論文 / 著書情報
Article / Book Information

題目(和文)	フィードバック無し分散ビデオ符号化の研究
Title(English)	Research on feedback free distributed video coding
著者(和文)	LOUW DANIEL JOHANNES
Author(English)	Daniel Johannes Louw
出典(和文)	学位:博士（工学）, 学位授与機関:東京工業大学, 報告番号:甲第9556号, 授与年月日:2014年3月26日, 学位の種別:課程博士, 審査員:金子　晴彦,横田　治夫,山田　功,杉山　将,吉瀬　謙二
Citation(English)	Degree:Doctor (Engineering), Conferring organization: Tokyo Institute of Technology, Report number:甲第9556号, Conferred date:2014/3/26, Degree Type:Course doctor, Examiner:,,,,
学位種別(和文)	博士論文
Type(English)	Doctoral Thesis

Research on feedback free distributed video coding

By

Daniel Johannes Louw

Academic Advisor: Dr. H. Kaneko

Submitted in partial fulfilment of the requirements for the degree

Doctor of Engineering

in the

Department of Computer Science

of the

Graduate School of Information Science and Engineering

TOKYO INSTITUTE OF TECHNOLOGY

February 2014

SUMMARY

RESEARCH ON FEEDBACK FREE DISTRIBUTED VIDEO CODING

by

Daniel Johannes Louw

Academic Advisor: Dr. H. Kaneko

Department of Computer Science

Doctor of Engineering

Due to the spread of low cost camera equipment and the significant increase in video content being produced on a daily basis, it is important to efficiently encode video signals so as to reduce the bandwidth and storage requirements. Conventional video coding requires significant computational resources at the encoder, whilst allowing for a simple decoder. This is convenient for broadcast type applications or applications where encoding is performed once and decoding many times. There are however applications where the encoding complexity or encoder energy consumption are constrained. Examples of these include wireless devices, surveillance systems, medical devices like capsule endoscopes, and satellites.

Single view distributed video coding (DVC) is a coding method that allows for the computational complexity of the system to be shifted from the encoder to the decoder. As a result, DVC is considered a prime candidate for low encoding complexity applications. However, currently DVC suffers from two main problems limiting its use. Firstly, DVC does not yet achieve similar compression efficiency to conventional coding methods. Specifically, the rate distortion (R-D) performance of DVC does not yet match the promise of the underlying information theoretic Wyner-Ziv and Slepian-Wolf coding theorems. This is primarily due to the difficulty in creating accurate side information at the decoder.

Secondly, the DVC encoder needs to know the required transmission rate for correct decoding at a desired distortion level. Since this rate is fundamentally unavailable in standard

DVC, current systems use a feedback path from the decoder to facilitate rate control. This disqualifies application scenarios where feedback paths do not exist, such as encode and store applications where decoding happens at a later time (ex. Recording video on a mobile phone). When the feedback path exists, its use for rate control introduces a range of practical problems such as increased latency and buffering requirements. This can negatively affect low-delay applications such as video conferencing. Alternative approaches, that do not employ feedback, suffer from either increased encoder complexity, due to performing motion estimation at the encoder, or an inaccurate rate estimate. Inaccurate rate estimates can result in a reduced average rate-distortion performance, as well as unpleasant visual artefacts.

This thesis considers the two problems of DVC as described above. Firstly, a multi-hypothesis transform domain single-view DVC system that performs symbol level coding with a non-binary low-density parity-check code is proposed to improve the R-D performance. The main contribution of the system relates to the method used for creating side information. Conventional methods use one side information hypothesis. The problem is that methods are typically accurate only when the underlying assumptions used for its derivation are accurate. In order to improve the performance over a range of different types of video conditions, multiple methods are required. In the proposed system, multiple hypotheses are created and combined using a novel Bayesian fusion method. The reconstruction process is also updated to take advantage of the multiple hypotheses. The system is then extended to combine interpolation and extrapolation in the side information creation process to improve the performance of the system over larger group-of-picture sizes. This allows for better performance at a lower encoding complexity.

Secondly, a single-view DVC system that does not require a feedback channel is proposed. The system relies on a computationally efficient method for estimating the rate. This method keeps the encoding complexity low. The consequences of inaccuracies in the rate estimate are then addressed by using codes defined over the real field instead of over finite fields. By using real fields the nature of the coding is changed. Conventional methods use finite fields to create discontiguous quantization cells. This minimizes the distortion introduced when the transmission rate is sufficiently high. However, if the rate is too low, a decoding failure occurs

resulting in a large increase in distortion. This increase in distortion creates very unpleasant visual artefacts. In contrast, real field coding creates contiguous quantization regions which allows for a more graceful degradation in the distortion when the rate is underestimated. The proposed system achieves further gains by employing a successive refinement strategy at the decoder. Finally, the proposed multi-hypothesis method is added to the proposed feedback free system in order to improve the R-D performance.

The result is a codec with average R-D performance that is comparable to that of a feedback based system at low rates without the use of motion estimation at the encoder or a feedback path. Furthermore, the use of the real field codes decreases the variance in the output distortion thus increasing the perceptual quality of the video sequence.

Keywords:
Distributed Video Coding, Non-binary LDPC, Side Information, Multi-hypothesis.

I dedicate this thesis to my loving parents

Contents

List of Figures

List of Tables

List of Abbreviations

AC	Alternating Current, refers to non-zero frequency subbands
AMP	Approximate Message Passing
BCH	Bose-Chaudhuri-Hocquenghem
BD	Bjontegaardt Delta
BM	Block Matching
BP	Belief Propagation
BSC	Binary Symmetric Channel
CS	Compressive Sensing
CIF	Common Intermediate Format
DC	Direct Current, refers to zero frequency subband
DCT	Discrete Cosine Transform
DE	Density Evolution
DFT	Discrete Fourier Transform
DVC	Distributed Video Coding
ECC	Error Correction Code
FF	Finite Field
FFT	Fast Fourier Transform
GAMP	Generalized Approximate Message Passing
GBP	Gaussian Belief Propagation
GF	Galois Field
GOP	Group of Pictures
IDCT	Inverse Discrete Cosine Transform
LDPC	Low Density Parity Check
LDPCA	Low Density Parity Check Accumulate
MAD	Mean of Absolute Differences
MAP	Maximum A Posteriori

MCE	Motion Compensated Extrapolation
MCI	Motion Compensated Interpolation
MF	Median Filter
MH	Multi Hypothesis
MSE	Mean Square Error
MV	Motion Vector
MVF	Motion Vector Field
NB	Non-binary
OBMC	Overlapped Block Motion Compensation
PDF	Probability Density Function
PMF	Probability Mass Function
PSNR	Peak Signal to Noise Ratio
QC	Quasi-Cyclic
QCIF	Quarter Common Intermediate Format
QP	Quality Parameter - CHECK
R-D	Rate-Distortion
SAD	Sum of Absolute Differences
SI	Side Information
SNR	Signal to Noise Ratio
SSIM	Structural Similarity Metric
SW	Slepian-Wolf
WZ	Wyner-Ziv

Commonly used symbols

N	The dimension size of the data signal
M	The dimension size of the encoded signal
$\mathbf{x}$	The data signal
$\mathbf{s}$	The encoded data signal, sometimes referred to as the syndrome
$\mathbf{y}$	The side information
$\mathbf{e}$	The error/noise signal
σ_e^2	The noise variance
σ_x^2	The data signal variance
N_H	The number of hypotheses
N_G	The Group-of-Picture size
q_b	The number of bits per symbol for sub-band b
$f[t]$	The pixel domain frame at time t
$F[t]$	The DCT domain frame at time t
$E[x]$	The expected value of x
$V[x]$	The variance of x

Chapter One

Introduction

Video has become ubiquitous in the modern world. A few short decades ago, video was the province of film makers and professionals. Today, the applications of video coding are extensive. With the advent of hand held video cameras and more recently smart phones, many people carry cameras in their pockets, recording and sharing all aspects of their lives. In the public arena, the need for safety leads to large scale video surveillance systems recording greater percentages of private and public spaces. In medicine, video is used in endoscopes to transmit images from inside the human body. Even in space, satellites and probes transmit back video from millions of kilometres away. As physical devices become cheaper this trend will only increase.

While all the video being created is improving society in many ways - education, communication, security, health - there comes a cost. The storage and transmission requirements for raw video signals are considerable. To counter this, video signals are often compressed. Compression reduces the space required to store, or the bandwidth required to transmit, the video signal. Compression can be lossless allowing the original signal to be reproduced exactly. It is also possible to further compress the signal by allowing small amounts of distortion to be introduced in the reconstructed signal. This is known as lossy compression. The distortion is typically added in a way that takes into account the limitations of the human visual system to reduce the negative effects on perceptual quality.

Compression incurs a cost in terms of computational complexity and energy use at both the encoder and the decoder. In a broadcast model, a video signal is encoded once and decoded many times (e.g. digital television). In this model the cost at the encoder is negligible compared to the cost at the decoder. Conventional video coding was developed to serve this need and

prioritised reducing decoder complexity at the expense of encoder complexity.

In contrast to the original broadcast model, some newer applications for video operate under different constraints. The first constraint is encoder complexity. High computational complexity increases the cost associated with the processor. This may be significant in applications such as surveillance systems where the cost of the device can be prohibitive. The second constraint is power consumption. Increased computational requirements and larger more advances processors require more power. For mobile devices, power usage affects the size, weight, cost and recharge time of the battery. It will similarly affect other devices, such as satellites, powered by solar cells. Finally, while a film maker can encode data one day and decode at some later point in time, many of the newer applications of video require a short decoder latency. As an example, consider a medical procedure where a doctor is expecting a video feed from inside a patient to provide real time responses. Similarly, in commercial applications like video conferencing there are also strong motivations to keep latency to a minimum.

Conventional lossy video compression algorithms focus on keeping the distortion to a minimum while reducing the storage requirements as much as possible. This is done without much consideration for the encoder complexity, as in a broadcast model it makes more sense to decrease the decoder complexity. Distributed Video Coding (DVC) is a recent development in video coding theory that aims to significantly reduce encoding complexity. DVC achieves this by shifting the complexity from the encoder to the decoder using the Slepian-Wolf [1] (SW) and Wyner-Ziv [2] (WZ) coding theorems.

While current DVC systems have shown that it is possible to operate at a much lower encoding complexity than conventional video coding, there are still some problems preventing DVC from adoption in industry.

- **Problem 1**: The rate-distortion performance of DVC is yet to approach the theoretically possible level of performance. As a result there is still room for improvement.

- **Problem 2**: Conventional DVC systems require the use of a feedback channel from the decoder. This creates problems in applications where the feedback channel may be missing or restricted. Examples include encode and store operations, where there is no feedback since decoding happens at some later stage (e.g. video surveillance) and real-time applications where the latency introduced by feedback processes make the system unsuitable for the application (e.g. video conferencing).

In this thesis, these two problems are investigated. Methods are proposed to improve the rate-distortion performance and remove the feedback path from the decoder.

1.1 BACKGROUND

1.1.1 Types of video coding

A digital video signal consists of a sequence of image frames made up of a grid of pixels (picture elements).

Conventional video coding can be split between intra coding and inter coding. Intra coding encodes each frame independent of the other frames. It is thus able to only take advantage of redundancy inside a frame. Inter-coding encodes groups of frames together. This allows inter-codes to take advantage of temporal redundancies as well. As the temporal sampling rate of the video signal (frame rate [Hz]) increases, so too does the amount of redundancy between frames and the effectiveness of inter-coding. In practical systems, inter codes will often use intra codes as a building block. DVC works on the principle of separately encoding frames (like an intra code) but jointly decoding frames (like an inter code). Figure 1.1 shows how the different types of video coding are related.

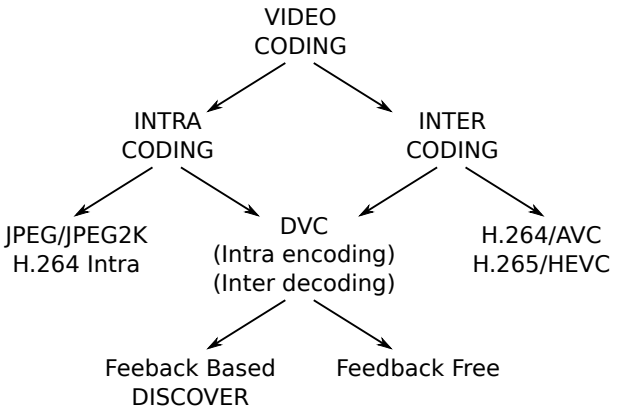

FIGURE 1.1: Classification of video coding methods

In general, inter coding will have better rate-distortion performance than intra coding at the cost of a much greater encoding complexity, and possibly a greater latency. The aim is for DVC to also achieve the same rate-distortion performance as an inter code. Though this has not yet been achieved, theory indicates that it is possible to get close. Since DVC separately encodes the frames in a sequence like an intra code, it is able to operate at a similar (or possibly lower)

encoding complexity. Figure 1.2 shows the performance complexity trade-off of the different types of video coding.

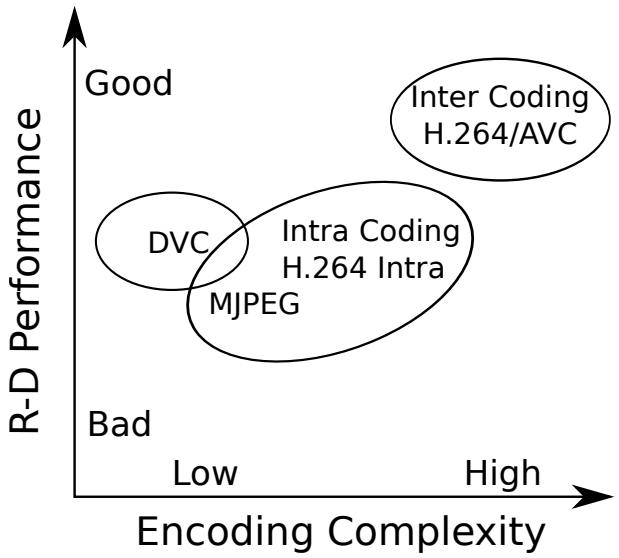

FIGURE 1.2: Comparison of video coding methods according to complexity and rate-distortion performance

1.1.2 DVC

DVC can be split between multi-view and single-view coding, where multi-view refers to considering the output from multiple video cameras as part of the same system. Single-view DVC considers only a single video signal at a time. In this thesis, only single-view DVC is considered.

DVC can also be split between pixel domain and transform domain systems. Pixel domain systems attempt to encode the original signal directly, while transform domain systems transform the signal to some other domain before encoding. Transform domain systems have better rate-distortion performance at a small increase in encoder complexity. In this thesis, we only consider transform domain DVC.

In a transform domain DVC system, frames are split into key frames and WZ frames. The key frames are intra coded, and the WZ frames are transformed, quantized and encoded using a systematic error correction code (ECC). The parity symbols from the encoding are then transmitted. At the decoder, advanced motion compensation methods are used to produce an estimate of the WZ frame, called side information (SI), from the intra coded key frames. The errors in the estimate are corrected using the received parity symbols. Improving the

performance of a DVC codec relies on improving the accuracy of the side information (the estimate of the WZ frame at the decoder).

1.1.3 Multi-hypothesis DVC

One method to improve the SI used at the decoder is to create multiple SI estimates. The idea of using multiple SI frames at the decoder (known as multi-hypothesis DVC) was first developed in 2005 [3]. In [3], the hypotheses are combined statistically by averaging the PMFs of the individual hypotheses before the error correction decoding step. Several competitive systems [4–6] have since been developed which are based on the method in [3]. A different method using an augmented decoding graph was presented in [7]. These improved methods suffer from either:

- A significantly increased decoding complexity [5–7]
- Using predefined scaling factors which do not adapt to new sequences [5, 6]

While these methods do improve the average R-D performance, the majority of the gain is from the use of advanced motion interpolation methods such as OBMC [8] and optical flow [6].

1.1.4 Feedback supressed DVC

In a standard transform domain DVC, the encoder must operate at a high enough rate (i.e. transmit enough parity symbols) to ensure that the errors can be corrected. However, calculating this rate at the encoder is fundamentally impossible without calculating the SI used at the decoder. Doing this at the encoder would defeat the purpose of DVC as it would increase the computational complexity of the encoder to the same level as that of conventional video coding. The conventional method used to solve this problem is to introduce a feedback path from the decoder. If the rate estimate is too low, more bits are requested. This requires the video sequence to be decoded in real time and renders the codec unsuitable for any application where the compressed sequence needs to be stored for decompression at a later time. It also introduces latency in the process which can become severe if the decoder is far away from the encoder. The second method used to determine the rate is called suppressed feedback DVC, where a low complexity estimate of the SI is created at the encoder and used to estimate the rate. If the rate estimate is too low, then decoding failures can lead to significant distortion in the decoded frame. Conversely, if the rate is overestimated, any extra bits are wasted. Typically, the rate

estimate is good for low motion sequences and poor for high or complex motion sequences. In order to achieve robust performance, extra bits may be transmitted to increase the probability of successfully decoding. However, a finite probability of a decoding failure remains. Increasing the number of extra bits to reduce this probability implies that a larger percentage of transmitted bits will be redundant, yielding reduced rate-distortion (R-D) performance.

Several methods for feedback free coding have been proposed, but they all suffer from either:

- Using an offline training stage that does not adapt to spatial and temporal variations in the video - [9–16].
- Unrealistic testing conditions that assume perfect key frames at the decoder - [10–13].
- A reliance on intra coding and skip modes - [14, 16].
- Increased encoder complexity due to the use of motion estimation at the encoder - [17–19].

1.2 PROPOSALS

The original research presented in this thesis can be summarised into two proposals. The first proposal is related to problem 1 and improves the rate distortion performance of DVC. For the second proposal a method is presented for feedback free DVC to address problem 2. The methods developed in the first proposal are also used as part of the system developed in the second proposal.

1.2.1 Proposal 1 : Multi-hypothesis DVC

The first proposal is related to improving the average R-D performance. The philosophy of the design is to give the belief propagation (BP) decoding algorithm access to as much information as possible. This is achieved in two ways.

Firstly, the system uses multiple hypotheses. The proposal relies on a novel way of combining hypotheses. The new method for combining SI leads to a different expression of optimal reconstruction, which is also derived in this thesis. The benefit of the proposed method for combining hypotheses is that it does not rely on pre-defined scaling factors and does not increase the decoding complexity. It also performs better than the conventional averaging approach.

Secondly, the system encodes symbols not bit-planes. This is done by employing non-binary (NB) quasi-cyclic (QC) low-density parity-check (LDPC) codes. NB codes are known to outperform binary codes for short to medium code lengths. Coding all the bit planes together also increases the effective code length, thus improving the performance of the codes.

The proposed system is based on the Stanford architecture. The key differences, as compared with conventional DVC, are the use of NB-LDPC codes and a new method for estimation of the SI from multiple hypotheses. Furthermore, interpolation and extrapolation based SI methods are combined at the decoder to improve the performance of the system over larger group-of-picture (GOP) sizes.

The result of combining all these methods is an improved average rate-distortion performance as compared with a state of the art DVC system.

1.2.2 Proposal 2 : A feedback free system using real field codes

The second proposal is related to removing the feedback path. The problem of operating without feedback is approached by proposing a novel codec structure so that the output distortion will be a smoother function of the rate.

As a result, any extra bits will continue to improve the distortion, and if the rate is underestimated, the image quality will degrade gradually. This is in contrast to conventional methods which do not benefit from over estimation and suffer from significant distortion in the case of underestimation.

The rate-distortion characteristics are modified by changing the nature of the quantization region that the Wyner-Ziv code imposes on the signal space. This change is implemented by reversing the order of the quantizer and the low density parity check (LDPC) encoder, and by defining the code over the real field instead of over a finite field.

In order to facilitate rate control without access to feedback, a low complexity rate estimation method is also proposed.

The resultant system is able to operate without feedback. In meaningful test conditions, the system is shown to operate with good rate-distortion performance. The proposed multi-hypothesis method from Proposal 1 is also added to the proposed feedback free system and shown to improve performance.

The system reduces the variance of the distortion and thus improves the perceptual video quality, in comparison with the a conventional finite field based system. This gain is

achieved without performing motion estimation at the encoder which would increase encoding complexity. The trade-off introduced by the proposed method is a reduction in the average R-D performance in the high rate region as compared to a feedback based system.

1.3 THESIS OUTLINE

Figure 1.3 shows an overview of the thesis structure.

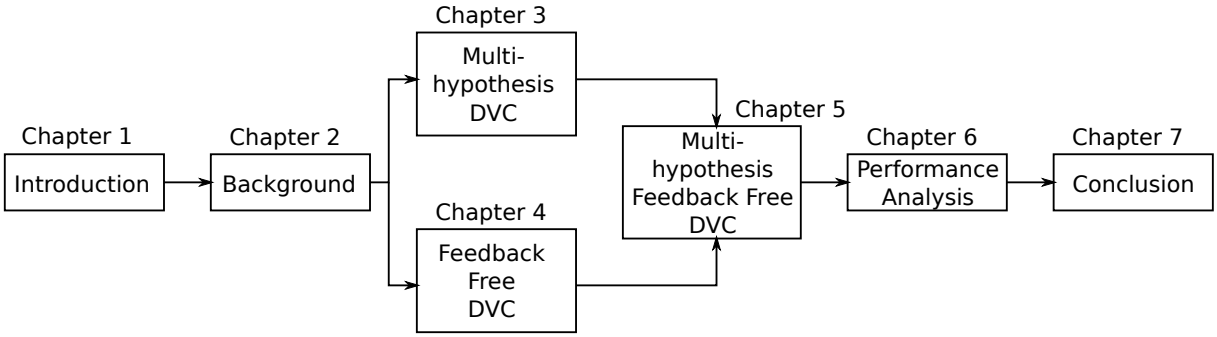

FIGURE 1.3: Overview of thesis structure.

Chapter 2 discusses the theoretical background of DVC as well as conventional DVC systems and methods for removing feedback.

Chapter 3 describes the proposed methods to improve the rate-distortion performance. This includes multi-hypothesis DVC and symbol based coding. The proposed methods are also evaluated in this chapter.

Chapter 4 describes the proposed method for removing feedback from DVC. This includes real field coding and low complexity rate estimation. The performance of the contributions are then analysed.

Chapter 5 combines the multi-hypothesis method presented in Chapter 3 with the feedback free system from Chapter 4 in order to improve the R-D performance.

In Chapter 6, the performance of both the proposed combined multi-hypothesis feedback free system described in Chapter 5 is analysed and compared with other video coding systems.

Chapter 7 concludes the thesis and provides a short summary of the contributions as well as possible areas for future research.

Chapter Two
Background

In this chapter the relevant background literature is discussed.

First, a broad overview and background on video coding is provided. Then, the foundational theorems of source coding and specifically distributed source coding as used in DVC are discussed. The specific components of conventional DVC are then individually described.

Next, the current theory and conventional approaches used in multi-hypothesis DVC as well as in feedback free DVC are discussed.

Finally, the metrics used to measure and compare different systems and solutions in this thesis are described.

2.1 Video Coding Background

2.1.1 Intra Coding

Intra coding encodes and decodes each frame independently. It works like an image or picture coding algorithm. For example, motion JPEG (MJPEG) is simply the JPEG [20] picture coding algorithm applied to each frame. Intra coding has the advantage that the encoding and decoding complexity are both reasonably low. Furthermore, since each frame is independently encoded and decoded, the delay of the system is very low and the coding process can be easily parallelised to allow different cores to encode and decode different frames. Commonly used intra codes include the intra mode of the H.264 codec [21], the intra mode of the VP8 codec, JPEG 2000 [22]. Recently (2013) the H.265/HEVC and VP9 codecs have been released, both of which contain intra modes.

2.1.2 Inter Coding

Inter coding encodes groups of pictures together. Different methods for grouping pictures have been used. In general, competitive systems rely on similar methods. Periodic, intra coded frames are used to form the backbone of the codec. The remaining frames are encoded by referring to the intra frames and possibly each other. The purpose of the inter coding is to allow the codec to take advantage of temporal redundancy. Typically, motion estimation is combined with transform coding, residual coding, entropy coding and sophisticated methods for selecting block sizes and GOP sizes. The codecs VP8, VP9, H.264 and H.265 all have similar structures differing primarily in sophistication and in smaller implementation details. Inter coding can achieve significantly better R-D performance than intra coding, but it does so at a significantly higher encoding complexity.

2.1.3 Distributed Video Coding

DVC is based on the WZ and SW coding theorems. It is called "distributed" because correlated signals can be encoded in a distributed manner. There are two types of DVC systems, single-view and multi-view. Multi-view considers video streams from different cameras pointing at the same scene', as the correlated sources that are separately encoded. Single-view considers different frames from the same video camera to be the correlated sources. In this thesis, only single-view DVC is considered.

The purpose of single-view DVC is to take advantage of the WZ coding theorem to reduce the encoding complexity, since separate encoding is less complex than joint encoding. The hope is that since the WZ and SW coding theorems indicate that there is no theoretical coding loss due to separate encoding and joint decoding, a system can be designed that will have the same R-D performance as inter coding with similar (or possibly even lower) encoding complexity as intra coding. In practice, though the encoding complexity of DVC is low, the average R-D performance is not yet as good as inter coding.

There have been several DVC systems proposed in literature along with many improvements to individual subsystems. The first practical systems were the Stanford system [23] and PRISM (Berkeley) [24]. Since then many improvements have been introduced and the majority of proposals have built on the architecture of the Stanford System. Notable milestones include the DISCOVER [25] and VISNET [26] codecs. The Stanford architecture has become the most popular approach for DVC and is also the basis of much of the work presented in this thesis.

2.2 THEORETICAL BACKGROUND

Video coding is an applied field of information theory. DVC is a subfield of video coding and relies on several specific results in information theory. In this section the necessary background will be covered.

2.2.1 Information theory concepts

Information theory was first developed in the seminal paper by Shannon [27]. In this paper, the term entropy was defined to represent the information content in a signal. Given a discrete random variable (RV) X with elements from alphabet $\mathcal{X}$, the entropy of X is defined as:

$$\begin{aligned} H(X) &= -E[\log P(X)] \\ &= -\sum_{x \in \mathcal{X}} P(X = x) \log P(X = x), \end{aligned}$$

where the base of the logarithm determines the units of entropy. Base 2 has the unit 'bits'. The entropy of an RV can be interpreted as the average number of bits required to store a single symbol of the signal. For continuous RV's, the entropy is infinite, since an infinite amount of information is required to store an analog value. To move past this problem, a continuous valued version of entropy was created. Given a continuous random variable X with a probability density function (PDF) $f_X(x)$, the differential entropy of X is defined as:

$$\begin{aligned} h(X) &= -E[\log f_X(x)] \\ &= -\int_{-\infty}^{\infty} f_X(x) \log f_X(x) \mathrm{dx} \end{aligned}$$

The interpretation of differential entropy is different to discrete entropy since the value can be negative and depends on scaling. A variable with a uniform distribution with unit width will give a differential entropy of 0. As such, the differential entropy may be interpreted as the number of extra bits per symbol required to store the signal compared to a signal with a uniform distribution.

When discussing the information content of signals, it is useful to also discuss the relationships between signals that are correlated or share information. Given discrete sources X and Y, mutual information represents the information (entropy) shared between the two sources. It is defined as:

$$I(X;Y) = -\sum_{x \in \mathcal{X}} \sum_{y \in \mathcal{Y}} P(X = x, Y = y) \log \frac{P(X = x, Y = y)}{P(X = x)P(Y = y)}$$

Another interpretation is that $I(X;Y)$ is the amount of information about X that Y provides. For independent sources the mutual information is zero. The total amount of information (entropy) contained in both sources is defined as the joint entropy:

$$H(X,Y) = -\sum_{x\in\mathcal{X}}\sum_{y\in\mathcal{Y}} P(X=x,Y=y)\log P(X=x,Y=y)$$

Finally, if one has already stored one signal, and one wishes to determine how much information remains in the other signal, conditional entropy is used:

$$H(X|Y) = -\sum_{x\in\mathcal{X}}\sum_{y\in\mathcal{Y}} P(X=x,Y=y)\log P(X=x|Y=y)$$

These terms are related as follows:

$$\begin{aligned} H(X,Y) &= H(X) + H(Y|X) \\ H(X,Y) &= H(Y) + H(X|Y) \\ H(X|Y) &= H(X) - I(X;Y) \\ I(X;Y) &= H(X) + H(Y) - H(X,Y) \end{aligned}$$

Analogous terms exist in the continuous domain, though some care should always be taken in interpreting these terms. For continuous X and Y with PDFs $f_X(x)$ and $f_Y(y)$ and joint distribution $f_{X,Y}(x,y)$, the mutual information, joint entropy and conditional entropy are defined as:

$$\begin{aligned} I(X;Y) &= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f_{X,Y}(x,y)\log\frac{f_{X,Y}(x,y)}{f_X(x)f_Y(y)}\mathrm{dxdy} \\ h(X,Y) &= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f_{X,Y}(x,y)\log f_{X,Y}(x,y)\mathrm{dxdy} \\ h(X|Y) &= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f_{X,Y}(x,y)\log f_{X|Y}(x|y)\mathrm{dxdy} \end{aligned}$$

2.2.2 Slepian-Wolf Coding

For a discrete source X the least number of bits per symbol required to represent X (rate) is given by the entropy $H(X)$. When encoding and decoding two sources X and Y separately, the required rate for each is lower bounded by $H(X)$ and $H(Y)$ respectively. This yields a total required rate of $H(X) + H(Y)$. When X and Y are correlated it is possible to reduce the total required rate further by taking advantage of the redundancy between sources. This is done by

jointly encoding and decoding the sources. In this scenario, given two discrete sources X and Y, the total required rate R is lower bound as:

$$R \geq H(X,Y)$$

In distributed source coding, the situation where the correlated sources are separately encoded is considered. For the case of lossless coding of discrete sources, Slepian and Wolf [1] showed that there is no performance loss as a result of not performing joint encoding as long as joint decoding is still performed. Thus, defining R_X and R_Y as the rates required for transmitting these sources, the SW theorem [1], provides the following bounds:

$$\begin{aligned} R_Y &\geq H(Y|X) \\ R_X &\geq H(X|Y) \\ R_Y + R_X &\geq H(X,Y) \end{aligned}$$

The rate region for the SW code can be seen in Figure2.1. This remarkable result forms the basis of DVC coding.

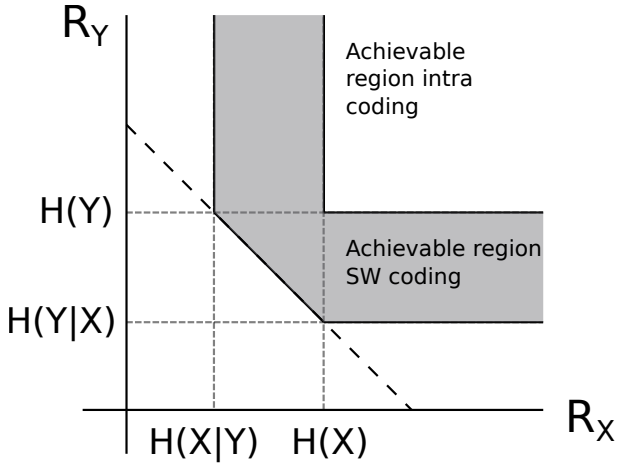

FIGURE 2.1: The achievable rate region for the Slepian-Wolf code.

2.2.3 Wyner-Ziv Coding

Up to this point we have assumed lossless coding of discrete random variables. In some applications it might be acceptable to incur some distortion if it can significantly reduce the rate requirements to store or transmit the information. Video coding is such an example

where significant coding efficiency is achieved by allowing a small amount of distortion in the reproduced signal. WZ coding is the extension of SW coding to the case of lossy compression.

Let X and Y be RVs with a joint distribution $f_{X,Y}(x, y)$. If X is the original signal and Y is the side information available only at the decoder, then $\hat{X}$ is the reconstructed signal at the decoder. Let $D = d(x, \hat{x})$ be a distortion function measuring the error in the reconstructed signal at the decoder. A Wyner-Ziv code [2] is defined as two measurable mappings F_E and F_D. F_E is the encoder and maps N input symbols to one of m output symbols. F_D is the decoder and maps one of m encoded symbols and N side information symbols to N decoded symbols:

$$\begin{aligned} F_E : \mathcal{X}^N &\rightarrow \mathcal{J}_m \\ F_D : \mathcal{J}_m \times \mathcal{Y}^N &\rightarrow \hat{\mathcal{X}}^N \end{aligned}$$

where $\mathcal{J}_m = \{0, 1, \ldots, m\}$. Thus $\hat{X}^N = F_D(Y^N, F_E(X^N))$. A rate distortion pair (R, D) is achievable if a code exists for an arbitrary ϵ and a sufficiently large N, where $E[d(x, \hat{x})] \leq D + \epsilon$ and $m \leq 2^{N(R+\epsilon)}$. If we define $\mathcal{R}$ as the set of achievable pairs, $(R, d) \in \mathcal{R}$, then the operational rate distortion function $R^{\mathrm{WZ}}_{X|Y}(D)$ is defined as:

$$R^{\mathrm{WZ}}_{X|Y}(D) = \min_{(R,D)\in\mathcal{R}} R$$

$R^{\mathrm{WZ}}_{X|Y}(D)$ describes the minimum rate required to transmit X given that Y is available at the decoder for an expected distortion D. The Wyner-Ziv theorem gives:

$$R^{\mathrm{WZ}}_{X|Y}(D) = \inf \left(I(X; Z) - I(Y; Z) \right),$$

where the infimum is taken over all random variables Z such that $Y \rightarrow X \rightarrow Z$ form a Markov chain and the distortion requirement is met. Let $R_{X|Y}(D)$ be the minimum required transmission rate for X at distortion D, given that Y is available at both the decoder and the encoder. The required rate when the side information is available only at the decoder can be lower bounded by the required rate when the side information is available at both the encoder and the decoder.

$$R^{\mathrm{WZ}}_{X|Y}(D) \geq R_{X|Y}(D)$$

What is remarkable is that the rate loss due to not having access to the side information at the encoder, $R^{\mathrm{WZ}}_{X|Y}(D) - R_{X|Y}(D)$, is zero for all D when X and Y are jointly Gaussian. It has been shown that in video coding, the rate loss can be up to 0.5 bit/sample for general source statistics and a mean square error distortion metric [28].

In practice, WZ codes are often implemented as quantizers followed by SW codes. The quantizer determines D and the SW code determines the minimum R required to transmit the quantized source losslessly.

2.2.4 DVC as an instance of Wyner-Ziv coding

DVC coding can occur either in the pixel domain or in the transform domain. Pixel domain DVC may have a lower encoding complexity, but typically will perform worse in terms of R-D performance. In this thesis only transform domain coding is considered. There are several candidate transforms for video coding. These will be discussed later in this chapter. In this thesis, as in most DVC literature, we will consider the discrete cosine transform (DCT).

Incoming frames in a video sequence are split between key frames and Wyner-Ziv frames. The key frames are encoded using an intra code. The key frames are encoded and decoded without reference to any other frames in the sequence.

The pixel domain WZ frames are transformed blockwise by the DCT transform. The coefficients corresponding to each subband of the transform are grouped together. In terms of WZ coding each subband can be considered as a continuous RV (X). Each subband is separately encoded.

At the decoder an estimate of the WZ frame is produced by some kind of motion compensated interpolation or extrapolation from the key frames. This estimate frame is transformed to produce an estimate of each subband. This estimate acts as side-information in the WZ sense (Y). Regarding the SI as the output of a virtual channel completes the model.

Each subband is thus encoded by a WZ code. The WZ code is implemented as a quantizer followed by a SW code. The SW code is implemented as an error correction code.

Pixel domain coding works in a similar fashion except that the vectorized form of the pixel domain signal is used as the data in the WZ code instead of the transformed subbands.

2.2.5 Theoretical Problem Description

Consider information sources X, Y_1 and Y_2. If an encoder were to have access to all three sources, and encode them jointly, then the lowest rate required for storage or transmission of these sources would be:

$$\begin{aligned} R &\geq H(X, Y_1, Y_2) \\ &= H(X|Y_1, Y_2) + H(Y_1, Y_2) \end{aligned}$$

If Y_1 and Y_2 were independent, then this becomes:

$$R \geq H(X|Y_1, Y_2) + H(Y_1) + H(Y_2)$$

This is the scenario for conventional video coding, where Y_1 and Y_2 are intra frames (or key frames) and X is an inter frame. The SW theorem indicates that this same performance is possible even when X is encoded separately from Y_1 and Y_2. In practice, it is not so simple. Conventional codes operate by calculating an estimate of X at the encoder ($\hat{X}$) by performing motion estimation:

$$\hat{X} = f(X, Y_1, Y_2)$$

This function has access to the original frame X to use in finding the best estimate which can be created from Y_1 and Y_2. Due to this access, the estimate can be very accurate. The rate bound then becomes:

$$R \geq H(X|\hat{X}) + H(Y_1) + H(Y_2)$$

Improving conventional coding is thus done by improving intra coding techniques to attempt to achieve $H(Y_1)$ and $H(Y_2)$, as well as improving $f(\cdot)$ so that $H(X|\hat{X})$ approaches $H(X|Y_1, Y_2)$ and then finding a way to achieve $H(X|\hat{X})$.

In the SW scenario, this scenario is different. Only the decoder has access to Y_1 and Y_2 to create an estimate $\hat{X}_2$. This estimate will be worse than the one created using conventional coding. This can be written in terms of entropy as:

$$H(X|\hat{X}) \leq H(X|\hat{X}_2)$$

As a result, the achievable rate for SW based systems will be higher (worse) than for conventional methods when the SI at the decoder is worse than the SI that can be created at the encoder. Typically, this means that DVC performs badly in high or complex motion sequences. Improving motion estimation methods at the decoder is thus the main method used to improve DVC performance. The first part of the proposals in this thesis attempt to improve the side information creation process.

The second problem faced by SW based systems can be found by looking at the constraints in which SW coding operates. Namely, X and (Y_1, Y_2) are encoded separately. How does the encoder for X know what rate to encode at? Theoretically it could operate at $H(X|\hat{X}_2)$, but it does not have access to $\hat{X}_2$. If the encoder guesses wrong and tries to operate at $R < H(X|\hat{X}_2)$, then there is no guarantee on performance, and there will certainly be errors. If it attempts to operate at $R > H(X|\hat{X}_2)$, then there will be no errors, but the code will not be achieving the bound and there will be a rate loss. Thus, if we know that the error in the rate estimate is anything other than zero, then there will be a non-zero loss in rate-distortion performance.

Current systems use two methods to bypass the rate knowledge issue:

1. Break down the independent coding assumption and use the key frames at the encoder to create $\hat{X}_2$ and estimate the rate. To some extent all DVC methods do this.

2. Introduce a feedback path from the decoder. In this scenario, the encoder keeps transmitting bits until the decoder indicates that it has received $H(X|\hat{X}_2)$.

The problems with these two methods are:

1. In order to calculate $\hat{X}_2$, motion estimation must be performed at the encoder. This can significantly increase the encoder complexity. The main reduction in encoding complexity that DVC gains over conventional coding comes from not performing motion estimation. Thus, adding motion estimation removes the purpose of DVC.

2. The problems with using a feedback path are that it limits the usefulness of the technology and increases technical requirements as described in the general problem description section. Furthermore, there are applications where the feedback path does not exist.

The proposed research attempts to solve the problem of rate knowledge at the encoder without using a feedback channel or increasing the computational complexity significantly relative to conventional DVC - thus no motion estimation is allowed at the encoder. While DVC decoding complexity is not considered as a constraint, it should still be kept as low as possible.

2.3 CONVENTIONAL DVC

In this section, an overview of conventional DVC systems will be provided. A description of the most common DVC structure, the Stanford architecture, will be given in Section 2.3.1.1 followed by a short description of the DISCOVER codec, which is based on the Stanford architecture, in Section 2.3.1.2.

Next, each of the sub-systems of DVC will be discussed. First, a description of conventional methods for rate control and quantization will be provided in Section 2.3.2. Then, error correction coding methods are described in Section 2.3.3. This is followed by a description of the side information creation process in Section 2.3.4. Finally, transforms are briefly discussed in Section 2.3.6 and then the specific quantizers are described in Section 2.3.7.

2.3.1 System Overview

2.3.1.1 Stanford Architecture

In a standard Stanford architecture based DVC system, the incoming frames are split between intra coded frames (called Key Frames) and WZ coded frames. The WZ frames will be independently encoded but jointly decoded with the key frames. If the key frames are spaced periodically then the GOP size is the number of WZ frames between key frames plus one.

The intra coded frames are encoded using an intra code. The WZ frames are then transformed and split into subbands of the transform. The transformed subbands are quantized, and then each bit plane of the quantized subbands are independently encoded using an error correction code.

At the decoder, the key frames are decoded using a decoder matched to the encoder intra code. These key frames are then used to create an estimate of the WZ frame called the SI. The ECC is then used to correct the errors in the SI. Typically if a soft-input ECC is used, the error in the SI must first be estimated. This process is called correlation noise estimation and can have a large affect on the performance. If the ECC fails to decode due to the rate being too low, extra bits are requested over the feedback path from the decoder to the encoder. After the decoding, the quantization process is inverted with a reconstruction algorithm. The reconstructed subbands are then combined and the inverse transform is applied to produce the decoded pixel domain WZ frame.

Figure 2.2 shows a block diagram of this basic structure as reproduced from [29].

2.3.1.2 DISCOVER Codec

The DISCOVER codec [25] is based on the Stanford architecture and follows the same structure. The specifics are provided below.

- 4×4 blockwise DCT transform
- Turbo Codes and LDPC Accumulate (LDPCA) codes (there are versions with either one)
- The H.264(intra) code is used to encode the key frames.
- An advanced Motion Compensated Frame Interpolation (MCFI) [30] method for SI creation.
- The correlation noise modelling from [31].

