Chapter 7, there are other alternative AI algorithms such as SVM, NN, FL and GA, which could be investigated as a means of comparison with the RST-based results. Given that AI has yet to gain wide use in DVC community, investigating the effect of

applying FL singleton, triangular or trapezoidal rule formation as opposed to RST, which induces the rules from the data presented, could be beneficial as this would dispense with the need to apply heuristics in the training processes which is inherent in RST.

Another AI method that can replace and enhance RST in the IMS algorithm of Chapter 7 is the SVM. It was particularly noted in Section 7.1 that the SVM has good classification potential, though it could have higher complexity implications even when linear hyper planes are employed. Furthermore, there is an inherent difficulty to select the appropriate kernel function as this could be application dependent. However where overhead complexity is not an overbearing concern as in the decoder side of DVC, and an appropriate kernel function is found, SVM gives very accurate classification results. Therefore, it is worthwhile to explore this AI technique in the *SI Generation and Improvement Framework* to improve SI generation and DVC output performance.

Thus a comparative analysis of different AI is merited and the corresponding impact upon SI and DVC performance should be critically evaluated.

8.3 HOPTTI and Intelligent MS in Multi-view DVC

One reason why DVC is a hot topic at the moment (Magli et al. 2013; Petrazzuoli et al 2013) is that it fits well into the predicted future of multimedia broadcasting for 3D TV and Free Viewpoint TV (Magli et al. 2013) which utilizes multiple views of the same scene. The DVC coding theory allows for a multi-view video context where spatial-temporal correlations between the various views of the same scene can be coded. Presently, multi-view DVC (Petrazzuoli et al 2013; Kodavalla and Mohan 2012) like its mono-view counterpart, utilizes LMCTI for SI generation. Therefore the development of a new *Higher Order Multi-view Piecewise Temporal Trajectory Interpolation* (HOMPTTI) within the *SI*

Generation and Improvement Framework is worthy of investigation as it could improve quantitative and qualitative multi-view DVC performance and multi-view SI quality in particular. Furthermore, video characteristics could be exploited in a HOMPTTI framework could be employed to intelligently determine the required number of views for a given quality of service level by using AI and IMS to optimize HOMPTTI. Thus, an extension of the *SI Generation and Improvement Framework* into multi-view DVC could enable both intelligent exploitation of the number of views, video characteristic and employment of more accurate, higher order, object motion model.

8.4 Channel Correlation Noise Model

The problem of how to effectively exploit intra-view and inter-view correlations, for efficient compression and joint decoding is another well-known DVC performance bottleneck (Brites et al., 2013; Petrazzuoli et al 2013; Roumy and Guillemot 2012; Yuan You and Fang 2012). In the DVC theory the channel model is assumed to be known and that it can thus be accurately modeled. Roumy and Guillemot (2012) and Yuan You and Fang (2012) showed the importance of the channel correlation model, SI and channel codes and their critical nature to the quality of decoded output. Leveraging from the research in this thesis, comprehensive investigation to determine how the spatial-temporal characteristics of sequences impact on key parameters of the various correlation noise models, especially the Laplacian model, which was implemented in the *SI Generation and Improvement Framework*. Furthermore, investigation and analysis of additional channel correlation models including, Gaussian and Dirichlet models is proposed. While a priori and post priori knowledge of the video has been proposed to improve the channel model, with Brites et al. (2013) proposing perceptually driven error correction, the effect of video characteristics on

each of the channel models and the intelligent switching of the correlation models to maximize DVC improvement gain is proposed.

Chapter 9

Conclusion

The basis for this thesis was the proven DVC theories of WZ and SW, which reverse the conventional coding paradigm of high-end encoders coupled with low-cost decoders to that of low-cost encoders coupled with high-end decoders. The premise is that despite these core theories stating DVC and conventional codecs should give the same performance quality, there is an acknowledged performance deficit which is primarily due to the low quality of SI at the DVC decoder.

The main objective in this thesis was to narrow this gap by introducing a new *SI Generation and Improvement Framework* which comprised four constituent modules, which represent original contributions to DVC. These are respectively:

(i) A cubic trajectory based HOPTTI framework for SI generation which significantly improves the accuracy of MV estimation particularly for fast moving objects, multiple objects and complex motion.

(ii) Development and incorporation of the AOBMC model to minimize the impact of BMA blocking and overlapping effects. The perceptual quality of both the SI and WZ outputs has been improved by employing this SI specific AOBMC solution.

(iii) An empirical MS algorithm that uses spatial-temporal video content parameters to switch between HOPTTI and HOPTTI-AOBMC macroblocks within frames to enhance SI generation.

(iv) An intelligent RST-based switching strategy which selects the best MBs between the HOPTTI and HOPTTI-AOBMC based intermediate frames and automatically determines key parameters.

