Distributed Video Coding for Wireless Lightweight Multimedia Applications

Thesis submitted in fulfillment of the requirements for the award of the degree of Doctor in Engineering (Doctor in de Ingenieurswetenschappen) by

ir. Nikolaos T. Deligiannis

June 2012

Advisors: Prof. dr. ir. Adrian Munteanu  
Prof. dr. ir. Peter Schelkens
Examining Committee

Prof. dr. ir. Adrian Munteanu – Vrije Universiteit Brussel – Promoter

Prof. dr. ir. Peter Schelkens – Vrije Universiteit Brussel – Promoter

Prof. dr. ir. Leo Van Biesen – Vrije Universiteit Brussel – Committee chair

Prof. dr. ir. Rik Pintelon – Vrije Universiteit Brussel – Committee vice-chair

Prof. dr. Ann Dooms – Vrije Universiteit Brussel – Committee secretary

Prof. dr. ir. Athanassios N. Skodras – Hellenic Open University – Member

Dr. ir. Nikolaos V. Boulgouris – Brunel University – Member

Prof. dr. ir. Jan Cornelis – Vrije Universiteit Brussel – Member
Ο ΉΛΙΟΣ Ο ΗΛΙΑΤΟΡΑΣ

Το Τρελοβάπορο

“Βαπόρι στολισμένο βγαίνει στα βουνά
κι αρχίζει τις μανούβρες «βίρα μάνα»

Την άγκυρα φοντάρει στις κουκουναριές
φορτώνει φρέσκο αέρα κι απ’ τις δύο μεριές

Είναι από μαύρη πέτρα κι είναι απ’ όνειρο
κι έχει λοστρόμο αθώο ναύτη πονηρό

Από τα βάθη φτάνει τους παλιούς καιρούς
βάσανα ξεφορτώνει κι αναστεναγμούς

Τέλος Χριστέ και Κύριε λέω κι απορώ
τέτοιο τρελό βαπόρι τρελοβάπορο

Χρόνους μας ταξιδεύει δε βουλιάζαμε
χίλιους καπεταναίους τους αλλάζαμε

Κατακλυσμούς ποτέ δε λογαρίσαμε,
μπήκαμε μες στα όλα και περάσαμε,

Κι έχουμε στο κατάρτι μας βιγλάτορα
Παντοτινό τον ‘Ηλιο τον Ηλιάτορα!”

THE SOVEREIGN SUN

The Crazy Boat

“A boat adorned and decked sails out for mountains oh
and there begins maneuvers with heave-to, heave-ho
weighs anchor by a pine tree grove and takes aboard
a cargo of fresh mountain air at lee and port.

She’s made of blackest stone,
she’s made of flimsy dream
her boatswain is naive, her sailors plot and scheme

she’s come from the deep depths of ancient bygone times
and here unloads her troubles and her trembling sighs.

O come my Lord and Jesus, I speak and am struck daft
on such a loony vessel on such a crazy craft

we’ve sailed for years on end,
and still we’ve kept afloat
we’ve changed a thousand skippers on this balmy boat

we never paid the slightest heed to cataclysms
but plunged headlong in everything with optimisms

and high up on our lookout mast we keep for sentry one
who ever and anon remains our Sun our Sovereign Sun!”

ΟΔΥΣΣΕΑΣ ΕΛΥΤΗΣ
Έλληνας Ποιητής (1911–1996)
Βραβείο Nobel Λογοτεχνίας 1979

Odysseas Elytis
Greek Poet (1911–1996)
Nobel Prize in Literature 1979
# Table of Contents

ACKNOWLEDGMENTS .......................................................... VII
SYNOPSIS ................................................................. IX
ACRONYMS ............................................................... XI

## CHAPTER 1: INTRODUCTION ............................................. 1

### 1.1 Motivation .......................................................... 1

#### 1.1.1 Wireless Lightweight Multimedia Applications .............. 1
#### 1.1.2 Digital Video Compression Basics ........................... 3
#### 1.1.3 Video Compression Requirements for Multimedia Sensors ... 4

### 1.2 Introduction to Conventional Video Coding ...................... 5

### 1.3 Introduction to Distributed Video Coding ....................... 5

#### 1.3.1 Distributed Video Coding Principles ........................ 6
#### 1.3.2 Distributed Video Coding Attributes ........................ 6

### 1.4 Outline and Major Contributions ................................ 7

## CHAPTER 2: INFORMATION THEORETIC FUNDAMENTALS ........... 11

### 2.1 Introduction ...................................................... 11

### 2.2 Source Coding .................................................. 12

#### 2.2.1 The Asymptotic Equipartition Property ..................... 13
#### 2.2.2 Data Compression with Fixed Length Codes ................. 14
#### 2.2.3 Data Compression with Variable Length Codes ............. 16

### 2.3 Rate Distortion Theory .......................................... 18

#### 2.3.1 Distortion Metrics .......................................... 19
#### 2.3.2 Rate-Distortion Function .................................... 20
#### 2.3.3 Achievability of the Rate-Distortion Function ............. 21

### 2.4 Successive Refinement of Information .......................... 24

#### 2.4.1 Formal Definition of Successive Refinement of Information 24
#### 2.4.2 Conditions for Achievability ................................ 25
#### 2.4.3 Practical Progressive Coding ................................ 26

### 2.5 Distributed Encoding of Correlated Sources .................... 27

#### 2.5.1 The Slepian-Wolf Coding Theorem .......................... 27
#### 2.5.2 Achievability of Slepian-Wolf Coding: The Random Binning Argument ................................................. 28
2.5.3 Practical Slepian-Wolf Coding ................................................................. 30

2.6 RATE-DISTORTION WITH SIDE INFORMATION ........................................... 34
  2.6.1 The Wyner-Ziv Coding Theorem ................................................................ 34
  2.6.2 Achievability of Wyner-Ziv Coding ............................................................. 36
  2.6.3 Practical Wyner-Ziv Coding ........................................................................ 38

2.7 SUCCESSIVELY REFINED WYNER-ZIV CODING ........................................ 40
  2.7.1 Definitions and Conditions for Achievability .............................................. 40
  2.7.2 Layered Wyner-Ziv Coding ....................................................................... 41

CHAPTER 3 CONVENTIONAL VS. DISTRIBUTED VIDEO CODING ....................... 43

3.1 INTRODUCTION .............................................................................................. 43

3.2 CONVENTIONAL VIDEO CODING ................................................................. 44
  3.2.1 Motion-Compensated Predictive Video Coding Architecture ..................... 45
  3.2.2 Video Coding with H.264/AVC ................................................................... 47

3.3 DISTRIBUTED VIDEO CODING ..................................................................... 49
  3.3.1 The PRISM Architecture .......................................................................... 50
  3.3.2 The Stanford Architecture ....................................................................... 55
  3.3.3 The DISCOVER Codec ............................................................................ 58
  3.3.4 Improvements over the DISCOVER Codec ............................................... 62

CHAPTER 4 OVERLAPPED BLOCK MOTION ESTIMATION AND COMPENSATION AT THE DECODER .................................................................................. 67

4.1 INTRODUCTION ................................................................................................ 67
  4.1.1 Related Work on Side Information Generation ........................................... 67
  4.1.2 Contributions ............................................................................................. 70

4.2 BIT-PLANE-BASED OBMEC ........................................................................ 72
  4.2.1 The Spatial-Domain Unidirectional DVC Codec ......................................... 72
  4.2.2 Bit-Plane Overlapped Block Motion Estimation ......................................... 74
  4.2.3 Probabilistic Motion Compensation ............................................................. 78

4.3 CODEC EXTENSION TO THE TRANSFORM DOMAIN ................................... 80

4.4 HASH-BASED DVC USING OBMEC/SSM .................................................... 81
  4.4.1 Hash Information Coding in HDVC ............................................................ 84
  4.4.2 Overlapped Block Motion Estimation and Compensation with Sub-Sampled Matching ................................................................. 86

4.5 OBMEC-BASED SIDE INFORMATION REFINEMENT .................................. 89
  4.5.1 Codec Architecture .................................................................................... 90
6.3 HASH FORMATION AND COMPRESSION 162
   6.3.1 Hash Encoding................................................................. 163
   6.3.2 Hash Decoding ............................................................... 164
6.4 SIDE INFORMATION GENERATION 165
   6.4.1 Modified OBME.............................................................. 165
   6.4.2 Reliability Screening...................................................... 166
   6.4.3 MSE-Optimal Motion Compensation............................. 167
6.5 EXPERIMENTAL EVALUATION 169
   6.5.1 Evaluation on Conventional Test Sequences .................. 170
   6.5.2 Evaluation on Endoscopic Video Sequences ................... 181
   6.5.3 Encoding Complexity .................................................... 190
   6.5.4 Encoder Buffer Requirements ....................................... 193
6.6 CONCLUSIONS 196

CHAPTER 7 EPILOGUE AND PROSPECTIVE WORK 199

7.1 CONCLUSIONS 199

7.2 PROSPECTIVE WORK 203

APPENDIX A CHANNEL CODES FOR DISTRIBUTED SOURCE COMPRESSION 207
   A.1. INTRODUCTION 207
   A.2. BLOCK CODES 208
      A.2.1. General Principles of Linear Block Codes .................... 209
      A.2.2. Low-Density Parity-Check Codes .............................. 209
   A.3. RATE-ADAPTIVE LDPC CODES FOR SLEPIAN-WOLF COMPRESSION 212

APPENDIX B MOTION-COMPENSATED INTERPOLATION 217
   B.1. INTRODUCTION 217
   B.2. FORWARD MOTION ESTIMATION 218
   B.3. BIDIRECTIONAL MOTION ESTIMATION 219
   B.4. SPATIAL MOTION VECTOR SMOOTHING 220
   B.5. MOTION COMPENSATION 221

APPENDIX C THEORETICAL FORMULATIONS 225
   C.1. PROOFS AND DERIVATIONS 225
      C.1.1. Proof of Lemma 5.1 ................................................... 225
      C.1.2. Proof of Lemma 5.2 ................................................... 226
C.1.3.  Proof of Theorem 5.1 ................................................................. 227
C.1.4.  Derivation and Uniqueness of the Root of Equation (5.18) .. 227

LIST OF PUBLICATIONS  229
REFERENCES  233
ACKNOWLEDGMENTS

This Ph.D. adventure would not have been the same without the support of special people that I have met along the way and who I wish to cordially thank here.