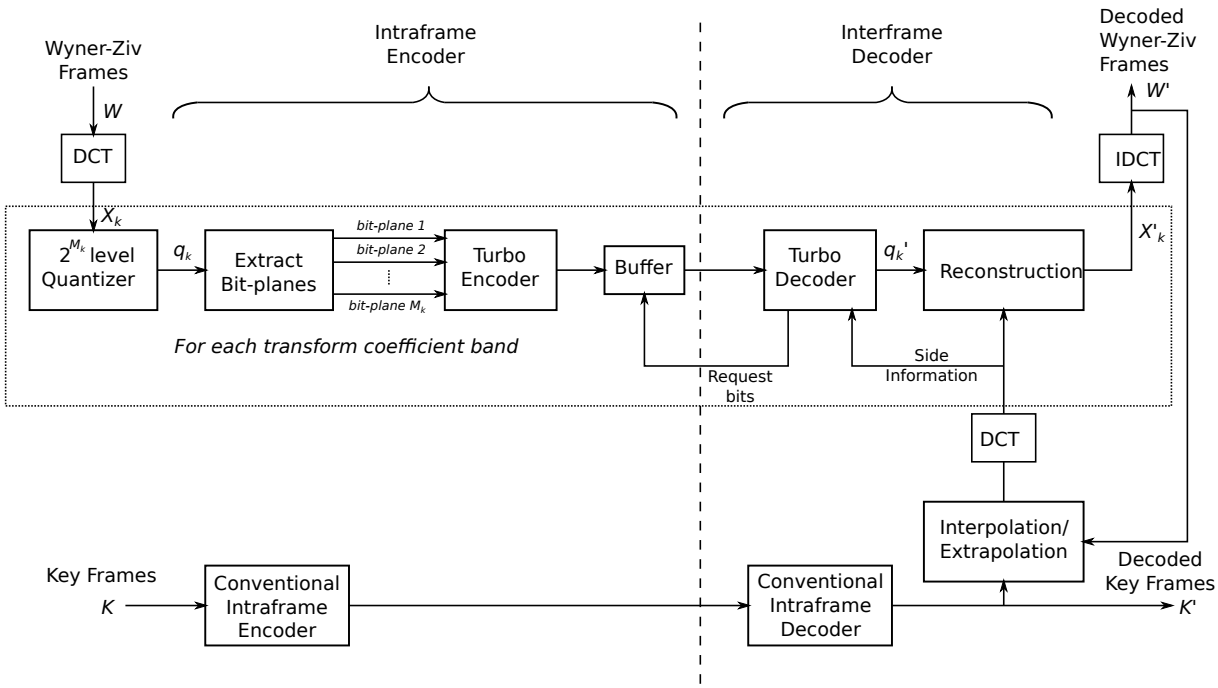

FIGURE 2.2: Block Diagram of the Stanford Architecture.

- MMSE Optimal reconstruction [4].

Due to the extensive reliable results, realistic well defined test conditions and good performance, this codec is used as the benchmark codec for comparisons in the rest of this thesis.

2.3.2 Rate Control and Quantization

In conventional DVC the WZ code is implemented as a quantizer followed by a SW code. In order to analyse the performance of DVC systems over a large range of rate and distortion values, eight profiles were developed. The first seven were developed in [29], and an eighth was added by [31]. Each profile defines the number of bits per symbol assigned to the quantizer, and thus the number of levels in the quantizer, for each subband. Table 2.1 shows the eight quantization profiles. Each profile table is a 4×4 matrix, where the position corresponds the subband of a 4×4 DCT transform.

The rate required by a DVC system can be lower bound by the SW theorem. This bound is accurate assuming that the length of the encoded sequence is infinite and that the chosen code is optimal.

There are two basic methods for rate control that rely on the use of a feedback channel:

1. **Decoder Rate Control** - Decoder rate control is the most accurate but incurs practical

4	3	0	0
3	0	0	0
0	0	0	0
0	0	0	0

(a) Profile 1

5	3	0	0
3	0	0	0
0	0	0	0
0	0	0	0

(b) Profile 2

5	3	2	0
3	2	0	0
2	0	0	0
0	0	0	0

(c) Profile 3

5	4	3	2
4	3	2	0
3	2	0	0
2	0	0	0

(d) Profile 4

5	4	3	2
4	3	2	2
3	2	2	0
2	2	0	0

(e) Profile 5

6	4	3	3
4	3	3	2
3	3	2	2
3	2	2	0

(f) Profile 6

6	5	4	3
5	4	3	2
4	3	2	2
3	2	2	0

(g) Profile 7

7	6	5	4
6	5	4	3
5	4	3	2
4	3	2	0

(h) Profile 8

Table 2.1: These eight tables represent the number of bits used to to quantize each subband for each of the eight quantization profiles.

costs. The basic idea is for the decoder to request bits from the encoder until the sequence is successfully decoded. This will allow for the best possible rate-distortion performance as successful decoding is guaranteed and no redundant bits will be transmitted. The main drawback of the system is the significant number of uses of the feedback channel [25, 32]. Each use of the feedback channel introduces more latency into the system and limits the applications for which the method is appropriate.

2. **Hybrid Rate Control** - Hybrid systems attempt to limit the number of feedback channel uses to reduce the latency. They can only be employed when the feedback path exists and so are still unsuitable for encode and store applications. Methods have been proposed for both transform domain systems [33, 34] as well as pixel domain systems [35].

In order to estimate the required rate, the conditional entropy as described by the SW theorem must be estimated.

2.3.2.1 Conventional bit-plane based conditional entropy estimate

Most DVC systems encode each bitplane of a DCT subband independently using binary codes. As a result, an estimate of the required rate for each bit-plane is required. The standard method for approaching this problem is to consider a binary symmetric channel (BSC) for each bit-plane

[18, 36]. The minimum rate for a given bitplane is calculated as:

$$\begin{aligned} R_b &\geq H_b(\epsilon) \\ &= -\epsilon \log \epsilon - (1-\epsilon)\log(1-\epsilon), \end{aligned}$$

where ϵ is the crossover probability of the virtual BSC. The crossover probability is defined as the probability that $x_b \neq \hat{x}_b$, where

$$\hat{x}_b = \arg\max_{i=0,1} Pr(x_b = i|y, x_{b-1}, \ldots, x_1),$$

and x_b is the b^{th} bit-plane of X. In practice the bitplane entropy is calculated as[18, 37]

$$\begin{aligned} p_n(i) &= \frac{Pr(x_b = i, x_{b-1}, \ldots, x_1|y)}{Pr(x_{b-1}, \ldots, x_1|y)} \\ H_b(p_n) &= -p_n(0)\log p_n(0) - p_n(1)\log p_n(1), \end{aligned}$$

where n is the n^{th} symbol in the subband. The final entropy is then calculated as:

$$H_b = \frac{1}{N}\sum_{WZ^j} H_b(p_n), \tag{2.1}$$

with WZ^j representing all the symbols in the j^{th} sub-band. This method requires N entropy calculations for each bit-plane of each subband.

2.3.2.2 Symbol-wise Conditional Entropy Estimation

It is also possible to calculate the conditional entropy on a symbol based level. This can be useful both for symbol based systems as well as bit-plane based DVC systems. The definition of symbol wise conditional entropy is:

$$\begin{aligned} H(X^Q|Y^Q) &= -\sum_{X^Q}\sum_{Y^Q} P(X^Q, Y^Q)\log P(X^Q|Y^Q) \\ &= -\sum_{X^Q}\sum_{Y^Q} P(X^Q, Y^Q)\log \frac{P(X^Q, Y^Q)}{P(Y^Q)}. \end{aligned} \tag{2.2}$$

This method was used for a pixel domain system [38], that assumed a uniform distribution for X. In general, estimating $H(X^Q|Y^Q)$ directly from Eq. 2.2, requires evaluating $P(X^Q, Y^Q)$ L^2 times, where $L = 2^{q_b}$ and q_b is the number of bits in the quantizer.

2.3.3 Error Correction Coding

An error correction code can correct errors in a signal by adding redundancy. If $\mathbf{x} \in \mathbb{F}^N$ is the signal, $\mathbb{F}$ is the field over which the signal is defined and N is the number of elements. Then the code is defined as the embedding of the N-dimensional signal space into an $N+M$-dimensional encoded signal space. For a linear code, the embedding is linear which means that the encoding can be performed using matrix multiplication:

$$\mathbf{y} = \mathbf{G}\mathbf{x}$$

where $\mathbf{G} \in \mathbb{F}^{(N+M)\times N}$ is the generator matrix of the code and $\mathbf{y} \in \mathbb{F}^{(N+M)}$ is the encoded signal. In this case, the columns of the generator matrix form the basis of the code. Any N non-zero vectors in the code can be used to form this basis, and all vectors in the code are linear combinations of other vectors. As a result, for a give code C, the generator matrix is not unique. Typically there may be benefits to using one version of the generator matrix over another. One example is the systematic form which allows for the data vector, $\mathbf{x}$, to be transmitted directly as part of $\mathbf{y}$. This method may reduce both encoding and decoding complexity in some cases. The parity check matrix of a code, $\mathbf{H} \in \mathbb{F}^{M\times(N+M)}$, is defined by:

$$\mathbf{H}\mathbf{G} = \mathbf{0},$$

and thus for any codeword $\mathbf{y} \in C$, $\mathbf{H}\mathbf{y} = \mathbf{0}$.

In DVC, error correction codes are used as part of the SW code. The signal is encoded using a systematic generator matrix. Only the parity symbols are transmitted. The SI at the decoder is a noisy version of the original signal. The error in the SI estimate is viewed as the noise of a virtual communication channel. The parity symbols are used to fix the errors in the SI in a similar way as for a conventional communication system using an ECC.

In order for the DVC code to approach the rate-distortion bound, the ECC should approach capacity [39]. Most systems use binary codes and encode bit-planes independently. This is because theoretically there is no gain from performing symbol based coding if the correlation noise modelling of the bit-plane based codes take into account the bit-plane dependencies [40]. The advantage of bit-plane based coding is reduced complexity decoding. Initially turbo codes [41] were used, but more recently, LDPC [42] codes have been used. Notably, a class of rate-adaptable LDPC codes (called LDPCA codes [43]) was shown to provide good performance and is used by most systems. Further work to improve rate-adaptable LDPC coding was done in [44].

2.3.3.1 LDPC code design

LDPC codes were first developed by Gallagher [42] and then rediscovered by MacKay and Neal [45]. LDPC codes are part of a more general group of codes called sparse graph codes (which also includes Turbo Codes), which have the property that the code can be represented with a sparse tanner graph. As a result of the sparsity, belief propagation (BP) can be used to efficiently decode LDPC codes.

Designing an LDPC code is equivalent to designing the decoding graph defined by the parity check matrix. In order for BP decoding to be MAP optimal, the graph should be a tree. However, codes with tree graphs are asymptotically bad. Thus, good codes will have graphs with cycles leading to suboptimal decoding. The length of the shortest cycle in the graph is known as the girth of the code. Increasing the girth, and optimizing the degree distribution of the graph can improve the performance. Density Evolution (DE) [46] is a method that has been successful in designing codes by optimising degree distributions.

The problem is that the performance of binary LDPC codes at short frame lengths can be poor. There is not enough "space" in the parity check matrix for all the ones required to maintain a good degree distribution without introducing a lot of short cycles. One method used to improve performance at short frame lengths is non-binary (NB) coding [47]. By defining the code over finite fields, the number of edges in the decoding graph can be reduced yielding larger girth codes with better BP decoding, while maintaining a high effective binary density, yielding a good code. The result is that NB-LDPC codes outperform binary LDPC codes at short to medium frame lengths [47, 48].

There are many methods for designing NB-LDPC codes. Quasi-Cyclic (QC) codes are codes for which each codeword can be shifted by l positions to yield another codeword. The parity check matrices of QC codes are made up of cyclically shifter identity matrices. The advantage of this is that the regular structure at both the encoder and the decoder can be exploited during implementation. This property is specifically useful in DVC where encoding complexity is critical. The structure of the encoding graph allows the QC-LDPC code to be stored with less space than a conventional code and there have also been proposals for fast encoders using shift-registers [49].

2.3.3.2 QC Design Method - Girth 6

In [50], a framework for designing QC-LDPC codes was developed. In this method the cyclic shift of an identity matrix is indicated by a finite field element. If GF(q) is the Galois Field used for construction, let α be the primitive element of GF(q). The powers of α ($\alpha^{-\infty}, \alpha^0, \ldots, \alpha^{q-2}$) will give all the elements of the field. From this, the location vector $\mathbf{z}$ is defined:

$$\mathbf{z}(\alpha^i) = \{z_0, z_1, \ldots, z_{q-2}\},$$

where $z_i = 1$ and $z_k = 0 \; \forall \; k \neq i$. For $\alpha^{-\infty}$, $\mathbf{z}$ is defined as the all-zero vector. Let σ be an element of GF(q). $\mathbf{z}(\sigma\alpha)$ will correspond to the location vector of σ cyclically shifted to the right by one position. A $(q-1) \times (q-1)$ sized matrix $\mathbf{A}(\sigma)$ can then be created by using $\mathbf{z}(\sigma)$, $\mathbf{z}(\sigma\alpha)$, $\mathbf{z}(\sigma\alpha^2)$, …, $\mathbf{z}(\sigma\alpha^{q-2})$ as the rows of the matrix. This matrix is thus the $(q-1) \times (q-1)$-fold matrix expansion of σ which results in a rotated identity matrix. The expansion of the zero element ($\alpha^{-\infty}$) is a $(q-1) \times (q-1)$ all zero matrix. Within this framework, one can design a QC-LDPC code by first creating a matrix consisting of elements in the construction field and then expanding the matrix by using the $(q-1) \times (q-1)$-fold matrix expansion of each element.

In order for a code to have no cycles of length 4 and thus a girth of at least 6, it must meet the row-column (RC) constraint. The RC constraint requires that no two rows have non-zero elements at the same positions in more than one column.

In [50], a method was shown to create a matrix of elements from the construction field such that after expansion, the resultant code is guaranteed to meet the RC constraint. Consider the construction field GF(q). Let m be the largest prime factor of $q-1$, then $cm = q-1$. Let α be a primitive element of $\mathbb{F}_q$ and $\beta = \alpha^c$. Then β is an element of GF(q) of order m (m is the smallest integer such that $\beta^m = 1$). The set $G_m = \{1, \beta, \beta^2, \ldots, \beta^{m-1}\}$ forms a cyclic subgroup of the multiplicative group $G_{q-1} = \{1, \alpha, \ldots, \alpha^{q-1}\}$ of $\mathbb{F}_q$. Let $\mathbf{W}$ be a $m \times m$ matrix over $\mathbb{F}_q$:

$$\mathbf{W} = \begin{pmatrix} \mathbf{w_0} \\ \mathbf{w_1} \\ \vdots \\ \mathbf{w_{m-1}} \end{pmatrix} = \begin{pmatrix} 1 & 1 & \ldots & 1 \\ \beta & \beta^2 & \ldots & \beta^{m-1} \\ \vdots & \vdots & \ddots & \vdots \\ \beta^{m-1} & (\beta^2)^{m-1} & \ldots & (\beta^{m-1})^{m-1} \end{pmatrix}.$$

After the $(q-1) \times (q-1)$-fold matrix expansion, as defined above, the resultant code will have a girth of at least 6. This method, and others based on algebraic geometry and finite fields, was further developed and extended to non-binary LDPC codes in [51].

A simple way to go from a binary code to a non-binary code is to replace each 1 in the expanded matrix with a non-zero element from the field over which the code should be defined.

2.3.3.3 NB-QC-LDPC code design method - Girth 12

A method for designing girth 12 QC-NB-LDPC codes was presented in [52]. The high girth is achieved by making the matrix ultra-sparse. A parity check matrix is ultra-sparse if it has a uniform column weight of 2. Ultra-sparse binary LDPC codes have poor performance. However, it has been shown that the performance of ultra-sparse NB-LDPC codes is excellent and improves with the field size. Let $\mathbf{P}$ be an $m \times m$ permutation matrix defined by:

$$P_{ij} = \begin{cases} 1 & \text{if } i+1 = j \bmod m \\ 0 & \text{otherwise.} \end{cases}$$

Taking powers of $\mathbf{P}$ results in cyclically shifted permutation matrices. $\mathbf{P}^0$ yields the identity matrix. The parity check matrix of a (J,L)-regular QC-LDPC code is made up of $J \times L$ permutation matrices :

$$\mathbf{H} = \begin{pmatrix} \mathbf{P}^{a_{11}} & \dots & \mathbf{P}^{a_{1L}} \\ \vdots & \ddots & \vdots \\ \mathbf{P}^{a_{J1}} & \dots & \mathbf{P}^{a_{JL}} \end{pmatrix}.$$

where a_{ij} is the shift value for each permutation matrix. For ultra-sparse codes $J = 2$ and the rate is controlled with L. It has been shown that the maximum girth for a code with $L \geq 3$ is 12. In [52] this bound is met by taking the top row as identity matrices and using the Hoey sequence for the shift values $\{a_{21}, \dots, a_{2L}\}$. The Hoey sequence is given below for the first 18 numbers:

$$0, 1, 3, 7, 12, 20, 30, 44, 65, 80, 96, 122, 147, 181, 203, 251, 289, 360.$$

Similar to the method above, an NB code is obtained from the binary code by substituting each 1 with an element from the desired code field.

2.3.4 Side Information

The performance of a DVC system depends primarily on the quality of the SI. SI creation is a combination of creating an estimate of the WZ frame and modelling the error between the estimate and the true WZ frame.

2.3.4.1 Wyne-Ziv frame estimation

The SI is an estimate of the WZ frame created from the key frames. If two key frames, before and after the WZ frame, are used then SI creation is an interpolation process. The simplest method for interpolation is to just average the pixels at each position in the two reference frames.

This estimate can be significantly improved by compensating for motion in the video sequence. Motion compensated interpolation (MCI) relies on the process of motion estimation (ME). There are many different methods for performing ME. Typically, image patches from one key frame are matched to image patches of the other key frame. The result is a motion vector (MV) which describes the motion of the patch. Typically the image patches are just small blocks.

An MCI method developed in [30], called motion compensated frame interpolation (MCFI), has been adopted in the DISCOVER codec and forms the basis of the side information creation method in this thesis. The method can be broken into the following steps:

- Filter both the reference frames using a mean filter. A 3×3 mean filter is passed over each frame to remove noise.

- Performing single directional block matching (BM) with large blocks and a modified mean of absolute differences (MAD) metric. The BM algorithm used to estimate the motion vectors is a full search pixel level accuracy algorithm. When comparing two image blocks, of size $M \times M$, related by the motion vector, (di, dj), in frame f_L and f_R, the MAD metric is defined as:

$$MAD(f_L, f_R, (di, dj)) = \frac{1}{M^2} \sum_{(i,j) \in B} |f_L(i, j) - f_R(i + di, j + dj)|$$

  where B is the set of all the pixels in the block being compared. The modified metric favours vectors that are smaller and is defined as:

$$\text{MAD}^*(f_L, f_R, (di, dj)) = \text{MAD}(f_L, f_R, (di, dj)) \times \left(1 + k\sqrt{di^2 + dj^2}\right) \quad (2.3)$$

  where $k = 0.05$ is typically used.

- Perform a directed search using an adaptive search size and smaller blocks.

- Filter the resultant motion vector (MV) field with an adaptive weighted median filter (AWMF) [53, 54].

Other methods that have been developed include mesh based ME [55], 3D recursive search (3DRS) [56, 57], overlapped block motion compensation (OBMC) [8, 58] and optical flow based ME [59, 60].

Implementation details that can affect performance include the size and shape of the blocks being matched. Larger blocks pick up structural details better and are less noisy, but smaller

blocks match up to details better (especially when there is complex motion). Due to the complexity involved there is typically a limited search range for the blocks being matched. Even if complexity is not a consideration, too large a search range can result in noisy motion vectors. Furthermore, when using filters (such as mean and median filters) different filter settings and window sizes will have different results depending on the video sequence being tested. Over-fitting to the test sequence is a known problem for video coding methods.

The performance of MCI based systems tends to decrease with an increase in the GOP size for most test sequences that contain any kind of complex motion. Systems have been proposed that use motion compensated extrapolation (MCE) in an attempt to reduce the delay in the system [61]. These systems generally perform worse than interpolation based systems. However, MCE based systems have smaller degradation in performance for larger GOP sizes and for a large enough GOP may even outperform MCI based systems [62].

2.3.4.2 Correlation Noise Modeling

Since the system proposed in this thesis is a transform domain system, only transform domain correlation noise modelling will be discussed. The noise experienced by each hypothesis is generally modelled using a Laplace distribution. The Laplace distribution provides for a good trade-off between performance and complexity [63, 64]. The distribution of a single co-efficient of the hypothesis, in the DCT domain, is given as

$$f_{Y|X}(y|x) = \frac{\alpha}{2}\exp(-\alpha|y-x|),$$

where y is the hypothesis, and x is the true value of the co-efficient. The only parameter required to be estimated is thus α. The method used by most systems was developed in [31] and estimates a noise parameter for each coefficient at the decoder. Define $f(i,j)$ as the original pixel domain WZ frame and $\hat{f}(i,j)$ as the pixel domain side information hypothesis frame. The noise parameter for the $(v,w)^{\text{th}}$ coefficient of the b^{th} subband, $\alpha_b(v,w)$, is estimated as follows [31]:

$$\hat{\alpha}_b(v,w) = \begin{cases} \hat{\alpha}_b, & [D_b(v,w)]^2 \leq \hat{\sigma}_b^2 \\ \sqrt{\frac{2}{[D_b(v,w)]^2}} & [D_b(v,w)]^2 > \hat{\sigma}_b^2 \end{cases}, \tag{2.4}$$

where

$$r(i,j) = f(i,j) - \hat{f}(i,j),$$
$$\hat{r}(i,j) = \frac{f_F(i+di_F, j+dj_F) - f_B(i+di_B, j+dj_B)}{2},$$

$$\hat{R} = DCT(\hat{r}), \qquad \hat{\sigma}_b^2 = E[\hat{R}_b^2] - E[|\hat{R}_b|]^2,$$

$$\hat{\mu}_b = E[|\hat{R}_b|], \qquad \hat{\alpha}_b = \sqrt{\frac{2}{\hat{\sigma}_b^2}},$$

$$D_b(v, w) = |\hat{R}_b|(v, w) - \hat{\mu}_b.$$

The residual frame, r, represents the true difference between the side information and the WZ frame. Its estimate is represented by $\hat{r}$. The forward and backward motion compensated frames are represented by $f_F(i + di_F, j + dj_F)$ and $f_B(i + di_B, j + dj_B)$ respectively, where (di_F, dj_F) and (di_B, dj_B) are the MVs associated with the pixel at (i, j). The b^{th} sub-band of $\hat{R}$ is given by $\hat{R}_b$.

2.3.5 Reconstruction

For every co-efficient, the output of the ECC decoder is a quantization index. The reconstruction process aims to reduce the quantization noise by using the side information to select the best value inside the bin corresponding to the index. Let L denote the number of quantizer bins and $z_l, l \in \{0, 1, \ldots, L\}$, denote the values of the bin borders. $\Delta = z_{l+1} - z_l$ is the quantization step size. The standard approach for reconstructing a coefficient x using side information y is:

$$\hat{x} = \begin{cases} z_l, & y < z_l \\ y, & y \in [z_l, z_{l+1}) \\ z_{l+1}, & y \geq z_{l+1} \end{cases},$$

where $\hat{x}$ is the reconstructed coefficient and l denotes the quantization index of x. The MMSE optimal approach for reconstructing a coefficient x using side information y and quantization bin $[z_l, z_{l+1})$ was given in [4]:

$$\begin{aligned} \hat{x}^{\text{opt}} &= E[x|x \in [z_l, z_{l+1}), y] \\ &= \frac{\int_{z_l}^{z_{l+1}} x f_{X|Y}(x|y)\,\mathrm{d}x}{\int_{z_l}^{z_{l+1}} f_{X|Y}(x|y)\,\mathrm{d}x}. \end{aligned}$$

2.3.6 Transforms

Transforms improve the performance of lossy compression implemented with quantizers by concentrating the energy of correlated signals into fewer coefficients. In image and video coding the DCT [65] is often used. In DVC the DCT has also proved the most popular [66]. Other transforms such as the wavelet transform have also been proposed for DVC coding. The DCT has however been the most successful due to low encoding complexity and high coding gain. In

this thesis the DCT is also used. The one dimensional type-II DCT can be defined by a $N \times N$ matrix A containing elements a_{jk} where $j, k \in \{0, \ldots, (N-1)\}$:

$$a_{jk} = \alpha_j \cos \frac{\pi(2k+1)j}{2N}$$

where $\alpha_0 = \sqrt{1/N}$ and $\alpha_j = \sqrt{2/N}\ \forall\ j \neq 0$. For a given $N \times N$ subblock of an image, the DCT is applied to each row, and then each column of the block.

2.3.7 Quantizers

DVC systems rely on quantizers to implement lossy compression. The most common quantizer used are simple uniform quantizers with slight modifications. The three main quantizers used are the mid-rise uniform quantizer, the mid-tread uniform quantizer and the dead-zone uniform quantizer.

If we are quantizing the signal $\mathbf{x}, x_i \in \mathbb{R}\ \forall i$, and the range of the quantizer is $A = 2\|\mathbf{x}\|_\infty$ with q bits, then the quantizer bin size $\Delta = A/L$, where $L = 2^q$ is the number of levels in the quantizer. The quantizer function for the mid-rise can be described as:

$$Q_{\mathrm{S}}(x_i) = \left\lfloor \frac{x_i}{\Delta} \right\rfloor + \frac{L}{2}. \tag{2.5}$$

For the mid-tread quantizer, the quantizer function can be described as:

$$Q_{\mathrm{M}}(x_i) = \mathrm{sign}(x_i) \left\lfloor \frac{|x_i|}{\Delta} + \frac{1}{2} \right\rfloor + \frac{L}{2}. \tag{2.6}$$

For the dead-zone quantizer, the bins around zero are grouped together. The quantizer function can be given as:

$$Q_{\mathrm{D}}(x_i) = \mathrm{sign}(x_i) \left\lfloor \frac{|x_i|}{\Delta} \right\rfloor + \frac{L}{2} \tag{2.7}$$

The Δ, parameters for each subband are transmitted along with the encoded sequence. Δ can be represented with 8 bits. As a result, the overhead from transmitting Δ is negligible.

2.4 MULTI-HYPOTHESES DVC

Multi-hypothesis DVC is when more than one SI hypothesis frame is used at the decoder to improve the R-D performance. For transform domain DVC this will correspond to multiple hypothesis for each coefficient in each subband. Assuming the correlation noise model can

be calculated for each individual hypothesis, the question of the MH is how to combine these separate distributions before performing the error correction decoding.

The first attempt to use MH in DVC was by Misra et al. in 2005 [3]. The proposed solution was to average the distributions. If $p[x|y_1]$ is the PMF for coefficient x given the first hypothesis y_1, and $p[x|y_2]$ is the PMF for coefficient x given the second hypothesis y_2, the combined distributions is given as:

$$p[x|y_1, y_2] = \frac{1}{2}\left(p[x|y_1] + p[x|y_2]\right).$$

This method can be easily extended to more hypotheses:

$$p[x|y_1, y_2, \ldots, y_{N_\mathrm{H}}] = \frac{1}{N_\mathrm{H}} \sum_{h=1}^{N_\mathrm{H}} p[x|y_h], \tag{2.8}$$

where N_H is the number of hypotheses. This method will be referred to as averaging in the rest of the thesis.

The problem with simple averaging was that it did not perform very well in general. In 2009, Huang et al. [5] proposed a system that considered averaging with different weights assigned to each hypothesis to improve performance. No method was proposed to calculate the weighting factors on-line. Instead, six different predefined weighting combinations were used to produce six different possible soft inputs. Each input was then separately decoded. As a result the system requires the use of multiple LPDC decoders which increases the decoding complexity six fold. The method is also difficult to extend to more hypotheses, and since the weighting factors are predefined there is a risk that the proposal will not adapt well to new video sequences.

In 2011, Huang et al. improved on the system in [5] by adding a sophisticated optical flow based interpolation hypothesis [6]. This improved the performance, but the method suffers from the same difficulty in terms of complexity and ease of extension to more hypotheses. The scaling factors were also determined experimentally and off-line, possibly affecting the performance of the proposed method on alternative sequences. Rate gains of between 1% and 14% depending on the test sequence were reported as compared to a single hypothesis system.

In 2011, Ascenco et al. [7] proposed a multiple hypothesis system that combined hypotheses using a different approach. By using an augmented graph in the BP decoding process, each hypothesis is independently decoded and the results from the BP decoding are fused together, and the hypotheses are updated. This process is iterated four times. The result is improved performance, but a significant increase in decoder complexity is expected from the use of

multiple decoders. Rate savings of between 0.34% and 4.24% as compared to a single hypothesis method were reported.

2.4.1 Reconstruction

The MMSE optimal reconstruction described in Section 2.3.5 for single hypotheses was extended for multiple hypotheses using averaging [4]:

$$\begin{aligned} \hat{x}^{opt} &= E[x|x \in [z_l, z_l+1), y_1, y_2, ..., y_{N_\mathrm{H}}] \\ &= \frac{\int_{z_l}^{z_{l+1}} u f_{X|Y_1,...,Y_{N_\mathrm{H}}}(x|y_1, y_2, ..., y_{N_\mathrm{H}})\,\mathrm{d}x}{\int_{z_l}^{z_{l+1}} f_{X|Y_1,...,Y_{N_\mathrm{H}}}(x|y_1, y_2, ..., y_H)\,\mathrm{d}x} \\ &= \frac{\sum_{h=1}^{N_\mathrm{H}} \int_{z_l}^{z_{l+1}} \frac{\alpha_h}{2} x \exp(-\alpha_h|x - y_h|)\,\mathrm{d}x}{\sum_{h=1}^{N_\mathrm{H}} \int_{z_l}^{z_{l+1}} \frac{\alpha_h}{2} \exp(-\alpha_h|x - y_h|)\,\mathrm{d}x}. \end{aligned} \tag{2.9}$$

In [4], this equation was simplified and solved in closed form.

2.5 FEEDBACK FREE SYSTEMS

The majority of competitive systems require a feedback path to facilitate rate control and guarantee a decoding success. Methods to remove feedback have been proposed and follow a similar pattern:

- Estimate the required rate at the encoder. This is known as encoder rate control (ERC).
- Attempt to mitigate the visual effects of decoding failures at the decoder.

There have been some notable proposals for feedback suppressed systems. Initially the majority of systems were developed for pixel domain DVC and relied on some kind of off-line training process. In 2005, Artigas and Torres [10] developed a rate control method based on a look-up table obtained through an off-line training stage. In 2007, Morbée et al. [11] also proposed a system based on look up tables with a training stage, and improved upon the method used in the training stage later in the same year [12]. In 2007, Weerakkoddy et al. [13] developed a system relying on predefined constant rates for all WZ frames and improved on this in 2009 [9] with an improved reconstruction method which is applicable to both pixel and transform domain DVC. In 2008, Martinez et al. [14] developed a method to learn the required rate from the SI estimate using machine learning in a training stage. Similarly, a method using neural networks to learn the rate was proposed by Nickaein et. al. in 2012 [15].

The problem with all these methods is that an off-line training based rate estimate is not capable of adapting to the changing spatial and temporal characteristics of a video sequence. The performance of a training based method is reliant on the videos used for training. As a result, there will be over and under estimation of the required rate, resulting in a loss of average performance and an increase in the variance of the image quality. Another problem with these proposals is the testing methodology used. The systems by Artigas and Torres, Morbée et. al and the first by Weerakkoddy et al. assumed perfect key frames at the decoder.

A system similar to [11] but for transform domain DVC was presented in [16], where a multi-mode SI method was used at the encoder. This allowed for the use of intra coding and skip modes to improve performance. While the method was successful in improving performance, the increased use of intra and skip modes in the WZ frame reduced the use of WZ coding. Furthermore, the method was only tested on low motion 30 Hz video sequences which are much easier to compress. The system also relies on a training stage and thus suffers from the same problems described above

A pixel domain system with a structure not relying on a training stage and look-up table was developed by Deligiannes et al. [67]. This system does not use conventional Wyner-Ziv techniques and exploits overlapped block motion estimation with probabilistic compensation (OBMEPC) and SI dependent correlation channel estimation (SID-CE). While this system produced good results at a low encoding complexity, it is perhaps more accurately classified as a low complexity intra code. In 2012, a system by Verbist et al. which built on the work in [67] was presented. This proposal included a hash to improve the motion estimation at the decoder as well as iterative successive refinement decoding to improve the quality of the decoded sequence.

Other than the methods by Weerakkody et al., Sheng et al. and Nickaein et al. already mentioned, there have been a few attempts for transform domain feedback free DVC. The first proposal was by Brites and Pereira [17] in 2007. This proposal relied on comparatively low complexity motion estimation at the encoder to improve the rate estimate, but still suffered from over and under estimation of the rate. The results indicate a loss of 1.2dB relative to a corresponding feedback based system. This proposal is notable for the realistic test conditions used in its evaluation. In 2011, the same authors proposed an improvement to this system in [18] relying on successive refinement decoding and a hash to improve performance. As a result it achieved performance similar to that of the feedback based DISCOVER codec.

A summary of the timeline is provided in Table 2.2. In summary, while several methods

Pixel Domain	Year	Transform Domain
Artigas and Torres [68]	2005	
Morbée et al.[11]	2007	Brites and Pereira[17]
Morbée et al.[12]	2007	
Weerakkody et al. [13]	2007	
Martinez et al.[14]	2008	
Weerakkody et al.[9]	2009	Weerakkody et al. [9]
Deligiannis et al. [67]	2009	
	2010	Sheng et. al[16]
	2011	Brites and Pereira[18]
Verbist et al. [19]	2012	Nickaein et al.[15]

Table 2.2: Summary of feedback free DVC systems

have been proposed, they suffer from either one or a combination of the following problems:

- Using an offline training stage that does not adapt to spatial and temporal variations in the video - [9–16].
- Unrealistic testing conditions that assume perfect key frames at the decoder - [10–13].
- A reliance on intra coding and skip modes - [14, 16].
- Increased encoder complexity due to the use of motion estimation at the encoder - [17, 18].

2.5.1 Hashing and side-information refinements procedures

The methods presented in the previous section with the best performance ([18] and [19]) both rely on the use of a hash and successive refinement decoding. The word "hash" has many meanings in computer science. In the case of DVC, it implies that extra bits are transmitted from the encoder to the decoder to improve the SI creation process. While hashing was not considered in this thesis, it improves the performance in all the systems where it has been implemented and could thus be added to the proposals in this thesis to achieve similar gains.

2.6 PERFORMANCE METRICS

Comparing the systems and methods in this thesis requires several metrics. A video sequence is made up of frames. Comparing two video sequences, for example the original sequence and a decoded version, is done by calculating a distortion metric for each frame.

The objective quality of two frames is measured using the Peak Signal-to-Noise Ratio (PSNR) metric. The PSNR between two discrete signals, $\mathbf{x} = \{x_1, \ldots, x_N\}$ and $\mathbf{y} = \{y_1, \ldots, y_N\}$ is measured in decibels (dB) and defined as:

$$\text{PSNR}(\mathbf{x}, \mathbf{y}) = 10 \log_{10} \frac{P^2}{\frac{1}{N} \sum_{i=1}^{N} (x_i - y_i)^2},$$

where P is the peak value that $\mathbf{x}$ or $\mathbf{y}$ can take. In standard video coding, 8 bit channels are used, thus $P = 255$. While easy to calculate, objective and widely used, the PSNR metric does not always correlate with human perception of distortion. As a result, we also use the SSIM metric [69], which was developed to improve the correlation between objective and subjective quality metrics.

Let $\mathbf{x} = \{x_1, \ldots, x_N\}$ and $\mathbf{y} = \{y_1, \ldots, y_N\}$ be two discrete non-negative signals that correspond to two image windows that are in the same location. Let μ_x, σ_x^2, μ_y and σ_y^2 be the mean and variance of x and y respectively. Let σ_{xy} be the covariance of $\mathbf{x}$ and $\mathbf{y}$. The SSIM metric between two windows of an image is composed of a luminance, contrast and structure component. These are defined as:

$$\begin{aligned} l(\mathbf{x}, \mathbf{y}) &= \frac{2\mu_x\mu_y + C_1}{\mu_x^2 + \mu_y^2 + C_1}, \\ c(\mathbf{x}, \mathbf{y}) &= \frac{2\sigma_x\sigma_y + C_2}{\sigma_x^2 + \sigma_y^2 + C_2}, \\ s(\mathbf{x}, \mathbf{y}) &= \frac{\sigma_{xy} + C_3}{\sigma_x\sigma_y + C_3}, \end{aligned}$$

where C_1, C_2 and C_3 are small constants to improve numerical stability. They are given by:

$$\begin{aligned} C_1 &= (K_1 L)^2, \; K_1 = 0.01 \\ C_2 &= (K_2 L)^2, \; K_2 = 0.03 \\ C_3 &= C_2/2, \end{aligned}$$

where L is the dynamic range of the pixels. The general form of the SSIM is then defined as:

$$\text{SSIM}(\mathbf{x}, \mathbf{y}) = [l(\mathbf{x}, \mathbf{y})]^\alpha [c(\mathbf{x}, \mathbf{y})]^\beta [s(\mathbf{x}, \mathbf{y})]^\gamma$$

The standard form (as used in this thesis) sets $\alpha = \beta = \gamma = 1$. The SSIM metric satisfies the following conditions:

1. symmetry: $\text{SSIM}(\mathbf{x}, \mathbf{y}) = \text{SSIM}(\mathbf{y}, \mathbf{x})$.
2. boundedness: $\text{SSIM}(\mathbf{x}, \mathbf{y}) \leq 1$
3. unique maximum: $\text{SSIM}(\mathbf{x}, \mathbf{y}) = 1$ if and only if $\mathbf{x} = \mathbf{y}$.

The SSIM metric is applied to a window of the two images being compared. This window is slid across the image in a pixel by pixel manner. The final average SSIM metric is just an average of all the SSIM values calculated in the sliding approach.

While PSNR and SSIM metrics are useful for images, they do not represent any of the dynamic aspects of visual quality. Two signals with the same average PSNR and SSIM values may have completely different perceptual qualities.

For this reason, the variance (or standard deviation) of the PSNR metric is also considered. Signals with greater quality fluctuation will typically score worse than signals with a more constant quality. Improving the average quality while increasing the variance may be counter productive. Furthermore, for systems where the average quality is similar, the one with the lower variance will be perceived as being of a higher quality.

Typically, the SSIM and PSNR metrics are plotted for a variety of rate values. The result is a Rate-Distortion (R-D) curve that provides insight into the performance of the system in different scenarios. Comparisons between systems are facilitated by using the Bjontegaardt [70] metrics. These metrics are calculated from the average PSNR results and create a single value which may be used to describe the difference between two systems. There are two Bjontegaard delta (BD) metrics, the percentage rate difference (BD-Rate) and the differential PSNR (BD-PSNR) metric. The BD metrics are calculated by considering 4 R-D points for each of the two systems begin compared. The process is as follows:

1. Convert the rate to the log domain.
2. Fit a third order polynomial through the points two yield two functions.
3. Integrating the functions in the region bounded by the points.
4. Calculate the difference between the integrals of the two systems.
5. Average the result

If $(\mathbf{r}_1, \mathbf{d}_1)$ are the set of R-D points for the first system then we first calculate $\mathbf{lr}_1 = \ln(\mathbf{r}_1)$. $D_1(R)$ is the curve fit through $(\mathbf{lr}_1, \mathbf{d}_1)$ where distortion is considered a function of the logarithmic rate. Similarly $(\mathbf{r}_2, \mathbf{d}_2)$ are the R-D points for the second system and $D_2(R)$ is the curve fit through $(\mathbf{lr}_2, \mathbf{d}_2)$ where $\mathbf{lr}_2 = \ln(\mathbf{r}_2)$. The BD-PSNR metric is defined as:

$$\text{BD-PSNR} = \frac{\int_{R_1}^{R_2} D_1(R)dR - \int_{R_1}^{R_2} D_1(R)dR}{R_2 - R_1}$$

where

$$\begin{aligned} R_1 &= \max\left(\min(\mathbf{lr}_1), \min(\mathbf{lr}_2)\right), \\ R_2 &= \min\left(\max(\mathbf{lr}_1), \max(\mathbf{lr}_2)\right), \end{aligned}$$

The BD-PSNR metric has the same units as the PSNR metric - decibels.

Curves can also be fit through the R-D points considering rate to be a function of distortion, to yield $R_1(D)$ and $R_2(D)$ for the first and second systems respectively. The BD-Rate metric is then calculated as:

$$\text{BD-Rate} = 100\left(\exp\left[\frac{\int_{D_1}^{D_2} R_1(D)dD - \int_{D_1}^{D_2} R_1(D)dD}{D_2 - D_1}\right] - 1\right)$$

where

$$\begin{aligned} D_1 &= \max\left(\min(\mathbf{d}_1), \min(\mathbf{d}_2)\right), \\ D_2 &= \min\left(\max(\mathbf{d}_1), \max(\mathbf{d}_2)\right). \end{aligned}$$

The unit of the BD-Rate metric is percentage.

The BD metrics are calculated in two ways. One which considers the full range of the results set, and one which considers the middle of the curve. It should be noted that the BD metrics should not be considered in isolation but only as a supplement to a full R-D curve. The reason, is that the BD metrics may give nonsensical results in some scenarios. For example, when the curves cross over one-another or if they exists in different PSNR or rate regions.

Complexity is also considered as a measure in this thesis. Complexity is compared by counting operations. Asymptotic order of complexity is considered when appropriate to evaluate the scalability of the different algorithms. Execution time experiments are also performed to provide some comparative measure of complexity. Execution time experiments are imperfect when considering algorithms since implementation optimizations such as compiler and programming optimization can create significant differences in run time. Nonetheless, these experiments provide insight and help to provide support for the proposals made in this thesis.

Chapter Three

Multi-hypothesis Distributed Video Coding

Given a specific distortion aim characterised by a quantizer, the rate required in DVC is controlled by the conditional entropy of the source given the side information, $H(X^Q|Y^Q)$. Improving R-D performance requires this value to be decreased. In other words, the closer the side information is to the original frame, the fewer bits are required to achieve a desired distortion. The difficulty lies in creating an estimate of the original frame at the decoder from the key frames without access to the original frame. Conventional video coding is not constrained in this way since the estimate is created at the encoder, which is partly why it performs better than DVC.

The main method for creating the SI in DVC is to interpolate a frame using motion estimation and compensation between the two key frames. This presents two difficulties. Firstly, as the GOP size increases, the quality of the interpolated frame decreases. This effectively limits the GOP size and as a consequence limits the reduction in the encoding complexity due to DVC as well as the average R-D performance. Secondly, without access to the original frame, motion estimated compensation methods struggle. There are many different MCE methods with varying levels of complexity and sophistication (as discussed in Section 2.3.4.1). Each of these methods typically makes assumptions on the type of motion that is likely to occur in a video sequence. As a result, they tend to work well in different scenarios. Using the most appropriate method at each point in time would result in improved performance, however, knowing which method will work well for any given sequence before decoding is difficult.

In DVC terminology the SI frame is known as a hypothesis. Using different methods to

produce SI frames results in multiple hypotheses. In this chapter, a method is proposed to combine multiple hypotheses. The philosophy behind the proposed method is to provide the decoding algorithm with as much information as possible.

This chapter will begin with a summarised description of the proposed system followed by a description of the main contribution. Next, the implementation details and smaller contributions are described. The complexity of the encoder and decoder is then discussed before some results and analyses are given.

3.1 SYSTEM DESCRIPTION

Fig. 3.1 shows a block diagram of the proposed system. A summary of the differences between the proposed system and the DISCOVER codec is provided in Table 3.1.

Table 3.1: A summary of the differences between the DISCOVER codec and the system proposed in this chapter.