The most notable contribution in the SI framework is the HOPTTI algorithm, because HOPTTI significantly improved the MV estimation accuracy for non-linear motions which occur in many natural sequences. HOPTTI addressed the problem differently by using a higher-order trajectory model, and even without the enhancements, consistently provided superior SI quality compared to linear-based models, especially for non-linear object motion. Improvement of at least 1.5dB and up to 5dB in PSNR was achieved for the sequences tested. The cost benefit of the HOPTTI model was analysed and revealed that SI quality generally improved as the trajectory model order increased, though the degree of improvement became negligible beyond cubic-order, while the computational cost rose exponentially.

Despite the improvements due to HOPPTI, artefacts appeared in the DVC codec output whose root causes were traced to the BMA applied during interpolation. The AOBMC algorithm was then integrated into HOPTTI and provided both quantitative and perceptual SI improvements. The results confirmed that frames produced by HOPPTI with AOBMC had lower artefacts than frames produced by HOPTTI alone.

It was also observed the AOBMC enhancement did not always work for all MBs in the frame. An investigation into selecting the better MBs between HOPTTI and HOPTTI-AOBMC frames was therefore conducted. The resulting MB selection MS mechanism was

guided by the spatial-temporal characteristics of a sequence on a MB basis, with the corresponding results confirming quantitative SI improvements of at least 1dB.

A constraint on the empirical MS algorithm was that parameters had to be set manually and different settings required for different sequences. A *proof-of-concept* intelligent MS algorithm was thus trialled and analysed. The RST-based method automated both parameter settings and MS between HOPTTI and HOPTTI-AOBMC and provided further SI improvement, with on average, 1.5dB gained in RD performance.

Overall, the findings from the novel SI generation improvement framework presented in this thesis have contributed to a narrowing of the performance gap between conventional codecs and DVC. While it is recognised that aside from the quality of SI generation, there are other DVC components which limit coding efficiency, however this thesis has explored the most fundamental bottleneck of them all whose outcome is the higher order *SI Generation and Improvement Framework* which have contributed on average 5 dB RD improvement when the improvement from each aspect of the framework is considered.

References

Aaron, A., Rane, S. and Girod, B. (2004), "Transform-domain Wyner-Ziv codec for video", Proc. Visual Communications and Image Processing, VCIP-2004, San Jose, CA.

Aaron, A., Setton, E. and Girod, B. (2003) “Toward practical Wyner-ziv coding of video,” in IEEE International Conference on Image processing, ICIP, Barcelona, Spain.

Aaron, A., Zhang, R. and Girod, B. (2002) “Wyner-Ziv coding for motion video,” in Proceedings of Asilomar Conf. Signals and Systems, ACSS, vol. 4, pp. 2002–2374.

Acticom GmbH, (2002). acticom.info - shortcuts, [website] Available at: < http://trace.eas.asu.edu/mirrors/h26l/1494.html> [21/11/2014].

Akinola, M.O., Dooley, L. S. and Wong, K. C. P. (2010) “Wyner-Ziv Side Information Using a Higher Order Piecewise Trajectory Temporal Interpolation Algorithm,” International Conf. on Graphic and Image Processing (ICGIP), Manila, Philippines, pp. 116-121.

Akinola, M.O., Dooley, L. S. and Wong, K. C. P. (2011) “Improved Side Information Generation Using Adaptive Overlapped block Motion Compensation and Higher-Order

Interpolation," 18th International Conference on Systems, Signals and Image Processing (IWSSIP), Sarajevo, Bosnia & Herzegovina, pp. 97-100.

Almalkawi, I., Zapata, M., Al-Karaki, J. and Morillo-Pozo, J. (2010) "Wireless multimedia sensor networks: current trends and future directions", Sensors, vol. 10, pp. 6662-6717.

Arizona State University, (2014). YUV Video Sequences, [website] Available at: < http://trace.eas.asu.edu/yuv/> [21/11/2014].

Artigas, X., Ascenso, J., Dalai, M., Klomp, S., Kubasov, D. and Ouaret, M. (2007) "The DISCOVER codec: architecture, techniques and evaluation," in Proc. of Picture Coding Symposium, vol 6, pp 14496-14410, Lisbon, Portugal, 2007.

Arvo, J., Torrance, T., and Smits, B. (1994) "A framework for the analysis of error in global illumination algorithms. Proceedings", SIGGRAPH 94, pp. 75-84.

Ascenso, J. and Pereira, F. (2009), "Complexity efficient stopping criterion for LDPC based distributed video coding" in proceedings of the 'MobiMedia', Kingston, UK, pp 28:1 – 28:7.

Ascenso, J., Brites, C. and Pereira, F. (2006) "Content Adaptive Wyner-Ziv Video Coding Driven by Motion Activity" in IEEE International Conference on Image Processing (ICIP), Atlanta, USA.