First of all, I wish to express my deepest gratitude to my promoters Prof. Adrian Munteanu and Prof. Peter Schelkens for their guidance and support throughout these years. On the one hand, Prof. Adrian Munteanu has been more like a true mentor to me rather than a promoter, always being there in moments of both joy and struggle. I cannot forget the limitless hours–even in weekends–he spent with me to discuss ideas and to review my publications and this thesis. On the other hand, Prof. Peter Schelkens has acted as a genuine tutor, providing me with the stimulus and encouragement for hard work. He has never spared his constructive comments and suggestions on my publications and this dissertation.

I am bound to Prof. Jan Cornelis who has been the most crucial factor on my decision to pursue a Ph.D. at the Vrije Universiteit Brussel. It is for his unceasing exertions that I have been part of a highly motivating research environment where premium research is conducted.

I am grateful to Dr. Nikolaos V. Boulgouris, Prof. Jan Cornelis, Prof. Ann Dooms, Prof. Rik Pintelon, Prof. Athanassios N. Skodras and Prof. Leo Van Biesen for accepting to be members of my Ph.D. jury and for their time devoted to studying and evaluating this dissertation.

I am indebted to my uncle Prof. Emeritus Theodore L. Deliyannis for his love and for his incessant guidance and support. His outstanding scientific achievements and attention to nonstop hard work, but primarily his benevolence and modesty have set an example I hope to match one day.

I owe a huge depth of gratitude to my good friend Dr. Joeri Barbarien for the countless hours we spent together on many subjects, for the many stimulating discussions and for proofreading this thesis. With his continuous encouragement and willingness to help, he has significantly contributed to this Ph.D. research work.

I am delighted to have collaborated with Prof. Rik Van de Walle, Prof. Peter Lambert, Dr. Jürgen Slowack, Dr. Jozef Škorupa and Stefaan Mys from the Department of Electronics and Information Systems at the Ghent University. Given their great expertise in the domain, the numerous brainstorming assemblies we had as well as their constructive comments on my publications have provided me with profound insights in the technical challenges of distributed video coding.
I am grateful to Prof. Stavros A. Kotsopoulos from the Department of Electrical and Computer Engineering at the University of Patras who has introduced me to the fascinating domain of wireless communications and has initiated me to the captivating world of research.

I would like to thank my colleagues Bruno Cornelis, Marc Jacobs, Shahid Satti and Frederik Verbist for their true friendship and for our intense collaboration. I am also thankful to Dr. Athanassios C. Iossifides and Dr. Spiros Louvros for endless discussions and our fruitful collaboration over the years.

I wish to sincerely thank Prof. Dr. Daniel Urbain, head of the Gastroenterology clinic of the Universitair Ziekenhuis Brussel, for providing numerous wireless capsule endoscopic video sequences and for appraising the visual quality delivered by the developed coding systems.

I am thankful to Karin Debruyn for the invaluable help in the administrative and organizational aspects of my Ph.D. studies.

I would like to acknowledge the financial support offered by the Fund for Scientific Research (FWO) Flanders – via projects G.0391.07, G.0146.10, and via the FWO postdoctoral fellowships of Peter Schelkens and Adrian Munteanu – and by the Interdisciplinary Institute for Broadband Technology (IBBT).

I wish to express great appreciation to my colleagues and friends at the Vrije Universiteit Brussel for establishing a creative but also enjoyable working environment. Many other friends located in Brussels and in Greece deserve more credit than I could give, particularly, Ioannis V., Manolis, Despoina, Alecos, Giannis G., Vicky N., Filippos G., Elias Gor., Elias K., Spiros, Aris, and Elias G.

I would like to thank my family, especially, my parents Theodore and Martha, my sister Christina and my brother Tasos, for their love, support and constant encouragement and for accepting my decision to pursue a Ph.D. abroad.

Last but not least, I am grateful to my significant other, Efthymia, for her unconditional love, her patience throughout my Ph.D. studies and for being an endless source of support and strength. She has helped me to achieve things I have never believed I was capable of.

Nikolaos T. Deligiannis
Brussels, June 2012
SYNOPSIS

In the modern wireless age, lightweight multimedia technology stimulates attractive commercial applications on a grand scale as well as highly specialized niche markets. In this regard, the design of efficient video compression systems meeting such key requirements as very low encoding complexity, transmission error robustness and scalability, is no straightforward task. The answer can be found in fundamental information theoretic results, according to which efficient compression can be achieved by leveraging knowledge of the source statistics at the decoder only, giving rise to distributed, or alias Wyner-Ziv, video coding. This dissertation engineers efficient lightweight Wyner-Ziv video coding schemes emphasizing on several design aspects and applications.

The first contribution of this dissertation focuses on the design of effective side information generation techniques so as to boost the compression capabilities of Wyner-Ziv video coding systems. To this end, overlapped block motion estimation and probabilistic compensation, a novel technique that performs advanced multi-hypothesis motion-compensated prediction at the decoder, is proposed. Using auxiliary (i.e., hash) information sent to the decoder, the proposed technique triggers the design of a novel efficient codec featuring very low encoding complexity and operating without a feedback channel. Adding a transform domain Wyner-Ziv coding layer with a feedback channel yields a novel codec that outperforms state-of-the-art Wyner-Ziv codecs. What is more, tailored to a novel hash design, an innovative modified version of the proposed technique is included in a new efficient hash-based Wyner-Ziv architecture. Furthermore, when coupled with an alternative side information creation method, the proposed technique enables side information refinement after decoding critical information, thereby further improving the performance of traditional Wyner-Ziv video codecs.

The second contribution of this dissertation constitutes the introduction of a novel correlation channel modeling concept that expresses the correlation noise as being statistically dependent on the side information. Compared to classical side-information-independent noise modeling adopted in traditional Wyner-Ziv coding solutions, it is theoretically proven that side-information-dependent modeling improves the compression performance. Anchored in this finding, a novel algorithm for online estimation of the side-information-dependent correlation channel parameters is presented. The proposed algorithm enables bit-plane-by-bit-plane
successive refinement of the channel estimation leading to progressively improved accuracy. Experimental results corroborate the theoretical coding gains brought by the side-information-dependent model and demonstrate the superior accuracy of the proposed online channel estimation algorithm over state-of-the-art approaches.

The third contribution of this dissertation intends to bridge the gap between distributed video coding and its practical applications. A keynote contribution in this direction is the expansion of the application domain of Wyner-Ziv coding from conventional video to medical imaging. Wireless capsule endoscopy in particular, which is essentially wireless video capturing and transmission by a pill, is proven to be a promising application field. Driven by such applications, a new Wyner-Ziv system, which generates a hash as a downscaled and subsequently intra coded version of the encoded frame, is proposed. Based on this hash, side information generation can adapt even to extreme spatial variations in temporal correlation, often appearing in endoscopic video content. By enabling low encoding complexity and scalable coding and by delivering improved compression performance compared to the state-of-the-art, the developed codec constitutes a strong candidate for lightweight (medical) imaging applications.
### ACRONYMS