DISCOVER	Proposed System
Bit-plane based	Symbol based
Binary LDPC/ Turbo Codes	Non-binary LDPC
Single Hypothesis	Multi-hypothesis
MCI only	MCI and MCE

Incoming frames are split into key frames, which are H.264 intra coded, and WZ frames.

Let the original pixel-domain WZ frame, at time t in the GOP, be referred to as $f[t]$. $f[t](i, j)$ will refer to the pixel at position (i, j). The system will be examined over varying GOP sizes. The GOP size (N_G) refers to the number of WZ frames in between the key frames plus one. $f[t-1]$ will refer to the previous frame, whether it be a key frame or a previously decoded WZ frame. The last key frame from the previous GOP will be referred to as $f[0]$ and the key frame at the end of the current GOP as $f[N_G]$.

3.1.1 Encoder

The WZ frame is transformed with the block based DCT to yield:

$$F[t] = \text{DCT}(f[t]).$$

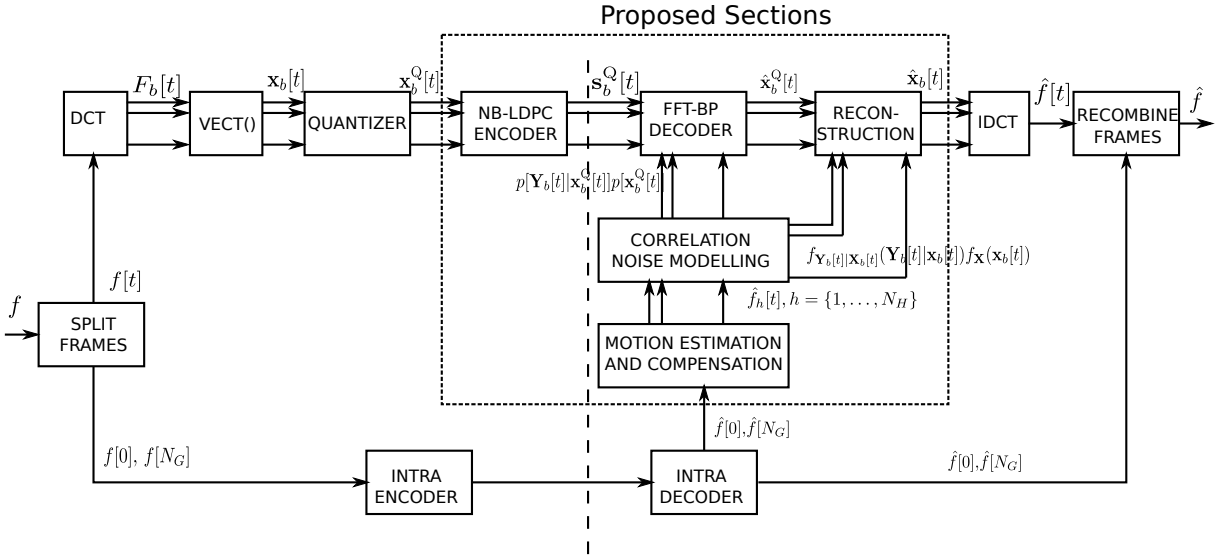

FIGURE 3.1: Diagram of the proposed Multi-hypothesis system

The b^{th} subband of the frame will be referred to as $F_b[t]$, and the coefficient corresponding to the $(v, w)^{th}$ subblock will be referred to as $F_b[t](v, w)$. Let $\mathbf{x}_b[t] = \text{vect}(F_b[t])$ be a $N \times 1$ vectorized version of the b^{th} subband.

Each subband is quantized. The DC subband ($b = 0$) is encoded using a mid-rise quantizer as described in Eq. (2.5), while the AC coefficients ($b \neq 0$) are encoded using a dead-zone quantizer as described in Eq. (2.7), where the size of the dead-zone around 0 is equal to twice the step size for the rest of the quantizer. The quantized subband will be referred to as:

$$\mathbf{x}_b^{\mathrm{Q}}[t] \quad = \quad Q_b(\mathbf{x}_b[t]),$$

where $Q_b(\cdot)$ is the quantizer for the b^{th} subband. The number of bits assigned to the quantizer is given by the quantization matrix and is related to the required minimum distortion. If q_b represents the number of bits assigned to $Q_b(\cdot)$, then $\mathbf{x}_b^{\mathrm{Q}}[t]$ will have elements in $\{0, 1, \ldots, L\}$, where $L = 2^{q_b} - 1$. The quantization matrix is chosen based on the required quality. For this paper, eight quantization profiles taken from [71] and [29] are used in conjunction with a 4 by 4 DCT as described in Section 2.3.7.

For notational brevity we will now drop the time indice and assume that each WZ frame in the GOP is treated in a similar manner. Each quantizer subband is then encoded using an NB-QC-LDPC code. The code is defined over a finite field $\mathbb{F} = \text{GF}(2^{q_b})$, which is a Galois field with 2^{q_b} elements. Before encoding, the elements of $\mathbf{x}_b^{\mathrm{Q}}$ are mapped from the quantization indices to elements in $\text{GF}(2^{q_b})$. Since only parity symbols are transmitted, the code must be in systematic form. Given the size $(N + M) \times N$ generator matrix used for subband b, $\mathbf{G}_b$, the

conventional ECC encoding is calculated as:

$$\begin{aligned}
\mathbf{G}_b &= \begin{bmatrix} \mathbf{I}_N \\ \mathbf{P}_b \end{bmatrix} \\
\mathbf{p}_b &= \mathbf{G}_b \mathbf{x}_b^{\mathrm{Q}} \\
&= \begin{bmatrix} \mathbf{x}_b^{\mathrm{Q}} \\ \mathbf{P}_b \mathbf{x}_b^{\mathrm{Q}} \end{bmatrix} \\
&= \begin{bmatrix} \mathbf{x}_b^{\mathrm{Q}} \\ \mathbf{s}_b^{\mathrm{Q}} \end{bmatrix},
\end{aligned}$$

where $\mathbf{I}_N$ is the $N \times N$ identity matrix. Only $\mathbf{s}_b^{\mathrm{Q}} \in \mathbb{F}^M$ is transmitted. $\mathbf{P}_b$ is a $M \times N$ parity matrix. In conventional ECC literature, the parity check matrix $\mathbf{H}_b$ is defined as the matrix that solves $\mathbf{H}_b \mathbf{G}_b = 0$. If $\mathbf{H}_b$ is in systematic form, $H_b = [\mathbf{P}_b | \mathbf{I}_M]$. Since only the parity symbols are transmitted, in DVC literature $\mathbf{P}_b$ is referred to as $\mathbf{H}_b$. For the purpose of this chapter, the rate is assumed to be known, and the design of the rate estimation algorithm will be considered as part of the feedback suppressed system. As a result, the system is comparable and functionally similar to feedback based rate control systems which also achieve near optimal rates.

3.1.2 Decoder

At the decoder, the key frames and possibly other decoded frames are used to create an estimate of f. In the multi-hypothesis case more than one estimate of f is created. The h^{th} hypothesis will be denoted by $\hat{f}_h$. Each hypothesis is transformed to produce:

$$\hat{F}_h = \mathrm{DCT}(\hat{f}_h).$$

If the error for each hypothesis is given as:

$$r_h = f - \hat{f}_h,$$

we calculate an estimate $\hat{r}_h$. The method used to calculate $\hat{r}_h$ will differ for each hypothesis and depend on the method used to produce the hypothesis. From the error estimates, noise parameters are calculated, at a co-efficient level using Eq. (2.4), assuming a Laplace noise distribution. In order to improve the relative quality of the noise estimation for the different hypotheses, a motion adaptive scaling factor is used to scale the α values of the key frame and motion based hypotheses.

For each subband, the error estimates and the N_{H} hypotheses are combined with the received parity information and decoded using the NB-FFT-BP algorithm [72]. The subband indice is now dropped for notational brevity. The BP decoding equation can be given as:

$$(\hat{x}_n^{\mathrm{Q}})^{\mathrm{MAP}} = \arg\max_{x_n^{\mathrm{Q}} \in \mathrm{GF}(2^q)} p[x_n^{\mathbf{Q}} | \mathbf{Y}, \mathbf{s}^{\mathrm{Q}}],$$

where n indicates the n^{th} coefficient in a subband with N coefficients. $\mathbf{Y}$ is an $N_{\mathrm{H}} \times N$ matrix with entries y_{hn} where the h^{th} row represents the N symbols of the h^{th} hypothesis. After decoding, $(\hat{x}_n^{\mathrm{Q}})^{MAP}$ will be an element in GF(2^q) which maps to a quantization bin defined by the relevant quantizer. Reconstruction is the process of selecting the best value within this bin for the final estimate of x_n. Optimal reconstruction is given as:

$$\hat{x}_n^{\mathrm{opt}} = E[x | x \in Q^{-1}((\hat{x}_k^{\mathrm{Q}})^{\mathrm{MAP}}), \mathbf{y}_k].$$

After reconstruction is performed for each band, $\hat{\mathbf{x}}^{\mathrm{opt}}$ is recompiled to produce $\hat{F}$, and then the final pixel domain output frame $\hat{f}$ is calculated:

$$\hat{f} = IDCT(\hat{F}).$$

3.2 MAIN CONTRIBUTIONS

The main contribution of this chapter is the method for combining multiple hypotheses. The proposed method also requires a new reconstruction equation which is derived in this section.

3.2.1 Multiple Hypotheses

The multi-hypothesis fusion equation is derived from the expression for the BP decoding. Let the encoded quantized data source be represented by the random variable X. Let the h^{th} side information hypothesis be represented by the random variable Y_h, $h \in \{1, \ldots, N_{\mathrm{H}}\}$ where N_{H} is the number of hypotheses. We wish to decode a specific set of encoded source symbols ($\{x_1^{\mathrm{Q}}, \ldots, x_N^{\mathrm{Q}}\}$) given the set of received symbols represented by the matrix $\mathbf{Y}$ and the M parity

symbols $\mathbf{s}^{\mathrm{Q}}$. The MAP decoded value can be given as:

$$\begin{aligned}(\hat{x}_n^{\mathrm{Q}})^{MAP} &= \underset{x_n^{\mathrm{Q}} \in \mathrm{GF}(2^q)}{\arg\max}\, p[x_n^{\mathrm{Q}}|\mathbf{Y}, \mathbf{s}^{\mathrm{Q}}] \\ &= \underset{x_n^{\mathrm{Q}} \in \mathrm{GF}(2^q)}{\arg\max} \sum_{\tilde{x}_n^{\mathrm{Q}}} p[\mathbf{x}^{\mathrm{Q}}|\mathbf{Y}, \mathbf{s}^{\mathrm{Q}}] \\ &= \underset{x_n^{\mathrm{Q}} \in \mathrm{GF}(2^q)}{\arg\max} \sum_{\tilde{x}_n^{\mathrm{Q}}} p[\mathbf{Y}, \mathbf{s}^{\mathrm{Q}}|\mathbf{x}^{\mathrm{Q}}] p[\mathbf{x}^{\mathrm{Q}}] \\ &= \underset{x_n^{\mathrm{Q}} \in \mathrm{GF}(2^q)}{\arg\max} \sum_{\tilde{x}_n} \left[\prod_{j=1}^{N} p[\mathbf{y}_j|x_j^{\mathrm{Q}}] p[x_j^{\mathrm{Q}}] \mathbb{1}_{\{\mathbf{H}\mathbf{x}^{\mathrm{Q}} = \mathbf{s}^{\mathrm{Q}}\}} \right],\end{aligned}$$

where $\mathbf{y}_j$ refers to the j^{th} column of $\mathbf{Y}$ and represents the N_{H} hypotheses for the j^{th} code symbol. $\mathbb{1}_{\mathrm{condition}}$ is the indicator function which equals one if its condition is met and zero otherwise. $\sum_{\tilde{x}_n^{\mathrm{Q}}}$ indicates the summation over all x_k^{Q} where $k \neq n$. The last step assumes a memoryless channel. This expression will now factor in the usual way to yield the update equations for the belief propagation decoding. The input message to the j^{th} variable node in the decoding graph will be:

$$\mu(j) = C \cdot p[\mathbf{y}_j|x_j^{\mathrm{Q}}] p[x_j^{\mathrm{Q}}]$$

where C is a normalizing constant. If we now assume independent hypotheses, this can be further reduced to:

$$\mu(j) = C \cdot p[x_j^{\mathrm{Q}}] \prod_{h}^{N_{\mathrm{H}}} p[y_{hj}|x_j^{\mathrm{Q}}]. \tag{3.1}$$

The different hypotheses are thus combined by multiplying their PMFs together. This allows the system to include extra hypotheses without increasing the decoding complexity of the belief propagation decoding step. The noise for each hypothesis is assumed to be Laplace distributed, and the noise parameter is calculated using (2.4). The PMF used above is calculated as:

$$p[y_{hj}|x_j^{\mathrm{Q}} = l] = \int_{z_l}^{z_{l+1}} f_{Y_h|X}(y_{hj}|x_j)\, \mathrm{d}x_j.$$

where $[z_l, z_{l+1}) = Q^{-1}(x^{\mathrm{Q}_j} = l)$ is the range of the l^{th} quantization bin. The accuracy of the independence assumption will affect the performance, and guide the selection of appropriate hypotheses. If the selected hypotheses are not independent enough, it will degrade the performance of the system.

The method presented here differs from the conventional averaging approach described in Section 2.4. The proposed method also differs from the methods presented in [5, 6] and

[7], since it does not rely on fixed weighting factors and does not increase the BP decoding complexity. While decoding complexity is typically not considered a limitation in DVC systems it is still important to keep it as low as possible. The method can also easily be extended to any number of hypotheses.

3.2.2 Reconstruction

Since a new method for combining hypotheses is used, a re-derivation of the reconstruction formula is required. Given the decoded value $\hat{x}_n^{\mathrm{Q}}$, the corresponding bin range $Q^{-1}(\hat{x}_n^{\mathrm{Q}}) = [z_l, z_{l+1})$ and the side information hypotheses, $\mathbf{y}_n$, for each coefficient, $\hat{x}_n$, the reconstruction can be calculated as (2.9):

$$\begin{aligned}
\hat{x}_n^{opt} &= \frac{\int_{z_l}^{z_{l+1}} x_n f_{X|\mathbf{Y}}(x_n|\mathbf{y}_n)\,\mathrm{d}x_n}{\int_{z_l}^{z_{l+1}} f_{X|\mathbf{Y}}(x_n|\mathbf{y}_n)\,\mathrm{d}x_n} \\
&= \frac{\int_{z_l}^{z_{l+1}} x_n f_{\mathbf{Y}|X}(\mathbf{y}_n|x_n) f_X(x_n)\,\mathrm{d}x_n}{\int_{z_l}^{z_{l+1}} f_{\mathbf{Y}|X}(\mathbf{y}_n|x_n) f_X(x_n)\,\mathrm{d}x_n} \\
&= \frac{\int_{z_l}^{z_{l+1}} x_n f_X(x_n) \prod_{h=1}^{N_\mathrm{H}} f_{Y_h|X}(y_{hn}|x_n)\,\mathrm{d}x_n}{\int_{z_l}^{z_{l+1}} f_X(x_n) \prod_{h=1}^{N_\mathrm{H}} f_{Y_h|X}(y_{hn}|x_n)\mathrm{d}x_n}.
\end{aligned}$$

Assuming a uniform prior, one could also cancel out the $f_X(x_n)$ priors. However, in this chapter, the output of the FFT-BP decoder is used as a prior. The PMF output of the FFT-BP decoder is mapped to a Laplace distributed PDF by calculating the mean and alpha parameter. For the sake of notational brevity, this will be indicated as $f_X(x_n) = f_{Y_{N_\mathrm{H}+1}|X}(y_{N_\mathrm{H}+1}|x_n)$, which allows the prior to be incorporated into the product term as the $(N_\mathrm{H}+1)^{th}$ hypothesis. Dropping the coefficient indice (n) and replacing each hypothesis with its Laplace PDF yields:

$$\begin{aligned}
\hat{x}^{opt} &= \frac{\int_{z_l}^{z_{l+1}} x \prod_{h=1}^{N_\mathrm{H}+1} \frac{\alpha_h}{2} \exp(-\alpha_h|x-y_h|)\,\mathrm{d}x}{\int_{z_l}^{z_{l+1}} \prod_{h=1}^{N_\mathrm{H}+1} \frac{\alpha_h}{2} \exp(-\alpha_h|x-y_h|)\,\mathrm{d}x} \\
&= \frac{\int_{z_l}^{z_{l+1}} x \exp\left(\sum_{h=1}^{N_\mathrm{H}+1} -\alpha_h|x-y_h|\right)\mathrm{d}x}{\int_{z_l}^{z_{l+1}} \exp\left(\sum_{h=1}^{N_\mathrm{H}+1} -\alpha_h|x-y_h|\right)\mathrm{d}x}.
\end{aligned}$$

In order to remove the absolute value signs and to yield a closed form expression, the integral must be subdivided into smaller ranges along which the signs of $(x-y_h)$ stay constant. We will define N_S ranges, where N_S equals the number of y_h inside the range $\{z_l, z_{l+1}\}$ plus one, and is limited by $N_\mathrm{S} \leq N_\mathrm{H}+2$. The boundaries of the ranges will be denoted by $q_0, q_1, \ldots, q_{N_\mathrm{S}}$. q_0 will equal z_l and q_{N_S} will equal z_{l+1}. The other boundaries will equal the values of the y_h that fall in

the range between z_l and z_{l+1}. For each range, s, we will select from the $N_{\mathbf{H}}+1$ hypotheses those hypotheses for which $(x - y_h) \geq 0$, and place them in a set $\mathbf{S}(s)$. All the remaining hypotheses are placed in the complement set $\mathbf{S}'(s)$. The expression above can then be written as:

$$\begin{aligned} \hat{x}_{\text{opt}} &= \frac{\sum_{s=0}^{N_{\text{S}}-1} \int_{q_s}^{q_{s+1}} x \exp(\lambda_s x - \gamma_s)\,\mathrm{d}x}{\sum_{s=0}^{N_{\text{S}}-1} \int_{q_s}^{q_{s+1}} \exp(\lambda_s x - \gamma_s)\,\mathrm{d}x} \\ &= \frac{\sum_{s=0}^{N_{\text{S}}-1} \left.\frac{(\lambda_s x-1)\exp(\lambda_s x-\gamma_s)}{\lambda_s^2}\right|_{q_s}^{q_{s+1}}}{\sum_{s=0}^{N_{\text{S}}-1} \left.\frac{\exp(\lambda_s x-\gamma_s)}{\lambda_s}\right|_{q_s}^{q_{s+1}}}, \end{aligned} \tag{3.2}$$

where λ_s and γ_s are defined as:

$$\begin{aligned} \lambda_s &= \sum_{h=1, h\in\mathbf{S}'(s)}^{N_{\text{H}}+1} \alpha_h - \sum_{h=1, h\in\mathbf{S}(s)}^{N_{\text{H}}+1} \alpha_h, \\ \gamma_s &= \sum_{h=1, h\in\mathbf{S}'(s)}^{N_{\text{H}}+1} \alpha_h y_h - \sum_{h=1, h\in\mathbf{S}(s)}^{N_{\text{H}}+1} \alpha_h y_h. \end{aligned}$$

When $\lambda_s = 0$, the integrals in Eq. (3.2) have different solutions which should be used instead:

$$\begin{aligned} \int_{q_s}^{q_{s+1}} x \exp(\lambda_s x - \gamma_s)\,\mathrm{d}x &= \frac{1}{2}\exp(-\gamma_s)x^2\Big|_{q_s}^{q_{s+1}} \\ \int_{q_s}^{q_{s+1}} \exp(\lambda_s x - \gamma_s)\,\mathrm{d}x &= \exp(-\gamma_k)x\Big|_{q_s}^{q_{s+1}} \end{aligned}$$

3.3 IMPLEMENTATION DETAILS AND SECONDARY CONTRIBUTIONS

The specific implementation details of the subsystems are described in this section. This includes a description of how the SI hypotheses are created and correlation noise model is estimated. Several smaller contributions are discussed in this section. This includes the use of NB-LDPC codes, the use of an adaptive scaling factor in the noise estimation process, as well as the use of extrapolation to improve the performance over larger GOP sizes.

3.3.1 Non-binary Codes

Due to there being theoretically no gain from performing symbol based coding if the correlation noise modelling of the bit-plane based codes take into account the bit-plane dependencies [40], most DVC systems are designed using bit-plane based codes.

However, LDPC codes are sub-optimal due to cycles in the decoding graph. At small code lengths this problem is especially pronounced. NB-LDPC codes however achieve good performance for much sparser graphs leading to improved performance at small code lengths. Thus, it is expected that using NB-LDPC codes for DVC, which operates at short code lengths, will yield improved performance.

The general class of QC-LDPC codes is known to hold two key advantages compared to random LDPC codes: reduced storage requirement at the encoder, and reduced encoding complexity [73]. For these reasons, NB-QC-LDPC codes are used in this system.

The design methods used were developed in [52] and [50, 51] and described in Section 2.3.3. The code construction parameters were optimized experimentally. Codes were designed with a parity length (M) starting at 70 symbols and increasing in increments of 10 symbols over Galois fields GF(2^2), GF(2^3), GF(2^4), GF(2^5), GF(2^6), GF(2^7) and GF(2^8). The code length (N) was fixed at 1584, which corresponds to a QCIF resolution video using a 4×4 DCT. The specific code used for a sub-band will depend on the required rate and the number of bits to which the sub-band has been quantized (required picture quality). The codes from [52] are girth-12 codes and will be referred to as the G12 code. The codes from [50] and [51] are girth-6 codes that are constructed using Galois fields. These codes will be referred to as C'n' codes, where 'n' refers to the exponent of the Galois field used for construction. For example, $C6$ was constructed using GF(2^6). Table 3.2 shows the code designs and the scenarios in which they were used.

Table 3.2: The code construction methods used for different scenarios.

Parity	Galois field size						
Length(M)	4	8	16	32	64	128	256
70 – 120	$C5$	$C5$	$C5$	$C5$	$C5$	$C5$	$C5$
130 – 250	$C6$	$C6$	$C6$	$C6$	$C6$	$C6$	$C6$
260 – 300	$C7$	$C7$	$C7$	$C7$	$C7$	$C7$	$C7$
310 – 400	$C7$	$C7$	$C7$	G12	G12	G12	G12
410 – 1000	$C7$	$C7$	G12	G12	G12	G12	G12

The number of parity symbols (column 1) corresponds to the number of transmitted symbols. The column headings indicate the Galois field over which the code is defined. For all the codes, the non-zero elements were selected at random from the code field. At the decoder, a NB-FFT-BP decoder [72] is employed since it allows for only a linear increase in complexity

with Galois field size.

3.3.2 Hypothesis selection

Each hypothesis should be as independent as possible as per the assumption made in the derivation of the joint density. For this thesis, five hypotheses are used. These hypotheses are two reference frames, $f[t-1]$ and $f[N_G]$; an MCI based frame using small blocks, $\hat{f}_S$, as well as an MCI based frame using bigger blocks, $\hat{f}_B$.

While interpolation yields accurate side information when the GOP size is small, for larger GOP sizes the MCI based SI can become very poor. The quality of extrapolation based SI on the other hand is primarily dependent on the temporal sampling rate, not the GOP size. To take advantage of this, an MCE based estimate, $\hat{f}_E$, is also used.

More hypotheses can be added, though it will require different or unique methods of estimation to ensure some level of independence. For example, by adding the same optical flow based interpolated hypothesis described in [6], the gains achieved in [6] could also be applied to the system in this chapter.

3.3.3 Motion Compensated Interpolation

The method used to perform MCI is a variant of the one described in section 2.3.4.1 [30]. The process starts by mean filtering the key frames. The filtered key frames are then used for bi-directional block matching using large blocks and the modified SAD metric in [30] as provided in Eq. (2.3). These vectors are then used as starting points for a block matching approach using smaller blocks with a fixed search range. The MV field generated with large blocks is then quadrupled in density (oversampled by a factor of two in both directions), before being passed through a median filter. By increasing the density, we allow the median filter to produce smoother edges and more precise vectors. The MV field generated with smaller blocks is then also quadrupled in density before being passed though a median filter. These two MV fields are then used to create two MCI hypotheses by interpolating from the original key frames. The exact parameters used are presented in Table 6.1

3.3.4 Motion Compensated Extrapolation

The method used to produce the extrapolated frame is an adaptation of the method normally used to produce the interpolated frame. If the WZ frame being estimated is $f[t]$, then the two frames being used for extrapolation ($f[t-1]$ and $f[t-2]$) are mean filtered to reduce the effect of noise. A block matching ME algorithm is then used to match subblocks in the filtered version of $f[t-1]$ with blocks in the filtered version of $f[t-2]$. This step will be referred to as central frame extrapolation (Figure 3.2).

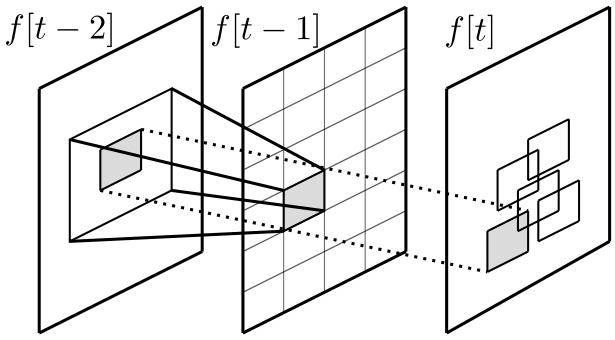

FIGURE 3.2: Central frame extrapolation

The resultant MV field is then median filtered [53] to remove noise. The smoothed MV field is projected onto the frame being constructed ($\hat{f}_{\mathrm{E}}[t]$). Due to the extrapolation, some of the projected blocks will overlap, and some areas of $\hat{f}_{\mathrm{E}}[t]$ will have no assigned MVs (holes). In the overlapped areas, the MV with the best metric is chosen, and for holes, the zero vector is assigned. Frame $\hat{f}_{\mathrm{E}}[t]$ is then broken up into subblocks. For each subblock there may be more than one MV that has been assigned. The vector with the best metric is chosen. These vectors are then refined by performing a backwards looking block matching algorithm over a very small range centred around the chosen vector (Figure 3.3).

The resultant MV field is then smoothed with a standard median vector filter[53]. The completed MV field is used to produce a pixel domain estimate by combining the motion compensated versions of $f[t-1]$ and $f[t-2]$. One method to produce this combination would be to linearly extrapolate the value of the pixel. This method is not very robust as the extrapolated value could grow very large. Instead, the linearly extrapolated value is averaged with the value

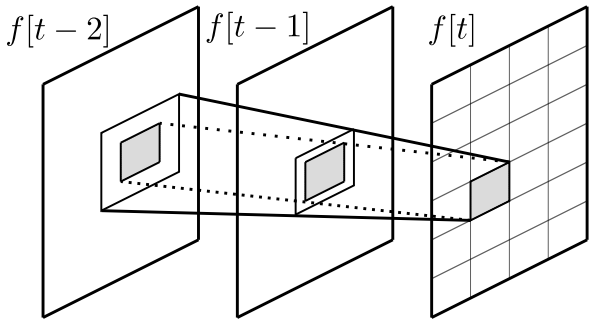

FIGURE 3.3: Backward Looking Extrapolation

from $f[t-1]$ to produce a less extreme extrapolation. The resultant equation is:

$$\begin{aligned}\hat{f}_{\mathrm{E}}[t](i,j) &= \frac{3}{2}f[t-1](i+\mathrm{d}i_{\mathrm{E}}, j+\mathrm{d}j_{\mathrm{E}}) \\ &\quad -\frac{1}{2}f[t-2](i+2\mathrm{d}i_{\mathrm{E}}, j+2\mathrm{d}j_{\mathrm{E}}),\end{aligned}$$

where $(\mathrm{d}x_{\mathrm{E}}, \mathrm{d}y_{\mathrm{E}})$ is the MV associated with pixel (i, j). Due to the backward looking extrapolation procedure, there are no overlapped or uncovered areas. The specific parameters used in the MCE process are discussed in the section 3.4.1 and can be seen in Table 3.3.

3.3.5 Residual frame estimation

For the multi-hypothesis system, a residual frame estimate is required for each hypothesis. For the MCI hypotheses, $\hat{f}_{\mathrm{S}}$ and $\hat{f}_{\mathrm{B}}$, the residual estimates, $\hat{r}_{\mathrm{S}}$ and $\hat{r}_{\mathrm{B}}$ respectively, are the difference between the motion compensated frames used to perform the interpolation. For frame t in the GOP this is given by:

$$\begin{aligned}\hat{r}_{\mathrm{B}}(i,j) &= f[t-1](i+\mathrm{d}i_{\mathrm{I_B}}, j+\mathrm{d}j_{\mathrm{I_B}}) \\ &\quad -f[N_{\mathrm{G}}](i-\phi\mathrm{d}i_{\mathrm{I_B}}, j-\phi\mathrm{d}j_{\mathrm{I_B}}), \\ \hat{r}_{\mathrm{S}}(i,j) &= f[t-1](i+\mathrm{d}i_{I_S}, j+\mathrm{d}j_{I_S}) \\ &\quad -f[N_{\mathrm{G}}](i-\phi\mathrm{d}i_{\mathrm{I_S}}, j-\phi\mathrm{d}j_{\mathrm{I_S}}), \\ \phi &= N_{\mathrm{G}}-t,\end{aligned}$$

where ϕ compensates for the possible unequal temporal distances between the interpolated frame and the reference frames due to the size of the GOP. $(\mathrm{d}i_{\mathrm{I_B}}, \mathrm{d}j_{\mathrm{I_B}})$ and $(\mathrm{d}i_{\mathrm{I_S}}, \mathrm{d}j_{\mathrm{I_S}})$ are the

motion vectors associated with the big block and small block MCI hypotheses respectively. For the reference frame hypotheses, the residual estimates ($\hat{r}_{\mathrm{L}}$ and $\hat{r}_{\mathrm{R}}$) are simply the difference between the two reference frames.

$$\begin{aligned}\hat{r}_{\mathrm{L}} &= f[t-1] - f[N_{\mathrm{G}}],\\ \hat{r}_{\mathrm{R}} &= f[t-1] - f[N_{\mathrm{G}}].\end{aligned}$$

For the MCE hypothesis, $\hat{f}_{\mathrm{E}}$, the residual frame, $\hat{r}_{\mathrm{E}}$, is twice the difference between the motion compensated previous frames:

$$\begin{aligned}\hat{r}_{\mathrm{E}}(i,j) &= 2\,(f[t-1](i+\mathrm{d}i_{\mathrm{E}}, j+\mathrm{d}j_{\mathrm{E}})\\ &\quad -f[t-2](i+2\mathrm{d}i_{\mathrm{E}}, j+2\mathrm{d}j_{\mathrm{E}}))\,.\end{aligned}$$

The factor of two is required since the two frames used for extrapolation are temporally very close which can result in underestimation of the error.

3.3.6 Time relative scaling

For the reference frames, it should be taken into account that the left reference frame is sometimes closer to the frame being decoded than the right reference frame and that the quality of the left reference frame degrades over the GOP due to reconstruction error. The method used to incorporate these two concepts is to simply scale the Laplacian noise parameter, α, of the reference hypotheses, as a function of the place in the GOP and the GOP size. The scaling parameters $\zeta_{\mathrm{L}}[t]$ and $\zeta_{\mathrm{R}}[t]$, corresponding to the left and right reference frames respectively, are given by:

$$\begin{aligned}\zeta_{\mathrm{L}}[t] &= 1-\eta_{\mathrm{L}}t,\\ \zeta_{\mathrm{R}}[t] &= 1-\eta_{\mathrm{R}}(N_{\mathrm{G}}-t),\end{aligned}$$

where η_{L} and η_{R} are tunable parameters. All simulations in this thesis use $\eta_{\mathrm{L}} = \eta_{\mathrm{R}} = 0.05$.

3.3.7 Joint adaptive noise modeling

To improve the combined noise modelling of the MH SI, the nature of the selected hypotheses is investigated and their typical behaviour described.

The key frame hypotheses add value when the motion estimated hypotheses fail. This occurs specifically when the failure is due to occlusion or due to new information in the second key

frame. Due to the nature of the estimation of the α parameters, the key frame hypotheses will always be considered to be more noisy, even when the block matching during ME is bad. It thus seems logical that when the motion estimation fails, decreasing the estimate of the noise of the key frame and increasing the estimate of the noise of the motion estimated hypotheses might yield a performance improvement. The questions that remain are: how to estimate whether motion estimation for a specific sub-block is bad, and which scaling factor for the noise would provide robust improvement.

The proposed scaling factor is based on the matching metric in the pixel domain. It thus uses all the information of the frame (not just the sub-band) and scales the alpha value by the relative quality of the match that produced the values. The scaling factor is calculated by breaking the residual frame, $\hat{r}(i, j)$, into sub-blocks and then calculating the mean square error value per sub-block as follows:

$$\mathrm{MSE}(v, w) = \frac{1}{n \times m} \sum_{i=vn}^{(v+1)n-1} \sum_{j=wm}^{(w+1)m-1} [\hat{r}(i, j)]^2,$$

where the sub-block is of size n by m. We match the size of the subblock with the size of the DCT ($n = m = 4$). The mean and the variance of the MSE metric is then estimated (over the frame) as:

$$\begin{aligned} \hat{\mu}_{\mathrm{MSE}} &= \frac{1}{V \times W} \sum_{v} \sum_{w} \mathrm{MSE}(v, w), \\ \hat{\sigma}^2_{\mathrm{MSE}} &= \frac{1}{V \times W} \sum_{v} \sum_{w} [\mathrm{MSE}(v, w) - \hat{\mu}_{\mathrm{MSE}}]^2, \end{aligned}$$

where V and W are the number of sub-blocks per column and row respectively. As the threshold for deciding between a 'good' and a 'bad' matching, we use the summation of the mean and standard deviation. Taking the ratio of the MSE metric with this threshold produces a value which will be larger than one for a bad match and less than one for a good match. To ensure robust performance, the scaling factor should not grow too large or too small. Thus, it is scaled to fit between 0 and 2 with a variable limit. Let this limit be denoted by η. The final scaling factor is then given as:

$$\zeta(v, w) = \eta \left[\frac{2 \frac{\mathrm{MSE}(v,w)}{\hat{\mu}_{\mathrm{MSE}} + \hat{\sigma}_{\mathrm{MSE}}}}{1 + \frac{\mathrm{MSE}(v,w)}{\hat{\mu}_{\mathrm{MSE}} + \hat{\sigma}_{\mathrm{MSE}}}} - 1 \right] + 1.$$

This factor will now be centred at 1 and be limited by $(1 - \eta)$ and $(1 + \eta)$. Experimentally, we found $\eta = 0.6$ to yield robust performance. For $\zeta(v, w)$, values larger than 1 indicates a poor match and values smaller than 1 indicates a good match.

3.3.8 Final correlation coefficients

Using the residual frame estimates described in Section 3.3.5, the Laplace correlation coefficients are calculated using Eq. (2.4) and then scaled using the parameters described in the previous sections. Let $\alpha_i^*(v,w)[t]$ be the parameter calculated with Eq. (2.4) for hypothesis i. If $\alpha_{\mathrm{L}}(v,w)[t], \alpha_{\mathrm{R}}(v,w)[t], \alpha_{\mathrm{S}}(v,w)[t], \alpha_{\mathrm{B}}(v,w)[t]$, and $\alpha_{\mathrm{L}}(v,w)[t]$ represent the Laplace parameters for the $f[t-1]$, $f[N_{\mathrm{G}}]$, $\hat{f}_{\mathrm{S}}$, $\hat{f}_{\mathrm{B}}$ and $\hat{f}_{\mathrm{E}}$ hypotheses respectively, the final values are:

$$\begin{aligned}
\alpha_{\mathrm{L}}(v,w)[t] &= \alpha_{\mathrm{L}}^*(v,w)[t]\zeta(v,w)\zeta_{\mathrm{L}}[t], \\
\alpha_{\mathrm{R}}(v,w)[t] &= \alpha_{\mathrm{R}}^*(v,w)[t]\zeta(v,w)\zeta_{\mathrm{R}}[t], \\
\alpha_{\mathrm{S}}(v,w)[t] &= \alpha_{\mathrm{S}}^*(v,w)[t](2-\zeta(v,w)), \\
\alpha_{\mathrm{B}}(v,w)[t] &= \alpha_{\mathrm{B}}^*(v,w)[t](2-\zeta(v,w)), \\
\alpha_{\mathrm{E}}(v,w)[t] &= \alpha_{\mathrm{E}}^*(v,w)[t].
\end{aligned}$$

MCE has the drawback in comparison with MCI that well matched blocks can still produce erroneous results. In order to allow for robust performance where the MCE hypothesis does not overwhelm the other hypotheses (for which more accurate error estimates are available), the variance values used in the calculation of $\alpha_{\mathrm{E}}^*(v,w)$ are hard limited to be no less than $\Delta_b/4$, where Δ_b is the size of the quantization bin for the given subband.

3.4 PERFORMANCE ANALYSIS OF PROPOSED MULTI-HYPOTHESIS SYSTEM

3.4.1 Simulation Setup

The methods proposed in this chapter to improve performance are tested on four standard test sequences: *foreman*, *coastguard*, *hall monitor* and *soccer*. The sequences are all QCIF format with a frame rate of 15 Hz. All the rate-distortion curves show results for the average bit rate and the distortion as calculated over all the frames in the entire sequence (key and WZ frames). All sequences consist of 150 frames.

The system will be compared against the H.264 intra codec as well as with the DISCOVER codec [25], which is one of the benchmark systems in DVC literature. DISCOVER is a single-hypothesis bit-plane based system. While some improvements have been made since, these rely on improving other aspects of the system such as exploiting information between

sub-bands, post-processing of decoded frames and encoder side motion estimation. These methods can also be added to system proposed in this thesis. Key frames were encoded using the reference implementation of the intra mode of the H.264 codec. The QP parameters used are similar to those used by the DISCOVER codec and are provided in Table 3.4. The other simulation parameters are provided in Table 3.3.

Table 3.3: The values of the codec parameters used for the experiments performed in this section.

Simulation parameter	Value
Mean filter window size	3×3
MCI large block median filter window size	5×5
MCI small block median filter window size	7×7
MCE block median filter window size	5×5
DCT block size	4×4
MCI large block size	16×16
MCI small block size	8×8
MCE block size	8×8
MCI large block search range	48
MCI small block search range	24
Central frame extrapolation search range	24
Backward extrapolation search range	4

Table 3.4: Key frame quantization parameters for the RD points. The same values are used for all GOP sizes.

RD number	1	2	3	4	5	6	7	8
Hall monitor	37	36	35	33	32	30	29	24
Coastguard	38	37	36	34	33	31	30	26
Foreman	40	39	38	34	33	31	29	25
Soccer	43	42	40	36	35	33	30	25

The performance of the system is also provided for different GOP sizes. The performance of DVC systems over different GOP sizes is important for several reasons. Firstly, since WZ

frames are faster to encode than key frames, an increase in the GOP yields a reduced encoding complexity. Secondly, when the motion content in a video sequence is low, or the temporal sampling rate is high, the correlation between frames increases. As a result, a larger GOP size can yield better rate-distortion performance. For general coding, the GOP size should adapt to the video sequence being encoded. However, by convention, when testing video coding systems, the GOP is fixed in order to provide a comparison of the coding system and not the adaptive GOP algorithm. Systems that perform better at a larger GOP size on average are also expected to perform better in an adaptive GOP scenario.

3.4.2 Comparison of proposed Multi-hypothesis system with DISCOVER

The main comparison shows the results for the proposed multi-hypothesis system combining extrapolation and interpolation (denoted by MH-IE). Figures 3.4, 3.5, 3.6 and 3.7 shows the performance on the *foreman*, *coastguard*, *hall monitor* and *soccer* sequences respectively.

The *foreman* and *soccer* sequences both contain significant motion, so the use of multiple hypotheses is expected to improve the performance. The *coastguard* sequence has a small amount of motion, and the changing texture of the water is difficult to predict, so only a small gain is expected. The *hall monitor* sequence has very little motion, so we do not expect much improvement. The results in general, meet these expectations. A small summary of the results is provided in table 3.5. In the summary, we include the results of the proposed multi-hypothesis system without the extrapolation based hypothesis (denoted by MH-I) to show the specific gain from the addition of an extrapolation based hypothesis. MH-I thus uses four hypotheses, the two reference frames and the two interpolated hypotheses. From the table one can see that the addition of the extrapolation based hypothesis improves the performance.

From the figures and the results in table 3.5, one can see that the multi-hypothesis system outperforms the DISCOVER system in almost all cases. The largest gain is seen in the sequences with motion and for larger GOP sizes. The addition of the extrapolation hypothesis further improves the performance of the system as the GOP increases. The gain is most pronounced for the higher motion sequences.

The system performs better than H.264 intra for the lower motion sequences *coastguard* and *hall monitor*. For the *foreman* sequence the proposed system performs similarly to H.264 (intra) and for the high motion *soccer* sequence the H.264 intra method performs best. It is apparent that the relative performance between DVC and H.264 intra depends on the motion content of

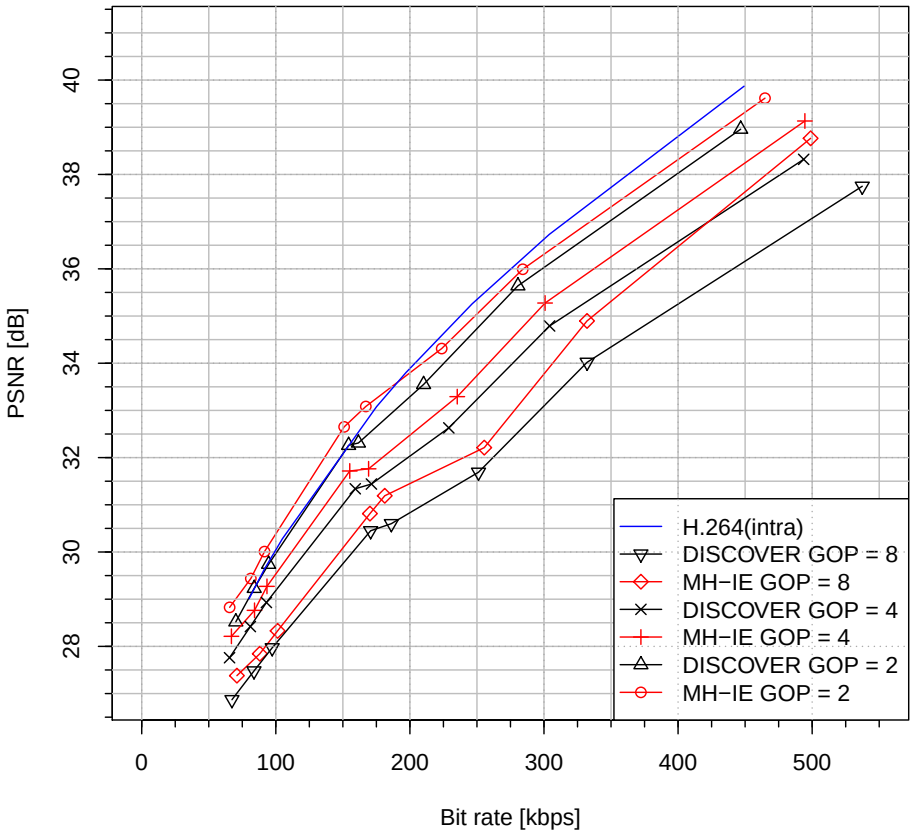

FIGURE 3.4: The performance of the system on the *foreman* sequence.