Ascenso, J., Brites, C. and Pereira, F. (2010), "Compression Efficiency Analysis of Wyner-Ziv Video Coding with Motion Compensated Side Information Interpolation", SPIE Visual Information and Communication (VPIC), San Jose, CA, USA.

Blanz, V., Scholkopf, B., Bulthoff, H., Burges, C., Vapnik, V. and Vetter, T., (1996) "Comparison of view–based object recognition algorithms using realistic 3d models". In

Artificial Neural Networks — ICANN'96, pages 251 – 256, Berlin. Springer Lecture Notes in Computer Science, Vol. 1112.

Borchert, S., Westerlaken, R. P., Gunnewiek, R. K. and Lagendijk, R. L. (2007) "On extrapolating side information in distributed video coding," in 26th Picture Coding System, Lisbon, Portugal.

Borchert, S., Westerlaken, R. P., Gunnewiek, R. K. and Lagendijk, R. L. (2008), "Motion compensated prediction in transform domain distributed video coding," in International Workshop on Multimedia Signal Processing, Cairns,Queensland, Australia.

Bosc, E., Pepion, R., Le Callet, P., Koppel, M., NdjikiNya, P., Pressigout, M., and Morin, L., (2011) "Towards a new quality metric for 3D synthesized views assessment" submitted to IEEE Journal of Selected Topics in Signal Processing, Volume 5 (7) pp. 1332 – 1343.

Boukerche, A., Feng, J., Werner, R., Du, Y. and Huang, Y. (2008) "Reconstructing the Plenoptic function from wireless multimedia sensor networks" In Proceedings of 33rd IEEE Conference on Local Computer Networks, LCN 2008, Montreal, Quebec, Canada, pp. 74–8

Bradski, G. (2000). The opencv library. Doctor Dobbs Journal, 25(11), 120-126

Brites, C. (2005) "Advances on Distributed Video Coding" M. Sc. Thesis, Instituto Superior Tecnico, Technical University of Lisbon, Lisbon, Portugal.

Brites, C., Ascenso, J., and Pereira, F. (2013). "Side information creation for efficient Wyner–Ziv video coding: Classifying and reviewing". *Signal Processing: Image Communication*.

Burges, C. J. C. and Scholkopf, B., (1997) "Improving the accuracy and speed of support vector learning machines". In Advances in Neural Information Processing Systems, 9pages, 375–381, Cambridge, MA. MIT Press.

Burges, C.J.C (1998) “A tutorial on support vector machines for pattern recognition” Data mining and Knowledge Discovery, 2(2): pp 121-167, 1998.

Cancellaro, M., Palma, V. and Neri, A. (2010) “Stereo video artifacts introduced by a distributed coding approach,” in Fifth International Workshop on Video Processing and Quality Metrics (VPQM), Arizona, U.S.A.

Chahine, M. (1995), “Motion compensation interpolation using trajectories with acceleration,” in IS&T/SPIE Symposium on Electronic Imaging Digital Video Compression, San Jose, CA, USA vol.2419.

Chahine, M. and Konrad, J. (1995) “Estimation and compensation of accelerated motion for temporal sequence interpolation,” in Signal Processing. Image Communication, vol. 7, pp. 503--527, Nov. 1995.

Chen, C-H., Lee S-C., and Chen J-J. (2011), "An improved block matching and prediction algorithm for multi-view video with distributed video codec", IEEE International Conference on Multimedia and Expo (ICME), vol., no., pp.1-6, 11-15.

Chen, Y., Wang, Y-K., Ugur, K., Hannuksela, M. M., Lainema, J. and Gabbouj, M. (2009). "The emerging MVC standard for 3D video services". *EURASIP J. Appl. Signal Processing.* 2009, Article 8, 13 pages.

Choi, B-D., Han, J-W., Kim, C-S. and Ko, S-J. (2007) “Motion Compensated Frame Interpolation and Adaptive Overlapped Block Motion Compensation” IEEE Transaction on Circuits and Systems for Video Technology, Vol. 17 (4), pp. 407-416.

Clarkson K. (2005) “Nearest-neighbor methods in learning and vision”, Cambridge, MA: MIT Press, ch. : Nearest-neighbor searching and metric space dimensions, pp. 15–60.

Cooper, M., Liu, T. and Rieffel, E., (2007) “Video segmemtation via temporal pattern classification” IEEE transaction on multimedia, vol. 9, no. 3, pp. 610-619.

Cortes, C. and Vapnik, V. N. (1995) "Support vector networks", Machine Learning, 20:273–297.

Craven, M. W., and Shavlik, J. W. (1997). Using neural networks for data mining. Future generation computer systems, 13(2), 211-229.