<table>
<thead>
<tr>
<th>Acronym</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-D</td>
<td>One-Dimensional</td>
<td>51</td>
</tr>
<tr>
<td>2-D</td>
<td>Two-Dimensional</td>
<td>39</td>
</tr>
<tr>
<td>3-D</td>
<td>Three-Dimensional</td>
<td>45</td>
</tr>
<tr>
<td>3GPP</td>
<td>3rd Generation Partnership Project</td>
<td>44</td>
</tr>
<tr>
<td>AC</td>
<td>Alternating Current</td>
<td>59</td>
</tr>
<tr>
<td>AEP</td>
<td>Asymptotic Equipartition Property</td>
<td>13</td>
</tr>
<tr>
<td>APP</td>
<td>A Posteriori Probability</td>
<td>210</td>
</tr>
<tr>
<td>AVC</td>
<td>Advanced Video Coding</td>
<td>44</td>
</tr>
<tr>
<td>BAC</td>
<td>Binary Asymmetric Channel</td>
<td>130</td>
</tr>
<tr>
<td>BCH</td>
<td>Bose-Chaudhuri-Hocquenghem</td>
<td>51</td>
</tr>
<tr>
<td>BD</td>
<td>Bjøntegaard Delta</td>
<td>100</td>
</tr>
<tr>
<td>bpp</td>
<td>bits per pixel</td>
<td>96</td>
</tr>
<tr>
<td>bps</td>
<td>bits per second</td>
<td>4</td>
</tr>
<tr>
<td>BSC</td>
<td>Binary Symmetric Channel</td>
<td>31</td>
</tr>
<tr>
<td>CABAC</td>
<td>Context-based Adaptive Binary Arithmetic Coding</td>
<td>49</td>
</tr>
<tr>
<td>CAVLC</td>
<td>Context-based Adaptive Variable-Length Coding</td>
<td>48</td>
</tr>
<tr>
<td>CC</td>
<td>Correlation Channel</td>
<td>84</td>
</tr>
<tr>
<td>CCE</td>
<td>Correlation Channel Estimation</td>
<td>133</td>
</tr>
<tr>
<td>CIF</td>
<td>Common Intermediate Format</td>
<td>3</td>
</tr>
<tr>
<td>CPU</td>
<td>Central Processing Unit</td>
<td>148</td>
</tr>
<tr>
<td>CRC</td>
<td>Cyclic Redundancy Check</td>
<td>52</td>
</tr>
<tr>
<td>DC</td>
<td>Direct Current</td>
<td>59</td>
</tr>
<tr>
<td>DCT</td>
<td>Discrete Cosine Transform</td>
<td>45</td>
</tr>
<tr>
<td>DISCOVER</td>
<td>DIStributed COding for Video sERvices</td>
<td>58</td>
</tr>
<tr>
<td>DISCUS</td>
<td>DIStributed Source Coding Using Syndromes</td>
<td>39</td>
</tr>
<tr>
<td>DSC</td>
<td>Distributed Source Coding</td>
<td>27</td>
</tr>
<tr>
<td>DVC</td>
<td>Distributed Video Coding</td>
<td>49</td>
</tr>
<tr>
<td>EDVC</td>
<td>Endoscopic Distributed Video Coding</td>
<td>158</td>
</tr>
<tr>
<td>GOP</td>
<td>Group Of Pictures</td>
<td>45</td>
</tr>
<tr>
<td>HDTV</td>
<td>High-Definition TV</td>
<td>45</td>
</tr>
<tr>
<td>HDVC</td>
<td>Hash-based Distributed Video Coding</td>
<td>81</td>
</tr>
<tr>
<td>HEVC</td>
<td>High Efficiency Video Coding</td>
<td>45</td>
</tr>
<tr>
<td>HPS</td>
<td>Hash-Predictor-Selection</td>
<td>166</td>
</tr>
<tr>
<td>IBBT</td>
<td>Interdisciplinary Institute for Broadband Technology</td>
<td>44</td>
</tr>
<tr>
<td>IDCT</td>
<td>Inverse Discrete Cosine Transform</td>
<td>53</td>
</tr>
<tr>
<td>Abbreviation</td>
<td>Full Name</td>
<td>Page</td>
</tr>
<tr>
<td>--------------</td>
<td>-----------</td>
<td>------</td>
</tr>
<tr>
<td>IEC</td>
<td>International Electrotechnical Commission</td>
<td>44</td>
</tr>
<tr>
<td>i.i.d.</td>
<td>independent identically distributed</td>
<td>13</td>
</tr>
<tr>
<td>ISO</td>
<td>International Organization for Standardization</td>
<td>44</td>
</tr>
<tr>
<td>ITU-T</td>
<td>International Telecommunication Union - Telecommunications sector</td>
<td>44</td>
</tr>
<tr>
<td>JPEG</td>
<td>Joint Photographic Experts Group</td>
<td>55</td>
</tr>
<tr>
<td>DJSCC</td>
<td>Distributed Joint Source-Channel Coding</td>
<td>83</td>
</tr>
<tr>
<td>LDPC</td>
<td>Low-Density Parity-Check</td>
<td>209</td>
</tr>
<tr>
<td>LDPCA</td>
<td>Low-Density Parity-Check Accumulate</td>
<td>212</td>
</tr>
<tr>
<td>LLR</td>
<td>Log-Likelihood Ratio</td>
<td>41</td>
</tr>
<tr>
<td>LT</td>
<td>Luby Transform</td>
<td>207</td>
</tr>
<tr>
<td>MB</td>
<td>Macroblock</td>
<td>45</td>
</tr>
<tr>
<td>MC</td>
<td>Motion Compensation</td>
<td>47</td>
</tr>
<tr>
<td>MCE</td>
<td>Motion-Compensated Extrapolation</td>
<td>57</td>
</tr>
<tr>
<td>MCI</td>
<td>Motion-Compensated Interpolation</td>
<td>55</td>
</tr>
<tr>
<td>ME</td>
<td>Motion Estimation</td>
<td>46</td>
</tr>
<tr>
<td>MMSE</td>
<td>Minimum Mean Squared Error</td>
<td>56</td>
</tr>
<tr>
<td>MPEG</td>
<td>Moving Picture Experts Group</td>
<td>45</td>
</tr>
<tr>
<td>MSB</td>
<td>Most Significant Bit-plane</td>
<td>70</td>
</tr>
<tr>
<td>MSE</td>
<td>Mean Squared Error</td>
<td>4</td>
</tr>
<tr>
<td>MVC</td>
<td>Multiview Video Coding</td>
<td>45</td>
</tr>
<tr>
<td>OBMC</td>
<td>Overlapped Block Motion Compensation</td>
<td>221</td>
</tr>
<tr>
<td>OBME</td>
<td>Overlapped Block Motion Estimation</td>
<td>74</td>
</tr>
<tr>
<td>OBMEC</td>
<td>Overlapped Block Motion Estimation and Compensation</td>
<td>72</td>
</tr>
<tr>
<td>OBMEC/SSM</td>
<td>Overlapped Block Motion Estimation and Compensation with Sub-Sampled Matching</td>
<td>86</td>
</tr>
<tr>
<td>pdf</td>
<td>probability density function</td>
<td>20</td>
</tr>
<tr>
<td>PDWZ</td>
<td>Pixel-Domain Wyner-Ziv</td>
<td>55</td>
</tr>
<tr>
<td>pmf</td>
<td>probability mass function</td>
<td>13</td>
</tr>
<tr>
<td>PRISM</td>
<td>Power-efficient Robust high-compression Syndrome-based Multimedia coding</td>
<td>50</td>
</tr>
<tr>
<td>PSNR</td>
<td>Peak Signal-to-Noise Ratio</td>
<td>4</td>
</tr>
<tr>
<td>QCIF</td>
<td>Quarter Common Intermediate Format</td>
<td>74</td>
</tr>
<tr>
<td>QM</td>
<td>Quantization Matrix</td>
<td>57</td>
</tr>
<tr>
<td>QP</td>
<td>Quantization Parameter</td>
<td>48</td>
</tr>
<tr>
<td>RD</td>
<td>Rate-Distortion</td>
<td>6</td>
</tr>
<tr>
<td>RDO</td>
<td>Rate-Distortion Optimization</td>
<td>83</td>
</tr>
<tr>
<td>RS</td>
<td>Reed-Solomon</td>
<td>207</td>
</tr>
<tr>
<td>RSD</td>
<td>Relative Standard Deviation</td>
<td>111</td>
</tr>
<tr>
<td>SAD</td>
<td>Sum of Absolute Differences</td>
<td>55</td>
</tr>
<tr>
<td>SAQ</td>
<td>Successive Approximation Quantization</td>
<td>26</td>
</tr>
<tr>
<td>SDUDVC</td>
<td>Spatial-Domain Unidirectional DVC</td>
<td>72</td>
</tr>
<tr>
<td>Abbreviation</td>
<td>Description</td>
<td>Page</td>
</tr>
<tr>
<td>--------------</td>
<td>--------------------------------------------------</td>
<td>------</td>
</tr>
<tr>
<td>SID</td>
<td>Side-Information-Dependent</td>
<td>128</td>
</tr>
<tr>
<td>SII</td>
<td>Side-Information-Independent</td>
<td>127</td>
</tr>
<tr>
<td>SISO</td>
<td>Soft-Input Soft-Output</td>
<td>193</td>
</tr>
<tr>
<td>SMV</td>
<td>Symmetric Motion Vector</td>
<td>55</td>
</tr>
<tr>
<td>SVC</td>
<td>Scalable Video Coding</td>
<td>44</td>
</tr>
<tr>
<td>SW</td>
<td>Slepian-Wolf</td>
<td>27</td>
</tr>
<tr>
<td>SWC-NQ</td>
<td>Slepian-Wolf Coded Nested Quantization</td>
<td>39</td>
</tr>
<tr>
<td>SWD</td>
<td>Slepian-Wolf Decoding</td>
<td>148</td>
</tr>
<tr>
<td>TCQ</td>
<td>Trellis Coded Quantization</td>
<td>38</td>
</tr>
<tr>
<td>TDWZ</td>
<td>Transform-Domain Wyner-Ziv</td>
<td>57</td>
</tr>
<tr>
<td>TDFDVC</td>
<td>Transform-Domain Feedback-channel-based DVC</td>
<td>80</td>
</tr>
<tr>
<td>VCEG</td>
<td>Video Coding Experts Group</td>
<td>45</td>
</tr>
<tr>
<td>VLSI</td>
<td>Very-Large-Scale Integration</td>
<td>194</td>
</tr>
<tr>
<td>WZ</td>
<td>Wyner-Ziv</td>
<td>52</td>
</tr>
<tr>
<td>XOR</td>
<td>eXclusive-OR</td>
<td>31</td>
</tr>
</tbody>
</table>
Chapter 1
INTRODUCTION

1.1 Motivation

In an assortment of domains ranging from entertainment, education and medicine to law enforcement and military authorities, a large number of modern-day applications make use of digital multimedia content. The resolution and the dynamic range of digital video content are steadily and significantly increasing. In addition, rapid developments in terms of display technologies are witnessed, with multi-view and 3D display devices swiftly gaining popularity in the market. Aside from these, communication media have gone through an explosion in terms of network heterogeneity, ranging from ultra-fast optical links to error-prone wireless channels.