Table 3.5: System gain in dB as measured against the DISCOVER codec. The bit rate at which the measurements were taken are indicated in the column headers.

	soccer *400kbps*		*hall* *200kbps*		*foreman* *300kbps*		*coastguard* *300kbps*	
GOP	MH-I	MH-IE	MH-I	MH-IE	MH-I	MH-IE	MH-I	MH-IE
2	0.5	0.6	0.1	0.3	0.1	0.2	0.3	0.3
4	0.6	0.8	0.4	0.3	0.05	0.55	0.25	0.4
8	0.5	0.6	0.5	0.25	0	0.6	0	0.25

the video sequence.

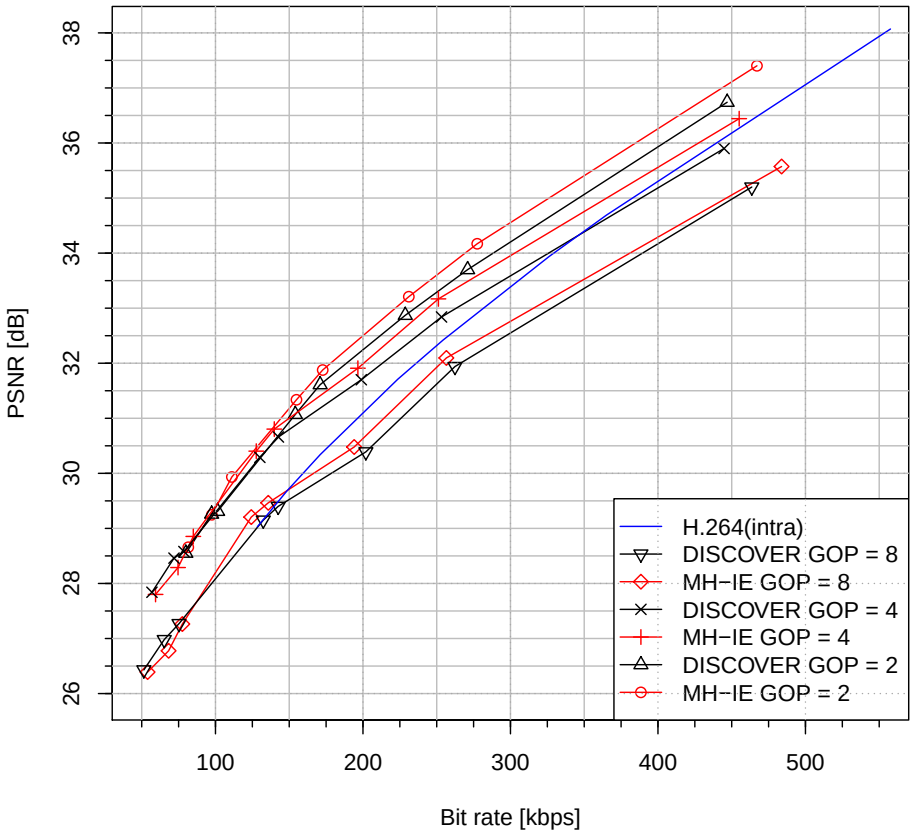

FIGURE 3.5: The performance of the system on the *coastguard* sequence.

3.4.3 Multi-hypotheses vs. Single Hypothesis

To show the effect of multi-hypotheses relative to using only a single hypothesis, the performance of the system on the *foreman* sequence is shown in Figure 3.8. In this case the single hypothesis is the small block MCI hypotheses as used for in the MH system. From the figure one can see that the multi-hypothesis method improves the performance. The gain is mostly in the lower and higher rate regions with similar performance in the middle rate region. The gain is not very large. This is expected since the figure shows average results over the entire sequence. Both sequences use the same key frames, thus the gain over the WZ frames is larger than the average gain. Furthermore, the SH uses the 'best' hypothesis available and this hypothesis will be accurate for some parts of the sequence. The gain for the MH system is also higher for larger GOP sizes which indicates that the MH system improves performance when the quality of a single hypothesis begins to degrade (due to the larger GOP size).

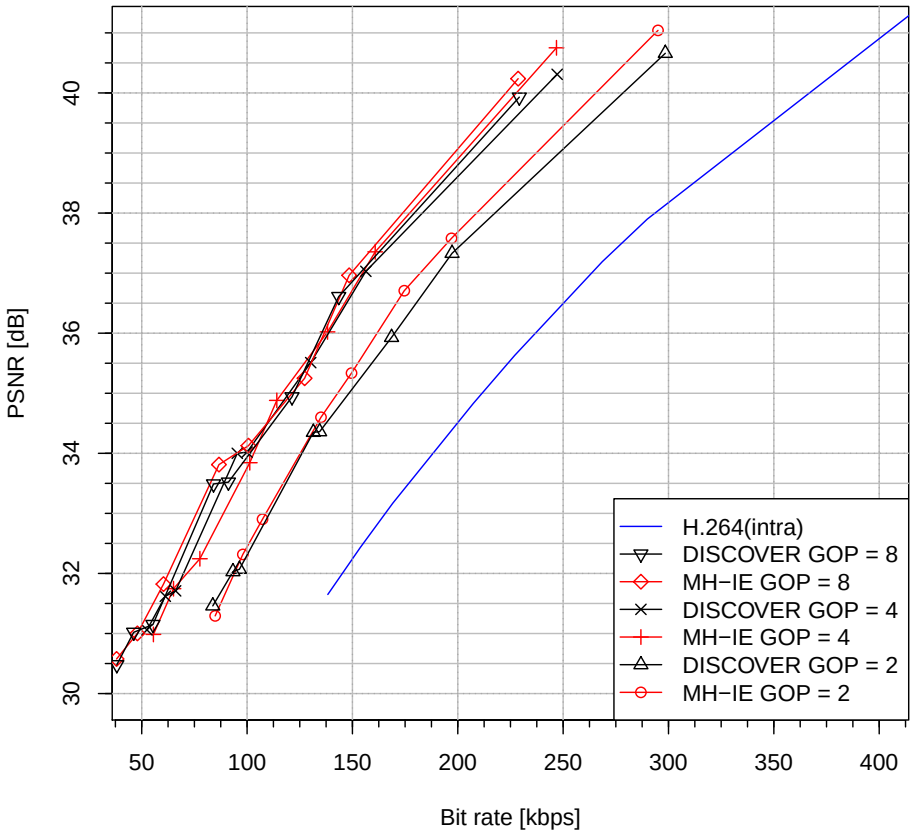

FIGURE 3.6: The performance of the system on the *hall monitor* sequence.

3.4.4 Bayesian Fusion vs. Averaging

To demonstrate the performance of proposed Bayesian fusion MH method relative to the averaging approach in Eq. (2.8) the performance of the two methods on the *foreman* sequence is shown in Figure 3.9. Both methods use all five hypotheses (two reference hypotheses, two MCI hypotheses and a MCE hypothesis). "MH-IE Bayesian Fusion" indicates the proposed method and "MH-IE averaging" indicates the averaging approach. For a GOP size of two, the proposed method performs better than the averaging method at all rates. For a GOP size of four, the two methods perform similarly at low and medium rates while the proposed method performs better at high rates. For a GOP size of eight, the averaging method performs better at low rates while the proposed method performs better at higher rates. Interestingly, the proposed method achieves a lower rate for all coding profiles, while occasionally achieving a lower PSNR. This indicates that using the conventional method for reconstruction yields better performance at

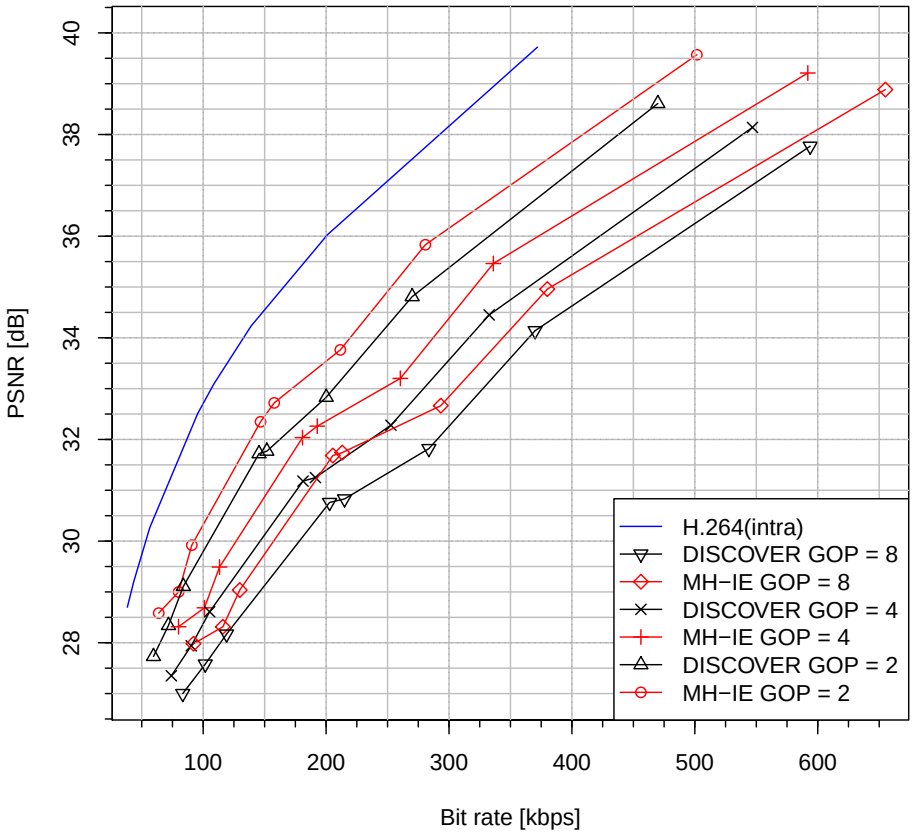

FIGURE 3.7: The performance of the system on the *soccer* sequence.

larger GOP sizes. A hybrid approach could perhaps be used in future to get the benefit of both methods.

3.5 COMPLEXITY

In this section the complexity of the proposed systems are evaluated with run-time experiments.

3.5.1 Simulation Setup

To experimentally evaluate the computational complexity execution-time experiments were performed. Simulation based comparisons are difficult and always imperfect, but may still provide some insight into the expected performance of the systems. For these simulations the reference implementation of the H.264(intra) codec was used. All the simulations were

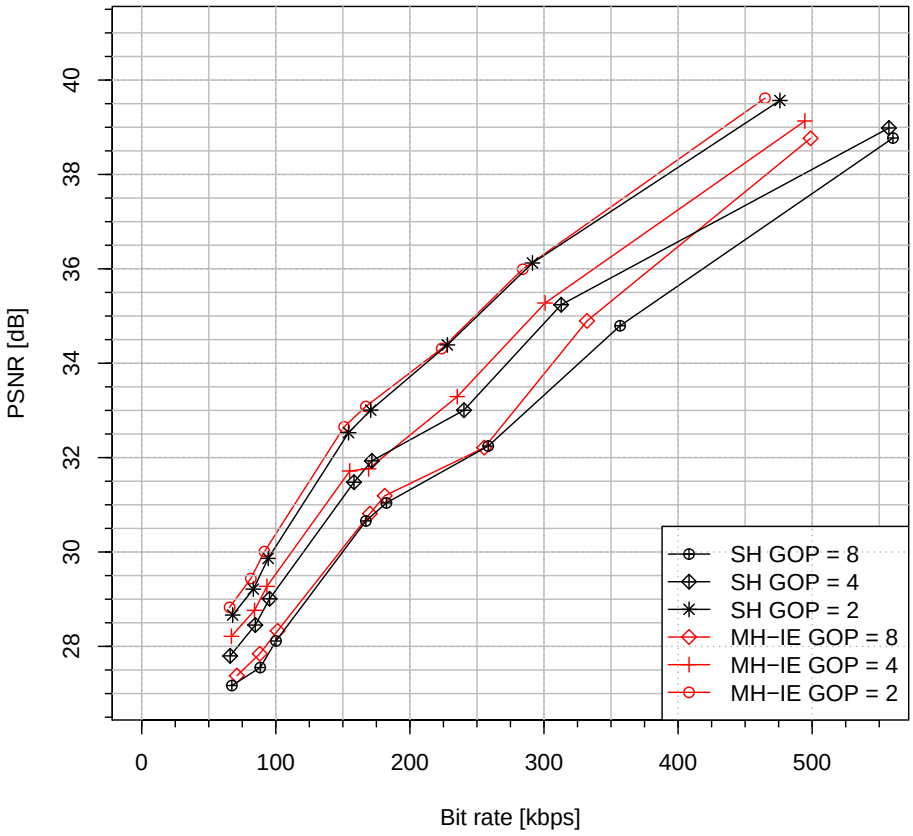

FIGURE 3.8: A comparison of the performance of the multi-hypothesis SI and the single hypothesis SI on the *foreman* sequence 15Hz. SH indicates the single hypothesis and MH-IE is the multi-hypothesis system with five hypotheses.

performed on a computer with a 2.93 GHz Intel Core 2 Duo processor and 2Gb RAM running Ubuntu Linux 12.04 LTS.

The encoder run-time of DISCOVER was analysed in [74]. Without access to the source code, run-time results for the DISCOVER encoder could not be accurately generated. To facilitate a comparison, the run-time of DISCOVER (on this computer) was predicted. This was done by calculating the ratio of the per frame run-time of DISCOVER to H.264(intra) from results presented in [74]. The run-time of H.264(intra), as produced in this simulation, was then multiplied by the ratio to predict the run-time of DISCOVER. While this is not ideal, it still provides a good idea of the relative performance of the different systems.

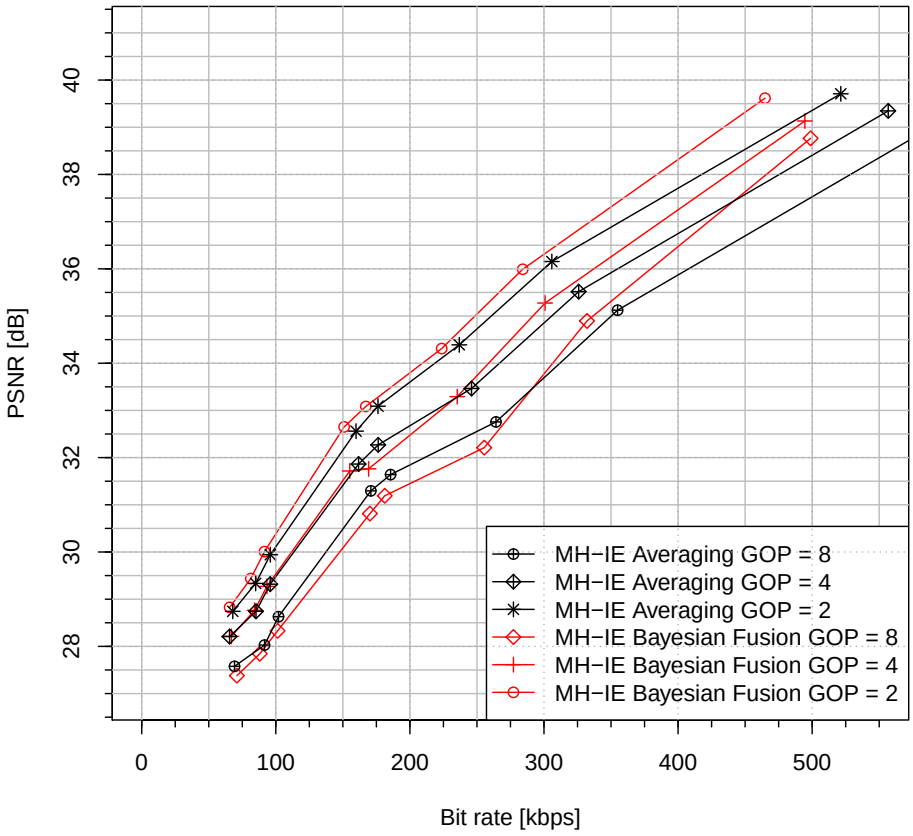

FIGURE 3.9: A comparison of the performance of Bayesian fusion vs. averaging multi-hypothesis on the *foreman* sequence 15Hz.

3.5.1.1 Encoder Complexity

The complexity of the encoder of the proposed system is similar to that of the DISCOVER codec since the main contributions of this proposal occur at the decoder. The main difference between the encoders of the proposed system and DISCOVER is non-binary vs. binary encoding, respectively.

To analyse the effect of the non-binary coding on the encoder complexity, CPU execution-time experiment results are provided. The time required for file IO was not considered.

Table 3.6 gives the average per-frame run-time. For the DISCOVER and proposed system, only the WZ frames were considered. The total run time will thus depend on GOP size and be a weighted sum between the intra and WZ frame run-times.

Table 3.6: Run-time per frame in milliseconds, for the encoder of the Proposed System (MH-IE), the reference implementation of H.264(intra) and the DISCOVER codec. The average run-time for the proposed and DISCOVER systems were measured only for the Wyner-Ziv frames.

	Foreman			Coastguard			Soccer			Hall Monitor		
RD	Prop	Disc	H.264	Prop	Disc	H.264	Prop	Disc	H.264	Prop	Disc	H.264
1	4.41	21	109	4.43	21	116	5.24	21	99.7	4.95	21	105
2	4.38	21	103	3.78	21	114	5.22	21	100	4.58	21	107
3	7.45	21	105	6.85	21	124	8.45	22	103	8.08	22	117
4	12.6	21	124	11.6	22	132	13.5	22	103	12.5	22	129
5	15.9	22	114	14.8	21	138	16.7	22	106	16.2	23	132
6	19.0	24	116	17.3	22	142	20.3	22	118	17.6	24	135
7	18.4	23	145	19.1	23	159	17.8	24	122	18.2	24	135
8	18.0	24	169	17.8	25	187	16.2	25	157	18.6	26	166

From the table, one can see that the WZ frames are encoded much faster than the H.264(intra) frames for both the proposed and DISCOVER systems. The proposed system is somewhat faster than DISCOVER, but this could be due to rate-estimation calculations performed by DISCOVER, whereas the proposed system assumes perfect rate knowledge.

Thus, one can conclude that the R-D gains of the proposed system over DISCOVER do not come at the expense of increased encoder complexity. Also, due to the faster run-time of the Wyner-Ziv frames compared to the H.264(intra) frames, we can conclude that the proposed system will operate at a lower complexity for larger GOP sizes than for smaller GOP sizes.

3.6 DISCUSSION AND SUMMARY

In this chapter methods were proposed to improve the average rate-distortion performance of DVC. The methods focused on providing the decoder with as much information as possible. This was achieved by doing symbol level coding and combining multiple hypotheses.

The simulations results showed that the proposed multi-hypothesis system performs better than the DISCOVER codec in almost all scenarios. The gains were most pronounced on sequences with high motion content. The use of extrapolation improved the performance

especially over larger GOP sizes. The proposed MH Bayesian fusion method was also shown to perform better than conventional averaging. The run-time simulation results show that the improved performance does not come at the cost of increased complexity at the encoder. The system was also shown to have similar average rate-distortion compared to H.264 intra coding at a reduced encoding complexity.

Chapter Four

Feedback Free DVC

4.1 Introduction

Systems that employ feedback from the decoder to the encoder are able to operate at the minimum required rate for a given distortion. However, the feedback path comes with several costs. Each of these costs makes the system unsuitable for some applications. Some examples include:

- Feedback requires real time decoding of the encoded video signal. This makes the system unsuitable for encode and store applications like large scale surveillance.
- Feedback increases latency. This makes the system unsuitable for low-delay applications like video conferencing.
- Feedback requires buffering at both the encoder and the decoder. The increased memory requirements can push up the costs of the encoder device. This makes the system unsuitable for some low cost applications.
- Use of the feedback requires bandwidth. In a video conferencing style connection, the feedback path is already being used. The required feedback bits should thus be added to the rate-distortion calculations.

In this chapter a method is developed that can operate completely without feedback, thus circumventing the above difficulties. It is not theoretically difficult to construct a video encoder that operates without feedback unless one considers specific limitations. The purpose of DVC is low complexity encoding. Feedback based DVC systems are able to encode WZ frames

approximately ten times faster than conventional (H.264) intra coding and approximately 100 times faster than conventional (H.264) inter coding [25]. A solution without a feedback path should ideally have similar complexity performance. The specific limitation that is defined for this chapter is that no motion estimation is allowed at the encoder.

Since the majority of the complexity of conventional inter and intra coding is related to motion estimation, adding motion estimation will remove the purpose of DVC coding. While hybrid systems, which attempt low complexity motion estimation, have been proposed as discussed in section 2.5, these systems fall outside the scope of interest.

The chapter will start by giving an overview of the proposed system in Section 4.2. This will be followed in Section 4.3 by a description of the main contributions of the proposed system, namely, the use of real field codes, and the rate estimation block. Implementation details of each sub-block in the system will then be individually discussed in Section 4.4. Next, the complexity of the proposed system is discussed in Section 4.5. This is followed in Section 4.6 by an evaluation of the proposals presented in this chapter.

4.2 PROPOSED SYSTEM

Figure 4.1 shows a block diagram of the proposed system. Incoming frames are split into key frames, which are H.264 intra coded, and WZ frames. Let the original pixel-domain WZ frame, at time t in the group-of-pictures (GOP), be referred to as $f[t]$. $f[t](i, j)$ refers to the pixel at position (i, j). The GOP size (N_{G}) refers to the number of WZ frames in between the key frames plus one. The last key frame from the previous GOP will be referred to as $f[0]$ and the key frame at the end of the current GOP as $f[N_{\mathrm{G}}]$.

4.2.1 Encoder

The WZ frame is transformed with the block based DCT to yield:

$$F[t] \quad = \quad \mathrm{DCT}(f[t]).$$

The b^{th} subband of the frame will be referred to as $F_b[t]$, and the coefficient corresponding to the $(v, w)^{th}$ subblock will be referred to as $F_b[t](v, w)$. For notational brevity we will now drop the time indices and assume that each WZ frame in the GOP is treated in a similar manner. The number of bits assigned to each subband is calculated using a rate estimation algorithm (Section

4.3.2) aiming for a specific distortion. The system attempts to achieve the same distortion as a conventional DVC system using a pre-defined quantization matrix. For this thesis, eight quantization profiles taken from [31] (as given in Table 2.1) are used in conjunction with a 4×4 DCT. After estimating the required rate for each subband according to the quantization profile, the bit budget ($\Re$) is allocated to a specific code size (M_b) and bits per quantized symbol (q_b) as described in Section 4.4.1.2. Let $\mathbf{x}_b = \text{vect}(F_b)$, be a $(N \times 1)$ vectorised version of F_b, then $\mathbf{x}_b$ is encoded:

$$\mathbf{s}_b \quad = \quad \mathbf{H}_b\mathbf{x}_b,$$

where $\mathbf{H}_b$ is the $M_b \times N$ parity check matrix used for the b^{th} subband. Each subband is then quantized with a symmetric uniform quantizer (Section 4.4.1.1). The quantized subband will be referred to as:

$$\mathbf{s}_b^{\text{Q}} \quad = \quad Q_b(\mathbf{s}_b),$$

where $Q_b(\cdot)$ is the quantizer for the b^{th} subband. If q_b represents the number of bits assigned to $Q_b(\cdot)$, then $\mathbf{s}_b^{\text{Q}}$ will have elements in $\{0, 1, \ldots, L_b - 1\}$, where $L_b = 2^{q_b}$. Each quantized subband

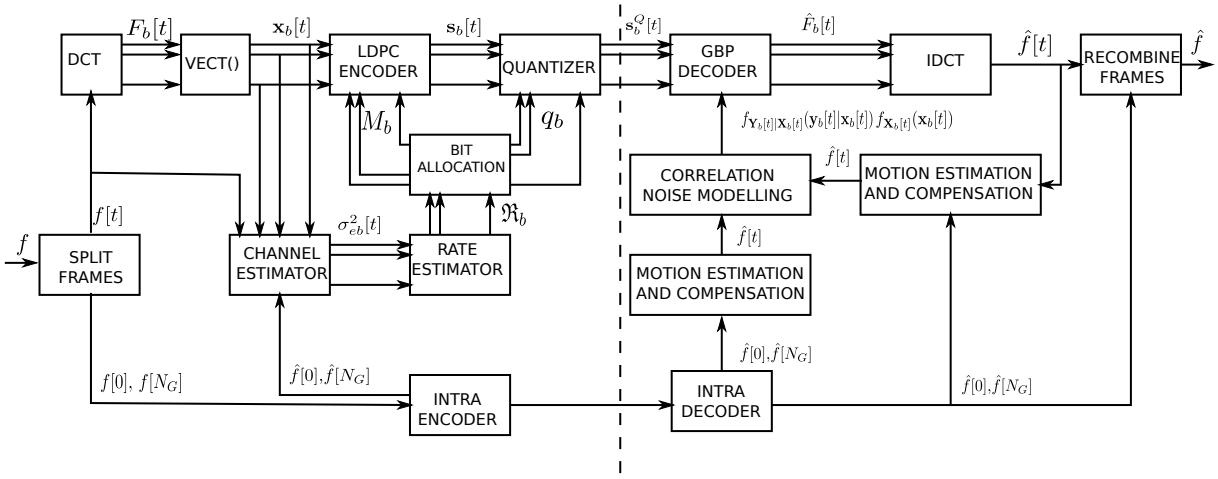

FIGURE 4.1: Diagram of proposed feedback free system

is then sent to the decoder along with the intra coded key frames. Due to the lack of a feedback path, frames do not need to be buffered after encoding. The parameters required to describe the quantizer, as discussed in Section 4.4.1.1, must also be transmitted.

4.2.2 Decoder

At the decoder, the key frames and possibly other decoded frames are used to create an estimate of f, denoted by $\hat{f}$. The SI hypothesis is transformed to produce $\hat{F}$ and each subband is vectorized to yield $\mathbf{y}_b$:

$$\begin{aligned} \hat{F} &= \mathrm{DCT}(\hat{f}) \\ \mathbf{y}_b &= vect(\hat{F}_b), \end{aligned}$$

where $\hat{F}_b$ is the b^{th} subband of $\hat{F}$. If the error for the SI hypothesis is given as:

$$r = f - \hat{f}_h,$$

we calculate an estimate $\hat{r}$. From the error estimate, noise parameters are calculated at a coefficient level using Eq. (2.4).

For each subband, the error estimate and SI hypothesis are combined with the received parity information and decoded using a Gaussian BP (GBP) algorithm which attempts to solve the equation:

$$\hat{x}_n = \arg\max_{x_n \in \mathbb{R}} f_{X|\mathbf{Y},\mathbf{S}^{\mathrm{Q}}}(x_n|\mathbf{y}, \mathbf{s}^{\mathrm{Q}}),$$

where n indicates the n^{th} coefficient in a subband with N coefficients and the subband indices are dropped for notational brevity. As opposed to standard DVC, no reconstruction is required. The decoded bands are recompiled to produce $\hat{F}$, and then the pixel domain output frame $\hat{f}$ is calculated:

$$\hat{f} = \mathrm{IDCT}(\hat{F}).$$

After decoding, the process is repeated in a method similar to successive refinement [75–78]. Using the decoded frame $\hat{f}$ a more accurate motion compensated interpolation (MCI) algorithm can be used to produce higher quality SI. New correlation noise estimates must then be produced for the SI, after which the GBP algorithm is used to decode the improved SI. This process may be iterated a number of times, though two iterations are typically sufficient. If the original decoding introduced an error, then the iterative process may worsen performance as the MCI process may produce worse quality SI. The subsections will now be discussed in more detail.

4.3 MAIN CONTRIBUTIONS

This section discusses the main contributions that allow for the proposed system to operate without feedback. This includes the restructuring of the encoder and the use of real field codes as discussed in Section 4.3.1 and a reduced complexity rate estimation method as discussed in Section 4.3.2.

4.3.1 Real Field Coding

4.3.1.1 Finite field systems and imperfect rate estimation

In this section, real field coding and its effect on Wyner-Ziv coding is described. Wyner-Ziv coding can be understood as a process of creating discontiguous quantization bins over the signal space. This can most easily be seen when the Wyner-Ziv code is implemented as a nested lattice quantizer [79]. A scalar quantizer followed by an error correction code performs the same role, where the syndrome describes a coset of codewords, each of which maps to a quantization cell in the signal space. The expected distortion, D_{WZ}, of a Wyner-Ziv code can be simplistically expressed as:

$$D_{\mathrm{WZ}} = E[D_{\mathrm{B}}]P[\epsilon] + E[D_{\mathrm{Q}}](1 - P[\epsilon]),$$

where $P[\epsilon]$ is the probability of a codeword error, D_B is the distortion introduced by a codeword error and D_{Q} is the distortion introduced by quantization. When the code is operating at a high enough rate $P[\epsilon] \rightarrow 0$ and the quantizer is the dominant distortion source. However, when the rate is too low, $P[\epsilon]$ can no longer be considered negligible and the distortion of a codeword error significantly affects performance.

The theoretic minimum required rate assuming a perfect SW coding after quantization is given by the SW theorem and requires knowledge of the channel statistics to calculate. Since the channel is unknown, and estimates will have some error, the estimated rate will have an error component.

As a result, even a perfect SW code will not be able to guarantee a distortion of $E[D_{\mathrm{Q}}]$. In the case of DVC, the quality of the channel knowledge depends on several factors. Firstly, it depends on the computational complexity of the algorithm used to perform the estimation. If full motion estimation is used, then the estimate can be exact. Without motion estimation, the estimate quality will depend significantly on the motion content of the sequence.

Furthermore, the SW code is not going to be able to achieve the SW bound for finite length sequences. This is analogous to the loss from capacity that finite length error correction codes suffer in communication systems [39]. Error correction codes (which are used to implement the SW code) also do not provide exactly equal error protection for all sequences and error patterns. These two concepts combine to add a further unknown to the required rate.

The problem can be summarised by saying that while we can calculate an upper bound on the lower bound of the required rate assuming perfect SW coding, using this value for conventional WZ coding may result in poor performance.

4.3.1.2 Proposed solution using real field coding

The proposed solution is to change the approach and the nature of the coding technique. Instead of performing conventional quantization followed by Slepian-Wolf coding, we encode first and then quantize.

A real field code creates a contiguous subspace over the signal space. This changes the distortion function to a more continuous and gradual function of the noise, as compared to a finite field code, since there is no codeword error based distortion. If $\mathbf{x} \in \mathbb{R}^N$ is the signal, encoding can be expressed in a similar manner as for conventional error correction codes:

$$\mathbf{p} = \mathbf{G}\mathbf{x},$$

where $\mathbf{G} \in \mathbb{R}^{(N+M)\times N}$ is the generator matrix and $\mathbf{y} \in \mathbb{R}^{(N+M)}$ is the encoded signal. Since only the parity symbols are transmitted, the generator matrix is in systematic form:

$$\mathbf{G} = \begin{bmatrix} \mathbf{I}_N \\ \mathbf{P} \end{bmatrix}$$

where $\mathbf{P} \in \mathbb{R}^{M\times N}$ is a parity matrix. Let $\mathbf{s}$ be the parity symbols (also referred to as the syndrome in Wyner-Ziv literature):

$$\begin{aligned} \mathbf{s} &= [y_{N+1} \dots y_{N+M}]^T \\ &= \mathbf{P}\mathbf{x}. \end{aligned}$$

The parity symbols identify a specific subspace with $(N - M)$ dimensions in $\mathbb{R}^N$ where $\mathbf{x}$ must lie. For example, if $N = 3$ and $M = 2$, then $\mathbf{s}$ will describe a specific line along which $\mathbf{x}$ lies. Decoding will find the most likely point on this line. In finite field coding the same is true over

the finite field based signal space, but when the correct codewords are mapped back to the signal space, they map to non-contiguous quantization regions.

Ignoring the prior distribution of the signal, and assuming a spherically symmetrical i.i.d Gaussian distributed error vector, the components of the error not along the line will be removed during decoding. Thus if the original error variance was σ_e^2, then the decoded error variance, σ_d^2, becomes:

$$\sigma_d^2 = \sigma_e^2 \frac{N-M}{N}. \tag{4.1}$$

If the noise is not Gaussian, the decoded variance can be reduced even further. For example, if the noise is K-sparse, which means $\|e\|_0 = K, K \ll N$, then with $M \geq K+1$, perfect reconstruction is possible, assuming an l_0-minimization decoder, though the problem is NP-complete [80]. If the noise is Laplace distributed, as is typically the case for DVC, then the performance lies between these two cases. Thus, decoding reduces the error variance in all cases and catastrophic decoding failures do not occur. Therefore, while there is no guarantee of a specific R-D performance, due to a lack of knowledge about the channel, the method will always reduce the distortion in the SI, and there will be no catastrophic decoding failures due to noise variance underestimation.

Before the real field encoded parity symbols can be transmitted or stored, they must be quantized. We thus calculate:

$$\mathbf{s}^{\mathrm{Q}} = Q(\mathbf{s}).$$

The quantized parity symbols describe a cell in $\mathbb{R}^N$ that is N dimensional, but bounded in M of those dimensions. The size of the bounded dimensions depends on the quantization bin size. There will thus be an additional quantization noise component added to the distortion. From this point we will refer to $\mathbf{P}$ as $\mathbf{H}$ to more closely align with notation used in the Wyner-Ziv coding literature and the rest of the thesis.

Figure 4.2 shows the difference in the induced quantization regions for a conventional WZ code vs. the proposed real field based code. The figure on the left shows conventional WZ coding where discontiguous quantization bins are grouped together (indicated by the grey boxes. The side information $\mathbf{y}$ is used to decode to one of the bins at the decoder. If the error is too large (which would correspond to an underestimated rate in the general case) a decoding error occurs which significantly increases the noise. For the figure on the right, a real field code followed by a quantizer created a contiguous quantization region. Since there is only one

region, there can be no decoding failure, only a varying degree of noise assuming the decoder converges.

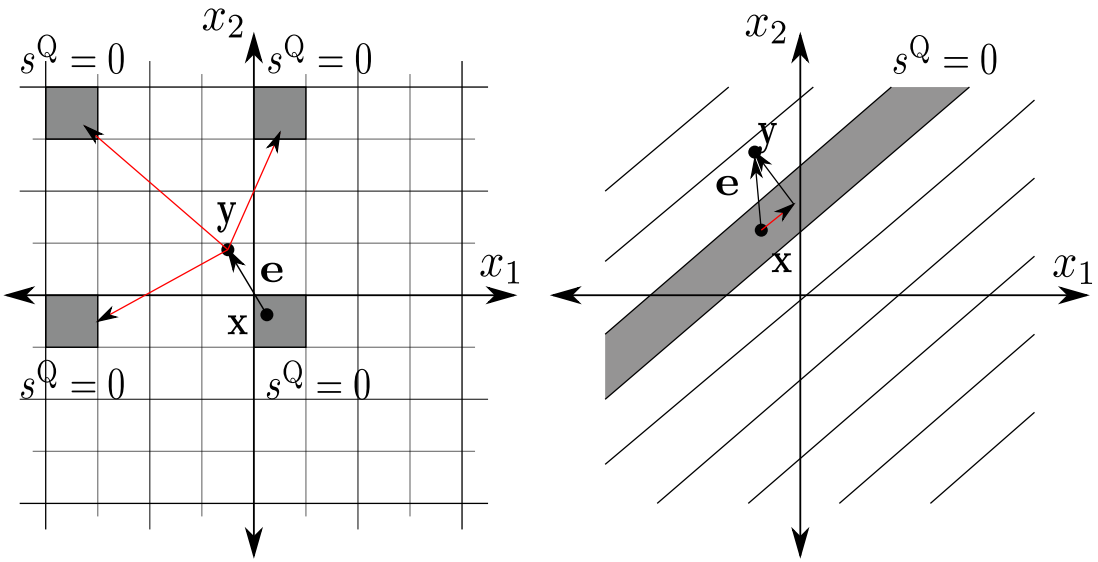

FIGURE 4.2: Two dimensional representations of contiguous vs. discontiguous quantization regions used for coding.

4.3.1.3 Real field Wyner-Ziv coding and Compressive Sensing

There is a link between the proposed method and Compressive Sensing. Encoding over the real field, quantizing the result, and decoding with side information can be shown to be mathematically similar to compressive sensing (CS), if the error signal is assumed to be sparse or compressible. If $\mathbf{x} \in \mathbb{R}^N$ is the original signal and $\mathbf{y}$ is the side information related as:

$$\mathbf{y} = \mathbf{x} + \mathbf{e},$$

encode $\mathbf{x}$:

$$\mathbf{s} = \mathbf{H}\mathbf{x}.$$

At the decoder, we would like to reconstruct $\mathbf{x}$, and thus we need $\mathbf{e}$.

$$\begin{aligned} \mathbf{H}\mathbf{y} &= \mathbf{H}\mathbf{x} + \mathbf{H}\mathbf{e} \\ \mathbf{H}\mathbf{y} &= \mathbf{s} + \mathbf{H}\mathbf{e} \\ \mathbf{H}\mathbf{y} - \mathbf{s} &= \mathbf{H}\mathbf{e} \\ \mathbf{v} &= \mathbf{H}\mathbf{e}, \end{aligned} \tag{4.2}$$

where $\mathbf{v} = \mathbf{Hy} - \mathbf{s}$, is known at the decoder. Eq. (4.2) is the same as CS, if $\mathbf{e}$ is assumed to be sparse or compressible.

The effects of quantization and the R-D performance of quantized CS were analysed in [81]. CS has been applied to video compression previously, for example in [82]. However, previous systems are different to the one presented here, since in these systems CS is used to sense a signal, which is sparse in a transform domain, directly to reduce the sampling requirements in the pixel domain. In contrast, the proposed method can be interpreted as sensing a sparse error vector at the decoder. Despite the link between the proposed method and CS, the results of CS are not necessarily directly applicable since for most cases in DVC, the error vector will not be very sparse.

4.3.1.4 Real Field Code Design

Coding over the real field as opposed to a finite field has not been considered much due to the prevalence of digital systems. Some early analog codes were based on Bose-Chaudhuri-Hocquenghem (BCH) codes and the DFT [83]. Large codes, analogous to finite field LDPC codes, were considered recently in the context of CS [84]. While exact design strategies are still being developed, it has been shown that codes that work well as binary LDPC codes also work well over the real field [84].

LDPC code design can be described as designing the decoding graph and choosing the connection weights. In this thesis, we use the quasi-cyclic (QC) codes designed in [51] to design the decoding graph (the parity check matrix structure). QC codes were chosen because of their fast encoding algorithms and reduced storage requirements at the encoder, as compared with random codes. For a code with an $M \times N$ parity matrix $\mathbf{H}$, the connection weights (non-zero elements) were selected at random from a Gaussian distribution such that the expected total energy of the encoded signal is the same as that of the original signal: $M\sigma_s^2 = N\sigma_x^2$.

The codes from [51] are girth-6 codes that are constructed using Galois fields. Depending on the number of parity symbols, M, a different construction field is used so as to ensure that the code remains girth-6 while being as dense as possible. If the construction field is the Galois field, $GF(2^{q_c})$, then the value for q_c is chosen as:

$$q_c = \begin{cases} 5 & (70 \leq M \leq 120) \\ 6 & (120 < M \leq 250) \\ 7 & (M > 250) \end{cases} .$$

M corresponds to the number of transmitted symbols. These values are sufficient for the video resolutions considered in this thesis, where $N = 1584$. If higher resolution frames are encoded, then N and M might increase. Codes should be designed with the required length in mind. Decoding is performed using a version of GBP described in Section 4.4.3.

4.3.2 Rate Estimation

The proposed system operates at an estimate of the theoretical Wyner-Ziv rate. This rate is calculated assuming a conventional system with a midrise symmetrical quantizer for the DC band and a deadzone uniform quantizer for the AC bands. The bits assigned to each band for each rate-distortion point, are taken from the quantization profiles in Table 2.1. While these quantizers are used to calculate the bit-budget assigned to each subband, a different quantizer is used after encoding. This is because we are quantizing the encoded signal and not the original signal. The code size M_b and number of quantization bits q_b are design parameters that may affect performance where $\Re = M_b q_b$ is the bit-budget.

We estimate the rate required to reconstruct the source $\mathbf{x}$ quantized with bin size Δ. If $\mathbf{x}$ is the signal, $\mathbf{e}$ is the error and $\mathbf{y}$ is the side information, then the system model for a given subband is:

$$\mathbf{y} = \mathbf{x} + \mathbf{e},$$

where the elements of $\mathbf{x}$ and $\mathbf{e}$ are Laplace distributed and drawn i.i.d from $X \sim \mathcal{L}(0, \alpha_x)$ and $E \sim \mathcal{L}(0, \alpha_e)$ respectively [63]. $\mathbf{x}$ is quantized to $\mathbf{x}^Q$ with a q-bit quantizer, where the value for q is taken from the quantization profile matrix. The required transmission rate is lower bounded by the entropy remaining in the quantized source given the side information [1]. The per symbol rate can thus be expressed as:

$$R \geq H(X^{\mathrm{Q}}|Y^{\mathrm{Q}}),$$

where $H(X^{\mathrm{Q}}|Y^{\mathrm{Q}})$ is the conditional entropy of quantized source, X^{Q}, given the quantized side information Y^{Q}. The final bit-budget per subband will be $\Re = NR$.

4.3.2.1 Correlation Channel Model

In order to estimate the rate, an estimate of the signal variance and the noise variance is required. The decoded key frames, $\hat{f}[0]$ and $\hat{f}[N_G]$, which are automatically available at the encoder due

to the intra coding algorithm are used to create an interpolated frame without motion estimation:

$$\hat{f}[t] = \tfrac{1}{2}\left(\hat{f}[0] + \hat{f}[N_G]\right).$$

An estimate of the residual, $\hat{r}$, is then used to calculate the required variance estimates:

$$\begin{aligned}
\hat{r}[t] &= f[t] - \hat{f}[t] \\
\hat{R}[t] &= \mathrm{DCT}[\hat{r}[t]] \\
\sigma_e^2(b) &= V[\hat{R}_b] \\
\sigma_x^2(b) &= V[\mathrm{DCT}[f](b)] \\
\sigma_y^2(b) &= \sigma_x^2(b) + \sigma_e^2(b).
\end{aligned}$$

While the variance estimation method works fairly well for the DC band, the AC bands are not as accurate. To improve the estimate, we estimate the noise variance of band (b) as usual and then average the value with a predicted variance:

$$\hat{\sigma}_e^2(b)^* = \frac{1}{2}\hat{\sigma}_e^2(b) + \frac{1}{2}\hat{\sigma}_e^2(b-1)^* \frac{\hat{\sigma}_x^2(b)}{\hat{\sigma}_x^2(b-1)}.$$

The bands are numbered by moving from top to bottom along the diagonals of the DCT subblock starting at the top left corner.