De-Bruyne, S., Van-Duersen, D. and De-Cock, J., (2008) “A compressed-domain approach for shot boundary detection on H.264/AVC bit streams” Image Communications, Vol. 23, no. 7, pp. 473-489.

DISCOVER codec evaluation (2007), [Online]. Available: <http://www.img.lx.it.pt/~discover/enc_time_complexity_gop248.html> [Accessed 12th January 2014.

Dong, B. A. and Thinh, H. Q. (2014). “An Implementation of High Speed DCT and Hadamard Transform for H. 264”. In AETA 2013: Recent Advances in Electrical Engineering and Related Sciences (pp. 329-338). Springer Berlin Heidelberg.

Fresia, M., Vandendorpe, L. and Poor, H.V.(2009) “ Distributed Source Coding using Raptor Codes for Hidden Markov Sources” IEEE Transaction on Signal Processing, vol. 57, no. 7, pp2868-2875.

Gall D. (1991) “MPEG: A Video Compression standard for Multimedia Applications”, Communications of the ACM, Vol.34, p.46-58.

Gallager, R. (1962), “Low-density parity-chechcodes”, IRE Transactions of Information Theory, vol. IT-8, pp. 21-28.

Ghanbari, M. (1999) “Video Coding, An Introduction to Standard Codecs” London: The Institute of Electrical Engineers.

Ghorbel, M., Derbel, A., Kallel, F., Samet, M. and Hamida, A. B. (2014). “Exploring Wavelet Transform Based Methodology for Cochlear Prosthesis Advanced Speech Processing Strategy”. Acta Acustica united with Acustica, 100(2), 341-352.

Girod, B. (1993) “Whats wrong with mean-squared error,”Digital Images and Human Vision, A. B. Watson, Ed., The MIT press, 1993, pp. 207–220.

Grouiller, F., Vercueil, L., Krainik, A., Segebarth, C., Kahane, P. and David O. “A comparative study of different artefact removal algorithms for EEG signals acquired during functional MRI” (2007) in Neuroimage. Volume 38(1) pp. 124-37.

Guo, X., Lu, Y., Wu, F., Gao, W. and Li, S. (2006) “Distributed Multi-viewVideo Coding” in SPIE Visual Communications and Image Processing (VCIP), San Jose, CA, USA.

Hansel, R., Richter, H., and Muller, E. (2011) “Incorporating feature point-based motion hypotheses in distributed video coding, 3rd International Congress on Ultra Modern Telecommunications and Control Systems and Workshops (ICUMT), Page(s): 1 – 6.

Hendrawan and Yusuf, N.I. (2012) "Impact analysis of bit error transmission on quality of H.264/AVC video codec," Telecommunication Systems, Services, and Applications (TSSA), 2012 7th International Conference on , vol., no., pp.314,317, doi: 10.1109/TSSA.2012.6366074.

HoangVan X and Jeon B. (2012), "Flexible Complexity Control Solution for Transform Domain Wyner-Ziv Video Coding," IEEE Transactions on Broadcasting, vol.58, no.2, pp.209 - 220.

Hore, A. and Ziou, D. (2010), "Image Quality Metrics: PSNR vs. SSIM." International Conference on Pattern Recognition, (ICPR 2010), 2366-2369.

Huang, X. and Forchhammer, S. (2008) "Improved Side Information Generation for Distributed Video Coding," IEEE 10th Workshop on Multimedia Signal Processing (MMSP), Cairns, Queensland, Australia, pp. 223–228.

Hur, A.B., Hom, D., Seigelmann, H.T. and Vapnik, V., (2001) " A Support Vector Method for Clustering" Advances in Neural Information Processing Systems, 13: pp 367-373.

Jacobs, M., Deligiannis, N., Verbist, F., Slowack, J., Barbarien, J., Van de Walle, R., Schelkens, P. and Munteanu, A. (2011), "Demo: Distributed video coding applications in wireless multimedia sensor networks," ACM/IEEE International Conference on Distributed Smart Cameras (ICDSC), pp.1-2, 22-25.

Jagnohan, A., Sehgal, A. and Ahuja, N. (2002) "Predictive encoding using coset codes," in Pmc. lClP '02, Rochester, NY, Sept. 2002, pp. 29-32

Javed, W., Khan, A., Shahzad, K., Asif, M. and Munir, F. (2012), " Analysis of Video Coding and Error Resilience Tools of H.264/AVC in Wireless Environment", International Journal of Computer Applications 50(1):1-7.

Jeon, G., Anisetti, M., Kim, D., Bellandi, V., Damiani, E., and Jeong, J., (2009) "Fuzzy rough sets for motion and scene complexity adaptive deinterlacing" Elsevier Image and Vision Computing Vol. 27, pp425-436.