The diversification of content and the increasing demand in mobility has led to a proliferation of miscellaneous multimedia terminals, such as high-resolution television sets, high-end graphics workstations, portable computers, game consoles, low-power mobile devices, etc. Wireless multimedia sensor technology, in particular, has lately been experiencing an enormous evolution, influencing the present standard of living in diverse fashions. Unlike conventional multimedia technology, wireless multimedia sensor networks are explicitly tailored to the target application, solely focusing on specific tasks.

1.1.1 Wireless Lightweight Multimedia Applications

Wireless sensor networks consist of a web of miniscule devices (i.e., sensors) obligated to (i) collect information from their environment, (ii) carry out simple processing operations and (iii) store or transmit the harvested data to a central base station. A consequence of precipitous advancements and miniaturization in hardware, single sensor devices can nowadays be equipped with visual information acquisition functionality. This has led to the development of wireless multimedia sensor networks [1], which bring in a new wave of wireless lightweight multimedia applications such as:

- **Wireless visual surveillance sensors**: Wireless video surveillance sensor networks comprise interconnected, battery-powered miniature sensors,
Chapter 1

composed of an integrated video camera, a module for processing and a low-
cost wireless transceiver [1]. For instance, the integrated mobile surveillance
and wireless sensor system (iMouse) in [2] consists of numerous static wireless
sensors and several more powerful mobile sensors. Conventional surveillance
solutions usually gather a large amount of data from cabled cameras, thereby
entailing notable computation and manpower resources for analysis.
Integrating wireless surveillance detection into these systems can reduce such
overhead. Additionally, operating either in a detached fashion or in tandem
with existing infrastructures, such sensors facilitate the implementation of
surveillance systems with advanced functionality. Assuming a security
scenario, when the system senses the presence of an intruder, it can for
example run processing algorithms to classify the most likely resource or cause
of invasion [2]. Hence, large-scale surveillance sensor networks can extend the
ability of law enforcement or military agencies to safeguard areas, public
events, private properties and national frontiers.

- **Wireless visual sensors for storage of germane incidents:** This type of
lightweight multimedia application envisages a single or a collection of visual
sensor recorders, operating on battery and being located on specific areas of
high importance [1]. Such sensors can be in sleep mode for most of the time
but, upon detection of a particular event they are automatically activated.
Relevant activation events can for instance include robberies, traffic violations,
restricted area infringements, etc. When switched on, these sensors can record
the event and store video streams or reports. If need be, the gathered
information remains available for future query, or it can be periodically
uploaded to a server in order to restrain the memory requirements of the
sensor.

- **Wearable/On body visual sensors:** Wearable wireless multimedia sensor
networks can find attractive application in diverse ambits, comprising sports
events, healthcare, remote assistance services, military or police force support,
etc. [1]. Helmet-mounted sensor cameras, for example, can be worn by athletes
of sports like water or snow ski, cycling, kite surfing, etc. Individual video
streams captured by each sensor can be uploaded to a website where they can
be purchased and viewed by sports fans. Alternatively, multimedia sensors can
be employed to monitor the activities and behaviors of people suffering from
specific diseases like epilepsy. The collected data can be screened and
analyzed by specialists that can identify the sources or the triggers (for
epileptic seizures) of the pathology. In such application scenarios, wireless
visual sensors are understood to be cheap, battery powered and modest in terms
of computational capacity and memory.

- **Implantable wireless sensors:** The flourishing applicability of wireless (wearable) sensors for clinical diagnoses has paved the road for the recent development of wireless implantable (in vivo) monitoring and intervention sensors. Although challenges concerning long-term stability and biocompatibility constitute topics of ongoing research, several promising prototypes are appearing for managing patients with acute diabetes, for treatment of epilepsy or other neurological conditions and for monitoring of patients with chronic cardiac diseases.

- **Wireless capsule endoscopy:** In order to diagnose diseases of the human gastrointestinal tract, standard examination techniques visualize the esophagus and stomach, i.e., gastroscopy, and the colon, i.e., colonoscopy. However, the lengthiest part of the human digestive system, namely, the small bowel, remains mainly inaccessible to being probed with such invasive diagnostic methods. Recent advances in electronics, micro-system manufacturing, medical imaging and wireless communications have led to the design of wireless capsule endoscopes. A capsule endoscope consists of a miniature battery, an effective illumination source, a video camera, and an RF transceiver. Once taken, the pill records video of the digestive track and transmits to a receiver mounted on the patient's body. Currently, a major aim of capsule endoscope vendors, e.g., Given Imaging, Olympus Optical, is to obtain highly efficient and error-resilient video compression, at low power consumption, and also to further battery life of the capsule.

### 1.1.2 Digital Video Compression Basics

Uncompressed video information demands high data rates which consume excessive bandwidth in a wireless link and vast energy resources of a sensor node. To set an illustrative example, just a single picture in the 4Common Intermediate Format (4CIF), with 704×576 pixels per frame and an 8-bit representation per sample in the 4:4:4 YCbCr sampling format, requires more than 1.21Mbytes; such that a video sequence at 30 frames per second (Hz) needs approximately 292Mbps. This simple example demonstrates that efficiently compressing, alias coding, digital image and video data is of paramount importance for storage and transmission in general and in wireless multimedia sensor networks in particular.

A video compression system is defined by an encoder, which translates the raw video information to a format of a smaller data rate that is suitable for storage and transmission, and the corresponding decoder, which reverses the encoder’s operations to decompress the video data. Compression is obtained by removing the
redundant information in the video signal. This redundancy is temporal, i.e., collocated pixels in neighboring frames have correlated values, and spatial, i.e., the values of adjacent pixels within a frame show strong resemblances.

When the decompressed data is identical to the raw video, the compression system is referred to as lossless. The compression performance of a lossless system is evaluated by means of the obtained rate to represent the coded video stream.

Nevertheless, when the decoded signal is somehow different from the original raw material, the compression is said to be lossy. By introducing distortion, lossy compression methods can achieve higher compression ratios than lossless algorithms. The performance of lossy compression methods is assessed by the coded bit rate – typically measured in bits per second (bps) – given a certain amount of introduced distortion. To express the distortion (i.e., quality loss) of the decoded video, objective fidelity criteria, like the mean squared error (MSE) or the peak signal-to-noise ratio (PSNR) are most commonly employed.

1.1.3 Video Compression Requirements for Multimedia Sensors

The functionality constraints imposed by the abovementioned heterogeneous wireless lightweight multimedia applications and the rapid increase in the amount of video data that is processed, transmitted or stored are calling for video compression systems with the following requirements:

- **High compression performance:** To constrain the required bandwidth and power resources, the volume of video information that a sensor node transmits needs to be limited. This is in need of a video coding system that delivers high compression performance.

- **Limitation of Resources:** Wireless sensors are constricted in terms of size, battery life, memory and processing capability [1]. These limitations confine the hardware and the computational demands of the employed video codec.

- **Variable bandwidth:** Although in wired networks the capacity of each link is assumed to be fixed, in wireless (sensor) networks, the achievable capacity of each wireless link is highly variable. To efficiently address this challenge, the generation of a scalable encoded representation of the video signal is needed. Using scalable coding techniques, only one encoding is performed that produces one single coded bit-stream containing the highest quality, resolution (and frame-rate) representation of the input image or video signal. Then, a very low-complexity extraction of a subset of the bit-stream can be performed. The subset bit-stream represents a valid encoded stream that allows for decoding the video with the best quality constrained by the available channel bandwidth.

---

1 This dissertation will revolve around lossy compression algorithms for video.
• **Communication error resilience**: Establishing dependable communication over error-prone wireless channels requires video compression architectures that provide transmission error robustness.

In order to simultaneously meet these stringent requirements, new video coding paradigms need to be considered and new ways of thinking have to be followed.

### 1.2 **Introduction to Conventional Video Coding**

Over the last decades, painstaking research efforts on digital video coding have resulted in well-established standardized video coding systems such as the MPEG standards, e.g., MPEG-2 [3], MPEG-4 Visual [4], and the H.26x recommendations, e.g., H.261 [5], H.263 [6], H.264/AVC [7, 8]. Focusing on optimizing the compression performance, these mainstream video compression architectures have managed to reduce the storage and bandwidth requirements of digital video with several orders of magnitude.

In these architectures, the redundancy in the video signal is exploited at the encoder by means of predictive coding. In this way, traditional video coding implies joint encoding and decoding of video. Namely, the encoder produces a prediction of the source and then codes the difference between the source and its prediction. Motion–compensated prediction in particular, a key algorithm to achieve high compression performance by removing the temporal correlation between successive frames in a sequence, is very effective but computationally demanding.