4.3.2.2 Proposed Method

Instead of calculating the conditional entropy from its definition, a different form of the expression is evaluated:

$$\begin{aligned}
H(X^Q|Y^Q) &= H(X^Q) - I(X^Q; Y^Q) \\
&= H(X^Q) - H(Y^Q) + H(Y^Q|X^Q),
\end{aligned}$$

where $I(X^Q; Y^Q)$ is the mutual information between X^Q and Y^Q. The proposed method has three parts:

1. Make a high rate assumption to simplify the expression.

2. Calculate closed form solutions for the simplified expression using the specific quantizer under consideration.

3. Improve the accuracy (reduce the effect of the high rate assumption) by changing the quantizer used for calculation of the individual entropies.

Beginning with step 1, make the simplifying assumption:

$$H(Y^{\mathrm{Q}}|X^{\mathrm{Q}}) \approx H(E^{\mathrm{Q}}).$$

This assumption allows the required rate to be calculated as a function of the entropies of three single variables, each of which is simpler to calculate than the conditional entropy. This is a high rate assumption which will typically be inaccurate for large Δ values.

Moving to step 2, closed form solutions are calculated for the single variable entropy expressions. Assuming that X is Laplace distributed, $H(X^{\mathrm{Q}})$ can be expressed as a summation that depends on the quantizer. For a deadzone quantizer, the entropy $H^{\mathrm{D}}(X^{\mathrm{Q}})$ is calculated as:

$$\begin{aligned} H^{\mathrm{D}}(X^{\mathrm{Q}}) &= -\sum_{k=-\infty}^{\infty} P(k\Delta)\log P(k\Delta) \qquad (4.3)\\ P(k\Delta) &= \begin{cases} \int_{k\Delta}^{(k+1)\Delta} f_X(x)\,\mathrm{d}x & (k>0)\\ \int_{(k-1)\Delta}^{(k+1)\Delta} f_X(x)\,\mathrm{d}x & (k=0)\\ \int_{(k-1)\Delta}^{k\Delta} f_X(x)\,\mathrm{d}x & (k<0), \end{cases} \end{aligned}$$

yielding:

$$P(k\Delta) = \begin{cases} \frac{1}{2}e^{-k\alpha_x\Delta}(1-e^{-\alpha_x\Delta}) & (k>0)\\ 1-e^{-\alpha_x\Delta} & (k=0)\\ \frac{1}{2}e^{k\alpha_x\Delta}(1-e^{-\alpha_x\Delta}) & (k<0). \end{cases} \qquad (4.4)$$

Calculating the infinite sum and recombining yields (a full derivation is given in Appendix A):

$$H^{\mathrm{D}}(X^{\mathrm{Q}}) = -(1-e^{-\alpha_x\Delta})\log(1-e^{-\alpha_x\Delta}) - e^{-\alpha_x\Delta}\left(\log\left(\frac{1-e^{-\alpha_x\Delta}}{2}\right) - \frac{\alpha_x\Delta\log(e)}{1-e^{-\alpha_x\Delta}}\right). \qquad (4.5)$$

For a uniform mid-rise quantizer that is symmetrical around zero, the entropy will be denoted by $H^{\mathrm{S}}(X^{\mathrm{Q}})$, and was calculated in a similar manner to yield:

$$\begin{aligned} H^{\mathrm{S}}(X^{\mathrm{Q}}) &= -2\left(\frac{1}{2}(1-e^{-\alpha_x\Delta})\right)\log\left(\frac{1}{2}(1-e^{-\alpha_x\Delta})\right) \qquad (4.6)\\ &\quad -e^{-\alpha_x\Delta}\left(\log\left(\frac{1-e^{-\alpha_x\Delta}}{2}\right) - \frac{\alpha_x\Delta\log(e)}{1-e^{-\alpha_x\Delta}}\right). \end{aligned}$$

For a uniform mid-tread quantizer the entropy, $H^{\mathrm{M}}(X^{\mathrm{Q}})$, is:

$$H^{\mathrm{M}}(X^{\mathrm{Q}}) = -\left(1-e^{-\frac{\alpha_x\Delta}{2}}\right)\log\left(1-e^{-\frac{\alpha_x\Delta}{2}}\right) - e^{-\frac{\alpha_x\Delta}{2}}\left(\log\left(\frac{e^{\frac{\alpha_x\Delta}{2}}-e^{-\frac{\alpha_x\Delta}{2}}}{2}\right) - \frac{\alpha_x\Delta\log(e)}{(1-e^{-\alpha_x\Delta})}\right). \qquad (4.7)$$

These closed form equations all assume that the quantizer is defined by the bin size, and that the range of the variable is theoretically infinite. In reality, there will be a limited number of

levels considered. We can thus also calculate $H(X^Q)$ by evaluating (4.3) and (4.4) over a finite number of levels. However, the difference between these is small enough to be ignored and we choose to use the closed form solutions provided by the infinite quantizer instead.

If we assume that both the side information and the noise are Laplace distributed, then the equations derived for the source signal, (4.5), (4.7) and (4.6), can be used to calculate $H(Y^Q)$ and $H(E^Q)$ as well.

In step 3, the accuracy of the simplification is improved by averaging the entropy estimates from different quantizers.

Figure 4.3 shows how different quantizers yield different entropy estimates for the same bin size. When the noise variance is large compared to the bin size Δ, the induced PMF looks similar irrespective of the quantizer used and as a result the entropy estimates will be similar for different quantizers. However, when the noise variance is small compared to the bin size, different quantizers will yield completely different PMFs (and entropy values). The true value will lie somewhere between the extremes of a mid-rise and a mid-tread quantizer. Thus, using an average of entropies from different quantizers for $H(E^Q)$ can reduce the error.

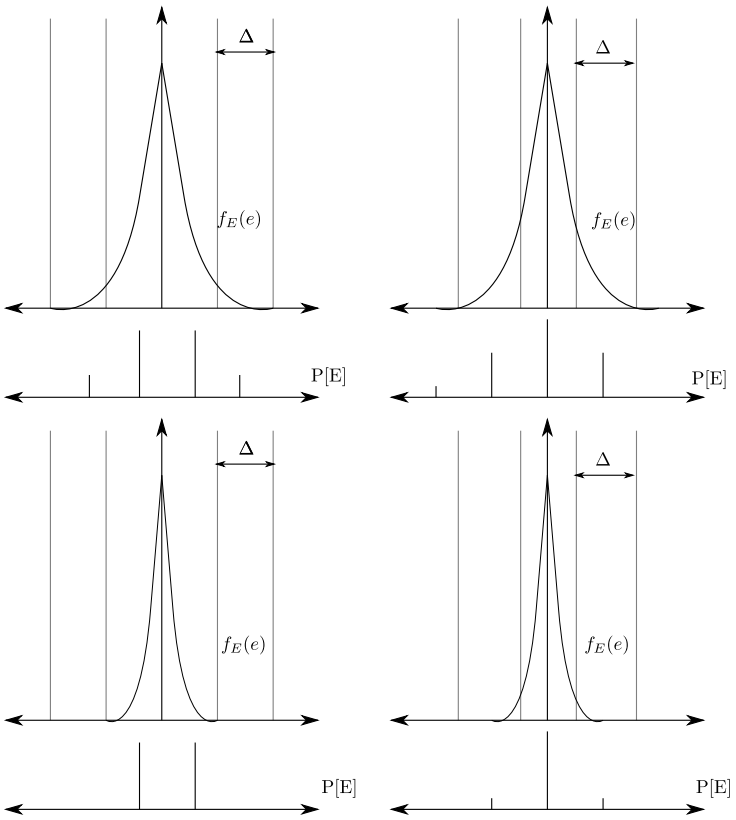

FIGURE 4.3: The effect of low rate quantizers on entropy calculation

The analysis on how averaging improves the accuracy is given in Section 4.6.2.

4.4 IMPLEMENTATION DETAILS

In this section the implementation details related to the subsystems of the proposed feedback free system are discussed.

4.4.1 Encoder

In this section the specifics of the different parts of the encoder are discussed. Both the time and subbands indices have been dropped from most of the equations since the same process is applied to all the subbands for each of the WZ coded frames.

4.4.1.1 Quantizer

The encoded signal must be quantized before transmission. If the bit budget for a subband is $\Re$, and the length of the code is M_b, then we use a $q_b = \Re/M_b$ bit uniform quantizer with $L = 2^{q_b}$ levels. If the range of the quantizer is $A = 2||\mathbf{s}||_\infty$, then the quantizer function can be described as:

$$Q(s) = \lfloor f_c(s)/\Delta \rfloor + \frac{L}{2},$$

where

$$f_c(s) = \begin{cases} s & \text{if } |s| \leq \beta A \\ \text{sign}(s)\beta A & \text{otherwise,} \end{cases}$$
$$\Delta = \frac{\beta A}{L}.$$

The scaling parameter β allows for the quantizer to clip some measurements. The Δ and q_b parameters for each subband are transmitted along with the encoded sequence. Δ can be represented with 8 bits and q_b with 3 bits. As a result, the overhead from transmitting these parameters is negligible.

4.4.1.2 Bit Allocation

The rate estimation algorithm provides a specific bit budget, $\Re$ for each subband. A method is required to allocate M_b and q_b , where $q_b M_b = \Re$.

The bit allocation should be chosen to minimize the expected distortion. In general, for a bounded quantization cell, the area in the cell and the resultant distortion can be halved by adding an additional quantization bit per symbol, or by doubling the number of constraints (M_b). This would indicate that the number of quantization bits per symbol (q_b) should be maximised. However, this is only true once the quantization region is bounded. This indicates that the allocation decision function should be piecewise defined. The number of constraints should be increased until the region is bounded at which point the number of bits per constraint should be increased.

Alternatively, since we are aiming for a specific distortion level, we can estimate the number of errors larger than the allowable distortion. Let this number be K:

$$K = NP(e > \epsilon_{\text{aim}}).$$

where e is the Laplace distributed error and ϵ_{aim} is the largest desired error. If we treat the dimensionality of the error signal as K, then we need at least K constraints. We thus set the minimum value for M_b, and the maximum value of q_b as:

$$\begin{aligned} M_b &\geq K + 1 \\ q_b &\leq \frac{\Re}{K+1}. \end{aligned}$$

This is only an approximation since the noise model at the encoder is not exact and the decoder is sub-optimal. M_b should be larger to improve the performance. The proposed approach is to calculate M_b and q_b as above but to limit q_b as:

$$q_b \leq q_{\text{aim}} + 2,$$

where q_{aim} is the number of bits used in the rate estimation derivation. We also require that $q_b \geq 2$. After adjusting q_b to meet the above criteria, M_b is recalculated.

4.4.2 Decoder

In this section the subsections and algorithms used in the decoder are discussed.

4.4.2.1 Side Information Creation

The decoder uses a successive refinement approach. This means that there is an initial SI creation process. The SI is then decoded. After decoding, the SI creation process is repeated and improved using the decoded frame.

The SI used in the initial decoding is the same as the interpolated hypothesis using small subblocks as described in Section 3.3.3.

After BP decoding, the subbands are recombined and a pixel domain version of the WZ frame is produced. Using this decoded frame, the motion estimation process is repeated to produce better quality side information. This is similar to [75–78] where successive refinement was used to improve reconstruction distortion. Due to the use of real field coding, the effect is slightly different from conventional DVC as improved side information can result in decreased distortion from the GBP decoding process, whereas for finite field coding there can be no further gain (other than improved reconstruction) after successful decoding of the bit planes.

In order to use the decoded frame, the MCI method needs to be adjusted. This is done by performing pixel level accuracy bilinear BM ME using the decoded frame as a hash to produce $\hat{f}_{\mathrm{I}}$. Figure 4.4 shows a representation for a GOP size of 2. The matching metric used during ME is:

$$M_{\mathrm{I}}(L, M, R) \quad = \quad (1 - 2\lambda_M) \sum |L - R| + \lambda_M \sum |L - M| + \lambda_M \sum |R - M|,$$

where L, M and R represent the sub-blocks in the left key frame, the reconstructed frame and the right key frame respectively. λ_M is a scaling factor that may be optimized. For this thesis $\lambda_M = 0.25$.

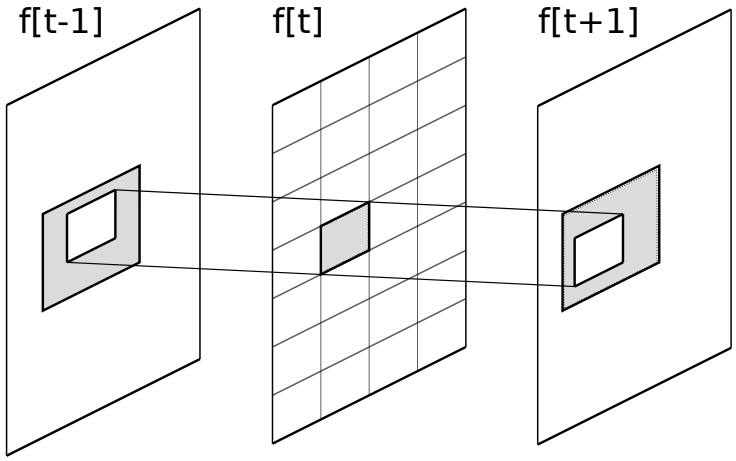

FIGURE 4.4: Bidirectional Motion Estimation

4.4.2.2 Correlation noise modeling

The system uses a coefficient level noise model [31] which models the correlation noise for a single coefficient of a subband of a hypothesis as being Laplace distributed.

For the first decoding, the correlation noise is estimated using the same method as for the system proposed in Chapter 3.

For the refinement stage, a different residual frame is calculated. If $\hat{f}_{\mathrm{I}}$ is the SI hypothesis, the residual frame is estimated as:

$$\hat{r}_{\mathrm{I}} = (1-\lambda_{\mathrm{I}})|\hat{f}_{\mathrm{L}^*} - \hat{f}_{\mathrm{R}^*}| + \lambda_{\mathrm{I}}|\hat{f}_{\mathrm{I}} - M|$$

Here, $\hat{f}_{\mathrm{L}^*}$ and $\hat{f}_{\mathrm{R}^*}$ indicate the motion compensated key frames used to create $\hat{f}_{\mathrm{I}}$, M is the previous decoded frame, and the λs are tunable parameters. While they may be optimized in future, we used $\lambda_{\mathrm{I}} = 0.5$.

From the residual estimate we calculate a correlation noise estimate for each hypothesis. The method from Eq (2.4) is used, but the output variance from the belief propagation decoder is included in calculating the value of the Laplace parameter α. The goal is to identify the areas where the decoder may have been in error, or where further refinement should occur. If $\sigma_b^{2(d)}(v,w)$ is the output variance from the GBP decoder, the new equation is:

$$\begin{aligned} D_b(v,w) &= |\hat{R}_b|(v,w) - \mu_b \\ \hat{\alpha}_b(v,w) &= \begin{cases} \sqrt{\frac{2}{\lambda\hat{\sigma}_b^2 + (1-\lambda)\sigma_b^{2(d)}(v,w)}}, & [D_b(v,w)]^2 \leq \hat{\sigma}_b^2 \\ \sqrt{\frac{2}{\lambda[D_b(v,w)]^2 + (1-\lambda)\sigma_b^{2(d)}(v,w)}}, & [D_b(v,w)]^2 > \hat{\sigma}_b^2 \end{cases}, \end{aligned}$$

where λ is a tunable variable, that is set to one half.

4.4.3 Decoding over real fields

Belief propagation (BP) is a decoding algorithm that calculates an estimate of a marginal probability by exploiting factorization to reduce computational complexity. BP can be performed over any arbitrary field. The update equations require the computation of the product and the convolution of the marginal PDFs being passed in the graph. When the coding is performed over the binary field, the PDF becomes a probability mass function (PMF) and can be represented using a single number. In the case of the real field, a full PDF is required. Calculating the convolution of a set of arbitrary functions is difficult. Thus, we use a version of Relaxed or Gaussian BP [85], which assumes that the messages internal to the graph are Gaussian distributed. This allows for simple closed form solutions of the update equations that rely only on the mean and the variance of the messages, reducing computational complexity. It also means that only two values need to be passed along each edge of the graph, reducing memory requirements.

Let μ_{ij} be the messages from variable node i to check node j and v_{ji} be the message from check node j to variable node i. Let $f_X(x_i)$ be the prior information of variable node i, and $f_{\mathbf{Y}|X}(\mathbf{y}_i|x_i)$ be the multi-hypothesis side information. The standard update equation at the variable node is:

$$\mu_{ij}(x_i) \quad \propto \quad f_X(x_i) f_{\mathbf{Y}|X}(\mathbf{y}_i|x_i) \prod_{k,k\neq j} v_{ki}\left(\frac{1}{H_{ki}} x_i\right),$$

while the update at the check node is:

$$v_{ji}(x_i) \quad \propto \quad f_S(s_j) \bigoplus_{k,k\neq i} \mu_{kj}(H_{jk} x_k),$$

where $\bigoplus$ indicates convolution, H_{ji} is the connection weight from the parity check matrix and $f_S(s_j)$ is the pdf of the j^{th} parity symbol. As mentioned, due to computational complexity we make the simplifying assumption that these messages are Gaussian distributed. Thus each message is represented with only a mean and a variance:

$$\mu_{ij}^G(x_i) \quad = \quad (E[\mu_{ij}(x_i)], V[\mu_{ij}(x_i)]) \tag{4.8}$$

$$v_{ji}^G(x_i) \quad = \quad (E[v_{ji}(x_i)], V[v_{ji}(x_i)]), \tag{4.9}$$

where (4.8) and (4.9) can be calculated using the error function. As a result of the Gaussian assumption for the marginals, the update equations are simplified. A product of Gaussian pdfs is a Gaussian pdf:

$$\begin{aligned} \prod_{i=1}^{I} G(\mu_i, \sigma_i^2) \quad &= \quad S\, G(\mu_p, \sigma_p^2) \\ \sigma_p^2 \quad &= \quad \left(\sum_{i=1}^{I} \frac{1}{\sigma_i^2}\right)^{-1} \\ \mu_p \quad &= \quad \frac{\sum_{i=1}^{I} \mu_i \sigma_i^{-2}}{\sum_{i=1}^{I} \sigma_i^{-2}}, \end{aligned}$$

where S is an irrelevant scaling factor, and I is the number of pdfs in the product. The convolution of Gaussian pdfs is also Gaussian:

$$\begin{aligned} \bigoplus_{i=1}^{I} G(\mu_i, \sigma_i^2) \quad &= \quad G(\mu_c, \sigma_c^2) \\ \mu_c \quad &= \quad \sum_{i=1}^{I} \mu_i \\ \sigma_c^2 \quad &= \quad \sum_{i=1}^{I} \sigma_i^2. \end{aligned}$$

Similarly, the connection weights are easily taken into account:

$$\begin{aligned} \mu_{ij}^{G}(H_{ji}x_i) &= (H_{ji}E[\mu_{ij}(x_i)], H_{ji}^2 V[\mu_{ij}(x_i)]) \\ v_{ji}^{G}\left(\frac{1}{H_{ji}}x_i\right) &= \left(\frac{1}{H_{ji}}E[v_{ji}(x_i)], \frac{1}{H_{ji}^2}V[v_{ji}(x_i)]\right). \end{aligned}$$

Decoding is the process of sequentially updating the messages at the variable and check nodes. Each update is an iteration. Decoding ends either after a set number of iterations or when the messages have converged (are no longer changing).

To help with convergence a damping parameter [86] is added at the variable node. If we are calculating the update for iteration $t+1$, $\mu_{ij}^{G}(x_i)^{t+1}$, then we combine the result of (4.8), represented by $\mu_{ij}^{G}(x_i)$, with the output of the previous iteration $\mu_{ij}^{G}(x_i)^{t}$:

$$\mu_{ij}^{G}(x_i)^{t+1} = \beta\mu_{ij}^{G}(x_i)^{t} + (1-\beta)\mu_{ij}^{G}(x_i).$$

The damping parameter may be optimized in future, but for this thesis, $\beta = 0.7$ was found experimentally to yield good results.

4.5 COMPLEXITY ANALYSIS

In this section the complexity of the proposed system relative to conventional DVC as well as H.264(intra) coding is discussed.

4.5.1 Encoder Time Complexity

The overall complexity is a function of the GOP size, since it will be a combination of the key frame encoding complexity as well as the WZ frame encoding complexity. Initially, only the complexity of the WZ frames is considered, since the key frames have the same complexity in H.264(intra) and in conventional DVC methods.

The complexity of the encoder can be described as the sum of the complexity of the subblocks. The main points where differences between the proposed method and conventional DVC may occur is in the conditional entropy estimation block, the quantizer and the real field encoder. The complexity of the DCT operation will be the same for all transform domain systems.

Real field encoding compared to finite field coding will have the same number of operations (assuming the same code size), but the type of operation will differ: Real field (floating point)

multiplication and addition versus Galois field multiplication and addition. The complexity of the different operations will be platform and implementation specific. Typically one would expect finite field operations to be faster than the real field equivalent, however many modern processors are optimized to perform floating point operations and can perform them at high speed.

A similar point can be made for bit-plane based systems with binary codes. These codes will have a larger number of operations compared to symbol based coding, but the operations will be simple binary XOR and AND operations which can be much faster.

The proposed conditional entropy estimator is faster than the conventional method (see Section 4.6.2), $O(1)$ vs. $O(q_B N)$ per subband, but the complexity of creating the side information estimate is also at least $O(N)$, so this does not represent a very large saving. The quantizers will also be similar in complexity though the proposed system only quantizes ($M < N$) values compared to the N values of a conventional system. Overall, the encoding complexity of the proposed system is expected to be similar to that of conventional DVC without motion estimation at the encoder. Exact comparisons will be implementation specific.

4.5.2 Decoder Time Complexity

Decoding complexity is generally not considered a limitation for DVC and is not often considered when evaluating performance in DVC systems. However, I will briefly discuss the effect of real field coding on the decoder complexity. The reduced complexity GBP algorithm used for decoding is similar to a binary BP decoding algorithm in terms of memory requirements and computational complexity. In GBP each edge requires two values to be transmitted while for binary BP one value is required. The update rules are also simple equations that scale with the check and variable node degree distributions. However, binary coded systems decode each bit-plane independently, thus it has to perform more than one decoding for each subband. The GBP also converges in less than 10 iterations, which is less than most binary BP algorithms. GBP is thus expected to compare favourably with binary BP in terms of decoding speed. Non-binary BP is typically slower than binary BP and requires more memory. Each edge must transmit the entire PMF (L values). The update equations at the check node are also slow. A faster FFT-BP algorithm still requires two FFT operations for every edge. Typically, FFT-BP is considered to be $O(q_b)$ times slower than binary BP. There are lower complexity non-binary BP algorithms in the literature, but discussing their relative merits is beyond the scope of this

thesis. In general, GBP is expected to be faster than non-binary BP.

4.5.3 Encoder Space Complexity

The theoretical space complexity or memory use of the encoder is dominated by the raw video input signal. For an input frame with $I \times J$ 8 bit pixels the required memory is $8IJ$ bits. The number of input frames in memory required for encoding is $N_G + 1$. Each input frame is individually processed. The processing requires an intermediate frame as part of the rate estimation as well as an intermediate floating point frame for the result of the DCT and another for the result of the real field encoding. Assuming a 32 bit floating point representation, this brings the required memory to $8IJ(N_G+2)+32IJ*2 = 8IJ(N_G+10)$ bits. Any other intermediate variables will be negligible by comparison. The space complexity will thus scale linearly with the number of pixels per frame. It will also scale linearly with the GOP size.

Due to the requirement of keeping the whole GOP in memory during encoding, the space complexity of DVC is larger than standard intra coding.

4.5.4 Decoder Space Complexity

The theoretical decoder space complexity is a combination of the memory required for the frames as well as the memory required by the GBP decoding algorithm.

The number of frames required to be stored in memory is $N_G + 1$ as well as one for the SI. A floating point frame is also required for the DCT of the frame being decoded. This yields a requirement of approximately $8IJ(N_G + 6)$ bits assuming a 32 bit floating point representation.

The soft input to the GBP decoder requires two floating point values per coefficient. The GBP decoder itself requires another 4 floating point values per edge in the decoding graph as well as Mq_b bits for the parity symbols. If N is the number of coefficients and d_v is the average number of edges per variable node, this amounts to $N(2+4d_v)$ floating point values. N is related to the number of pixels by $N = IJ/N_B$ where N_B is the number of subbands in the DCT. For the 4×4 DCT used in this thesis, and a 32 bit floating point representation, this implies a storage requirement of $2IJ(2+4d_v)+Mq_b$ bits for the GBP decoder. This is equivalent to approximately d_v frames.

Thus, the total number of bits is approximately $8IJ(N_G+6+d_v)$ bits. The space requirements thus scales linearly with the GOP size and linearly with the density of the decoding graph.

4.6 ANALYSIS OF PROPOSALS

This section starts by analysing the performance of the real field codes. Next, the proposed rate estimation method is analysed. Finally, the performance of the full system is analysed.

4.6.1 Performance of Real Field Codes

4.6.1.1 Simulation Setup

In this section the real field coding method is evaluated using synthetic data so as to characterise the performance of the system. A Laplace distributed data signal, with variance 1 and $N = 1584$ is generated. A Laplace distributed noise signal with variance σ_e^2 is then added to yield the SI. The signal to noise ratio is calculated as :

$$\text{SNR} = 10 \log_{10} \frac{1}{\sigma_e^2}$$

The original signal is then encoded to M symbols and quantized to q_b bits per symbol as described in this chapter. The SI is then decoded along with the quantized encoded signal using the GBP algorithm. In this section, the performance of the proposed method is measured for different values for M and q_b at different SNR values and number of decoding iterations. The performance gain of decoding will be indicated by the ratio of the output noise variance (σ_d^2) to the input noise variance ($\alpha = \sigma_d^2/\sigma_e^2$) - smaller values are better.

4.6.1.2 Performance of real field codes

In order to demonstrate the ability of real field codes to reduce the noise variance in the side information, the ratio of the noise variance after decoding (σ_d^2) to the noise variance of the side information (σ_e^2) is analysed as a function of the number of parity symbols. Figure 4.5 shows this ratio, with the number of bits per symbol (q_b) set to 12 so as to remove the effect of quantization noise, for 10 and 100 decoding iterations.

From the figure, one can see that the output variance is always lower than the input noise variance. For lower SNR values (more noise), the ratio is lower indicating a greater gain. This is partly due to the use of the signal prior. In general, the system performs better than predicted by Eq. (4.1) which is also plotted in the figure. This is due to the use of the prior and the Laplace distributed noise instead of the Gaussian distributed noise. From the figure it can also be seen that the number of decoding iterations can affect the performance for higher values of

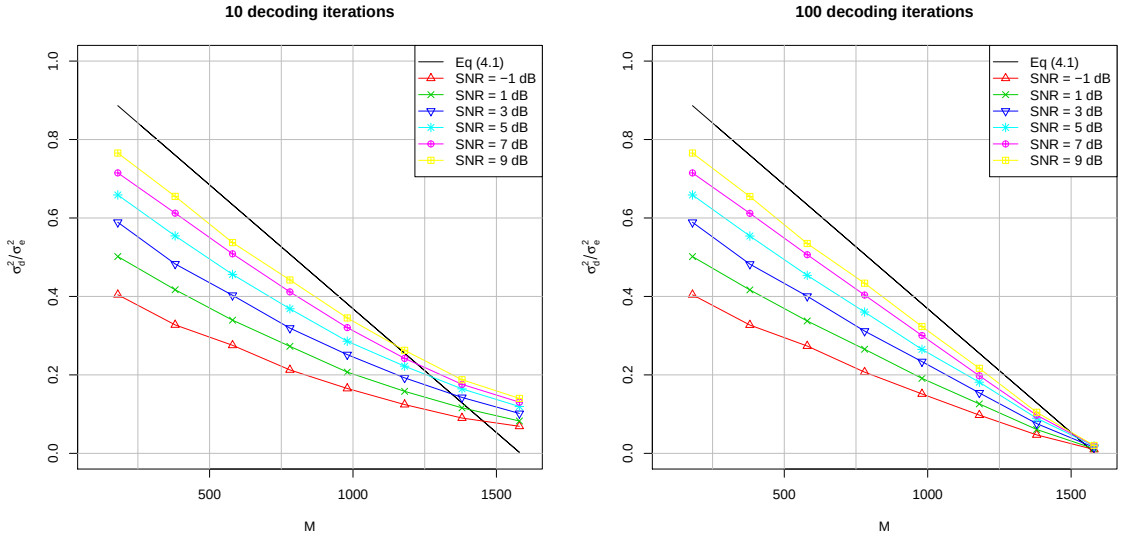

FIGURE 4.5: The ratio of the output decoded noise variance to the input noise variance as a function of the number of parity symbols (M). The number of bits per symbol was set to $q_b = 12$, to remove the effect of the quantization. The number of iterations refers to the number of decoding iterations in the GBP decoder.

M. Specifically, the codes take longer to converge for larger M. More decoding iterations will increase the decoding complexity but not the encoding complexity.

To see whether the quantization of the encoded signal affects the functioning of the real field code, the same ratio is plotted for $q_b = 4$ and $q_b = 6$ in Figures 4.6 and 4.7 respectively.

From Figure 4.6 one can see that when the signal is significantly quantized the performance improvement plateaus. In contrast to the case for $q_b = 12$, increasing the number of decoding iterations does not significantly improve performance. When $q_b = 6$, there is less of a plateauing effect and similar to the case for $q_b = 6$, increasing the number of decoding iterations improves performance. Important to notice, is that in all cases the gain achieved by the real field coding is a smooth function of the rate.

Figure 4.8 shows the same ratio as above ($\alpha = \sigma_d^2/\sigma_e^2$) for a range of q_b and SNR values using 10 decoding iterations.

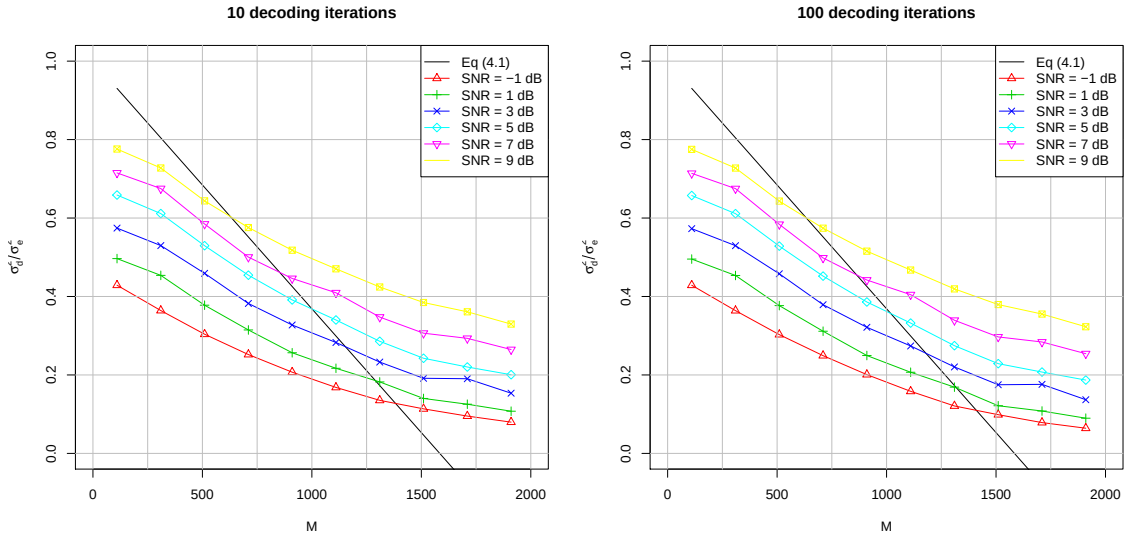

FIGURE 4.6: The ratio of the output decoded noise variance to the input noise variance as a function of the number of parity symbols (M). The number of bits per symbol $q_b = 4$. The number of iterations refers to the number of decoding iterations in the GBP decoder.

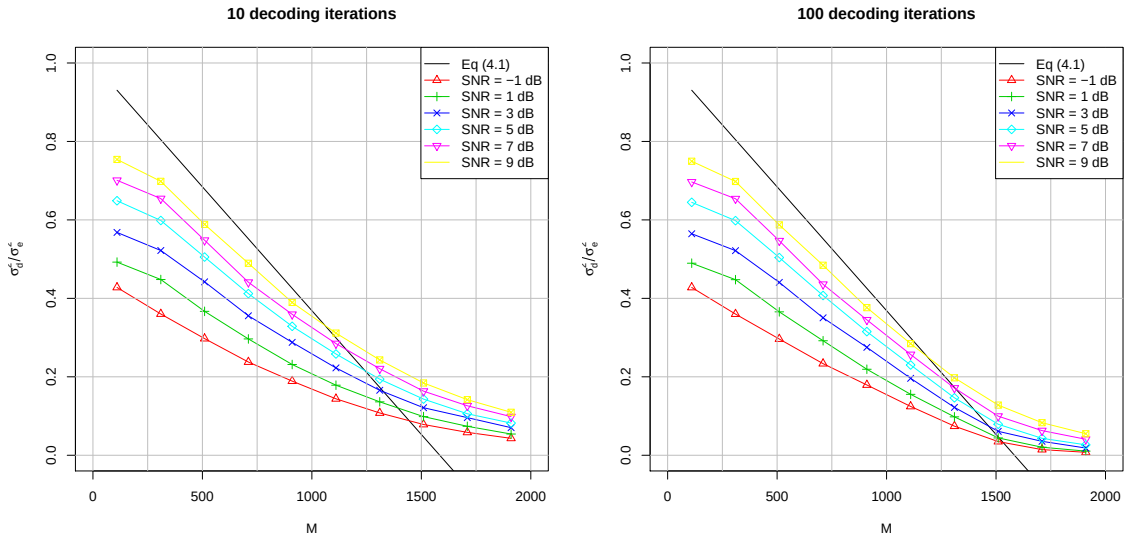

FIGURE 4.7: The ratio of the output decoded noise variance to the input noise variance as a function of the number of parity symbols (M). The number of bits per symbol $q_b = 6$. The number of iterations refers to the number of decoding iterations in the GBP decoder.

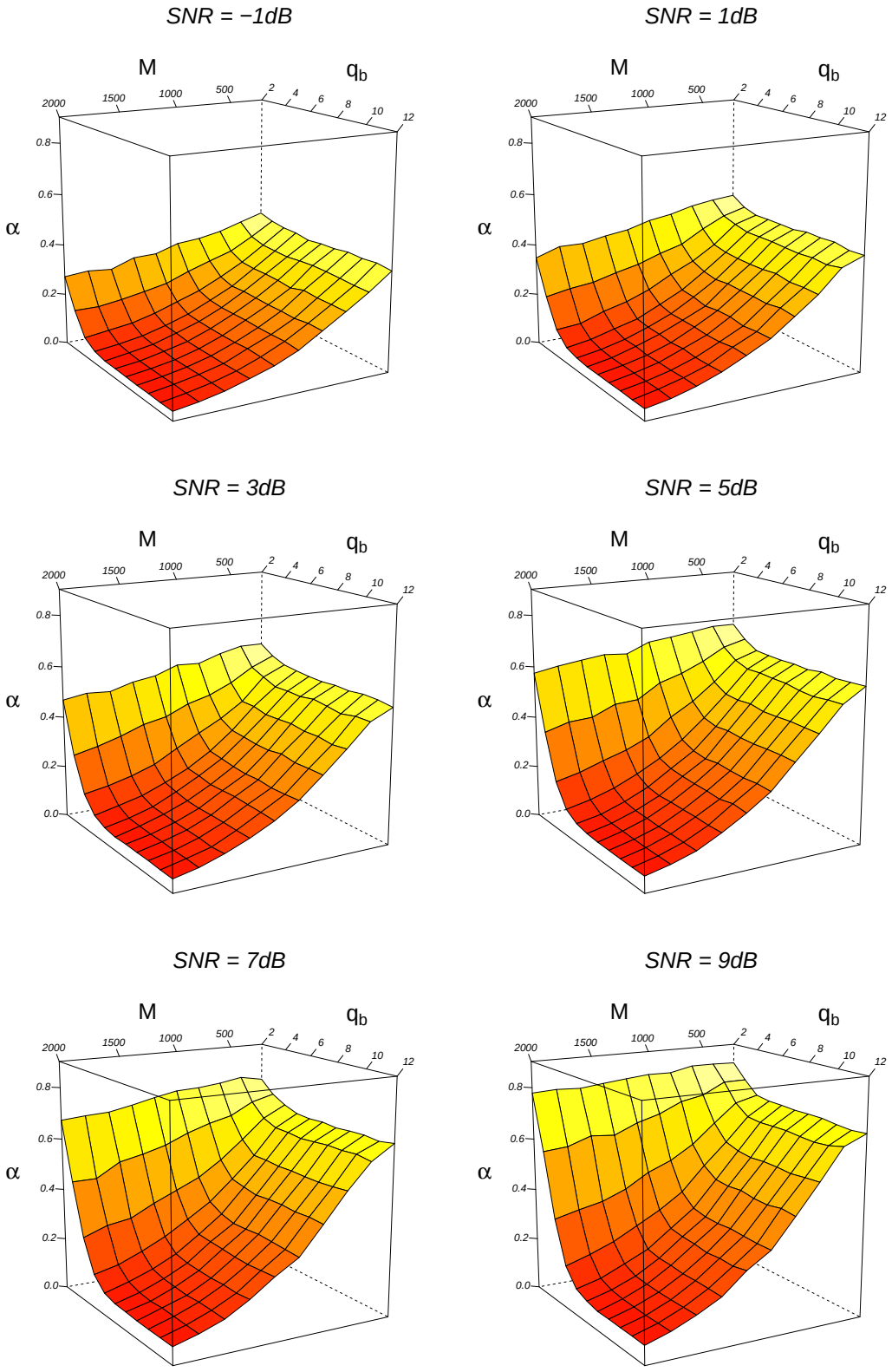

FIGURE 4.8: The ratio of the output decoded noise variance to the input noise variance ($\alpha = \sigma_d^2/\sigma_e^2$) as a function of the number of parity symbols (M) and the number of bits per symbol (q_b). The number of iterations in the GBP decoder is 10.

4.6.2 Analysis of Proposed Rate Estimation Method

In this section we compare the proposed method with the theoretically calculated method and the bit-plane method from [18] described in section 2.3.

Figure 4.9 shows the error of the different methods used to estimate the rate for a mid-rise quantizer as measured against the reference rate. From the figure one can see that when the number of bits in the quantizer is high (high rate assumption, small Δ) both the mid-tread and the mid-rise quantizer entropy estimates give good results. However, when the number of bits in the quantizer is small (large Δ) both methods are inaccurate. However, averaging the entropies as calculated using the mid-tread and mid-rise quantizers yields a reasonably accurate result even for large bin sizes. This corresponds to the "Proposed adjusted" curve in Figure 4.10.

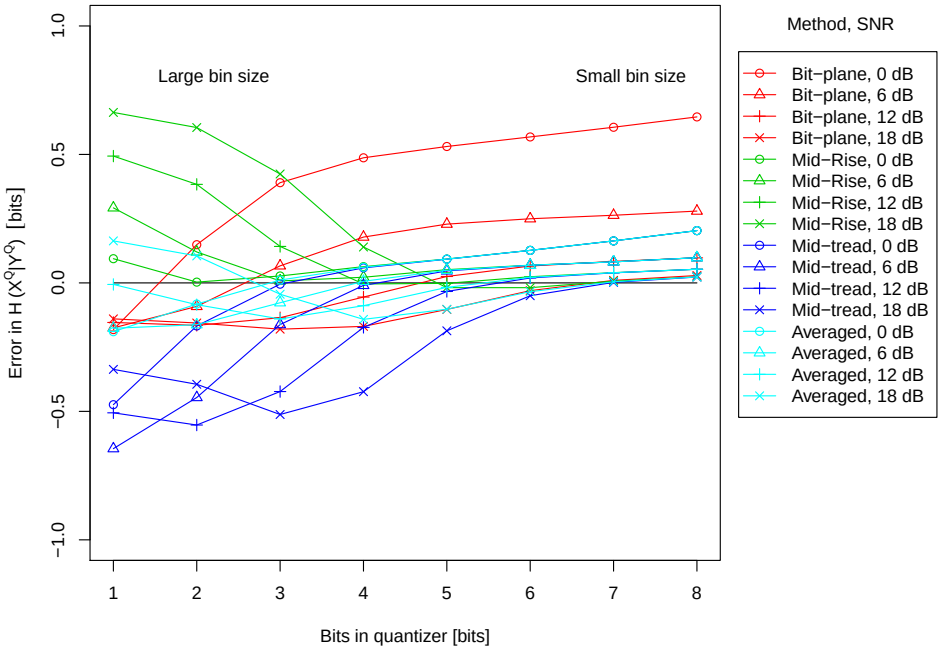

FIGURE 4.9: The error in the estimate of the conditional entropy. The curves show the error for different numbers of bits, from 1 to 8, for different SNR values.

Figure 4.10 shows $H(X^{\mathrm{Q}}|Y^{\mathrm{Q}})$ as a function of the signal-to-noise ratio (SNR) for symbols with an increasing number of bits, from 1 to 8. The figure shows the results for the mid-rise uniform quantizer. The proposed method uses the mid-rise quantizer function $H^{S}(\cdot)$, for $H(X^{\mathrm{Q}})$, $H(Y^{\mathrm{Q}})$ and $H(E^{\mathrm{Q}})$ as this allows for the lowest computational complexity. The proposed adjusted curve uses $H^{\mathrm{S}}(\cdot)$, for $H(X^{\mathrm{Q}})$ and $H(Y^{\mathrm{Q}})$ and $(H^{\mathrm{S}}(\cdot) + H^{\mathrm{M}}(\cdot))/2$ for $H(E^{\mathrm{Q}})$. From this

figure one can see that the proposed adjusted curve yields more accurate results.

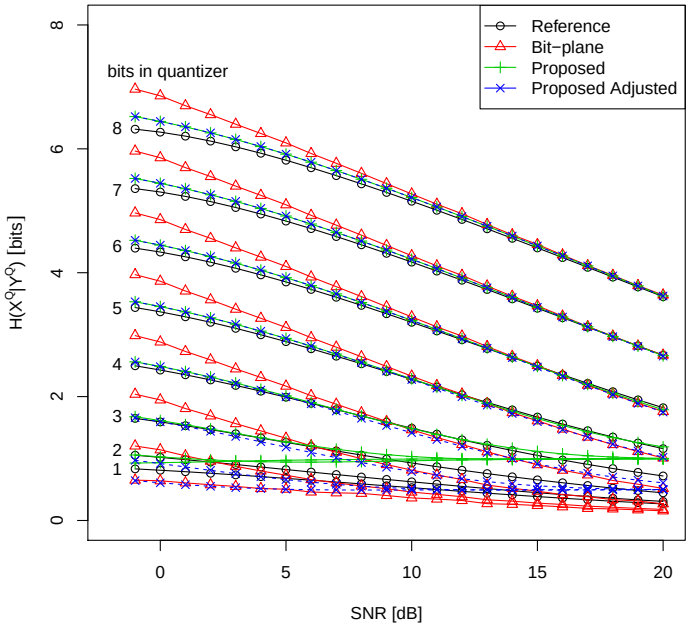

FIGURE 4.10: The conditional entropy as a function of the signal to noise ratio. The curves show $H(X^Q|Y^Q)$ for increasing numbers of bits, from 1 to 8.