Jeon, G., Won, J., Lee, R. and Jeong, J.,(2007) “A rough set approach for video deinterlacing” IEEE International conference on Multimedia and Expo (ICME) Beijing, China, July 2007, pp. 1942-1945.

Jian-Qiang, L.(2009) "3-D Hyperintegration and Packaging Technologies for Micro-Nano Systems," Proceedings of the IEEE , vol.97, no.1, pp.18,30, doi: 10.1109/JPROC.2008.2007458.

Joachims, T. (1997) "Text categorization with support vector machines", Technical report, LS VIII Number 23, University of Dortmund. ftp://ftp-ai.informatik.uni-dortmund.de/pub/Reports/report23.ps.Z.

Katsoyiannis, A. and Breivik, K. (2014). “Model-based evaluation of the use of polycyclic aromatic hydrocarbons molecular diagnostic ratios as a source identification tool”. Environmental pollution, 184, 488-494.

Kim, B. S. and Park, S. (1986) “A fast *k* nearest neighbor finding algorithm based on the ordered partition”, IEEE Trans. on Pattern Analysis and Machine Intelligence, vol. 8, no. 6, pp. 761–766.

Knoche, H. and Sasse M.A. (2006), “Breaking News on Mobile TV: User requirements of a popular mobile content”, Proceedings of IS&T/SPIE Symposium on Electronic Imaging San Jose, CA, USA. Pp 15-19.

Kodavalla, V.K. and Mohan, P.G.K. (2012), "Multi-view distributed video coding", International Conference on Devices, Circuits and Systems (ICDCS), vol., no., pp.614-618, 15-16.

Komatsu, T., Aizawa, K., Igarashi, T. and Saito, T. (1993) "Signal-processing based method for acquiring very high resolution image with multiple cameras and its theoretical analysis", Proc. Inst. Elec. Eng., vol. 140, no. 1, pp.19 - 25 .

Kubasov, D., Lajnef, K. and Guillemot, C. (2007) "A Hybrid Encoder/Decoder Rate Control for Wyner-Ziv Video Coding with a Feedback Channel", Int. Workshop on Multimedia Signal Processing (MMSP), Crete, Greece.

Kuo, T. Y. and Kuo, C. –C. J, (1998) "Motion-compensated interpolation for low bit rate quality enhancement" Proceeding SPIE visual Communication and Image Processing, Vol. 3460, pp. 277-288, July 1998.

Lee, M-H., Yoo, H-W. and Jang, D-S., (2006) "Video scene change detection using neural network: improved ART2" Expert System with Application, Vol. 31, no.1, pp. 13-25.

Li, R., Zeng, B. and Liou, M. L. (1994) "A New Three-Step Search Algorithm for Block Motion Estimation", IEEE Trans. Circuits And Systems For Video Technology, vol 4., no. 4, pp. 438-442

Li, X. and Orchard, M. (2001) "New Edge-Directed Interpolation "IEEE Trans. on Image Processing, vol. 10, no. 10, pp. 1521-1527.

Li, X., (2008) "On the importance of source classification in Wyner-Ziv video coding," in SPIE conf. on Visual Communications and Im age Processing, Jan. Vol. 5308, pp. 520-528.

Lie, W-N. and Lai, C-M., (2004) "News video summarization based on spatial and motion feature analysis" PacificpRim Conference on Multimedia (PCM 2004), Tokyo, Japan.

Liu, R., Yue, Z. and Chen, C. (2009) “Side information generation based on hierarchical motion estimation in distributed video coding,” in Chinese Journal of Aeronautics, Volume 22, Issue 2, pp. 167-173.

Liu, T., Moore, A. W. and Gray, A. (2006) “New algorithms for efficient high-dimensional nonparametric classification”, Journal of Machine Learning Research, vol. 7, pp. 1135–1158.

Liu, X., Zhao, D., Ma, S. and Gao, W . (2010) “Side information via texture and motion activity analysis in distributed video coding”, in Visual Communications and Image Processing (VCIP), Proceedings of SPIE, Volume 7744, pp77442E-77442E-8.

Magli, E., Mea, W., Frossard, P., and Markopoulou, A. (2013) "Network Coding Meets Multimedia: A Review," IEEE Transactions on Multimedia, vol.15, no.5, pp.1195-1212.

Martins R., Brites C., Ascenso J. and Pereira F. (2009) "Refining Side Information for Improved Transform Domain Wyner-Ziv Video Coding", IEEE Transactions on Circuits and Systems for Video Technology, Vol. 19, no. 9, pp. 1327-1341.

Martins R., Brites C., Ascenso J. and Pereira F. (2010) "Statistical Motion Learning for Improved Transform Domain Wyner-Ziv Video Coding", IET Image Processing, Vol. 4, no. 1, pp. 28-41.