Hence, in traditional video coding architectures, the encoder is the computational pillar of the codec and its computational complexity is dominated by the motion estimation operation. On the other hand, the decoder is a relatively dull component that follows the encoder’s instructions. Therefore, such systems are mainly suitable for one-to-many application scenarios, e.g., video broadcasting, where video content is encoded by a powerful encoder and distributed to multiple users with light decoding devices. However, mostly due to this complexity imbalance between encoder and decoder, traditional video coding finds it difficult to simultaneously fulfill all the aforementioned requirements imposed by wireless multimedia sensor applications.

### 1.3 **Introduction to Distributed Video Coding**

The need for efficient video compression architectures maintaining lightweight encoding remains challenging in the context of wireless video sensor devices that have only modest computational capacity or operate on limited battery life.
1.3.1 Distributed Video Coding Principles

The solution to reduce the encoding complexity can be found in the fundamentals of information theory, which constitute an original coding perspective, known as distributed source coding (DSC). The latter stems from the theory of Slepian and Wolf [9] on lossless separate encoding and joint decoding of correlated sources. Subsequently, Wyner and Ziv [10] extended the DSC problem to the lossy case, deriving the rate-distortion (RD) function with side information at the decoder. Driven by these principles, the distributed, alias Wyner-Ziv, video coding paradigm has arisen [11, 12].

Distributed video coding (DVC) offers an attractive route for designing appropriate video coding systems that meet the austere constraints of wireless multimedia sensors. Unlike traditional video coding, in DVC the source redundancies are exploited at the decoder, implying separate encoding and joint decoding. Specifically, a prediction of the source, named side information, is generated at the decoder using already decoded information. By expressing the statistical dependency between the source and the side information in the form of a virtual correlation channel (see for example [11, 12]), compression can be achieved by transmitting parity or syndrome bits of a channel code, which are used to decode the source with the aid of the side information. Hence, computationally expensive tasks, like motion estimation, could be relocated to the decoder, allowing for a flexible sharing of the computational complexity between the encoder and the decoder and enabling the design of lightweight video encoding architectures.

1.3.2 Distributed Video Coding Attributes

Conversely to mainstream predictive video coding systems, DVC schemes facilitate uplink video transmission at the benefit of key advantages:

- First, DVC offers low complexity encoding, which boosts the development of coding architectures that diminish the power processing requirements of the encoder (i.e., the sensor node) [11, 12].
- Second, DVC enables adaptable architectures that effectively switch encoder-decoder complexity [13, 14], thereby increasing the flexibility of video communications.
- Third, since efficient distributed source codebooks are rooted in channel coding constructions [12, 15, 16], distributed joint-source channel coding (DJSCC) of video offers resilience against communication channel errors [12]. This feature is very attractive to achieve reliable communications over wireless error-prone links.
- Fourth, based on layered Wyner-Ziv coding [17], DVC supports (codec-
independent) scalability [18]. This is a key trait for heterogeneous wireless sensor networks, where resources and bandwidth are regularly fluctuating.

- Fifth, DVC facilitates efficient multi-view/stereo coding enabling exploitation of inter-view correlation at the decoder, which unburdens the encoder and suppresses inter-camera communication [19, 20].

Table 1-I attempts a comparison between predictive and distributed video coding tabulating the characteristics, benefits and drawbacks of both paradigms.

**Table 1-I: Characteristics, benefits and drawbacks of traditional predictive and distributed video coding.**

<table>
<thead>
<tr>
<th>Traditional Predictive Video Coding (e.g., H.264/AVC)</th>
<th>Distributed Video Coding</th>
</tr>
</thead>
<tbody>
<tr>
<td>Highly optimized video coding framework</td>
<td>A promising video coding paradigm yet in its early life</td>
</tr>
<tr>
<td>Complex encoder vs. simple decoder</td>
<td>Simple encoder vs. complex decoder</td>
</tr>
<tr>
<td>Targets downlink-driven applications</td>
<td>Suitable for uplink-tuned applications</td>
</tr>
<tr>
<td>High compression efficiency</td>
<td>Good compression efficiency (Lower than the one obtained with traditional video coding)</td>
</tr>
<tr>
<td>Scalable coding significantly increases encoding complexity</td>
<td>Providing scalability incurs a limited cost in encoding complexity.</td>
</tr>
<tr>
<td>Fixed complexity distribution between encoder-decoder</td>
<td>Flexible complexity distribution between encoder-decoder</td>
</tr>
<tr>
<td>Resilience against transmission errors is not part of the standard and requires the use of additional error-protection mechanisms</td>
<td>Built-in error protection – distributed joint source channel coding</td>
</tr>
<tr>
<td>Several codec profiles</td>
<td>Limited codec profiles</td>
</tr>
<tr>
<td>Default use of a rich set of coding modes</td>
<td>DVC with coding modes shows high potential</td>
</tr>
<tr>
<td>Heavily evolved and standardized</td>
<td>There is no standardization effort initiated yet</td>
</tr>
</tbody>
</table>

Thanks to its attractive traits, DVC has already been recognized as a promising technology for wireless multimedia sensor networks and is regarded to play a key role in the design of future wireless visual sensors protocols [1, 21, 22].

### 1.4 Outline and Major Contributions

While offering exceptionally low encoding complexity by shifting computationally expensive operations at the decoder, DVC architectures suffer from lower compression performance with respect to systems following the conventional video coding paradigm. The scope of this dissertation is therefore to bring in novel concepts and tools that enhance the compression capacity of contemporary DVC systems. Envisioning bridging the gap between academic achievements and practical domains, this dissertation also aims at introducing hands-on DVC systems attractive to important applications.

The remainder of this dissertation is structured as follows.
Chapter 1

Chapter 2 introduces the reader to the information theoretic fundamentals related to both conventional and distributed source coding. Specifically, this chapter theoretically formulates the topics of conventional source compression, rate-distortion theory and successive refinement of information. Furthermore, their equivalents in distributed source compression, i.e., the Slepian-Wolf coding argument, the rate-distortion with side information at the decoder problem (a.k.a., the Wyner-Ziv coding problem) and the problem of successively refined Wyner-Ziv coding, are presented.

In the subsequent preparatory chapter, that is, Chapter 3, a comprehensive overview of the most important developments in the domains of conventional and distributed video coding is given. As being the scope of this dissertation, emphasis is put on the most representative DVC systems in the literature, namely, the pioneering architectures developed at Berkeley University [11, 23] and Stanford University [12], as well as the state-of-the-art DISCOVER [24] codec. Furthermore, Chapter 3 comments in brief on the advancements over the state-of-the-art, as introduced by the IBBT DVC research group (to which I belong). The remaining chapters describe in depth the contributions that have led to this thesis.

Unlike traditional codecs, which effectively exploit the temporal correlation in the video sequence at the encoder, DVC schemes carry out motion-compensated prediction at the decoder, which, however, does not have access to the original frame to be encoded. As a consequence, the quality of the temporal prediction, i.e., the side information, in contemporary DVC systems is lower than that of their traditional video coding counterparts. Being identified as a prime factor on the compression performance of DVC [25], the first part of this dissertation, covered in Chapter 4, focuses on the problem of generating high-quality motion-compensated prediction at the decoder. The principal contribution in Chapter 4 is the development of overlapped block motion estimation and compensation (OBMEC), a new side information generation concept that advances over existing approaches. Employed as a hash-based motion estimation tool or as a side information refinement technique, OBMEC yields novel DVC designs that deliver superior compression performance over state-of-the-art DVC systems.

Chapter 5 expands on the second major challenge in DVC, namely, the accurate prediction of the statistical dependency, i.e., the correlation, between the original frame to be encoded and the side information at the decoder. In this context, unlike existing works in the literature, Chapter 5 introduces a novel correlation channel modeling concept, which expresses the correlation noise as being statistically dependent on the side information. Chapter 5 provides a theoretical comparison of the proposed model against the common side-information-independent model in the
literature, proving that it enhances the compression performance of DVC. To accurately estimate the parameters of the proposed model at the decoder, a novel correlation channel estimation technique is described in Chapter 5. The proposed method overcomes the limitations of state-of-the-art correlation channel estimation methods, while delivering higher rate-distortion performance.

Despite the fact that it was launched almost a decade ago, DVC has not yet found its place in the market. Aspiring to alter this, the third key contribution of this dissertation targets the successful application of DVC in a niche medical imaging field. Intrinsically, Chapter 6 presents a novel codec specifically designed for an exciting application of the DVC technology, namely, wireless capsule endoscopy. Conversely to contemporary architectures, the codec described in Chapter 6 includes novel features that can cope with the irregular traits of the motion content acquired with a wireless capsule endoscope. Evaluating its performance on conventional endoscopic and capsule endoscopic video material, the proposed DVC system is shown to bring significant compression improvements over several relevant state-of-the-art codecs.

Chapter 7 draws the overall conclusions of this dissertation and sketches further research possibilities and directions.

At the end of this thesis, a collection of appendixes providing supplementary information and clarifications is included. Since they constitute the platform for efficient DSC, Appendix A gives a brief overview of channel coding principles. Appendix B describes the motion-compensated interpolation method used to generate side information in state-of-the-art codecs, including extensions introduced by this research. Appendix C provides a series of mathematical proofs related to the development of the correlation channel model discussed in Chapter 5.
Chapter 2
INFORMATION THEORETIC FUNDAMENTALS

2.1 INTRODUCTION

This chapter serves as an introduction to the information theoretic principles of distributed data compression. Since their birth in the early 70’s, these fundamental principles have been given a remarkable attention by the information theory community and have gained an important place in the broad subject of information theory. The theory of distributed source coding (DSC) originates from two landmark information theory papers; the first [9] by Slepian and Wolf in 1973, and the second [10] by Wyner and Ziv in 1976. Nevertheless, distributed compression did not exactly derive from parthenogenesis. In fact, DSC concepts build on conventional information theoretic problems in source and channel coding, featuring many analogies and even sharing various dualities with them.