4.6.3 Performance of Proposed Feedback Free System

4.6.3.1 Simulation Setup

The simulation setup is similar to to the one presented in Chapter 3, but repeated here for convenience. The proposed feedback free system is tested on four standard test sequences: *foreman*, *coastguard*, *hall monitor* and *soccer*. The sequences are all QCIF format with a frame rate of 15 Hz. All the rate-distortion curves show results for the average bit rate and the distortion as calculated over all the frames in the entire sequence (key and WZ frames). All sequences consist of 150 frames.

The systems will be compared against the H.264 intra codec as well as with the DISCOVER codec [25], which is one of the benchmark systems in DVC literature. DISCOVER is a single-hypothesis bit-plane based system. Key frames were encoded using the reference implementation of the intra mode of the H.264 codec. The QP parameters used are similar to those used by the DISCOVER codec and are provided in Table 4.2. The other simulation parameters are provided in Table 4.1.

Finally, the system will be compared with other feedback free systems in the literature ([17], [18] and [9]), thought this comparison is not as complete due to a lack of available results for the other systems.

Table 4.1: The values of the codec parameters used for the experiments performed in this section.

Simulation parameter	Value
Mean filter window size	3×3
MCI large block median filter window size	5×5
MCI small block median filter window size	7×7
DCT block size	4×4
MCI large block size	16×16
MCI small block size	8×8
MCI large block search range	48
MCI small block search range	24

4.6.3.2 Comparison of Proposed Method with DISCOVER

The performance results for the proposed feedback free systems on the *foreman*, *coastguard*, *soccer* and *hall monitor*, sequences can be seen in Figures 4.11, 4.12, 4.13 and 4.14, respectively.

The same trends are apparent in all the sequences. The proposed multi-hypothesis system performs comparably with the DISCOVER codec in the low rate regime, but then deviates and performs less well in the high rate regime. In general, the performance loss of the proposed system as measured against the DISCOVER codec increases with the GOP size. For the low motion sequence *hall monitor* the proposed method performs slightly better than the DISCOVER codec in the low and medium rate regions for a GOP size of 2.

Compared to the H.264(intra) codec, the proposed method performs worse in the higher motion sequences (*soccer* and *foreman*) and better in the low motion sequences (*coastguard* and *hall monitor*) in the low to medium rate regime for a GOP size of 2.

Table 4.2: Key frame quantization parameters for the RD points. The same values are used for all GOP sizes.

RD number	1	2	3	4	5	6	7	8
Hall monitor	37	36	35	33	32	30	29	24
Coastguard	38	37	36	34	33	31	30	26
Foreman	40	39	38	34	33	31	29	25
Soccer	43	42	40	36	35	33	30	25

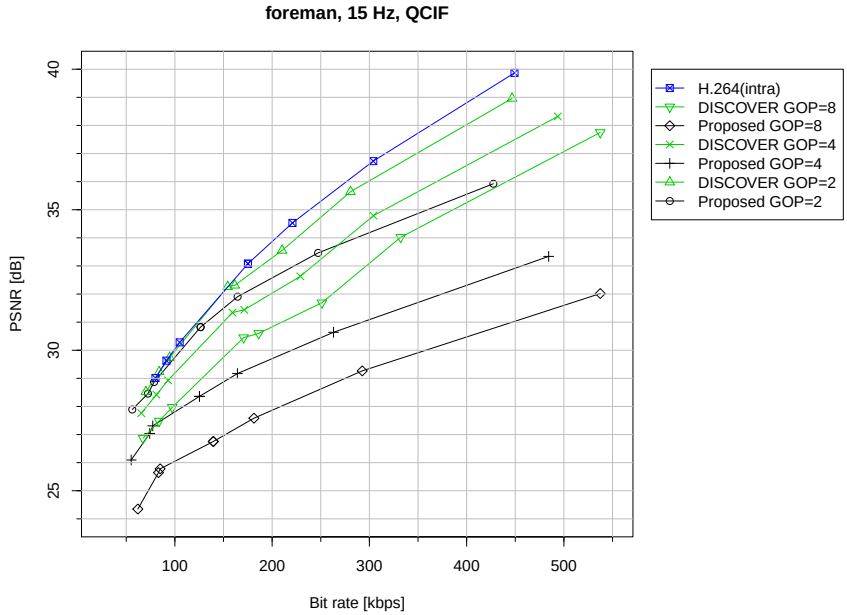

FIGURE 4.11: The performance of the proposed feedback free system on the *foreman* sequence compared to the DISCOVER codec as well as the H.264 intra codec.

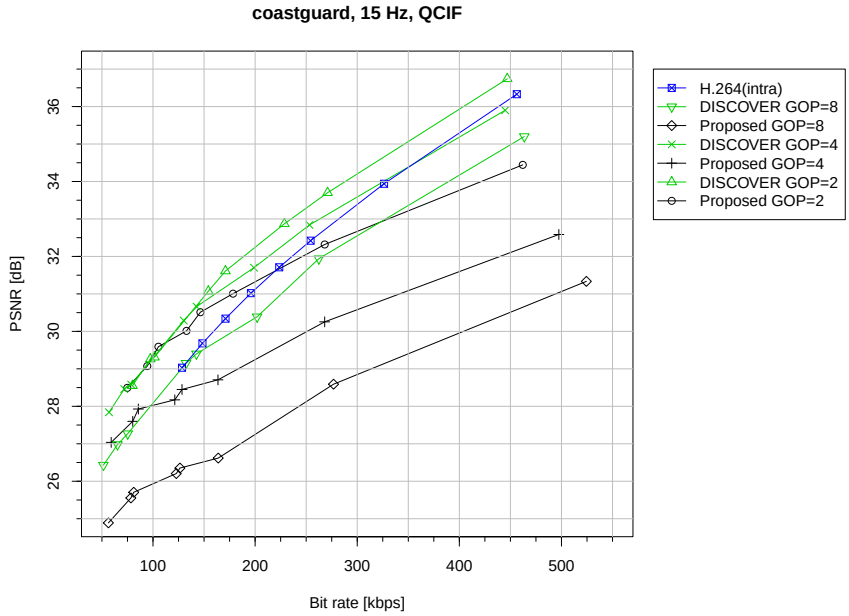

FIGURE 4.12: The performance of the proposed feedback free system on the *coastguard* sequence compared to the DISCOVER codec as well as the H.264 intra codec.

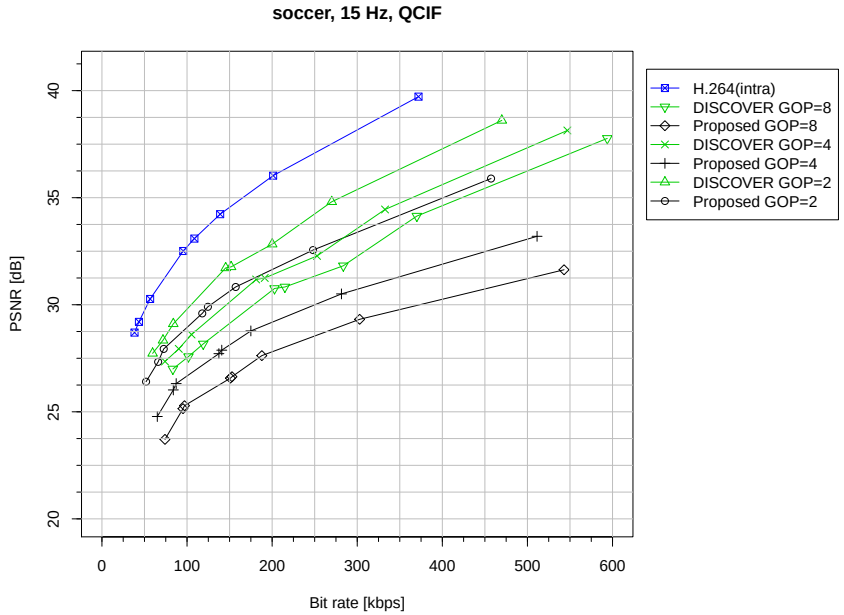

FIGURE 4.13: The performance of the proposed feedback free system on the *soccer* sequence compared to the DISCOVER codec as well as the H.264 intra codec.

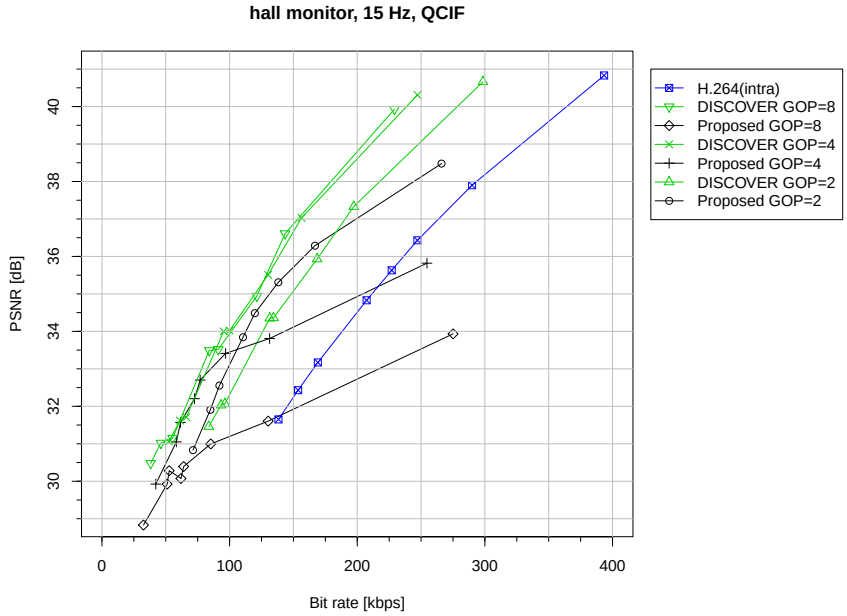

FIGURE 4.14: The performance of the proposed feedback free system on the *hall monitor* sequence compared to the DISCOVER codec as well as the H.264 intra codec.

4.6.3.3 Comparison of Proposed Feedback Free System with conventional methods

Comparisons with previously developed feedback free systems are problematic due to several reasons. Firstly, inconsistent testing conditions used by different proposals make it difficult to facilitate fair comparisons. Some methods for example do not consider the effect of imperfect key frames at the decoder or alternatively use high temporal sampling rates to improve performance. Results are also not available for larger GOP sizes.

However, the methods proposed by Brites and Pereira in 2007[17] and 2011[18] use similar testing conditions to the ones presented in this thesis. Results are only available for a GOP of 2 for the *foreman*, *coastguard* and *hall monitor* sequences. Results were also obtained for the system presented by Weerakkody et al. in [9] for the *foreman* and *coastguard* sequences for a GOP of 2. The comparison of the results for the *foreman*, *coastguard* and *hall monitor* sequences can be seen in Figures 4.15, 4.16, and 4.17 respectively.

From the results one can see that the proposed system performs better that the system in [9] in all scenarios. For the *foreman* and *coastguard* sequences, the proposed codec performs better than [17] and similar to [18] at low to medium rates and worse at higher rates. For the *hall monitor* sequence the proposed method performs better than [17] at all rates. For the *foreman* and *coastguard* sequences, the proposed codec performs similar to [18] at low to medium rates and worse at higher rates. For the *hall monitor* sequence the proposed method performs better than [18] at low to medium rates with equivalent performance at higher rates.

While the methods in [17] and [18] perform well, they both employ motion estimation at the encoder which increases encoder complexity.

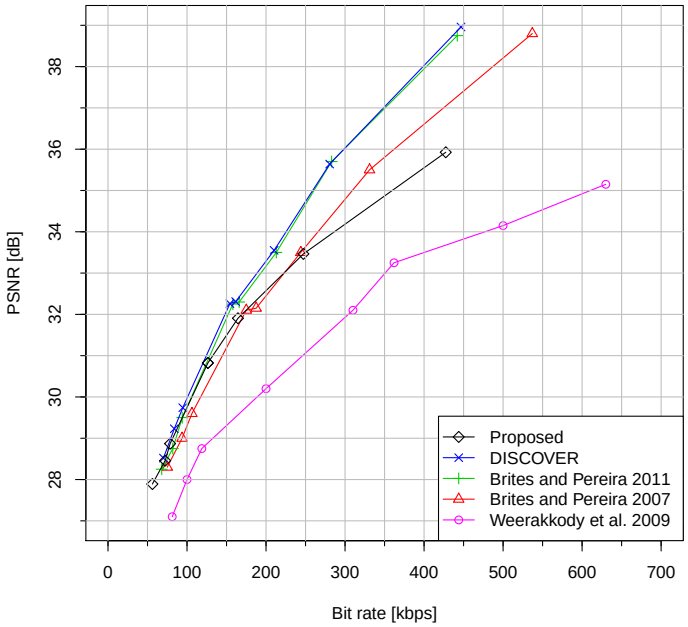

FIGURE 4.15: A comparison of the proposed feedback free system with conventional methods on the *foreman* sequence.

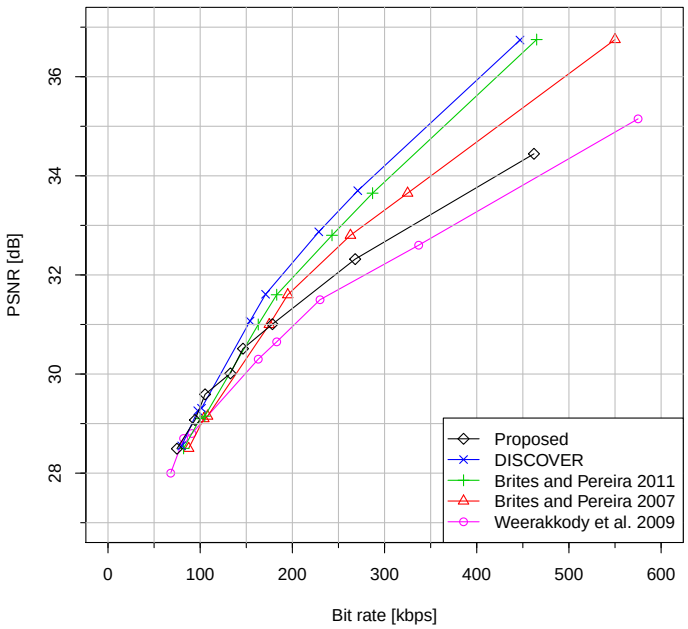

FIGURE 4.16: A comparison of the proposed feedback free system with conventional methods on the *coastguard* sequence.

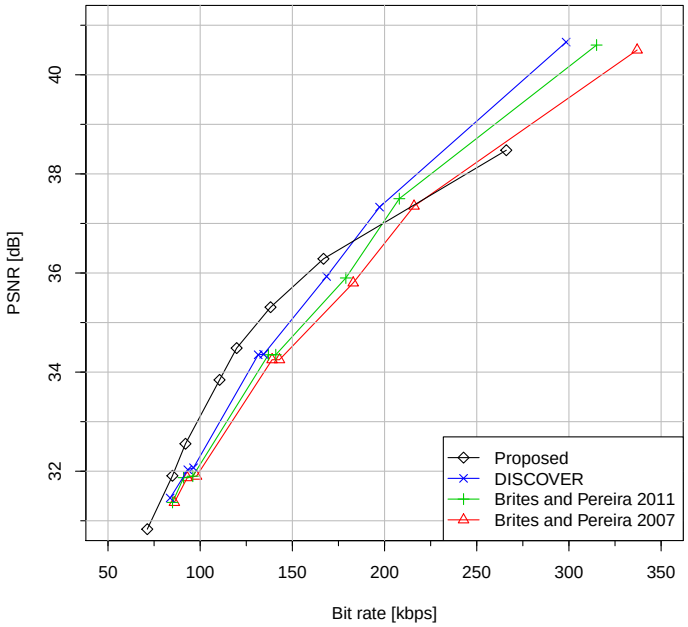

FIGURE 4.17: A comparison of the proposed feedback free system with conventional methods on the *hall monitor* sequence.

4.7 CONCLUSIONS

In this chapter, a feedback free DVC system was proposed. The system is able to operate without feedback due to two factors:

1. The nature of the coding operation was changed to allow the system to operate even when the estimated rate estimate is inaccurate. This was done by swapping the encoder and the quantizer around and defining the encoder over the real field. It was shown that the real field code yields a smooth distortion as a function of the rate which makes the system more robust to rate estimation errors.

2. A low complexity method for estimating the required rate was developed. The proposed method is both more accurate and less computationally expensive than conventional methods.

The performance analysis showed that the system was able to operate without feedback at a similar quality to feedback based systems in the low and medium rate regime at a GOP size of 2.

Chapter Five

Multi-hypothesis Feedback Free DVC

In this chapter the feedback free system developed in Chapter 4 is improved by using multiple hypotheses. The hypotheses are combined by using the Bayesian fusion method developed in Chapter 3. The specific hypotheses used in the initial decoding are the same as in Chapter 3, but different methods are used in the refinement stage.

This chapter starts with a quick description of the overall system. This is followed by describing the side information creation process as far as it differs from the work in Chapters 3 and 4.

5.1 Proposed System

Figure 5.1 shows a block diagram of the proposed system. The encoder of the system is the same as for the system presented in Chapter 4. In order to use the multiple hypotheses as described in Chapter 3, the decoder is adapted.

At the decoder, the key frames and possibly other decoded frames are used to create an estimate of f. In the multi-hypothesis case more than one estimate of f is created. The h^{th} hypothesis will be denoted by $\hat{f}_h$. For this system we use four hypotheses as described in Section 3.2.1. Each hypothesis is transformed to produce $\hat{F}_h$ and each subband is vectorized to yield $\mathbf{y}_{hb}$:

$$\begin{aligned} \hat{F}_h &= \mathrm{DCT}(\hat{f}_h) \\ \mathbf{y}_{hb} &= vect(\hat{F}_{hb}), \end{aligned}$$

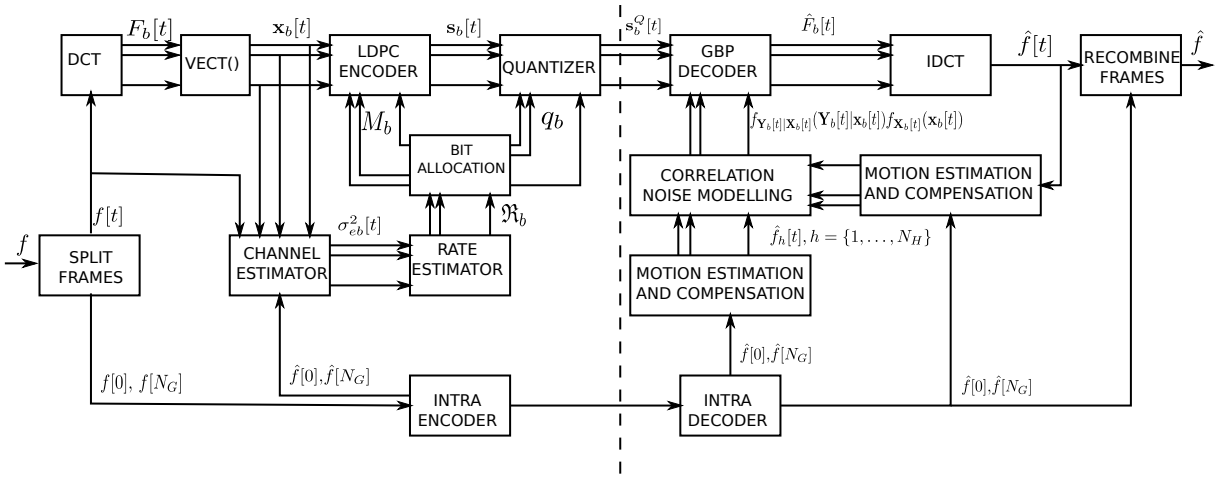

FIGURE 5.1: Diagram of proposed multi-hypothesis feedback free system.

where $\hat{F}_{hb}$ is the b^{th} subband of $\hat{F}_h$. If the error for each hypothesis is given as:

$$r_h \quad = \quad f - \hat{f}_h,$$

we calculate an estimate $\hat{r}_h$. The method used to calculate the estimated error will differ for each hypothesis and depend on the method used to produce the hypothesis. From the error estimates, noise parameters are calculated at a coefficient level using Eq. (2.4). In order to improve the relative quality of the noise estimation for the different hypotheses, a motion adaptive scaling factor is used to scale the Laplace noise parameters of the key frame and motion based hypotheses as described in 3.3.7.

For each subband, the error estimates and the N_{H} hypotheses are combined with the received parity information and decoded using a Gaussian BP (GBP) algorithm which attempts to solve the equation:

$$\hat{x}_n \quad = \quad \arg\max_{x_n \in \mathbb{R}} f_{X|\mathbf{Y},\mathbf{S}^{\mathrm{Q}}}(x_n|\mathbf{Y},\mathbf{s}^{\mathrm{Q}}),$$

where n indicates the n^{th} coefficient in a subband with N coefficients and the subband indices are dropped for notational brevity. $\mathbf{Y}$ is an $N_{\mathrm{H}} \times N$ matrix with entries y_{hn} where the h^{th} row represents the N received symbols of the h^{th} hypothesis. As opposed to standard DVC, no reconstruction is required. The decoded bands are recompiled to produce $\hat{F}$, and then the pixel domain output frame $\hat{f}$ is calculated:

$$\hat{f} \quad = \quad \mathrm{IDCT}(\hat{F}).$$

After decoding, the process is repeated in a method similar to successive refinement [75–78]. Using the decoded frame, $\hat{f}$, a more accurate motion compensated interpolation (MCI)

algorithm can be used to produce higher quality SI. New correlation noise estimates must then be produced for the SI, after which the GBP algorithm is used to decode the improved SI. This process may be iterated a number of times, though two iterations are typically sufficient. If the original decoding introduced an error, then the iterative process may worsen performance as the MCI process may produce worse quality SI. We will now discuss the subsections in more detail.

5.2 IMPLEMENTATION DETAILS

In this section the subsections and algorithms used in the decoder are discussed.

5.2.1 Side Information Creation

The decoder uses a successive refinement approach. This means that there is an initial SI creation process. The SI is then decoded. After decoding, the SI creation process is repeated and improved using the decoded frame.

5.2.1.1 Initial Decoding

A multi-hypothesis SI creation method using four hypotheses is used. This method is similar to the one developed in Chapter 3. The hypotheses are the two reference frames, $\hat{f}[0]$ and $\hat{f}[N_G]$; a motion compensated interpolation (MCI) based frame using small blocks, $\hat{f}_S$, as well as an MCI based frame using big blocks, $\hat{f}_B$. The MCI hypotheses are the same as those described in Section 3.3.3.

5.2.1.2 Successive Refinement

After BP decoding, the subbands are recombined and a pixel domain version of the WZ frame is produced. Using this decoded frame, the motion estimation process is repeated to produce better quality side information. This is similar to [75–78] where successive refinement was used to improve reconstruction distortion. Due to the use of real field coding, the effect is slightly different from conventional DVC as improved side information can result in decreased distortion from the GBP decoding process, whereas for finite field coding there can be no further gain (other than improved reconstruction) after successful decoding of the bit planes.

In order to use the decoded frame, the MCI method needs to be adjusted. Three different hypothesis frames are created. The first method performs pixel level accuracy bilinear BM ME

using the decoded frame as a hash to produce $\hat{f}_{\mathrm{I}}$. This is the same as the single hypothesis used in the refinement phase for the proposed single hypothesis feedback free system in Chapter 4.

The other two hypotheses, $\hat{f}_{\mathrm{L_r}}$ and $\hat{f}_{\mathrm{R_r}}$, are created by matching the key frames individually with the reconstructed frame. Figure 5.2 shows a representation for a GOP size of 2. This allows non-linear motion to be tracked across the GOP. For these hypotheses, a simple SAD

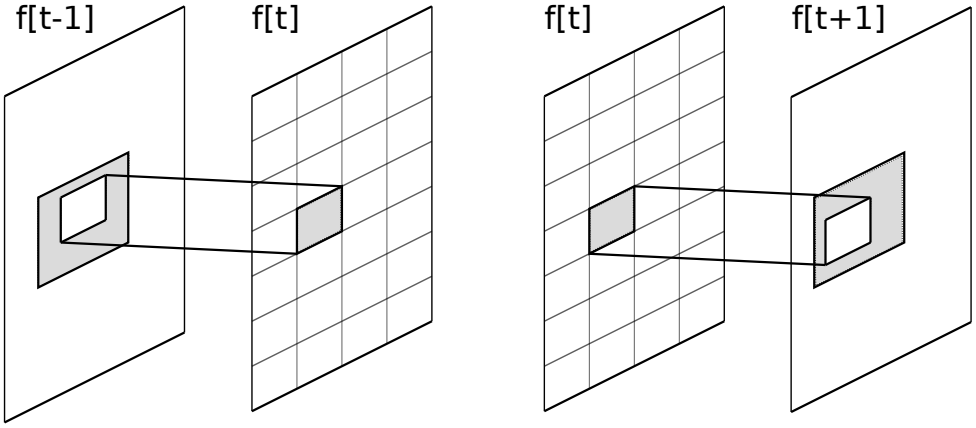

FIGURE 5.2: Single Directional Motion Estimation

metric is used:

$$\begin{aligned} M_{\mathrm{L_r}}(L, M) &= \sum |L - M| \\ M_{\mathrm{R_r}}(R, M) &= \sum |R - M|. \end{aligned}$$

The metrics are all adjusted by including a motion penalising term as described in Eq. (2.3), which helps to ensure less noisy motion vectors.

5.2.2 Correlation noise modeling

For the multi-hypothesis system, a residual frame estimate is required for each hypothesis. For the initial decoding, the residual frame estimates are calculated in the same way as for the system proposed in Chapter 3. The correlation noise for each hypothesis is also estimated using the same method as for the system proposed in Chapter 3.

All the hypotheses are then combined at a coefficient level using Bayesian fusion:

$$f_{\mathbf{Y}|X}(\mathbf{y}|x) = \prod_{h=1}^{N_H} f_{Y_h|X}(y_h|x),$$

before being passed to the GBP algorithm for decoding.

5.2.2.1 Residual estimation for later iterations

After the frame has been decoded the first time, we calculate new side information frames using the methods discussed in the previous section. For $\hat{f}_{\mathrm{I}}$ the same residual estimate is used as described in Chapter 4. For the two new hypotheses, $\hat{f}_{\mathrm{L_r}}$ and $\hat{f}_{\mathrm{R_r}}$ we estimate a residual frame for each as:

$$\begin{aligned}\hat{r}_{\mathrm{L_r}} &= (1-\lambda_{\mathrm{L_r}})|\hat{f}_{\mathrm{L_r}} - \hat{f}_{\mathrm{R_r}}| + \lambda_{\mathrm{L_r}}|\hat{f}_{\mathrm{L_r}} - M| \\ \hat{r}_{\mathrm{R_r}} &= (1-\lambda_{\mathrm{R_r}})|\hat{f}_{\mathrm{L_r}} - \hat{f}_{\mathrm{R_r}}| + \lambda_{\mathrm{R}}|\hat{f}_{\mathrm{R_r}} - M|.\end{aligned}$$

Here, M is the previous decoded frame, and the λs are tunable parameters. While they may be optimized in future, we used $\lambda_{\mathrm{L_r}} = \lambda_{\mathrm{R_r}} = 0.5$.

From the residual estimate we calculate a correlation noise estimate for each hypothesis. The same method as described in Chapter 4 is used for each hypothesis. The hypotheses are then combined using Bayesian fusion in the same was as for the initial decoding.

5.3 DISCUSSION AND SUMMARY

In this chapter, the multi-hypothesis method described in Chapter 3 was applied to the proposed feedback free system in Chapter 4. Multiple hypotheses were used for both the initial decoding as well as for the refinement stage. The encoder is the same for both the single and multi-hypothesis methods. The performance analysis of the proposed multi-hypothesis system as well as a comparison with the single-hypothesis system is provided in Chapter 6.

Chapter Six

Performance analysis of Proposed Multi-hypothesis Feedback-Free DVC System

In this chapter, the performance of the proposed multi-hypothesis feedback free system will be analysed.

First the simulation setup is described in Section 6.1.

Then the proposed multi-hypothesis feedback free system from Chapter 5 is compared with the single hypothesis feedback free system from Chapter 4 in Section 6.2.

Next, the proposed multi-hypothesis feedback free system is compared with a conventional finite field based system operating at the same rate as well as the DISCOVER codec and the H.264(intra) codec in Section 6.3.

The perceptual quality of the proposed multi-hypothesis feedback free system is then analysed in Section 6.4.

This is followed in Section 6.5 by a comparison of the proposed system with other feedback free proposals in the literature.

Finally, the complexity of the proposed system is evaluated with run-time experiments in Section 6.6.

6.1 Simulation Setup

The simulation setup is similar to to the one presented in Chapter 4, but repeated here for convenience. The proposed feedback free system is tested on four standard test sequences:

foreman, *coastguard*, *hall monitor* and *soccer*. The sequences are all QCIF format with a frame rate of 15 Hz. All the rate-distortion curves show results for the average bit rate and the distortion as calculated over all the frames in the entire sequence (key and WZ frames). All sequences consist of 150 frames.

The systems will be compared against the H.264 intra codec as well as with the DISCOVER codec [25], which is one of the benchmark systems in DVC literature. DISCOVER is a single-hypothesis bit-plane based system. Key frames were encoded using the reference implementation of the intra mode of the H.264 codec. The QP parameters used are similar to those used by the DISCOVER codec and are provided in Table 6.2. The other simulation parameters are provided in Table 6.1.

The systems will also be compared against the 'FF system' which is the system developed in Chapter 3, except that it also uses the same estimated rate without feedback as the proposed feedback free system. The purpose of using the FF system is to highlight the effect of using real field coding while keeping most other aspects, such as the SI, constant. Comparisons with competing systems, while interesting, are more difficult to interpret.

Finally, the system will be compared with other feedback free systems in the literature ([17], [18] and [9]), though this comparison is not as complete due to a lack of available results for the other systems.

Table 6.1: The values of the codec parameters used for the experiments performed in this section.

Simulation parameter	Value
Mean filter window size	3×3
MCI large block median filter window size	5×5
MCI small block median filter window size	7×7
DCT block size	4×4
MCI large block size	16×16
MCI small block size	8×8
MCI large block search range	48
MCI small block search range	24

Table 6.2: Key frame quantization parameters for the RD points. The same values are used for all GOP sizes.

RD number	1	2	3	4	5	6	7	8
Hall monitor	37	36	35	33	32	30	29	24
Coastguard	38	37	36	34	33	31	30	26
Foreman	40	39	38	34	33	31	29	25
Soccer	43	42	40	36	35	33	30	25

6.2 COMPARISON OF PROPOSED SYSTEM WITH SINGLE AND MULTIPLE HYPOTHESES

The performance results for the proposed single hypothesis (SH Proposed) and multi hypothesis (MH Proposed) feedback free systems on the *foreman*, *coastguard*, *soccer* and *hall monitor*, sequences can be seen in Figures 6.1, 6.2, 6.3 and 6.4, respectively. From the figures, one can see that the MH system performs better than the SH system in all sequences and for all GOP sizes.

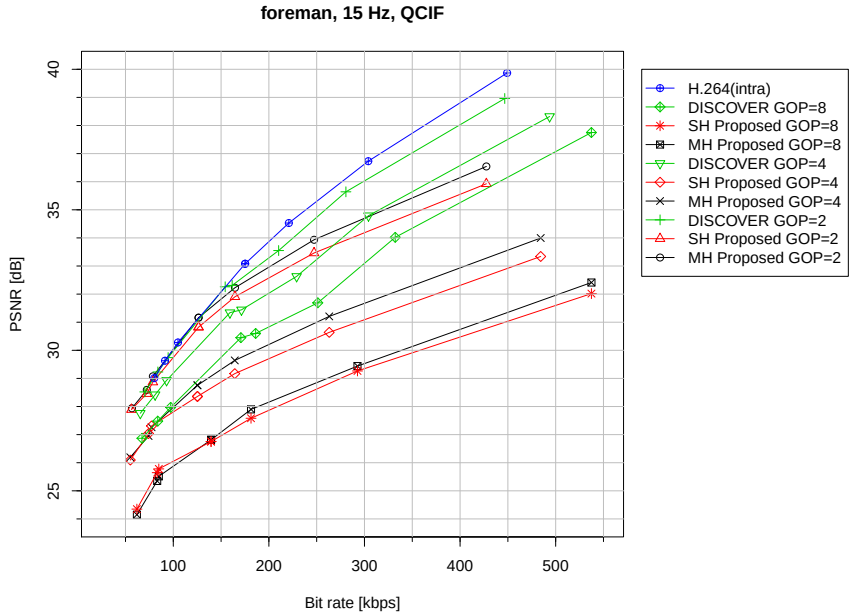

FIGURE 6.1: The performance of the proposed feedback free system with a single hypothesis (SH Proposed) and with multiple hypotheses (MH Proposed) on the *foreman* sequence compared to the DISCOVER codec as well as the H.264 intra codec.

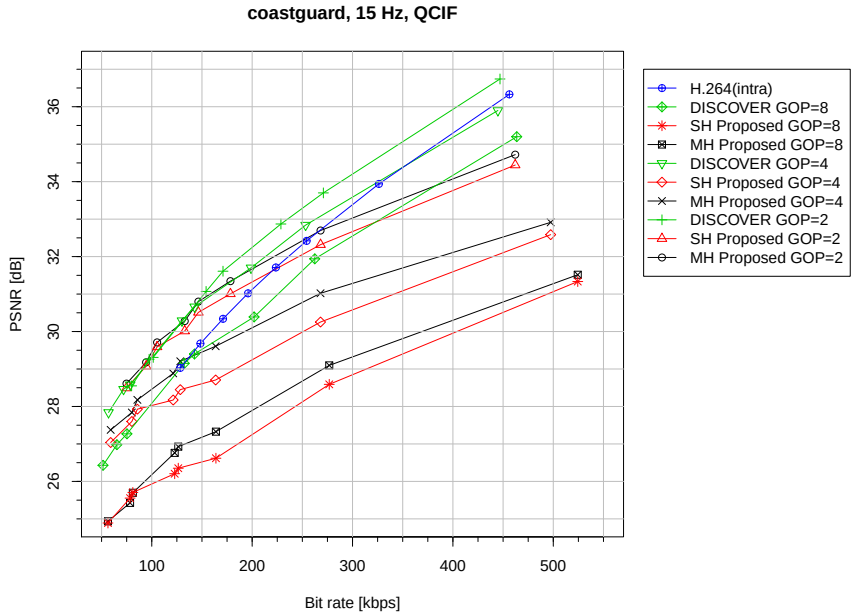

FIGURE 6.2: The performance of the proposed feedback free system with a single hypothesis (SH Proposed) and with multiple hypotheses (MH Proposed) on the *coastguard* sequence compared to the DISCOVER codec as well as the H.264 intra codec.

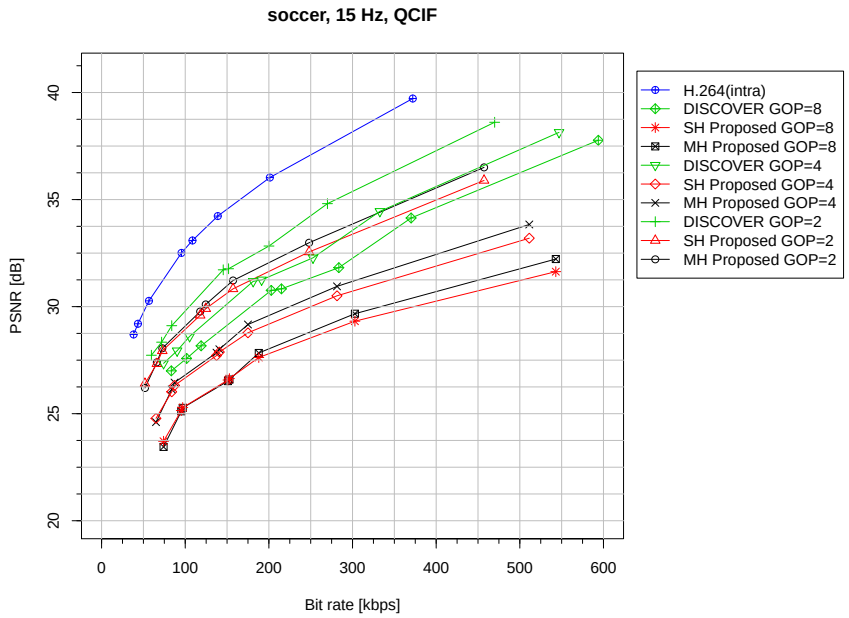

FIGURE 6.3: The performance of the proposed feedback free system with a single hypothesis (SH Proposed) and with multiple hypotheses (MH Proposed) on the *soccer* sequence compared to the DISCOVER codec as well as the H.264 intra codec.

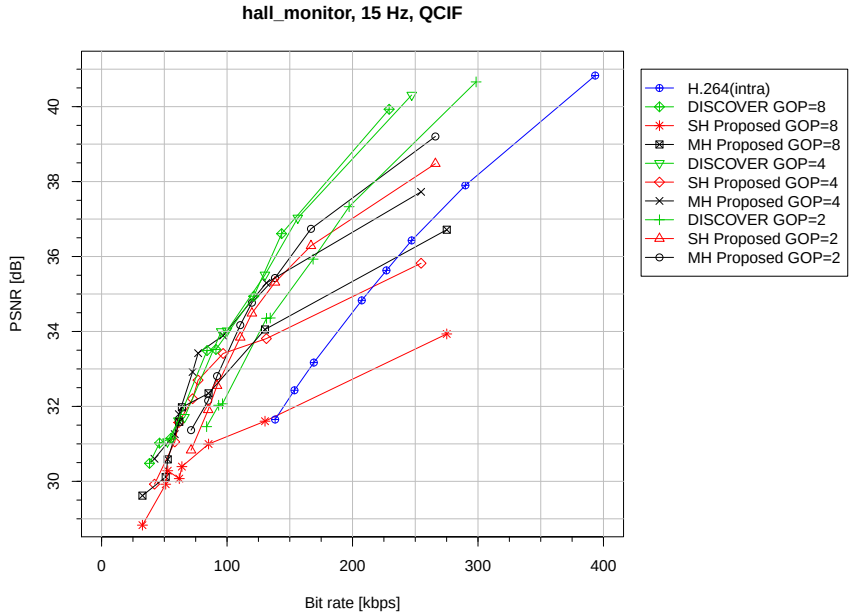

FIGURE 6.4: The performance of the proposed feedback free system with a single hypothesis (SH Proposed) and with multiple hypotheses (MH Proposed) on the *hall monitor* sequence compared to the DISCOVER codec as well as the H.264 intra codec.

6.3 COMPARISON OF PROPOSED SYSTEM WITH FINITE FIELD BASED SYSTEM

The performance results for the *foreman*, *coastguard*, *soccer* and *hall monitor*, sequences can be seen in Figures 6.5, 6.6, 6.7 and 6.8, respectively.

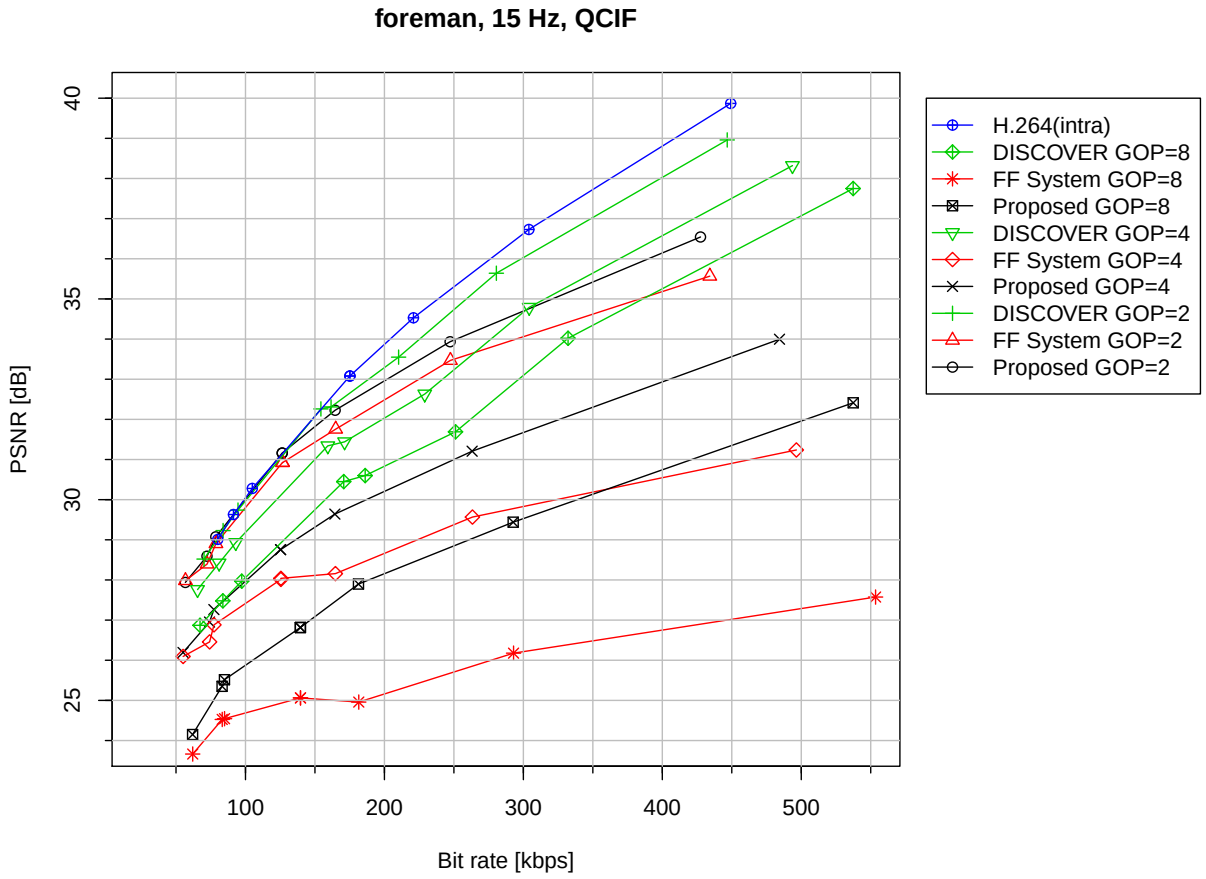

FIGURE 6.5: The performance of the proposed MH feedback free system on the *foreman* sequence compared to the DISCOVER codec as well as the H.264 intra codec and the FF system.