McNames, J. (2001) “A fast nearest-neighbor algorithm based on a principal axis search tree,” IEEE Trans. on Pattern Analysis and Machine Intelligence, vol. 23, no. 9, pp. 964–976.

Natario, L., Brites, C., Ascenso, J. and Pereira, F. (2005) “ Extrapolating Side Information for Low-Delay Pixel-Domain Distributed Video Coding,” International Workshop on Very Low Bit rate Video Coding, Sept 200, pp

Nath, P. K., and Datta, D. (2014). “Multi-objective hardware–software partitioning of embedded systems: A case study of JPEG encoder”. Applied Soft Computing, 15, 30-41.

Nie, Y. and Ma, K-M. (2002) “Adaptive Rood Pattern Search for Fast Block-Matching Motion Estimation”, IEEE Trans. Image Processing, vol 11, no. 12, pp. 1442-1448.

Nie, Y. and Ma, K-K., (2002) “Adaptive Rood Pattern Search for Fast Block-matching Motion Estimation,” IEEE Trans. Image Processing, Vol. 11, Issue 12, pp.

Orchard, M. T. and Sullivan, G. J., (1994) “Overlapped Block Motion Compensation” IEEE Transaction of Image processing, Vol. 3, no. 5, pp 693-699, September 1994.

Ortega, A. (2007), "Video Coding: Predictions Are Hard To Make, Especially About The Future", IEEE Distinguished Lecturers: Distributed Estimation Using Wireless Sensor Networks, 2007.

Ostermann J., Bormans, J., List, P., Marpe, D., Narroschke, M., Pereira, F., Stockammer, T. and Wedi, T. (2004) “Video Coding with H.264/AVC: Tools, Performance, and Complexity”, IEEE Circuits and Systems, Vol. 4, No. 1. Pp 7 – 28.

Osuna E., Freund, R. and Girosi, F. (1997) "An improved training algorithm for support vector machines", In Proceedings of the IEEE Workshop on Neural Networks for Signal Processing, pages 276 – 285, Amelia Island, FL.

Ouret, M., Dufaux, F. and Ebrahimi, T. (2009), "Iterative Multiview Side Information for Enhanced Reconstruction in Distributed Video Coding", EURASIP Journal on Image and video processing Vol 2009, Article ID 591915, 17 pages.

Park J., Shim H. J. and Jeon B. (2011), "Mobile to mobile video communication using DVC framework with channel information transmission", IEEE International Symposium on Broadband Multimedia Systems and Broadcasting (BMSB), pp.1-5.

Pawlak z, (1982) “Rough sets” International Journal of Computer and Information Sciences, 11, pp 341-356.

Pereira, F., Brites, C., Ascenso, J. O. and Tagliasacchi, M. (2008), “Wyner-Ziv Video coding; A Review of Early Architectures and Further Developments”, Multimedia and Expo, 2008 IEEE International Conference, pg. 625-628.

Pereira, F., Torres, L., Guillemot, C., Ebrahimi, T., Leonardi, R. and Klomp, S. (2008) “Distributed video coding: selecting the most promising application scenarios,” in Elsevier, Signal Processing: Image Communication, vol. 23, pp. 339-352.

Petrazzuoli, G., Cagnazzo, M. and Pesquet-Popescu, B. (2010) "High order interpolation for side informationimprovement in DVC" in IEEE International Conference on Acoustics Speech and Signal Processing (ICASSP), pp 2342 – 2345, Dallas Texas, USA.

Petrazzuoli, G., Cagnazzo, M. and Pesquet-Popescu, B. (2010) “Fast and Efficient Side Information Generation in Distributed Video Coding by Using Dense Motion Representation,” European Signal Processing Conference (EUSIPCO), Denmark, pp. 2156-2160.

Petrazzuoli, G., Macovei, C., Nicolae, I.-E., Cagnazzo, M., Dufaux, F., Pesquet-Popescu, B., (2013) "Versatile multiview layered video based on distributed source coding," Image Analysis for Multimedia Interactive Services (WIAMIS), 2013 14th International Workshop on , vol., no., pp.1-4.

Pieters, B., Van Rijsselbergen, D., De Neve, W., & Van de Walle, R. (2007). “Performance Evaluation of H. 264/AVC Decoding and Visualization using the GPU”. In Optical Engineering+ Applications (pp. 669606-669606). International Society for Optics and Photonics.

Po, L.-M. and Ma, W.-C. (1996) “A Novel Four-Step Search Algorithm for Fast Block Motion Estimation”, IEEE Trans. Circuits And Systems For Video Technology, vol 6, no. 3, pp. 313-317.

Pradhan, S. S. and Ramchandran, K. (2003) “Distributed source coding using syndromes(DISCUS): design and construction,” IEEE Transactions on Information Theory, vol. 49, no. 3, pp. 626 – 643.