Driven by this spirit, this chapter initially digs into the fundamental notions of conventional source coding, rate-distortion theory and successive refinement of information. Understanding the very basics of these conventional coding concepts will guide the reader through the exploration of their fascinating DSC counterparts, namely, Slepian-Wolf coding, Wyner-Ziv coding and successively refined Wyner-Ziv coding. Although some mathematical formalism is inevitable, the key intention of this chapter is to put emphasis on the philosophy behind the information theoretic problems of coding. With this aim in mind, these information theoretic insights are delivered using the concept of typicality, which is formally defined by the asymptotic equipartition property [26, 27]. Furthermore, simple geometric parallelisms and toy examples (involving bins and spheres) will often be employed to sketchily explain fundamental coding ideas.

Armed by a good comprehension of the information theoretic essentials of DSC, the reader can more easily grasp the basic ideas behind the application scenarios and
the associated coding architectures, as presented in this dissertation. What is more, the information theoretic analyses, presented in this chapter, will provide valuable support to appreciate the implications of input-dependent, or alias, side-information-dependent correlation channel modeling in the context of DSC. This modeling approach, which sees the correlation channel in DSC as being asymmetric, is one of the foremost contributions of this dissertation and is unique in the DSC-related literature.

As regards the structure of this chapter, Section 2.2 provides the fundamentals of lossless source coding. The definition of weakly typical sequences is first introduced, and then based on it, fixed and variable source coding is explained. Moving towards the lossy compression paradigm, Section 2.3 defines the rate-distortion theorem and demonstrates its achievability using typicality. Our study on conventional source coding is completed in Section 2.4, which refers to the important topic of successive refinement of information. After this section, our focus is put on the equivalent information theoretic basics of distributed source coding. Specifically, Section 2.5 elaborates on the topic of distributed compression of correlated information sources. This section presents the definition, the achievability and practical realizations of the Slepian Wolf coding theorem. Subsequently, Section 2.6 discusses the problem of rate-distortion with side information at the decoder. This problem is covered by the definition of the Wyner-Ziv coding theory and the proof of its achievability, as well as, by reviewing practically realizable codes. Finally, this chapter is concluded by Section 2.7, in which we define the problem of successive refinement in the Wyner-Ziv setting and present practical layered Wyner-Ziv codes.

Before we begin, let us say a word about notation. We employ capital italic letters, e.g., $X$, to denote random variables and small italic letters, e.g., $x$, for their realizations or samples. The alphabet of a random variable is denoted by the capital letter $\mathcal{X}$ adding a subscript index referring to the random variable. An array or string of $n$ random variables is signified by $X^n \equiv X_1, X_2, ..., X_n$, while the alphabet of this is given by $\mathcal{X}_n$. Furthermore, we use the notation $x^n$ (also using vector representation $\mathbf{x}^n$ when necessary) to note the $n$-tuple of realizations $x_1, x_2, ..., x_n$.

### 2.2 Source Coding

Primarily, we properly state the mathematical formalism of the data compression problem. Let a random source represented by a random variable $X$ which takes one of possible values in the alphabet $\mathcal{X}_X = \{x_1, x_2, ..., x_n\}$, having probabilities $0 \leq p_X (X = x_i) \leq 1$, with $\sum_{i=1}^{n} p_X (X = x_i) = 1$.

The fundamental question is how can we express the information content of a
realization \(X = x_i\) of such a random variable \(X\). Information theory [26-28] defines two fundamental information content metrics, as follows:

A. The **Shannon information content**, namely, \(- \log p_X(X = x_i)\), which is a measure of the information content of the particular realization \(x_i\) of the random variable \(X\).

B. The **entropy**, i.e., \(H(X) = - \sum_{i=1}^{n} p_X(X = x_i) \log p_X(X = x_i)\) of the random variable \(X\), which expresses the average information content of the random variable (or the source).

The theoretical foundations of source coding have been laid by the asymptotic equipartition property (AEP), which is one of the basic pillars of information theory.

### 2.2.1 The Asymptotic Equipartition Property

The AEP, which is the information theoretic equivalent of the law of large numbers, stems from the weak law of large numbers. As first stated by Shannon in his groundbreaking paper [28], the AEP is defined by the subsequent theorem:

**Theorem 2.1** [26]: Let \(X^n = X_1, X_2, \ldots, X_n\) be independent and identically distributed (i.i.d.) random variables, with marginal probability mass function \(p_X(x)\). Also, let \(p_{X_1X_2\ldots X_n}(x_1, x_2, \ldots, x_n)\) be the probability of observing a specific sequence \(x^n = (x_1, x_2, \ldots, x_n)\). Then, the expression 
\[- \frac{1}{n} \log p_{X_1X_2\ldots X_n}(x_1, x_2, \ldots, x_n)\]
is close to the entropy \(H(X)\), alias,
\[- \frac{1}{n} \log p_{X_1X_2\ldots X_n}(x_1, x_2, \ldots, x_n) \rightarrow H(X), \quad (2.1)\]

**Proof:** Rigorous proofs of the theorem can be found in [26, 27]. The original proof is given in [28].

Analogously to the AEP, the law of large numbers states that for i.i.d. random variables,
\[
\frac{1}{n} \sum_{i=1}^{n} X_i \rightarrow E[X], \quad \text{when} \quad n \rightarrow \infty. \quad (2.2)
\]
where \(E[\cdot]\) is the expectation operator.

Based on the AEP, one concludes that the probability \(p_{X_1X_2\ldots X_n}(x_1, x_2, \ldots, x_n)\) to detect a particular sequence \(x^n = (x_1, x_2, \ldots, x_n)\) is asymptotically approaching the value \(2^{-nH(X)}\). This enables the partition of all possible sequences \(x^n\) of the alphabet \(\mathcal{X}^n\) into two disjoint sets, namely, the typical and the non-typical set. The typical set, denoted by \(\mathcal{A}^{(n)}\) contains the sequences \(x^n \in \mathcal{A}^{(n)}\) of which the probability complies with the following principle [26]:
\[
2^{-n(H(X)+\delta)} \leq p_{X_1X_2\ldots X_n}(x_1, x_2, \ldots, x_n) \leq 2^{-n(H(X)-\delta)}, \quad (2.3)
\]
where \(\delta\) is arbitrarily small. The other sequences of the \(\mathcal{A}^{(n)}\) alphabet, which do
not fall into the above definition, form the set of non-typical sequences.

Anchored in the AEP, the set of typical sequences exhibits the following properties [26, 27]:

**Theorem 2.2 [26]:**

1. If \( \left( x_1, x_2, \ldots, x_n \right) \in \mathcal{A}_\delta^{(n)} \), then
   \[
   H(X) - \delta \leq \frac{1}{n} \log \frac{1}{p_{X_1, X_2, \ldots, X_n}(x_1, x_2, \ldots, x_n)} \leq H(X) + \delta.
   \]

2. If \( n \to \infty \), then \( \Pr\left( \mathcal{A}_\delta^{(n)} \right) > 1 - \delta \).

3. For the number of elements (cardinality) in the typical set, we have
   \[
   \left| \mathcal{A}_\delta^{(n)} \right| \leq 2^{n(H(X) + \delta)}.
   \]

4. If \( n \to \infty \), then
   \[
   \left| \mathcal{A}_\delta^{(n)} \right| \geq (1 - \delta) 2^{n(H(X) - \delta)}.
   \]

**Remark 2.1:** Based on the above-stated properties of the typical set sequences one can note the following [26]:

1. If a sequence belongs to the typical set, then the average Shannon information content [27] of the sequence is asymptotically close to the entropy of the source.

2. The probability of typical set is approximately one. That is, asymptotically \( (n \to \infty) \), the probability that a sequence produced by the source is typical is almost one.

3. The number of sequences in the typical set is nearly equal to \( 2^{nH(X)} \).

4. All elements of the typical set are asymptotically equally probable, each of which having a probability of appearance almost \( 2^{-nH(X)} \).

Notice also that the abovementioned properties of the typical sequences are true with high probability and determine the average behavior of a large sample, i.e. when \( n \to \infty \).

### 2.2.2 Data Compression with Fixed Length Codes

The AEP provides the theoretical means to design an efficient data compression code. Such a code can compress data from a particular source into symbols belonging to an alphabet of a smaller size, and still be able to recover the original data reliably. Namely, an efficient data compression code aims to find short descriptions of the \( X^n \) sequences of a random variable \( X \).