The same trends are apparent in all the sequences. The proposed multi-hypothesis system performs comparably with the DISCOVER codec in the low rate regime, but then deviates and performs less well in the high rate regime. The FF based system and the proposed system perform similarly in the *coastguard* sequence for the GOP size equal to 2 and 4. In all other cases the proposed system has better performance. The performance gain when the GOP size is equal to 2 is small, but increases at higher rates. The performance gain is also larger for GOP

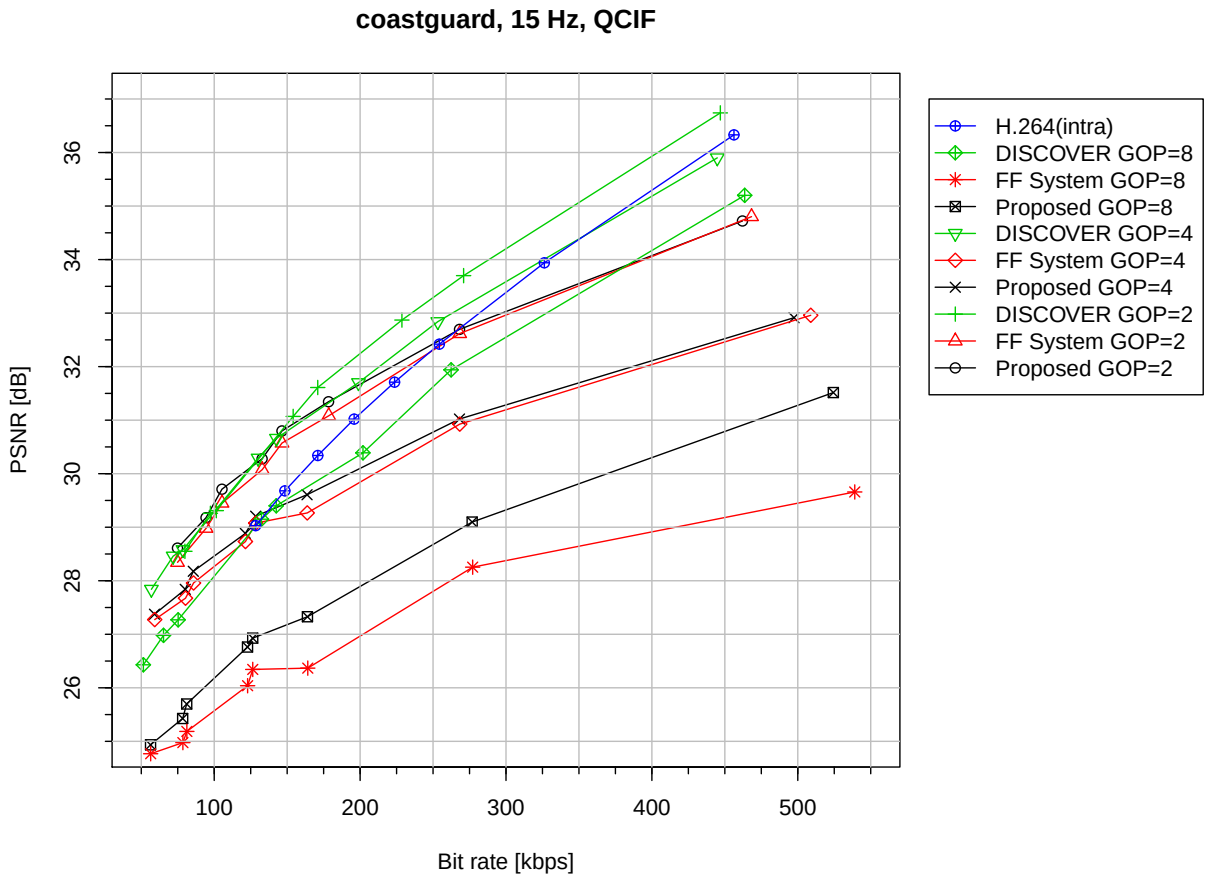

FIGURE 6.6: The performance of the proposed MH feedback free system on the *coastguard* sequence compared to the DISCOVER codec, as well as the H.264 intra codec and the FF system.

sizes equal to 4 and 8. In general, the performance loss of the proposed system as measured against the DISCOVER codec increases with the GOP size.

The Bjontegaard differential rate and PSNR metrics [70, 87] were calculated for the proposed SH feedback free system, the proposed MH feedback free system, the FF system and the DISCOVER codec as measured against the H.264(intra) results. The results can be seen in Table 6.3 for the *foreman* and *coastguard* sequences and in Table 6.4 for the *soccer* and *hall monitor* sequences.

From the tables one can see that the proposed SH and MH systems both perform better than the FF System in all conditions but especially in the high motion *foreman* and *soccer* sequences. The MH system also performs better than the SH system in all conditions with average gains of approximately 0.3dB.

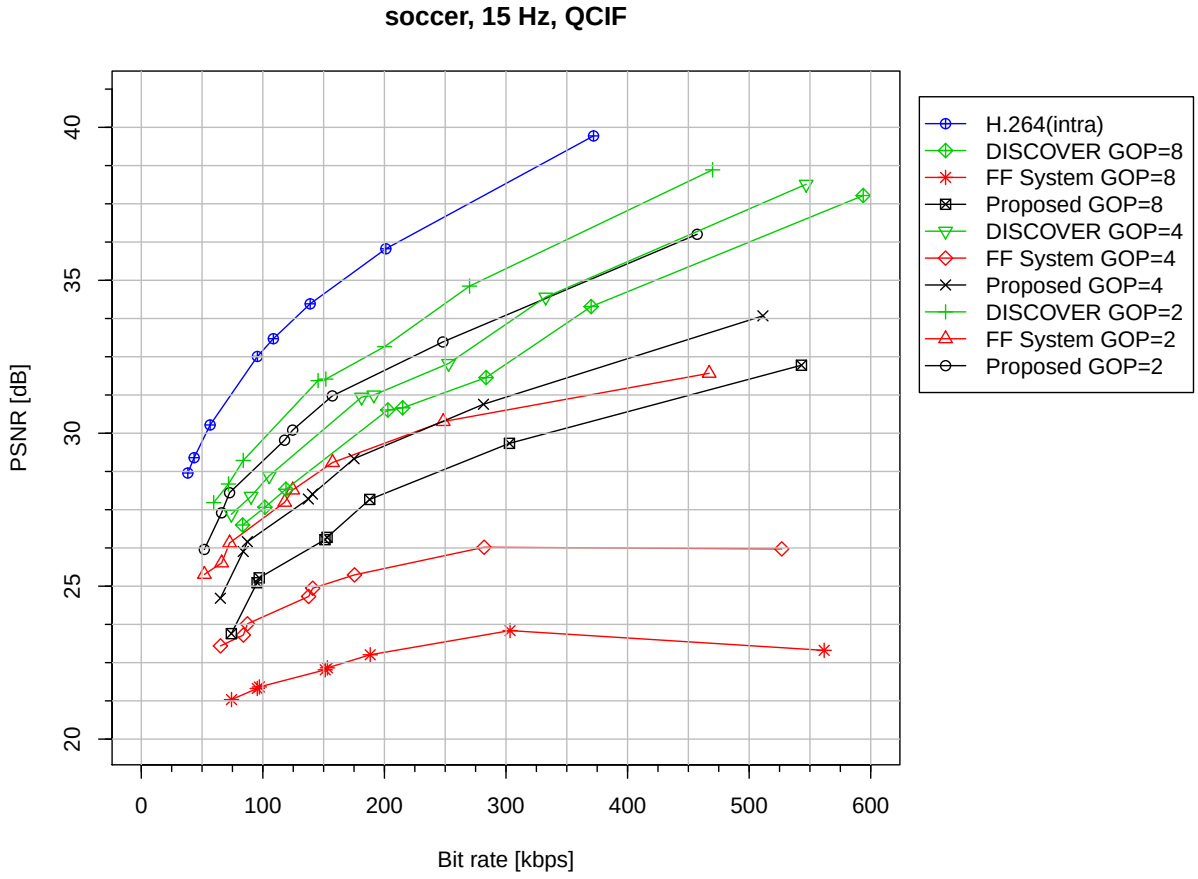

FIGURE 6.7: The performance of the proposed MH feedback free system on the *soccer* sequence compared to the DISCOVER codec, as well as the H.264 intra codec and the FF system.

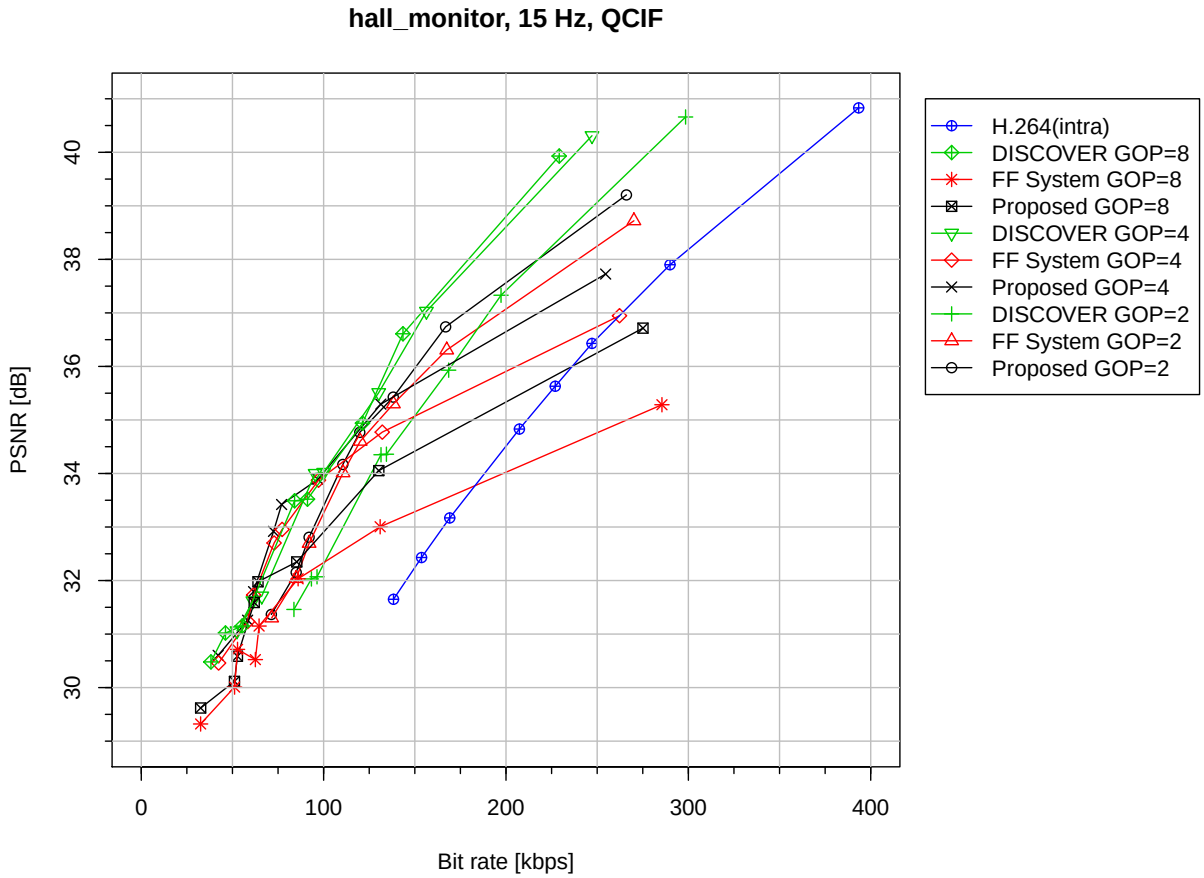

FIGURE 6.8: The performance of the proposed MH feedback free system on the *hall monitor* sequence compared to the DISCOVER codec, as well as the H.264 intra codec and the FF system.

Table 6.3: This table show the Bjontegaard Delta metrics for R-D performance as measured against the H.264(intra) code for the *foreman* and *coastguard* sequences. The BD-Rate metric is given in (%) and the BD-PSNR metric in (dB). 'DISC' refers to the DISCOVER codec, 'Prop SH' refers to the proposed SH feedback free codec, 'Prop MH' refers to the proposed MH feedback free codec and 'FF' refers to the Finite Field codec. The symbol '-' indicates that the metric could not be calculated.

		foreman				coastguard			
		Full-range		Mid-range		Full-range		Mid-range	
GOP Size	System	BD-Rate	BD-PSNR	BD-Rate	BD-PSNR	BD-Rate	BD-PSNR	BD-Rate	BD-PSNR
2	Prop SH	20.8	-1.25	16.0	-0.85	-3.08	-0.20	-6.32	0.14
	Prop MH	13.8	-0.89	8.37	-0.50	-9.88	0.14	-12.7	0.50
	FF	26.1	-1.47	18.5	-0.91	-3.41	-0.08	-7.42	0.27
	DISC	10.4	-0.40	5.13	-0.25	-17.9	0.98	-19.3	1.07
4	Prop SH	135	-4.09	127	-3.61	93.1	-2.50	69.5	-2.02
	Prop MH	107	-3.60	99.3	-3.16	37.5	-1.61	30.6	-1.14
	FF	1570	-5.35	-48.1	-4.49	77.8	-2.03	48.5	-1.47
	DISC	42.2	-1.55	33.9	-1.43	-9.04	0.41	-12.9	0.58
8	Prop SH	262	-5.99	292	-5.64	353	-4.58	30.7	-4.10
	Prop MH	207	-5.70	277	-5.45	194	-3.89	48.8	-3.44
	FF	-	-8.74	-	-8.00	378	-5.20	-91.2	-4.45
	DISC	91.3	-2.76	78.6	-2.65	26.7	-0.90	18.7	-0.75

Table 6.4: This table show the Bjontegaard Delta metrics for R-D performance as measured against the H.264(intra) code for the *soccer* and *hall monitor* sequences. The BD-Rate metric is given in (%) and the BD-PSNR metric in (dB). 'DISC' refers to the DISCOVER codec, 'Prop SH' refers to the proposed SH feedback free codec, 'Prop MH' refers to the proposed MH feedback free codec and 'FF' refers to the Finite Field codec. The symbol '-' indicates that the metric could not be calculated.

		soccer				hall monitor			
		Full-range		Mid-range		Full-range		Mid-range	
GOP Size	System	BD-Rate	BD-PSNR	BD-Rate	BD-PSNR	BD-Rate	BD-PSNR	BD-Rate	BD-PSNR
2	Prop SH	140	-4.02	145	-3.85	-34.8	2.85	-37.8	3.19
	Prop MH	116	-3.74	123	-3.61	-36.6	3.04	-39.5	3.68
	FF	310	-5.89	321	-5.65	-35.4	2.77	-38.4	3.20
	DISC	90.0	-2.86	85.6	-2.69	-28.4	2.64	-30.5	2.83
4	Prop SH	339	-6.65	363	-6.37	-37.6	1.36	-41.5	0.98
	Prop MH	280	-6.36	305	-6.20	-40.3	2.26	-45.5	4.24
	FF	-	-9.99	-	-9.41	-41.5	1.98	-46.0	2.75
	DISC	168	-4.35	164	-4.07	-42.0	3.94	-44.8	4.58
8	Prop SH	530	-8.19	666	-7.91	-1.84	-1.09	-93.8	-1.55
	Prop MH	449	-8.06	572	-7.95	-15.2	0.63	-28.3	3.23
	FF	-	-12.8	-	-12.1	-30.3	0.66	-24.2	-0.47
	DISC	234	-5.24	230	-4.93	-41.7	3.96	-44.8	5.27

6.4 COMPARISON OF PERCEPTUAL QUALITY OF PROPOSED SYSTEM WITH FINITE FIELD BASED SYSTEM

In order to objectively analyse the perceptual quality of the proposed system we include two sets of results. First, we include the performance of the proposed MH system and the FF system on all three sequences as measured using the SSIM metric in Figure 6.9 as it is considered to measure perceptual quality better than the PSNR metric for a given frame. From the figure one can see that the performance is similar for all three sequences when the GOP size is equal to 2. However, when the GOP size increases to 4 the proposed system performs better.

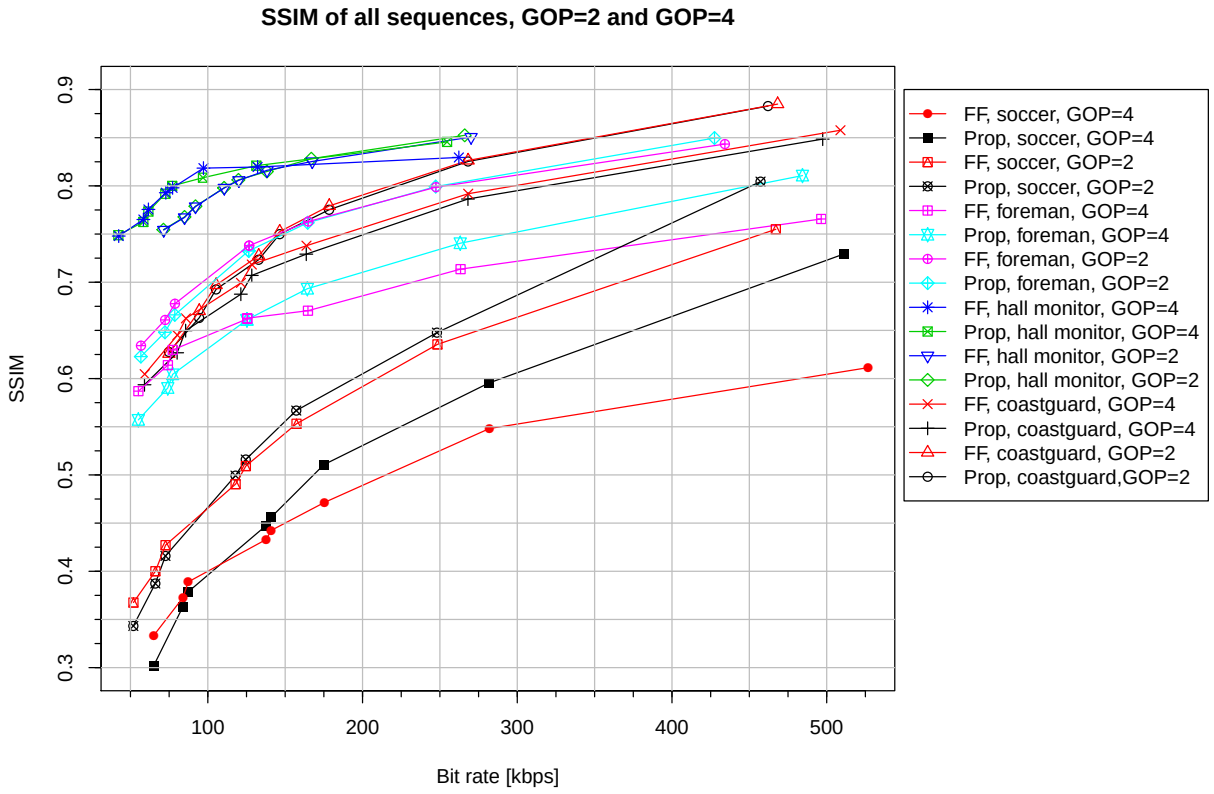

FIGURE 6.9: The performance of the proposed MH system on all four sequences compared to the FF system using the SSIM metric. The results are shown for a GOP size of 2 and 4.

Secondly, to evaluate the perceptual quality of the entire sequence, we analyse the variance of the PSNR for each sequence. Figure 6.10 shows the standard deviation of the PSNR for the two systems over all profiles and all test sequences for a GOP size of 2. From the figure it can be seen that in all cases, the standard deviation of the proposed multi-hypothesis system is less than that of the FF system. Figures 6.11 and 6.12 show the standard deviation of the PSNR for GOP

sizes of 4 and 8 respectively. A similar trend is visible for these cases as well. A large variance in distortion results in unpleasant flicker in the video sequence. While the proposed system and the FF system have similar average R-D performance for GOP equal to 2, subjectively the quality of the FF system appears worse because of image flickering. The proposed system on the other hand has a gentler degradation in image quality as evidenced by a reduced standard deviation in PSNR, resulting in less flickering.

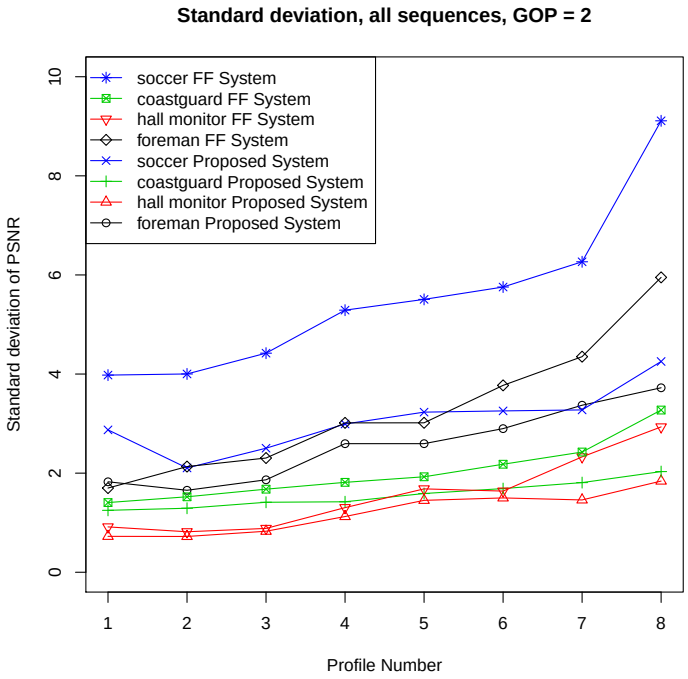

FIGURE 6.10: The standard deviation of the PSNR over the four test sequences for both the proposed multi-hypothesis system as well as the FF system as a function of the profile number (rate). The same rate is used for both systems and the GOP size is 2.

As an example, Figure 6.13 shows the same frame coded at the same rate by the proposed multi-hypothesis system, the proposed single-hypothesis system and the FF system. The estimated rate is obviously lower than required. As a result, the quality in the FF system degraded significantly while the visual quality in the proposed systems degraded much less. The multi-hypothesis system can also be seen to have a better quality than the single-hypothesis system. As a caveat, no attempt was made in the FF system to detect and ameliorate decoding failures. In cases where the estimated rate was high enough, the FF system decoded correctly and produced a higher quality frame than the proposed system.

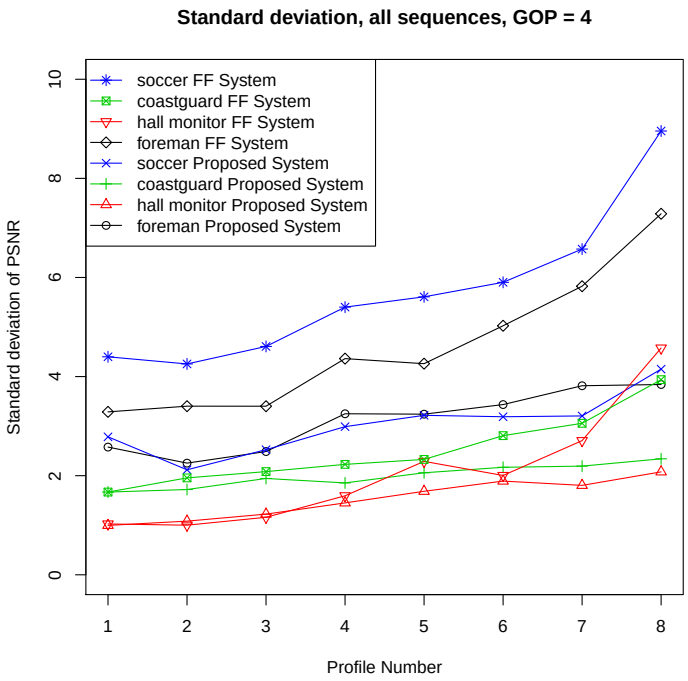

FIGURE 6.11: The standard deviation of the PSNR over the four test sequences for both the proposed multi-hypothesis system as well as the FF system as a function of the profile number (rate). The same rate is used for both systems and the GOP size is 4.

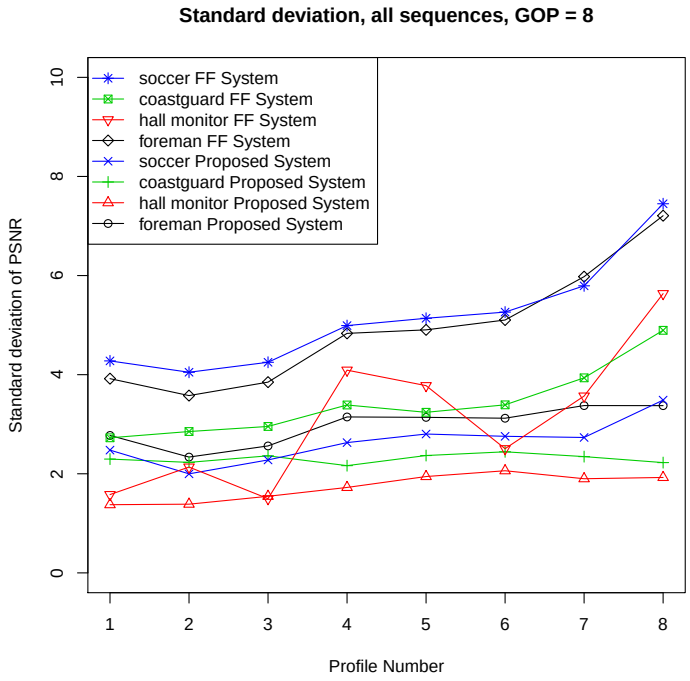

FIGURE 6.12: The standard deviation of the PSNR over the four test sequences for both the proposed multi-hypothesis system as well as the FF system as a function of the profile number (rate). The same rate is used for both systems and the GOP size is 8.

FIGURE 6.13: Comparison of the same frame. (A) Original frame. (B) Proposed multi-hypothesis feedback free system. (C) FF system. (D) Proposed single-hypothesis feedback free system. B, C and D all used the same rate.

6.5 COMPARISON OF PROPOSED SYSTEM WITH OTHER FEEDBACK FREE SYSTEMS

Comparisons with previously developed feedback free systems are problematic due to several reasons. Firstly, inconsistent testing conditions used by different proposals make it difficult to facilitate fair comparisons. Some methods for example do not consider the effect of imperfect key frames at the decoder or alternatively use high temporal sampling rates to improve performance. Secondly, currently none of the proposed methods have made the source code of their implementation publicly available. As a result, it is not possible to produce results under the same conditions considered in this thesis. For example, the variance of the PSNR and the SSIM is not reported for any of the conventional methods. Results are also not available for larger GOP sizes. Furthermore, the run-time is not supplied making it difficult to establish the effect of the proposed methods on the encoder complexity.

However, the methods proposed by Brites and Pereira in 2007 [17] and 2011 [18] use similar testing conditions to the ones presented in this thesis. Results are only available for a GOP of 2 for the *foreman*, *coastguard* and *hall monitor* sequences. Results were also obtained for the system presented by Weerakkody et al. in 2009 [9] for the *foreman* and *coastguard* sequences for a GOP of 2. The comparison of the results for the *foreman*, *coastguard* and *hall monitor* sequences can be seen in Figures 6.14, 6.15, and 6.16 respectively.

From the results one can see that the proposed multi-hypothesis system performs better that the system in [9] in all scenarios. For the *foreman* and *coastguard* sequences, the proposed codec performs better than [17] and similar to [18] at low to medium rates and slightly worse at higher rates. For the *hall monitor* sequence the proposed method performs better than [17] at all rates. For the *foreman* and *coastguard* sequences, the proposed codec performs similar to [18] at low to medium rates and worse at higher rates. For the *hall monitor* sequence the proposed method performs better than [18] at low to medium rates with equivalent performance at higher rates.

While the methods in [17] and [18] perform well, they both employ motion estimation at the encoder which increases encoder complexity.

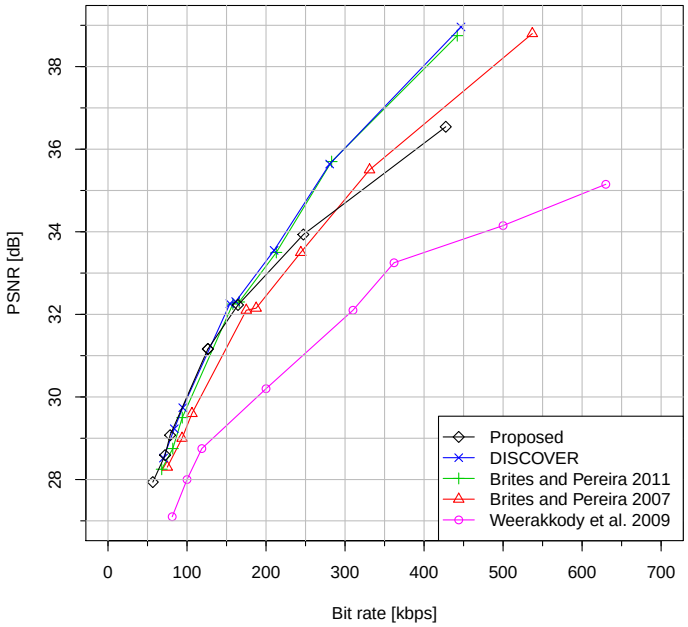

FIGURE 6.14: A comparison of the proposed multi-hypothesis feedback free system with conventional methods on the *foreman* sequence.

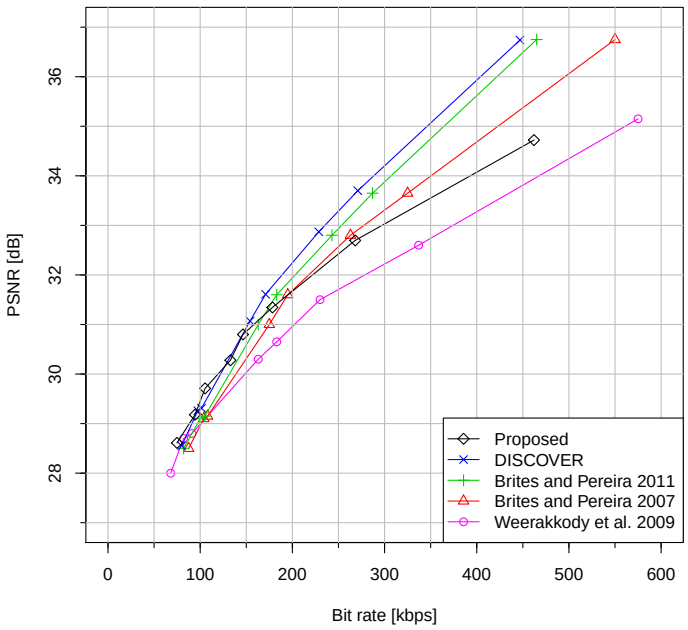

FIGURE 6.15: A comparison of the proposed multi-hypothesis feedback free system with conventional methods on the *coastguard* sequence.

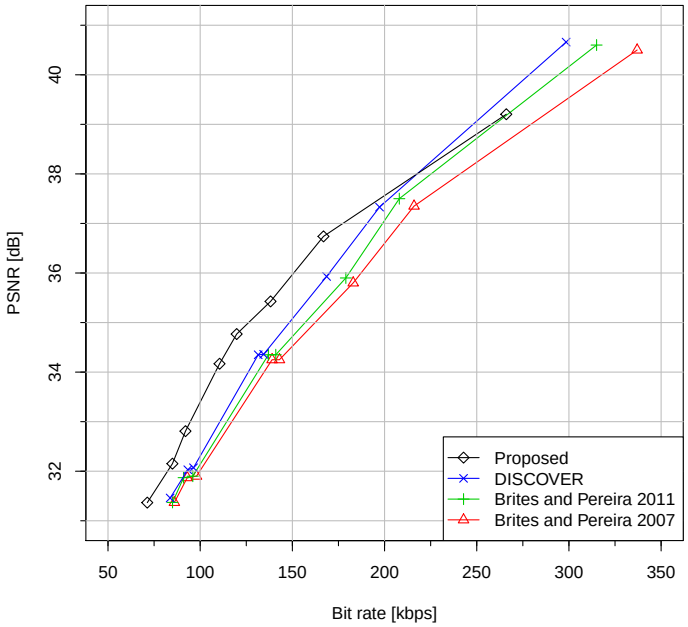

FIGURE 6.16: A comparison of the proposed feedback free system with conventional methods on the *hall monitor* sequence.

6.6 EXPERIMENTAL ANALYSIS OF COMPLEXITY

In this section the complexity of the proposed system is evaluated with run-time experiments.

6.6.1 Simulation Setup

To experimentally evaluate the computational complexity we performed execution-time experiments. Simulation based comparisons are difficult and always imperfect, but may still provide some insight into the expected performance of the systems. For these simulations we used the reference implementation of the H.264(intra) codec. All the simulations were performed on a computer with a 2.93 GHz Intel Core 2 Duo processor and 2Gb RAM running Ubuntu Linux 12.04 LTS.

The encoder run-time of DISCOVER was analysed in [74]. Without access to the source code, run-time results for the DISCOVER encoder could not be accurately generated. To facilitate a comparison, the run-time of DISCOVER (on this computer) was predicted. This was done by calculating the ratio of the per frame run-time of DISCOVER to H.264(intra) from results presented in [74]. The run-time of H.264(intra), as produced in this simulation, was then multiplied by the ratio to predict the run-time of DISCOVER. While this is not ideal, it still provides a good idea of the relative performance of the different systems.

For the decoder run-time of DISCOVER, the publicly available executable could be used to produce run-time results.

6.6.2 Encoder run-time complexity analysis

The results of the encoder run-time simulation can be seen in Table 6.5. The table shows the average encoding time for a WZ frame as measured when $N_G = 2$. The overall encoding time for any GOP size can be estimated by combining the running time for the key frames and the WZ frames. From the table, one can see that the proposed system has a reduced run time when measured against the H.264(intra) codec. The running time for the proposed system increases as the rate increases (larger RD points), since more subbands are encoded. A rate increase for a given subband also increases the size of the encoding matrix, which leads to a run time increase.

The proposed system has approximately the same run time as the DISCOVER codec in the low rate region. However, for the higher RD points, the proposed system has a longer run time than the DISCOVER codec. It should be mentioned that no attempt was made to optimise the

encoder implementation.

Table 6.5: Encoder Run-time Simulation for feedback free system. Run time comparison of the encoder for the proposed system with the H.264(intra) method as well as the DISCOVER codec. All results are in milliseconds per frame. Only WZ frames were considered for the DISCOVER codec and the proposed system. 'Disc' refers to the DISCOVER codec and 'Prop' refers to the proposed codec.

	Foreman			Coastguard			Soccer			Hall Monitor		
RD	Prop	Disc	H.264	Prop	Disc	H.264	Prop	Disc	H.264	Prop	Disc	H.264
1	14	21	112	10	21	120	23	21	104	8	21	121
2	16	21	114	13	21	124	24	21	104	9	21	124
3	16	21	116	13	21	127	24	22	107	9	22	126
4	26	21	128	25	22	136	38	22	114	10	22	133
5	26	22	131	25	21	141	39	22	116	10	23	136
6	35	24	140	39	22	152	48	22	122	11	24	143
7	49	23	148	52	23	157	69	24	132	20	24	147
8	86	24	172	98	25	184	110	25	159	43	26	170

6.6.3 Decoder Run-Time Complexity Analysis

We include some execution-time experimental results showing the decoding time in Table 6.6.

The table shows the average decoding time for a WZ frame as measured when $N_G = 2$. From the table one can see that the proposed system runs faster than the DISCOVER codec in all simulations. While this is not proof that the system will always be faster, it does indicate that real field decoding does not represent a complexity increase.

6.6.4 Experimental analysis of memory usage

The software used for the results in this thesis was not designed with the intention of measuring the memory usage. As a result, measurement of the memory usage is of limited value. For example, standard 32 bit integer variables were used to store the 8 bit pixel input values and 64 bit double variables were used to store floating point values. In many cases containers copies were preferred over performing computations in place. Furthermore, the encoder and decoder

Table 6.6: Decoder Run-time Simulation for feedback free system. Run time comparison of the decoder of the proposed multi-hypothesis system with the DISCOVER codec. All results are in seconds per frame. Only WZ frames were considered. 'Disc' refers to the DISCOVER codec and 'Prop' refers to the proposed codec.

	Foreman		Coastguard		Soccer		Hall Monitor	
RD	Prop	Disc	Prop	Disc	Prop	Disc	Prop	Disc
1	2.133	6.528	1.830	4.835	2.910	8.323	1.580	4.984
2	2.164	7.541	1.830	5.780	2.940	8.955	1.590	5.780
3	2.180	8.533	1.850	6.291	2.940	10.869	1.790	6.394
4	4.990	12.933	5.400	8.927	6.430	15.586	2.950	7.964
5	5.080	13.909	5.480	9.138	6.560	16.249	4.070	8.018
6	7.000	17.825	8.180	11.821	8.950	20.301	6.120	9.820
7	10.040	21.682	11.190	14.349	12.570	23.737	6.450	11.251
8	17.200	31.127	18.910	21.704	20.130	29.926	10.880	14.061

were run within the same software making it difficult to measure the contribution of each. Table 6.7 shows the maximum measured memory use of the encoder and the decoder combined. From the table one can see that the memory usage increased with an increase in the GOP size as expected.

Table 6.7: Maximum measured memory use of the encoder and the decoder. The results are given in Megabytes (MB).

GOP size	2	4	8
Memory Usage	47.5	57.3	65.1

6.7 SUMMARY

In this chapter the proposed multi-hypothesis feedback free system was evaluated experimentally. The system was compared with the single hypothesis feedback free system from Chapter 4, the H.264 intra code, the DISCOVER code, the system presented in Chapter 3 using rate estimation, as well as alternative feedback free proposals ([17], [18], [9]).

It was shown that the multi-hypothesis method improved performance by approximately 0.3dB in all cases. It was also shown that the proposed system can achieve average rate-distortion performance that is similar to a feedback based DISCOVER codec at low to medium rates. At high rates the system is slightly worse. The system was shown to be robust to rate estimation errors since it performed better than a conventional finite field based system operating at the same rate. The average rate-distortion performance was better than the finite field system especially for larger GOP sizes. More importantly, the standard deviation in the PSNR was lower for the proposed system relative to the finite field system indicating better perceptual video quality.

The run-time simulations show that the proposed system is able to achieve this performance without a feedback path without significantly increasing the encoder complexity. Specifically, the proposed system has a lower encoding run-time than H.264(intra).

Chapter Seven

Conclusions and future research

7.1 Conclusion

In this dissertation two problems in DVC theory were considered.

The first problem considered was that DVC has not yet achieved its theoretically possible R-D performance.

In order to improve the R-D performance, a symbol based multi-hypothesis DVC system was presented. The main concept behind the proposal was to provide the decoder with as much information as possible. As such, binary-level coding was replaced with symbol level coding. This was implemented using NB-QC-LDPC codes. QC codes were chosen for their low encoding complexity and encoder storage requirements.

The main contribution was a new method for combining hypotheses and a reconstruction algorithm matched to the proposed method. A system combining five hypotheses was developed and tested to show the efficacy of the proposal. The proposed method for combining hypotheses improved the rate-distortion performance when the extra hypotheses were independent and added new information. Since the multi-hypothesis method is implemented at the decoder, the gain in performance can be achieved without increasing the encoder complexity.

Extrapolation and interpolation were also combined in the system to facilitate better R-D performance over larger GOP sizes. Since the encoding complexity of DVC is related to the GOP size, combining interpolation and extrapolation at the decoder can decrease encoder complexity.

Overall, the proposed symbol-based multi-hypothesis DVC system had better rate-distortion performance than the benchmark DVC system (DISCOVER) in all conditions. For sequences with motion (*foreman* and *soccer*) the gain was between 0.2dB and 0.8dB depending on the GOP size. For sequences with little motion content (*hall monitor* and *coastguard*) the gain was between 0.25dB and 0.4dB. Thus the proposed system improved the R-D performance of DVC.

The second problem considered was that of the feedback channel. Conventional DVC systems require a feedback channel to operate and this limits their practical usability. Current methods to remove the feedback path increase the encoder complexity by adding motion estimation at the encoder to facilitate rate estimation.

In order to solve this problem, a new approach to feedback suppression in DVC systems was proposed. The proposed method changes the encoder structure and relies on codes defined over the real field. Despite the removal of the feedback path, the encoder complexity was not significantly increased, since no motion estimation was performed at the encoder. A new, faster, and more accurate method was also presented to estimate the conditional entropy used for encoder side rate estimation. Finally, the multi-hypothesis method was also applied to the proposed feedback free system to improve R-D performance.

The system showed average R-D performance comparable to that of a feedback based system at low to medium rates. At high rates, while there was a reduction in performance compared to feedback based systems, compared to a conventional finite field based feedback free system operating at the same rate, the performance was improved by between 0.22dB and 2.15dB depending on the sequence. Furthermore, the variance in the distortion was reduced. This resulted in improved perceptual visual quality. The use of the multi-hypothesis method also improved the performance of the proposed system in all conditions by 0.3dB on average.

In conclusion, the proposed method allows DVC to operate without a feedback channel, without increasing the encoding complexity by performing motion estimation at the encoder, and while maintaining reasonable perceptual quality.

This method can thus be used to expand the possible applications for DVC. Specifically, applications where feedback does not exist, such as encode and store type surveillance, or low-delay applications, such as video conferencing on wireless devices, could benefit from the proposals.

7.2 FUTURE RESEARCH

The work presented in this thesis can be used as a basis for future research. There are several possible avenues for improving the performance of the proposed system, and the methods proposed are general enough to find applications in other systems. Specific examples of possible future work are discussed in the following sections.

7.2.1 Sophisticated Motion Estimation

The proposals in this thesis focused on improving the structure of DVC systems. As such only conventional block matching motion estimation was considered. Advanced methods for performing motion estimation such as OBMC and optical flow have been shown to improve the performance of video coding in general, and are likely to yield improvements if combined with the proposals in this thesis.

7.2.2 Hash

Hashing has been shown to significantly improve the performance in a number of DVC systems. It is expected that hashing can be used to similarly improve the performance of the proposed system. It may also be possible to use hashes to improve the decoding algorithm of the real field codes.

7.2.3 Code Design

Designing LDPC codes for use over real fields is not yet a mature topic. Binary code designs appear to work well, but it is possible that the performance can be improved with real field specific design methods.

7.2.4 Block based coding

The proposed method encodes at a frame level. Conventional video coding has shown improvements by encoding different parts of a frame using different methods. Expanding the proposed method to encode blocks in an adaptive manner may improve performance.

7.2.5 Integer Encoding

The proposed method relies on real field coding. It is possible that the same effect can be obtained by encoding over integers instead. Combining an integer DCT with an integer LDPC code could result in reduced encoding run times.