Pradhan, S. S., Chou, J. and Ramchandran, K. (2003), “Duality between source coding and channel coding with side information”, IEEE transactions on information theory, Vol. 49, no. 3, pp. 1181 – 1203.

Puri, R. and Ramchandran, K. (2002), "PRISM: a new robust video coding architecture based on distributed compression principles", Proc. of 40th Allerton Conference on Communication, Control, and Computing, Allerton, IL, pp.

Qin, Z.R., Wu, Y and Wang, G.Y., (2006) “ A Partition Algorith for Huge Data Sets Based on Rough Sets” Mode Identification and Artificial Intelligence, pp249-256.

Saponara S., Blanch, C., Denolf, K. and Bormans, J. (2003) “The JVT advanced video coding standard: Complexity and performance analysis on a tool-by-tool basis,” Packet Video Workshop (PV’03), Nantes, France. Pp 98 – 109.

Schmidt, M. (1996) "Identifying speaker with support vector networks". In Interface ’96 Proceedings, Sydney.

Scholkopf, B., Burges C. and Vapnik N.. (1995) "Extracting support data for a given task", Proceedings of First International Conference on Knowledge Discovery & Data Mining. AAAI Press, Menlo Park, CA, 1995.

Scholkopf, B., Burges, C. and Vapnik, V. (1996) "Incorporating invariances in support vector learning machines", In Artificial Neural Networks — ICANN’96, pages 47 – 52, Berlin. Springer Lecture Notes in Computer Science, Vol. 1112.

Semsarzadeh, M., Lotfi, A., Hashemi, M. R., and Shirmohammadi, S. (2013). "A fine-grain distortion and complexity aware parameter tuning model for the H. 264/AVC encoder". *Signal Processing: Image Communication*.

Shapiro, J. M. (1993) “Embedded image coding using zerotrees of wavelet coefficients” in IEEE Trans. on ASSP, 41(12): pp. 3445-3462.

Shen, L., Liu, Z., Zhang, Z. and Wang, G. (2007) “An adaptive and fast multiframe selection algorithm for H.264video coding,” IEEE Signal Processing Letters, vol. 14, no. 11, pp.836-839.

Shirahama, K., and Uehara, K. (2010) “Video Retrieval from Few Examples Using Ontology and Rough Set Theory” 12th International Workshop on Semantic Multimedia Database Technologies SMDT 2010),Saarbrucken, Germany.

Shuping, Y. and Xinggang, L., (2005) “Key Frame Extraction Using Unsupervised Clustering Based on Statistical Model” Tsingba Science and Technology, Vol. 10, no. 2, pp. 169-172.

Slepian, D. and Wolf, J. (1973) “Noiseless coding of correlated information sources,” in IEEE Transactions on Information Theory, vol. 19, pp.471-480.

Sohel, F.A., Dooley, L.S. and Karmakar, G.C. (2004), “A modified distortion measurement algorithm for shape coding,” Proceedings of the 3rd Workshop on the Internet, Telecommunication and Signal Processing 2004 (WITSP04), Adelaide, Australia.

Stiller, C. and Konrad, J. (1999) “Estimating motion in image sequences,” in IEEE Signal Processing Magazine, Vol. 16, no. 4, pp 70-91.

Stockhammer, T., Hannuksela, M. M. and Wiegand T. (2003) “H.264/AVC in wireless Environments” in IEEE Transactions on Circuits and Systems for Video Technology, Vol. 13, no. 7, pp. 657-673.

Tagliasacchi, M., Trapanese, A., Tubaro, S., Ascenso, J., Brites, C. and Pereira, F. (2006) “Exploiting spatial redundancy in pixel domain Wyner-Ziv video coding,” in IEEE International Conference on Image Processing, Atlanta, USA.

Taguchi, A. and Kimura, T., (2001) “Edge-preserving interpolationby using the fuzzy technique” Proceedings of the SPIE4304, Nonlinear Image Processing and Pattern Analysis XII, pp 98-105.

Tariq, F., Dooley, L. S. and Poulton, A. (2011) “Virtual clustering for resource management in cognitive femtocell networks” in 3rd International Congress on Ultra Modern Telecommunications and Systems (ICUMT), Budapest, Hungary.

Ting, H.-C. and Hang, H.-M., (1997) “Edge preserving interpolation of digital images using fuzzy inference” Journal of Visual Communications and Image Representation 8 (4) pp. 338-355.

Toto-Zarasoa, V., Roumy, A. and Guillemot, C. (2012), "Source Modeling for Distributed Video Coding," IEEE Transactions on Circuits and Systems for Video Technology, vol.22, no.2, pp.174-187.

Vapnik, V. N. (1995) "The Nature of Statistical Learning Theory", Springer-Verlag, New York."

Vapnik, V. N. (1998) "Statistical Learning Theory", John Wiley and Sons, Inc., New York.