To do so, based on the AEP, Shannon divided all sequences in the random variable’s alphabet \( \mathcal{A}_X \) into two sets, that is, the typical set \( \mathcal{A}_\delta^{(n)} \) and its complement, a.k.a., the non-typical set. This separation is schematically illustrated in Figure 2.1. To derive his coding argument, Shannon ordered all the elements in each set according to some order, e.g., lexicographic order.
Then, he indicated each sequence in the typical set by a specific index corresponding to this particular sequence in the set only. Asymptotically, as there are approximately less than \(2^{n(H(X) + \delta)}\) sequences in the typical set \(\mathcal{A}_b^{(n)}\) (see Theorem 2.2 above), Shannon’s indexing requires no more than \(n(H(X) + \delta) + 1\) bits\(^2\) to catalogue the typical sequences. Furthermore, the defined code symbols of the typical sequences are prefixed by one bit (e.g., 0), which indicates the set they belong to. As a consequence, Shannon’s asymptotic code assigns a total length of no more than \(n(H(X) + \delta) + 2\) bits to represent each sequence in the typical set.

Likewise, Shannon indexed each sequence of the non-typical set by using no more than \(n\log|\mathcal{A}_X| + 1\) bits. Moreover, similar to the code for the typical sequences, the defined code symbols for the non-typical sequences are prefixed by for example the bit 1, to uniquely identify the set. Notice that this initial bit also acts as a flag bit to indicate the length of the codeword that follows.

The above-explained procedure defines an one-to-one, straightforwardly decodable code to represent all the sequences in \(\mathcal{A}_X\). One observes that Shannon used short descriptions to label the sequences of the typical set. On the other hand, Shannon used a brute-force indexing of the sequences in the non-typical set. In his indexing, he completely “neglected” the fact that the number of non-typical sequences is definitely less than the total number of sequences in \(\mathcal{A}_X\).

However, recall from Theorem 2.2 that the probability of the non-typical set is nearly \(\delta\), which can become arbitrarily small when \(n \rightarrow \infty\). Therefore, asymptotically, the contribution of the non-typical sequences in the average length of Shannon’s code is negligible.

It can be proven that the expected codeword length of Shannon’s code is given by [26]

\(^2\) The additional bit is included in case \(n(H(X) + \delta)\) is not an integer.
\[
\frac{1}{n} E \left[ l(X^n) \right] \leq H(X) + \delta ,
\]  
(2.4)

where \( l(X^n) \) represents the codeword length and \( n \to \infty \) [26].

### 2.2.3 Data Compression with Variable Length Codes

According to the previously detailed fixed length coding paradigm, one can represent \( X^n \) sequences using \( nH(X) \) bits on average. Notice that such a coding argument is asymptotically efficient, that is, when the number of source samples goes to infinity.

However, common sense dictates that improbable source outcomes (i.e., realizations) convey more information than probable ones, and hence they should be described using a longer codeword. Indeed, in practice, efficient data compression is realized by using variable length coding. The latter assigns short descriptions to the most common outcomes of the source, and longer descriptions to the less recurrent outcomes. To describe the basic principles of variable length coding, we use the following notation [26]:

A code \( \mathcal{C} \), for a source represented by the random variable \( X \), is a mapping from \( \mathcal{X} \), i.e., the alphabet of \( X \), to a group of finite-length strings of symbols taken from an alphabet with cardinality \( \Omega \). Also, let \( C(x) \) signify the codeword corresponding to the source realization \( x \) and \( l(x) \) represent the length of this codeword.

Furthermore, the subsequent definitions are necessary to characterize variable length codes [26]:

- A source code is called *non-singular*, if it assigns a different codeword \( C(x_i) \) to every different realization \( x_i \) of the source.
- A source code is referred to as *uniquely decodable*, if sequences consisting of the code’s codewords are non-singular.
- A code is said to be *prefix* or *instantaneous*, if it does not contain a codeword that is a prefix of another codeword. In this way, using an instantaneous code, one can straightaway identify the different codewords of the code included in a string of codewords.

The collection of codeword lengths that are achievable for instantaneous codes is constrained by the ensuing inequality, which is known as Kraft’s inequality:

**Theorem 2.3 [26]:** For any prefix code \( \mathcal{C} \) over an alphabet of cardinality \( \Omega \), the codeword lengths \( l(x_i) \), \( i = \{1, 2, \ldots, n\} \) must satisfy the inequality

\[
\sum_{i=1}^{n} \Omega^{-l(x_i)} \leq 1.
\]  
(2.5)

Conversely, given a set of codeword lengths that satisfy Kraft inequality, there
exists a prefix code with these word lengths.

Proof: A meticulous proof can be found in [26].

It is important to note that the converse of Theorem 2.3 states that, given the codewords’ lengths $l(x_i)$, $i = \{1, 2, ..., n\}$, there exists an instantaneous code, not that every such code is instantaneous.

We now turn to the problem of determining the most efficient prefix code, i.e., the prefix code with the minimum expected length. Anchored in Theorem 2.3, this problem is equivalent to finding the set of lengths $l(x_i)$, $i = \{1, 2, ..., n\}$ satisfying the Kraft inequality and whose expected length $L = \sum_{i=1}^{n} p_X(x_i) l(x_i)$ is less than the expected length of any alternative prefix code. The following stands:

**Theorem 2.4 [27]:** (Lower bound on expected length) The expected length $L$ of a uniquely decodable code is bounded below by the average Shannon’s information content, a.k.a., the entropy. Namely,

$$L = \sum_{i=1}^{n} p_X(x_i) l(x_i) \geq H(X).$$ (2.6)

Proof: A meticulous proof can be found in [27].

Observe that in Eq. (2.6), equality holds when the codeword lengths are equal to the Shannon information contents of the source realizations. This means that the lower expected length, i.e., $H(X)$, is achieved when

$$l(x_i) = -\log p_X(x_i).$$ (2.7)

A question that is immediately raised is how close can we go to the entropy? The answer to this question is given by the fundamental source coding theorem, as stated below:

**Theorem 2.4 [27]:** (Source coding theorem) To compress a random source described by a random variable $X$, one can construct an instantaneous source code $C$ with expected length $L$ satisfying the following double inequality

$$H(X) \leq L \leq H(X) + 1.$$ (2.8)

Proof: A comprehensive proof is sketched in [26, 27].

This code will be composed of codewords with lengths $l(x_i) = \lceil \log(1/p_X(x_i)) \rceil$, where $\lceil \cdot \rceil$ denotes the ceiling function.

What is then the cost of using a source code of which the codeword lengths are not optimal? Suppose that the source probability mass function (pmf) is described by $p_X(x_i)$, $i = \{1, 2, ..., n\}$, and instead of the optimal $l(x_i)$, $i = \{1, 2, ..., n\}$ codeword lengths, the designed code employs $l'(x_i)$, $i = \{1, 2, ..., n\}$. We can view these lengths as defining another implicit pmf $p'_X(x_i)$, $i = \{1, 2, ..., n\}$, where $p'_X(x_i) = 2^{-l'(x_i)}$ or $l'(x_i) = -\log p'_X(x_i)$. Then, from Eq. (2.6), the average code length is
\[ L = \sum_{i=1}^{n} p_X(x_i) I'(x_i) \]
\[ = -\sum_{i=1}^{n} p_X(x_i) \log p'_X(x_i) \]
\[ = -\sum_{i=1}^{n} p_X(x_i) \log \left( \frac{p_X(x_i) p'_X(x_i)}{p_X(x_i)} \right) \]
\[ = -\sum_{i=1}^{n} p_X(x_i) \log p_X(x_i) + \sum_{i=1}^{n} p_X(x_i) \log \left( \frac{p_X(x_i)}{p'_X(x_i)} \right) \]
\[ = H(X) + D\left( p_X(x_i)\|p'_X(x_i) \right) \]

Namely, such a code exceeds the entropy by the relative entropy, or else the Kullback-Leibler distance (see [26, 27] for a comprehensive explanation of the latter information theoretic metric).

The source coding theorem can be extended to a generic stochastic process, i.e., a sequence of random variables \( X^n = X_1, X_2, \ldots, X_n \). In this case Eq. (2.8) is modified to:
\[
\frac{H(X_1, X_2, \ldots, X_n)}{n} \leq L_n \leq \frac{H(X_1, X_2, \ldots, X_n)}{n} + \frac{1}{n} (2.9)
\]
where \( L_n = \frac{1}{n} \sum p_{X_1, X_2, \ldots, X_n}(x_1, x_2, \ldots, x_n) I(x_1, x_2, \ldots, x_n) \) is the expected codeword length associated with the \( (x_1, x_2, \ldots, x_n) \) outcome \( n \)-tuples.

Obviously, if the random variables \( X^n = X_1, X_2, \ldots, X_n \) are considered i.i.d., then Eq. (2.9) yields
\[
H(X) \leq L_n \leq H(X) + \frac{1}{n}, \quad (2.10)
\]
as in this case \( H(X_1, X_2, \ldots, X_n) = H(X^n) = nH(X) \).

Practical realizations of variable length coding include Fano [26], Huffman [29], Golomb [30], Lempel-Ziv [31], and arithmetic codes [32]. For sources with memory, e.g., image and video data, efficient variable lengths codes enable context adaptation leading to state-of-the-art entropy coding schemes, like context-adaptive variable-length coding (CAVLC) [33], and context-based adaptive binary arithmetic coding (CABAC) [34].

### 2.3 Rate Distortion Theory

Up to this point, our analysis has concentrated on lossless compression of sources with finite alphabets. However, losslessly compressing a continuous random source would require infinite precision, which cannot be reproduced using finite-rate codes. What is more, as noted in Section 1.1.2, lossy compression algorithms can achieve...
higher compression ratios than lossless ones. Practically, lossless compression is typically used for scientific applications in various domains such as medical or satellite imaging. In essence, state-of-the-art image and video codecs are performing lossy compression.