Appendix A
Entropy derivations

Estimating $H(X^{\mathrm{Q}}|Y^{\mathrm{Q}})$ directly, requires evaluating $P(X^{\mathrm{Q}}, Y^{\mathrm{Q}})$ L^2 times, where $L = 2^q$. Instead, the proposal is to evaluate a different form of the expression:

$$\begin{aligned} H(X^{\mathrm{Q}}|Y^{\mathrm{Q}}) &= H(X^{\mathrm{Q}}) - I(X^{\mathrm{Q}}; Y^{\mathrm{Q}}) \\ &= H(X^{\mathrm{Q}}) - H(Y^{\mathrm{Q}}) + H(Y^{\mathrm{Q}}|X^{\mathrm{Q}}), \end{aligned}$$

where $I(X^{\mathrm{Q}}; Y^{\mathrm{Q}})$ is the mutual information between X^{Q} and Y^{Q}. We now make the simplifying assumption:

$$H(Y^{\mathrm{Q}}|X^{\mathrm{Q}}) \approx H(E^{\mathrm{Q}}),$$

which allows the required rate to be calculated as a function of the entropies of three single variables, each of which is simpler to calculate than the conditional entropy.

I now present derivations and equations for $H(X^{\mathrm{Q}})$ assuming a Laplace distribution and several possible quantizers. When a deadzone quantizer is used:

$$H^{\mathrm{D}}(X^{\mathrm{Q}}) = -\sum_{k=-\infty}^{\infty} P(k\Delta) \log P(k\Delta) \tag{A.1}$$

$$P(k\Delta) = \begin{cases} \int_{k\Delta}^{(k+1)\Delta} f_X(x)\,\mathrm{d}x & (k > 0) \\ \int_{(k-1)\Delta}^{(k+1)\Delta} f_X(x)\,\mathrm{d}x & (k = 0) \\ \int_{(k-1)\Delta}^{k\Delta} f_X(x)\,\mathrm{d}x & (k < 0) \end{cases},$$

yielding

$$P(k\Delta) = \begin{cases} \frac{1}{2} e^{-k\alpha_x \Delta}(1 - e^{-\alpha_x \Delta}) & (k > 0) \\ 1 - e^{-\alpha_x \Delta} & (k = 0) \\ \frac{1}{2} e^{k\alpha_x \Delta}(1 - e^{-\alpha_x \Delta}) & (k < 0) \end{cases}. \tag{A.2}$$

Therefore

$$H^{\mathrm{D}}(X^{\mathrm{Q}}) = -P(0)\log P(0) - 2\sum_{k=1}^{\infty} P(k\Delta)\log P(k\Delta)$$

Deriving the infinite sum yields:

$$\begin{aligned}
& -2\sum_{k=1}^{\infty} P(k\Delta)\log P(k\Delta) \\
= & -2\sum_{k=1}^{\infty} \frac{1}{2}e^{-k\alpha_x\Delta}(1-e^{-\alpha_x\Delta})\log\left(\frac{1}{2}e^{-k\alpha_x\Delta}(1-e^{-\alpha_x\Delta})\right) \\
= & -(1-e^{-\alpha_x\Delta})\sum_{k=1}^{\infty} e^{-k\alpha_x\Delta}\left(\log\left(\frac{1}{2}(1-e^{-\alpha_x\Delta})\right)+\log(e^{-k\alpha_x\Delta})\right) \\
= & -(1-e^{-\alpha_x\Delta})\left(\log\left(\frac{1-e^{-\alpha_x\Delta}}{2}\right)\sum_{k=1}^{\infty} e^{-k\alpha_x\Delta} - \alpha_x\Delta\log(e)\sum_{k=1}^{\infty} ke^{-k\alpha_x\Delta}\right) \\
= & -(1-e^{-\alpha_x\Delta})\left(\log\left(\frac{1-e^{-\alpha_x\Delta}}{2}\right)\frac{1}{e^{\alpha_x\Delta}-1} - \alpha_x\Delta\log(e)\frac{e^{-\alpha_x\Delta}}{(1-e^{-\alpha_x\Delta})^2}\right) \\
= & -\left(\frac{1-e^{-\alpha_x\Delta}}{e^{\alpha_x\Delta}-1}\right)\log\left(\frac{1-e^{-\alpha_x\Delta}}{2}\right)+\left(\frac{(1-e^{-\alpha_x\Delta})e^{-\alpha_x\Delta}}{1-e^{-\alpha_x\Delta}}\right)\left(\frac{\alpha_x\Delta\log(e)}{1-e^{-\alpha_x\Delta}}\right) \\
= & -\left(\frac{e^{-\alpha_x\Delta}(e^{\alpha_x\Delta}-1)}{e^{\alpha_x\Delta}-1}\right)\log\left(\frac{1-e^{-\alpha_x\Delta}}{2}\right)+e^{-\alpha_x\Delta}\left(\frac{\alpha_x\Delta\log(e)}{1-e^{-\alpha_x\Delta}}\right) \\
= & -e^{-\alpha_x\Delta}\left(\log\left(\frac{1-e^{-\alpha_x\Delta}}{2}\right)-\frac{\alpha_x\Delta\log(e)}{(1-e^{-\alpha_x\Delta})}\right)
\end{aligned}$$

Recombining yields the final result for a deadzone quantizer:

$$H^{\mathrm{D}}(X^{\mathrm{Q}}) = -(1-e^{-\alpha_x\Delta})\log(1-e^{-\alpha_x\Delta}) - e^{-\alpha_x\Delta}\left(\log\left(\frac{1-e^{-\alpha_x\Delta}}{2}\right)-\frac{\alpha_x\Delta\log(e)}{1-e^{-\alpha_x\Delta}}\right).$$

For a mid-tread quantizer, the same process yields:

$$P(k\Delta) = \begin{cases} \frac{1}{2}e^{-k\alpha_x\Delta}(e^{\frac{\alpha_x\Delta}{2}}-e^{-\frac{\alpha_x\Delta}{2}}) & (k>0) \\ 1-e^{-\frac{\alpha_x\Delta}{2}} & (k=0) \\ \frac{1}{2}e^{k\alpha_x\Delta}(e^{\frac{\alpha_x\Delta}{2}}-e^{-\frac{\alpha_x\Delta}{2}}) & (k<0) \end{cases}$$

Therefore

$$\begin{aligned}
H^{\mathrm{M}}(X^{\mathrm{Q}}) &= -P(0)\log P(0) - 2\sum_{k=1}^{\infty} P(k\Delta)\log P(k\Delta) \\
&= -\left(1-e^{-\frac{\alpha_x\Delta}{2}}\right)\log\left(1-e^{-\frac{\alpha_x\Delta}{2}}\right) - 2\sum_{k=1}^{\infty} P(k\Delta)\log P(k\Delta)
\end{aligned}$$

The infinite sum can be calculated as:

$$
\begin{aligned}
& -2\sum_{k=1}^{\infty} P(k\Delta)\log P(k\Delta) \\
= \; & -2\sum_{k=1}^{\infty} \frac{1}{2}e^{-k\alpha_x\Delta}(e^{\frac{\alpha_x\Delta}{2}} - e^{-\frac{\alpha_x\Delta}{2}})\log\left(\frac{1}{2}e^{-k\alpha_x\Delta}(e^{\frac{\alpha_x\Delta}{2}} - e^{-\frac{\alpha_x\Delta}{2}})\right) \\
= \; & -(e^{\frac{\alpha_x\Delta}{2}} - e^{-\frac{\alpha_x\Delta}{2}})\sum_{k=1}^{\infty} e^{-k\alpha_x\Delta}\left(\log\left(\frac{e^{\frac{\alpha_x\Delta}{2}} - e^{-\frac{\alpha_x\Delta}{2}}}{2}\right) + \log(e^{-k\alpha_x\Delta})\right) \\
= \; & -(e^{\frac{\alpha_x\Delta}{2}} - e^{-\frac{\alpha_x\Delta}{2}})\left(\log\left(\frac{e^{\frac{\alpha_x\Delta}{2}} - e^{-\frac{\alpha_x\Delta}{2}}}{2}\right)\sum_{k=1}^{\infty} e^{-k\alpha_x\Delta} - \alpha_x\Delta\log(e)\sum_{k=1}^{\infty} ke^{-k\alpha\Delta}\right) \\
= \; & -(e^{\frac{\alpha_x\Delta}{2}} - e^{-\frac{\alpha_x\Delta}{2}})\left(\log\left(\frac{e^{\frac{\alpha_x\Delta}{2}} - e^{-\frac{\alpha_x\Delta}{2}}}{2}\right)\frac{1}{e^{\alpha_x\Delta}-1} - \alpha_x\Delta\log(e)\frac{e^{-\alpha_x\Delta}}{(1-e^{-\alpha_x\Delta})^2}\right) \\
= \; & -\frac{(e^{\frac{\alpha_x\Delta}{2}} - e^{-\frac{\alpha_x\Delta}{2}})}{e^{\alpha_x\Delta}-1}\log\left(\frac{e^{\frac{\alpha_x\Delta}{2}} - e^{-\frac{\alpha_x\Delta}{2}}}{2}\right) + \frac{\alpha_x\Delta\log(e)}{1-e^{-\alpha_x\Delta}}\frac{(e^{\frac{\alpha_x\Delta}{2}} - e^{-\frac{\alpha_x\Delta}{2}})e^{-\alpha_x\Delta}}{1-e^{-\alpha_x\Delta}} \\
= \; & -e^{-\frac{\alpha_x\Delta}{2}}\log\left(\frac{e^{\frac{\alpha_x\Delta}{2}} - e^{-\frac{\alpha_x\Delta}{2}}}{2}\right) + e^{-\frac{\alpha_x\Delta}{2}}\frac{\alpha_x\Delta\log(e)}{1-e^{-\alpha_x\Delta}} \\
= \; & -e^{-\frac{\alpha_x\Delta}{2}}\left(\log\left(\frac{e^{\frac{\alpha_x\Delta}{2}} - e^{-\frac{\alpha_x\Delta}{2}}}{2}\right) - \frac{\alpha_x\Delta\log(e)}{1-e^{-\alpha_x\Delta}}\right)
\end{aligned}
$$

Recombining yields the final result for a uniform midtread quantizer:

$$
\begin{aligned}
H^{\mathrm{M}}(X^{\mathrm{Q}}) \; = \; & -\left(1-e^{-\frac{\alpha_x\Delta}{2}}\right)\log\left(1-e^{-\frac{\alpha_x\Delta}{2}}\right) \\
& -e^{-\frac{\alpha_x\Delta}{2}}\left(\log\left(\frac{e^{\frac{\alpha_x\Delta}{2}} - e^{-\frac{\alpha_x\Delta}{2}}}{2}\right) - \frac{\alpha_x\Delta\log(e)}{1-e^{-\alpha_x\Delta}}\right)
\end{aligned}
\tag{A.3}
$$

This differs from the equation in Altunbasak and Kamachi [64], which contained a typographical error. For a mid-rise uniform quantizer that is symmetrical around zero, a similar process is followed:

$$
\begin{aligned}
H^{\mathrm{S}}(X^{\mathrm{Q}}) \; = \; & -(1-e^{-\alpha_x\Delta})\log\left(\frac{1}{2}(1-e^{-\alpha_x\Delta})\right) \\
& -e^{-\alpha_x\Delta}\left(\log\left(\frac{1-e^{-\alpha_x\Delta}}{2}\right) - \frac{\alpha_x\Delta\log(e)}{1-e^{-\alpha_x\Delta}}\right).
\end{aligned}
$$

These closed form equations all assume that the quantizer is defined by the bin size, and that the range of the variable is theoretically infinite. In reality, there will be a limited number of levels considered. We can thus also calculate $H(X^{\mathrm{Q}})$ by evaluating (A.1) and (A.2) over a finite number of levels.

By assuming that Y and E are Laplace distributed, with $\sigma_y^2 = \sigma_e^2 + \sigma_x^2$, we can use the equations derived for $H(X^{\mathrm{Q}})$ to calculate $H(Y^{\mathrm{Q}})$ and $H(E^{\mathrm{Q}})$. We can also calculate $H(Y^{\mathrm{Q}})$

explicitly from the true distribution of Y:

$$\begin{aligned} f_Y(y) &= (f_X \circledast f_E)(y) \\ &= \int_{-\infty}^{\infty} \frac{\alpha_x}{2} e^{-\alpha_x|y-e|} \frac{\alpha_e}{2} e^{-\alpha_e|e|} \,\mathrm{d}e \\ &= \frac{\alpha_x^2 \alpha_e}{2(\alpha_x^2 - \alpha_e^2)} e^{-\alpha_e|y|} - \frac{\alpha_e^2 \alpha_x}{2(\alpha_x^2 - \alpha_e^2)} e^{-\alpha_x|y|}. \end{aligned}$$

If $\alpha_e = \alpha_x$, then

$$f_Y(y) = \frac{\alpha_x^2}{4}\left[\frac{e^{-\alpha_x|y|}}{\alpha_x} - |y|e^{-\alpha_x|y|}\right].$$

This expression can be used to calculate $H(Y^Q)$. We only show $P_j = P(q_j < Y < q_{j+1})$, which can easily be adapted to a specific quantizer. Assuming $\alpha_x \neq \alpha_e$:

$$\begin{aligned} P_j &= \int_{q_j}^{q_{j+1}} f_Y(y) \,\mathrm{d}y \\ &= \int_{q_j}^{q_{j+1}} \frac{\alpha_x^2 \alpha_e}{2(\alpha_x^2 - \alpha_e^2)} e^{-\alpha_e|y|} - \frac{\alpha_e^2 \alpha_x}{2(\alpha_x^2 - \alpha_e^2)} e^{-\alpha_x|y|} \,\mathrm{d}y. \end{aligned}$$

There are three options for this integral, $q_{j+1} \leq 0$, $q_j \geq 0$, and $q_j \leq 0, q_{j+1} \geq 0$. Due to symmetry, P_j is the same for $q_{j+1} \leq 0$ and $q_j \geq 0$. We thus only consider the latter. If $q_j \geq 0$

$$P_j = \frac{-\alpha_x^2}{2(\alpha_x^2 - \alpha_e^2)} e^{-\alpha_e y} + \frac{\alpha_e^2}{2(\alpha_x^2 - \alpha_e^2)} e^{-\alpha_x y}\Bigg|_{q_j}^{q_{j+1}}.$$

If $q_j \leq 0$ and $q_{j+1} \geq 0$:

$$\begin{aligned} P_j = &\frac{\alpha_x^2}{2(\alpha_x^2 - \alpha_e^2)} e^{\alpha_e y} - \frac{\alpha_e^2}{2(\alpha_x^2 - \alpha_e^2)} e^{\alpha_x y}\Bigg|_{q_j}^{0} \\ &+ \frac{-\alpha_x^2}{2(\alpha_x^2 - \alpha_e^2)} e^{-\alpha_e y} + \frac{\alpha_e^2}{2(\alpha_x^2 - \alpha_e^2)} e^{-\alpha_x y}\Bigg|_{0}^{q_{j+1}}. \end{aligned}$$

If $\alpha_e = \alpha_x$, then we have a similar set of two equations. If $q_j \geq 0$:

$$P_j = \frac{1}{4}(-2 - \alpha_x y) e^{-\alpha_x y}\Bigg|_{q_j}^{q_{j+1}},$$

while if $q_j \leq 0$ and $q_{j+1} \geq 0$,

$$P_j = \frac{1}{4}(2 - \alpha_x y) e^{\alpha_x y}\Bigg|_{q_j}^{0} + \frac{1}{4}(-2 - \alpha_x y) e^{-\alpha_x y}\Bigg|_{0}^{q_{j+1}}.$$

A.1 REFERENCE RESULTS

To test whether the estimates of $H(X^{\mathrm{Q}}|Y^{\mathrm{Q}})$ are accurate, we also calculate the true values as from (2.2). Thus we need an expression for $P(X^{\mathrm{Q}}, Y^{\mathrm{Q}})$.

$$\begin{aligned} P(X^{\mathrm{Q}}, Y^{\mathrm{Q}}) &= \int_{Q(x)} \int_{Q(y)} f_{X,Y}(x,y)\,\mathrm{d}y\,\mathrm{d}x \\ &= \int_{Q(x)} \int_{Q(y)} f_{Y|X}(y|x) f_X(x)\,\mathrm{d}y\,\mathrm{d}x \\ &= \int_{Q(x)} \int_{Q(y)} \frac{\alpha_e}{2} e^{-\alpha_e|x-y|} \frac{\alpha_x}{2} e^{-\alpha_x|x|}\,\mathrm{d}y\,\mathrm{d}x, \end{aligned}$$

where $Q(x) = [q_i, q_{i+1})$ and $Q(y) = [q_j, q_{j+1})$ represents the quantization bins for X and Y respectively. Due to the absolute value signs the integrals must be piecewise defined. There are three options $Q(x) = Q(y)$, $Q(x) < Q(y)$ and $Q(x) > Q(y)$. For each of these cases, we also consider $Q(x) > 0$, $Q(x) < 0$ and $Q(x)$ spanning zero (which happens for dead-zone and mid-tread quantizers. When $Q(x) > Q(y)$ or $Q(x) < Q(y)$, the resulting integrals will have the form:

$$\begin{aligned} P &= \int_a^b \int_c^d \frac{\alpha_e}{2} e^{s_1\alpha_e(x-y)} \frac{\alpha_x}{2} e^{s_2\alpha_x x}\,\mathrm{d}y\,\mathrm{d}x \\ &= \frac{\alpha_x}{4 s_1 \alpha_p} \left(e^{-s_1\alpha_e d} - e^{-s_1\alpha_e c}\right)\left(e^{-\alpha_p b} - e^{-\alpha_p a}\right), \end{aligned}$$

where $\alpha_p = -s_1\alpha_e - s_2\alpha_x$, and s_1 and s_2 are signs. If $\alpha_p = 0$, the result is:

$$P = -\frac{\alpha_x}{4 s_1} \left(e^{-s_1\alpha_e d} - e^{-s_1\alpha_e c}\right)(b-a).$$

When $Q(x) > Q(y)$, $s_1 = -1$ and when $Q(x) < Q(y)$, $s_1 = 1$. When $Q(x) < 0$, $s_2 = 1$ and when $Q(x) > 0$, $s_2 = -1$. If $Q(x)$ spans zero, the integral is evaluated twice. Once from q_i to 0 and once from 0 to q_{i+1}, with the result being the sum.

When $Q(x) = Q(y)$, the integrals have the form:

$$\begin{aligned} P &= \int_a^b \int_c^x \frac{\alpha_e}{2} e^{s_1\alpha_e(x-y)} \frac{\alpha_x}{2} e^{s_2\alpha_x x}\,\mathrm{d}y + \\ &\quad \int_x^d \frac{\alpha_n}{2} e^{-s_1\alpha_e(x-y)} \frac{\alpha_x}{2} e^{s_2\alpha_x x}\,\mathrm{d}y\,\mathrm{d}x \\ &= -\frac{\alpha_x}{4 s_1} \left(\frac{2}{s_2\alpha_x} \left(e^{s_2\alpha_x b} - e^{s_2\alpha_x a}\right) - \right. \\ &\quad \left. \frac{e^{-s_1\alpha_e c}}{\alpha_{p_1}} \left(e^{\alpha_{p_1} b} - e^{\alpha_{p_1} a}\right) - \frac{e^{s_1\alpha_e d}}{\alpha_{p_2}} \left(e^{\alpha_{p_2} b} - e^{\alpha_{p_2} a}\right) \right), \end{aligned}$$

where $\alpha_{p_1} = s_1\alpha_e + s_2\alpha_x$ and $\alpha_{p_2} = -s_1\alpha_e + s_2\alpha_x$. If $\alpha_{p_1} = 0$

$$\begin{aligned} P &= -\frac{\alpha_x}{4s_1}\left(\frac{2}{s_2\alpha_x}\left(e^{s_2\alpha_x b} - e^{s_2\alpha_x a}\right) - e^{-s_1\alpha_e c}(b-a)\right. \\ &\quad \left. -\frac{e^{s_1\alpha_e d}}{\alpha_{p_2}}\left(e^{\alpha_{p_2} b} - e^{\alpha_{p_2} a}\right)\right), \end{aligned}$$

or if $\alpha_{p_2} = 0$:

$$\begin{aligned} P &= -\frac{\alpha_x}{4s_1}\left(\frac{2}{s_2\alpha_x}\left(e^{s_2\alpha_x b} - e^{s_2\alpha_x a}\right) - \right. \\ &\quad \left. \frac{e^{-s_1\alpha_e c}}{\alpha_{p_1}}\left(e^{\alpha_{p_1} b} - e^{\alpha_{p_1} a}\right) - e^{s_1\alpha_e d}(b-a)\right). \end{aligned}$$

In this case $s_1 = -1$ and $s_2 = 1$ if $Q(x) < 0$ and $s_2 = -1$ if $Q(x) > 0$. If $Q(x)$ spans zero, the integral is evaluated twice. Once from q_i to 0 and once from 0 to q_i, with the result being the sum.

With these equations it is possible to calculate the conditional entropy exactly using the defining equation.

REFERENCES

[1] D. Slepian and J. Wolf, "Noiseless coding of correlated information sources," *IEEE Transactions on Information Theory*, vol. 19, pp. 471 – 480, July 1973.

[2] A. Wyner and J. Ziv, "The rate-distortion function for source coding with side information at the decoder," *IEEE Transactions on Information Theory*, vol. 22, pp. 1 – 10, Jan. 1976.

[3] K. Misra, S. Kar, and H. Radha, "Multi-hypothesis based distributed video coding using LDPC codes," in *in Proc. Allerton Conf. Commun., Control Comput*, 2005.

[4] D. Kubasov, J. Nayak, and C. Guillemot, "Optimal Reconstruction in Wyner-Ziv Video Coding with Multiple Side Information," in *IEEE 9th Workshop on Multimedia Signal Processing, 2007. MMSP 2007.*, pp. 183 –186, oct. 2007.

[5] X. Huang, C. Brites, J. Ascenso, F. Pereira, and S. Forchhammer, "Distributed Video Coding with multiple side information," in *Picture Coding Symposium, 2009. PCS 2009*, pp. 1 –4, may 2009.

[6] X. Huang, L. Raket, H. V. Luong, M. Nielsen, F. Lauze, and S. Forchhammer, "Multi-hypothesis transform domain Wyner-Ziv video coding including optical flow," in *IEEE 13th International Workshop on Multimedia Signal Processing (MMSP)*, pp. 1 –6, oct. 2011.

[7] J. Ascenso, C. Brites, and F. Pereira, "Augmented LDPC graph for distributed video coding with multiple side information," in *2011 IEEE 13th International Workshop on Multimedia Signal Processing (MMSP)*, pp. 1 –6, oct. 2011.

[8] X. Huang and S. Forchhammer, "Improved side information generation for distributed video coding," in *2008 IEEE 10th Workshop on Multimedia Signal Processing*, pp. 223–228, IEEE, 2008.

[9] W. Weerakkody, W. Fernando, and A. Kondoz, "Enhanced reconstruction algorithm for unidirectional distributed video coding," *IET image processing*, vol. 3, no. 6, pp. 329–334, 2009.

[10] X. Artigas and L. Torres, "Iterative generation of motion-compensated side information for distributed video coding," in *IEEE International Conference on Image Processing (ICIP)*, vol. 1, pp. I–833, IEEE, 2005.

[11] M. Morbée, J. Prades-Nebot, A. Pizurica, and W. Philips, "Rate allocation algorithm for pixel-domain distributed video coding without feedback channel," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, vol. 1, pp. I–521, IEEE, 2007.

[12] M. Morbée, J. Prades-Nebot, A. Roca, A. Pižurica, and W. Philips, "Improved pixel-based rate allocation for pixel-domain distributed video coders without feedback channel," in *Proceedings of the 9th international conference on Advanced concepts for intelligent vision systems*, ACIVS'07, (Berlin, Heidelberg), pp. 663–674, Springer-Verlag, 2007.

[13] W. A. R. J. Weerakkody, W. A. C. Fernando, and A. B. B. Adikari, "Unidirectional Distributed Video Coding for Low Cost Video Encoding," *IEEE Transactions on Consumer Electronics*, vol. 53, no. 2, pp. 788–795, 2007.

[14] J. Martinez, G. Fernandez-Escribano, H. Kalva, W. A. R. J. Weerakkody, W. A. C. Fernando, and A. Garrido, "Feedback free DVC architecture using machine learning," in *15th IEEE International Conference on Image Processing, 2008. ICIP 2008.*, pp. 1140–1143, 2008.

[15] I. Nickaein, M. Rahmati, S. Ghidary, and A. Zohrabi, "Feedback-free and hybrid distributed video coding using neural networks," in *2012 11th International Conference on Information Science, Signal Processing and their Applications (ISSPA)*, pp. 528–532, 2012.

[16] T. Sheng, X. Zhu, G. Hua, H. Guo, J. Zhou, and C. Chen, "Feedback-free rate-allocation scheme for transform domain Wyner-Ziv video coding," *Multimedia Systems*, vol. 16, no. 2, pp. 127–137, 2010.

[17] C. Brites and F. Pereira, "Encoder rate control for transform domain Wyner-Ziv video coding," in *IEEE International Conference on Image Processing (ICIP)*, vol. 2, pp. II–5, IEEE, 2007.

[18] C. Brites and F. Pereira, "An Efficient Encoder Rate Control Solution for Transform Domain Wyner-Ziv Video Coding," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 21, no. 9, pp. 1278–1292, 2011.

[19] F. Verbist, N. Deligiannis, S. M. Satti, A. Munteanu, and P. Schelkens, "Iterative Wyner-Ziv decoding and successive side-information refinement in feedback channel-free hash-based distributed video coding," pp. 84990O–1–84990O–10, 2012.

[20] G. Wallace, "The JPEG still picture compression standard," *IEEE Transactions on Consumer Electronics*, vol. 38, no. 1, pp. xviii–xxxiv, 1992.

[21] G. Sullivan and T. Wiegand, "Video Compression - From Concepts to the H.264/AVC Standard," *Proceedings of the IEEE*, vol. 93, no. 1, pp. 18–31, 2005.

[22] A. Skodras, C. Christopoulos, and T. Ebrahimi, "The JPEG 2000 still image compression standard," *Signal Processing Magazine, IEEE*, vol. 18, no. 5, pp. 36–58, 2001.

[23] A. Aaron, R. Zhang, and B. Girod, "Wyner-Ziv coding of motion video," in *Conference Record of the Thirty-Sixth Asilomar Conference on Signals, Systems and Computers*, vol. 1, pp. 240–244, nov. 2002.

[24] R. Puri and K. Ramchandran, "PRISM: A New Robust Video Coding Architecture based on Distributed Compression Principle," in *Annual Allerton Conference on Communication, Control, and Computing (ACCC)*, (Urbana, IL, USA), October 2002.

[25] X. Artigas, J. Ascenso, M. Dalai, S. Klomp, and M. Ouaret, "The DISCOVER Codec: Architecture, Techiniques and Evaluation," in *Picture Coding Symposium (PCS)*, (Lisbon, Portugal), November 2007.

[26] J. Ascenso, C. Brites, F. Dufaux, A. Fernando, T. Ebrahim, F. Pereira, and S. Tubaro, "The VISNET II DVC Codec: Architecture, Tools and Performance," in *European Signal Processing Conference (EUSIPCO)*, (Aalborg, Denmark), August 2010.

[27] C. E. Shannon, "A mathematical theory of communication," *Bell system technical journal*, vol. 27, 1948.

[28] R. Zamir, "The rate loss in the Wyner-Ziv problem," *IEEE Transactions on Information Theory*, vol. 42, no. 6, pp. 2073–2084, 1996.

[29] A. Aaron, S. D. Rane, E. Setton, and B. Girod, "Transform-domain wyner-ziv codec for video," in *Visual Communications and Image Processing*, vol. 5308, pp. 520–528, 2004.

[30] J. Ascenso, C. Brites, and F. Pereira, "Content Adaptive Wyner-ZIV Video Coding Driven by Motion Activity," in *IEEE International Conference on Image Processing*, pp. 605 –608, oct. 2006.

[31] C. Brites and F. Pereira, "Correlation Noise Modeling for Efficient Pixel and Transform Domain Wyner-Ziv Video Coding," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 18, pp. 1177 –1190, sept. 2008.

[32] C. Brites, J. Ascenso, J. Quintas Pedro, and F. Pereira, "Evaluating a feedback channel based transform domain Wyner–Ziv video codec," *Signal Processing: Image Communication*, vol. 23, no. 4, pp. 269–297, 2008.

[33] D. Kubasov, K. Lajnef, and C. Guillemot, "A hybrid encoder/decoder rate control for a wyner-ziv video codec with a feedback channel," in *in Proc. IEEE Multimedia Signal Processing Workshop, MMSP, Chania*, 2007.

[34] J. Areia, J. Ascenso, C. Brites, and F. Pereira, "Low complexity hybrid rate control for lower complexity wyner-ziv video decoding," in *16th European Signal Processing Conference (EUSIPCO), Lausanne, Switzerland, August*, 2008.

[35] M. Morbée, A. Roca, J. Prades-Nebot, A. Pižurica, and W. Philips, "Reduced decoder complexity and latency in pixel-domain Wyner–Ziv video coders," *Signal, Image and Video Processing*, vol. 2, no. 2, pp. 129–140, 2008.

[36] C. Fu and J. Kim, "Encoder rate control for block-based distributed video coding," in *2010 IEEE International Workshop on Multimedia Signal Processing (MMSP)*, pp. 333–338, 2010.

[37] S. Cheng and Z. Xiong, "Successive refinement for the Wyner-Ziv problem and layered code design," *IEEE Transactions on Signal Processing*, vol. 53, no. 8, pp. 3269–3281, 2005.

[38] C. Yaacoub, J. Farah, and B. Pesquet-Popescu, "Feedback Channel Suppression in Distributed Video Coding with Adaptive Rate Allocation and Quantization for Multiuser Applications," *EURASIP Journal on Wireless Communications and Networking*, vol. 2008, 2008.

[39] S. Pradhan, J. Chou, and K. Ramchandran, "Duality between source coding and channel coding and its extension to the side information case," *IEEE Transactions on Information Theory*, vol. 49, no. 5, pp. 1181–1203, 2003.

[40] R. P. Westerlaken, S. Borchert, R. K. Gunnewiek, and R. L. Lagendijk, "Analyzing Symbol and Bit Plane-Based LDPC in Distributed Video Coding," in *Proceedings of the International Conference on Image Processing (ICIP)*, pp. 17–20, September 2007.

[41] C. Berrou, A. Glavieux, and P. Thitimajshima, "Near shannon limit error-correcting coding and decoding: Turbo-codes," in *Proc. of IEEE International Conference on Communications (ICC)*, vol. 2, (Geneva), pp. 1064–1070 vol.2, 1993.

[42] R. Gallager, "Low-Density Parity-Check Codes." Cambridge MA, MIT Press, 1963.

[43] D. Varodayan, A. Aaron, and B. Girod, "Rate-Adaptive Distributed Source Coding using Low-Density Parity-Check Codes," in *39th Asilomar Conference on Signals, Systems and Computers*, (Pacific Grove, CA, USA), Oct/Nov 2005.

[44] J. Ascenso, C. Brites, and F. Pereira, "Design and performance of a novel low-density parity-check code for distributed video coding," in *Image Processing, 2008. ICIP 2008. 15th IEEE International Conference on*, pp. 1116–1119, IEEE, 2008.

[45] D. J. MacKay and R. M. Neal, "Near shannon limit performance of low density parity check codes," *Electronics letters*, vol. 33, no. 6, pp. 457–458, 1997.

[46] T. J. Richardson and R. L. Urbanke, "The capacity of low-density parity-check codes under message-passing decoding," *Information Theory, IEEE Transactions on*, vol. 47, no. 2, pp. 599–618, 2001.

[47] M. C. Davey and D. J. MacKay, "Low density parity check codes over GF (q)," in *Information Theory Workshop, 1998*, pp. 70–71, IEEE, 1998.

[48] C. Poulliat, M. Fossorier, and D. Declercq, "Design of non binary LDPC codes using their binary image: algebraic properties," in *IEEE International Symposium on Information Theory*, pp. 93–97, IEEE, 2006.

[49] Z. Li, L. Chen, L. Zeng, S. Lin, and W. H. Fong, "Efficient encoding of quasi-cyclic low-density parity-check codes," *IEEE Transactions on Communications*, vol. 54, no. 1, pp. 71–81, 2006.

[50] L. Lan, L. Zeng, Y. Y. Tai, L. Chen, S. Lin, and K. Abdel-Ghaffar, "Construction of quasi-cyclic LDPC codes for AWGN and binary erasure channels: A finite field approach," *IEEE Transactions on Information Theory*, vol. 53, no. 7, pp. 2429–2458, 2007.

[51] S. Song, B. Zhou, S. Lin, and K. Abdel-Ghaffar, "A unified approach to the construction of binary and nonbinary quasi-cyclic LDPC codes based on finite fields," *IEEE Transactions on Communications*, vol. 57, pp. 84 –93, jan. 2009.

[52] X. Ge and S.-T. Xia, "Structured non-binary LDPC codes with large girth," *Electronics Letters*, vol. 43, oct. 2007.

[53] L. Alparone, M. Barni, F. Bartolini, and V. Cappellini, "Adaptively weighted vector-median filters for motion-fields smoothing," in *Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP-96*, vol. 4, pp. 2267–2270, may 1996.

[54] J. Astola, P. Haavisto, and Y. Neuvo, "Vector median filters," *Proceedings of the IEEE*, vol. 78, no. 4, pp. 678–689, 1990.

[55] D. Kubasov and C. Guillemot, "Mesh-Based Motion-Compensated Interpolation for Side Information Extraction in Distributed Video Coding," in *2006 IEEE International Conference on Image Processing*, pp. 261 –264, oct. 2006.

[56] G. de Haan, P. Biezen, H. Huijgen, and O. Ojo, "True-motion estimation with 3-D recursive search block matching," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 3, no. 5, pp. 368–379, 388, 1993.

[57] W.-J. Chien, L. J. Karam, and G. P. Abousleman, "Distributed video coding with 3D recursive search block matching," in *Proc. of IEEE International Symposium on Circuits and Systems, (ISCAS)*, pp. 4–pp, IEEE, 2006.

[58] B.-D. Choi, J.-W. Han, C.-S. Kim, and S.-J. Ko, "Motion-compensated frame interpolation using bilateral motion estimation and adaptive overlapped block motion compensation," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 17, no. 4, pp. 407–416, 2007.

[59] A. Verri and T. Poggio, "Motion field and optical flow: qualitative properties," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 11, no. 5, pp. 490–498, 1989.

[60] L. L. Rakêt, L. Roholm, A. Bruhn, and J. Weickert, "Motion compensated frame interpolation with a symmetric optical flow constraint," *Advances in Visual Computing,?(Springer, 2012), Lecture Notes in Computer Science*, 2012.

[61] L. Natrio, C. Brites, J. Ascenso, and F. Pereira, "Extrapolating Side Information for Low-Delay Pixel-Domain Distributed Video Coding," in *Visual Content Processing and Representation* (L. Atzori, D. Giusto, R. Leonardi, and F. Pereira, eds.), vol. 3893 of *Lecture Notes in Computer Science*, pp. 16–21, Springer Berlin / Heidelberg, 2006.

[62] S. Borchert, R. Westerlaken, R. K. Gunnewiek, and R. Lagendijk, "On Extrapolating Side Information in Distributed Video Coding," in *26th Picture Coding System*, (Lisbon, Portugal), November 2007.

[63] E. Lam and J. Goodman, "A mathematical analysis of the DCT coefficient distributions for images," *IEEE Transactions on Image Processing*, vol. 9, pp. 1661 –1666, oct 2000.

[64] Y. Altunbasak and N. Kamaci, "An analysis of the DCT coefficient distribution with the H.264 video coder," in *Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing, (ICASSP '04).*, vol. 3, pp. iii–177–80 vol.3, 2004.

[65] N. Ahmed, T. Natarajan, and K. R. Rao, "Discrete cosine transfom," *IEEE Trans. Comput.*, vol. 23, pp. 90–93, Jan. 1974.

[66] D. Rebollo-Monedero, A. Aaron, and B. Girod, "Transforms for high-rate distributed source coding," in *Conference Record of the Thirty-Seventh Asilomar Conference on Signals, Systems and Computers*, vol. 1, pp. 850–854, IEEE, 2003.

[67] N. Deligiannis, A. Munteanu, T. Clerckx, J. Cornelis, and P. Schelkens, "Overlapped Block Motion Estimation and Probabilistic Compensation with Application in Distributed Video Coding," *IEEE Signal Processing Letters*, vol. 16, no. 9, pp. 743–746, 2009.

[68] X. Artigas and L. Torres, "Improved signal reconstruction and return channel suppression in distributed video coding systems," in *47th International Symposium ELMAR-2005, Croatia*, 2005.

[69] Z. Wang, A. Bovik, H. Sheikh, and E. Simoncelli, "Image quality assessment: from error visibility to structural similarity," *IEEE Transactions on Image Processing*, vol. 13, no. 4, pp. 600–612, 2004.

[70] G. Bjontegaard, "Calculation of average PSNR differences between RD-curves," in *the 13th VCEG-M33 Meeting*, (Austin, Texas, USA), Apr. 2001.

[71] C. Brites, J. Ascenso, and F. Pereira, "Improving Transform Domain Wyner-Ziv Video Coding Performance," in *in Proc. IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP*, vol. 2, pp. II–II, 2006.

[72] D. Declercq and M. Fossorier, "Decoding Algorithms for Nonbinary LDPC Codes Over GF(q)," *IEEE Transactions on Communications*, vol. 55, pp. 633–643, April 2007.

[73] K. Andrews, S. Dolinar, and J. Thorpe, "Encoders for block-circulant LDPC codes," *Proc. International Symposium on Information Theory, ISIT 2005.*, pp. 2300 –2304, sep. 2005.

[74] F. Pereira, J. Ascenso, and C. Brites, "Studying the GOP size impact on the performance of a feedback channel-based Wyner-Ziv video codec," in *Advances in Image and Video Technology*, pp. 801–815, Springer, 2007.

[75] R. Martins, C. Brites, J. Ascenso, and F. Pereira, "Refining Side Information for Improved Transform Domain Wyner-Ziv Video Coding," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 19, pp. 1327 –1341, sept. 2009.

[76] A. A.-E. Ailah, G. Petrazzuoli, J. Farah, M. Cagnazzo, B. Pesquet-Popescu, and F. Dufaux, "Side Information Improvement in Transform-Domain Distributed Video Coding," in *SPIE - Applications of Digital Image Processing*, (San Diego, CA (USA)), 2012.

[77] N. Deligiannis, F. Verbist, J. Slowack, R. Van De Walle, P. Schelkens, and A. Munteanu, "Joint successive correlation estimation and side information refinement in distributed video coding," in *2012 Proceedings of the 20th European Signal Processing Conference (EUSIPCO)*, pp. 569–573, 2012.

[78] S. Ye, M. Ouaret, F. Dufaux, and T. Ebrahimi, "Improved side information generation for distributed video coding by exploiting spatial and temporal correlations," *EURASIP Journal on Image and Video Processing*, vol. 2009, no. 1, p. 683510, 2009.

[79] Z. Liu, S. Cheng, A. Liveris, and Z. Xiong, "Slepian-Wolf Coded Nested Lattice Quantization for Wyner-Ziv Coding: High-Rate Performance Analysis and Code Design," *IEEE Transactions on Information Theory*, vol. 52, no. 10, pp. 4358–4379, 2006.

[80] D. Donoho, "Compressed sensing," *IEEE Transactions on Information Theory*, vol. 52, no. 4, pp. 1289–1306, 2006.

[81] V. Goyal, A. Fletcher, and S. Rangan, "Compressive Sampling and Lossy Compression," *Signal Processing Magazine, IEEE*, vol. 25, no. 2, pp. 48–56, 2008.

[82] Y. Baig, E. Lai, and A. Punchihewa, "Distributed video coding based on compressed sensing," in *2012 IEEE International Conference on Multimedia and Expo Workshops (ICMEW)*, pp. 325–330, 2012.

[83] J. Marshall, T., "Coding of Real-Number Sequences for Error Correction: A Digital Signal Processing Problem," *IEEE Journal on Selected Areas in Communications*, vol. 2, no. 2, pp. 381–392, 1984.

[84] A. Dimakis, R. Smarandache, and P. Vontobel, "LDPC Codes for Compressed Sensing," *IEEE Transactions on Information Theory*, vol. 58, no. 5, pp. 3093–3114, 2012.

[85] S. Rangan, "Estimation with random linear mixing, belief propagation and compressed sensing," in *2010 44th Annual Conference on Information Sciences and Systems (CISS)*, pp. 1–6, 2010.

[86] M. Pretti, "A message-passing algorithm with damping," *Journal of Statistical Mechanics: Theory and Experiment*, vol. 2005, no. 11, p. P11008, 2005.

[87] G. Sullivan and G. Bjontegaard, "Recommended Simulation Common Conditions for H.26L Coding Efficiency Experiments on Low-Resolution Progressive-Scan Source Material," in *ITU-T VCEG, DOC. VCEG-N81*, Sept. 2001.

PUBLICATIONS

The author wrote several papers based on the research work presented in this thesis. The following two journal articles were published:

1. Louw D.J and Kaneko H, "Suppressing feedback in a distributed video coding system by employing real field codes," EURASIP Journal on Advances in Signal Processing, 19 pages, 2013:181, Dec. 2013

2. Louw D.J and Kaneko H, "A symbol based distributed video coding system using multiple hypotheses", IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences, Vol.E97-A, No.2, pp.632-641, Feb. 2014.

The following three conference papers were presented at international peer reviewed conferences:

1. Louw D.J and Kaneko H, "Efficient conditional entropy estimation for distributed video coding" in *Proc. 30th Picture Coding Symposium* (PCS), San Jose, USA, Dec., 2013

2. Louw D.J and Kaneko H, "A system combining extrapolated and interpolated side information for single view multi-hypothesis distributed video coding," in *Proc. Int. Symp. on Information Theory and its Applications* (ISITA), Honolulu, USA, Oct., 2012

3. Louw D.J and Kaneko H, "A Multi-Hypothesis Non-Binary LDPC Code based Distributed Video Coding System," in Proc. *13th IASTED Int. Conf. On Signal and Image Processing* (SIP), Dallas, USA, Dec., 2011

Acknowledgements

I would like to thank all the individuals who helped and supported me.

First and foremost, I would like to thank my parents for their endless support and encouragement. I would also like to thank my long suffering partner Sandra Thompson for the significant moral support during the difficult times.

I am especially grateful to my supervisor, Dr. Haruhiko Kaneko, for welcoming me into his laboratory. His patience, guidance and insight was invaluable.

I would also like to thank Prof. Yokota, Prof. Kise, Prof. Sugiyama, and Prof. Yamada for their constructive comments during the evaluation phase which significantly improved the structure and presentation of the thesis.

Credit goes to the Japanese Ministry of Education, Culture, Sports, Science, and Technology (MEXT) for financially supporting the research in the form of a MEXT Scholarship.