Varodayan, D., Aaron, A. and Girod, B. (2006), "Rate-Adaptive Codes for Distributed Source Coding", EURASIP Signal Processing Journal, Special Section on Distributed Source Coding, vol. 86, no. 11.

Verbeek, J., Vlassis, N. and Krose, B. (2003) "Efficient greedy learning of Gaussian mixture models" Neural Computation, Vol. 15, no. 2, pp. 469-485.

Wang, Y. and Zhu, Q.-F. (1998) "Error Control and Concealment for Video Communication: a Review" in Preceedings of IEEE, Vol. 86, no. 5, pp. 974-997.

Weerakkody W., Fernando W., Martinez J., Cuenca P. and Quiles F. (2007) "An Iterative Refinement Technique for Side Information Generation in DVC", IEEE International Conference on Multimedia and Expo (ICME), Beijing, China.

Wiegand, T., Sullivan, G.J., Bjontegaard, G. and Luthra, A. (2003) "Overview of the H.264/AVC video Coding Standard", IEEE Transactions on Circuits and Systems for Video Technology, Vol.13 (7), pp 560-576.

Witten, I. H., and Frank, E. (2005). Data mining: Practical machine learning

Wu, B., Ji, X., Zhao, D. and Gao, W. (2009) “Spatial-aided low-delay Wyner-Ziv video coding” in Journal of Image Video Processing, Hindawi Publishing Corp., Volume 2009, Article 6, 11 pages.

Wyner, A.D. and Ziv, J. (1973) “The rate-distortion function for source coding with side information at the decoder,” in IEEE Transaction on Information Theory, vol. 22, pp.1-10.

Xiph.org, (2013). Xiph.org Video Test Media [derf's collection] [website] Available at: <http://media.xiph.org/video/derf/> [21/11/2014].

Yang, X. K., Lin, W., Lu, Z. K., Ong, E. P. and Yao, S. S. (2005) “Just noticeable distortion model and its applications in video coding,” Signal Process.: Image Commun., vol. 20, pp. 662–680, 2005.

Ye, S., Ouaret, M., Dufaux, F. and Ebrahimi, T. (2009) “Improved side information generation for distributed video coding by exploiting spatial and temporal correlations,” in EURASIP journal on Image and Video Processing, Vol. 2009, Article ID 683510, 15 pages.

Yeo, C. and Ramchandran, K. (2007) “Robust Distributed Multi-view Video Compression for Wireless Camera Networks” in SPIE Visual Communication and Image Processing (VCIP), San Jose, CA, USA.

Yuan, C., You, X. and Fang, S., (2012), "Analysis of Laplacian distribution model for virtual channel in distributed video coding," Consumer Electronics, Communications and Networks (CECNet), 2012 2nd International Conference on , vol., no., pp.881-884, 21-23.

Yuan, Z., Wu, Y., Wang, G.Y. and Li, J.B., (2006) “Motion-Information-Based Video Retrieval System Using Rough Pre-classification” Lecture Notes in Computer Science, Book Transactions on Rough Sets V, 4100:pp 306-333.

Zadeh, L. A. (1996) “Fuzzy logic= computing with words” in IEEE Transactions on Fuzzy Systems, 4(2), pp. 103-111.

Zhai, J., Yu, J. L. K. and Li, S. (2005) “A low complexity motion compensated frame interpolation method,”. ISCAS, vol. 2, pp. 4927–4930.

Zhao, H., Yu, X., Sun, J., Sun, C. and Cong, H. (2008) “An enhanced adaptive rood pattern search algorithm for fast block-matching motion estimation”, in Congress on Image and Signal Processing, Vol. 1, pp.416-420.

Zhao, Y., Zhu, C., Yu, L., & Tanimoto, M. (2013). "An Overview of 3D-TV System Using Depth-Image-Based Rendering". In *3D-TV System with Depth-Image-Based Rendering* (pp. 3-35). Springer New York.

Zhu, S. and Ma, K.-K. (2000) “A New Diamond Search Algorithm for Fast Block-Matching Motion Estimation” IEEE Transactions on Image Processing, vol 9, no. 2, pp. 287-290.

Appendix

PUBLICATIONS

Akinola, M., Dooley, L., & Wong, P. (2010). “Wyner-Ziv side information generation using a higher order piecewise trajectory temporal interpolation algorithm”. International Conference on Graphic and Image Processing (ICGIP 2010), 4-5 Dec 2010, Manila, Philippines 2010.

Akinola, M., Dooley, L., & Wong, P. (2011). “Improved side information generation using adaptive overlapped block motion compensation and higher-order interpolation”. 18th International Conference on Systems, Signals and Image Processing (IWSSIP 2011), 16-18 June 2011, Sarajevo, Bosnia and Herzegovina 2011.