The fundamental question in lossy coding is to determine the best possible representation of the encoded source (i.e., the least possible distortion) for any given data rate.

2.3.1 Distortion Metrics

To appropriately answer to this question, it is essential to first express the goodness of a representation of a source. This is accomplished by defining a distortion metric \( d(x, \hat{x}) \), that is, a mapping

\[
d : \mathcal{A}_X \times \mathcal{A}_{\hat{X}} \to \mathbb{R}^+
\]

(2.11)

from the group of source alphabet (e.g., \( \mathcal{A}_X \)) and reconstruction alphabet (e.g., \( \mathcal{A}_{\hat{X}} \)) couples to the set of non-negative real numbers. Namely, a distortion metric \( d(x, \hat{x}) \) is a measure of the distance between the random variable (i.e., \( X \)) and its representation (i.e., \( \hat{X} \)).

In the following, we provide examples of common distortion metrics. For binary sources, the Hamming distortion metric is given by

\[
d(x, \hat{x}) = \begin{cases} 0, & x = \hat{x} \\ 1, & x \neq \hat{x} \end{cases}
\]

(2.12)

Another example includes the \( n \)-th order distortion metric, which is defined as

\[
d(x, \hat{x}) = |x - \hat{x}|^n.
\]

(2.13)

A particular case is obtained when \( n = 2 \) in (2.13), leading to the squared-error distortion metric, i.e.,

\[
d(x, \hat{x}) = (x - \hat{x})^2.
\]

(2.14)

The latter is the most common distortion measure for continuous sources, and is typically used in image and video coding applications.

Extending the aforementioned metric to measure the distortion between sequences of samples, the mean squared-error (MSE) distortion metric is given by

\[
d(x^n, \hat{x}^n) = \frac{1}{n} \sum_{i=1}^{n} (x_i - \hat{x}_i)^2.
\]

(2.15)

That is, the MSE distortion of a sequence is given by the mean of the per-symbol distortion of the elements of the sequence.
2.3.2 Rate-Distortion Function

The essential problem in *rate-distortion theory* can thereafter be phrased as follows: Given a source distribution and a distortion metric, what is the minimum expected distortion attainable at a specific rate? Otherwise, what is the minimum rate required to achieve a certain amount of distortion?

![Schema of a rate-distortion coding system](image)

We continue with the definitions of source coding (see Figure 2.2), as given by Cover and Thomas [26]:

A \((2^{nR}, n)\) rate-distortion code, namely, a source code that represents sequences \(X^n\) of \(n\) random variables \(X_1, X_2, \ldots, X_n\) into one out of \(2^{nR}\) codewords, comprises of an encoding function, i.e.,

\[ f_n : \mathcal{A}_{X^n} \rightarrow \{1, 2, \ldots, 2^{nR}\} \]  

and a decoding (alias, reconstruction) function,

\[ g_n : \{1, 2, \ldots, 2^{nR}\} \rightarrow \mathcal{A}_{\hat{X}^n} . \]  

The distortion corresponding to this code is given by

\[ D = E\left[ d(X^n, g_n(f_n(X^n))) \right], \]  

where the expectation operation is in regard to the source pmf (or probability density function, a.k.a., pdf, for continuous sources), that is,

\[ D = \sum_{x^n} p_{X^n}(x^n) d\left(X^n, g_n\left(f_n\left(X^n\right)\right)\right). \]  

A rate-distortion pair \((R, D)\) is said to be achievable [26] if there is a \((2^{nR}, n)\) source code with encoding and decoding functions \(f_n, g_n\), respectively, that satisfy

\[ \lim_{n \rightarrow \infty} E\left[ d\left(X^n, g_n\left(f_n\left(X^n\right)\right)\right) \right] \leq D . \]  

Based on the previous definitions [26], the rate-distortion region for a source is then defined as the entire set of achievable rate-distortion pairs \((R, D)\). It follows that, the rate distortion function is the infimum of rates such that \((R, D)\) is in the rate distortion region of the source for a given distortion \(D\). Using an austere mathematical formalism, the rate-distortion function is defined by the following
Theorem 2.5 [26]: The rate distortion function for an i.i.d. source $X$ with pmf $p_X(x)$ and distortion metric $d(x, \hat{x})$ is equal to the minimum achievable rate at distortion $D$, that is,

$$ R(D) = \inf_{p_{\hat{X} | X}(\cdot | x) \sum_{x} p_X(x) p_{\hat{X} | x}(\cdot | x) d(x, \hat{x}) \leq D} I(X; \hat{X}), $$

(2.21)

where $I(X; \hat{X})$ denotes the mutual information between the random source and the reconstruction alphabet.

Proof: For a thorough proof the reader is referred to [26, 27].

Note that, in Eq. (2.21), the minimization is over all conditional pmfs $p_{\hat{X} | X}(\cdot | x)$ for which the joint pmf, i.e., $p_{\hat{X} X}(\cdot, x) = p_X(x) p_{\hat{X} | x}(\cdot | x)$, satisfies the distortion constraint. Notice also that the lower the mutual information between the $X$ and $\hat{X}$ random variables, the lower the achievable rate for a specific distortion constraint. Conversely, the upper bound in the rate-distortion region is given by setting the rate to the entropy of the source, i.e., $H(X)$, which induces a zero distortion.

2.3.3 Achievability of the Rate-Distortion Function

An essential and illustrative part of the proof of Theorem 2.5 is the achievability of the rate-distortion function, which is briefly sketched in this section. Before we start, we need to explain a modified version of the joint AEP, which involves the concept of a pair of sequences that are typical with respect to the distortion measure [26]. Specifically, similar to the definition of typicality in Section 2.2.1, a pair of sequences $(x^n, \hat{x}^n)$ is called distortion typical [26] if

$$ \left| \frac{1}{n} \log \frac{1}{p_{X^n}(x^n)} - H(X) \right| < \delta $$

(2.22)

and

$$ \left| \frac{1}{n} \log \frac{1}{p_{\hat{X}^n}(\hat{x}^n)} - H(\hat{X}) \right| < \delta $$

(2.23)

and

$$ \left| \frac{1}{n} \log \frac{1}{p_{X^n, \hat{X}^n}(x^n, \hat{x}^n)} - H(X, \hat{X}) \right| < \delta $$

(2.24)

and

$$ d(x^n, \hat{x}^n) - E[d(X, \hat{X})] < \delta $$

(2.25)

In this context, the collection of distortion typical sequences is termed distortion typical set and is denoted by $\mathcal{A}_{\delta}^{(n)}$ [26].

3 A comprehensive proof of the achievability of the rate-distortion function can be found in [26].
In the following, we briefly show that the rate distortion pair \((R,D)\) is achievable by demonstrating the existence of a source code with rate \(R\) that asymptotically provides distortion \(D\). According to the argument detailed in [26], consider a conditional pmf \(p_{\hat{X}|X}(\hat{x}|x)\), which achieves the bound in Eq. (2.21), namely, \(R(D) = I(X; \hat{X})\). Also, using the definition, compute the marginal pmf of the reconstruction points \(p_{\hat{X}}(\hat{x}) = \sum_x p_X(x) p_{\hat{X}|X}(\hat{x}|x)\). Prior to encoding, let us randomly generate a rate-distortion code consisting of \(2^{nR}\) i.i.d. sequences \(\hat{X}^n\) drawn by \(\prod_{i=1}^n p_{\hat{X}_i}(\hat{x}_i)\). Catalog these sequences (i.e., codewords) by \(\omega \in \{1,2,\ldots,2^{nR}\}\) and then reveal this codebook to the encoder and decoder. During encoding, let the encoder represent a sequence \(X^n\) by \(\omega\) if there exists an \(\omega\) such that \(X^n\) and \(g_n(\omega)\) are distortion typical sequences, i.e., if \((X^n, g_n(\omega)) \in \mathcal{A}_{d,\delta}^{(n)}\). If there is more than one such \(\omega\), send the least. If there is no such \(\omega\), send the least \(\omega\), i.e., \(\omega = 1\). Hence, \(nR\) bits are used to encode the source. At the decoder, the reconstructed sequence is found as \(\hat{X}^n = g_n(\omega)\).

Analyzing the aforementioned coding scheme, we divide the source sequences \(x^n \in \mathcal{A}_{x^n}\) into two classes [26]:

(i) Sequences for which there exists a distortion typical codeword. As according to the definition of distortion typicality the overall probability of these sequences is at most 1, these sequences contribute maximum \(D + \delta\) to the expected distortion.

(ii) Sequences for which there was no distortion typical codeword found. In this case, denote by \(P_{\text{error}}\) the overall probability of these sequences. Based on the definition of distortion typicality the total probability \(P_{\text{error}}\) of these sequences is asymptotically equal to zero (see [26] for further details).

Hence, the expected distortion induced by the afore-explained coding scheme is asymptotically bounded by

\[
E\left[d\left(x^n, \hat{x}^n\right)\right] \leq D + \delta. \tag{2.26}
\]

Let us now provide a geometric interpretation of the rate-distortion theorem. Assume a zero-mean Gaussian source \(X\) that generates source samples \(x_i\) subject to a power constraint \(P = E\left[X^2\right] = \sigma^2\). With high probability, sequences \(x_1, x_2, \ldots, x_n\) of source samples lie within a sphere of radius \(\sqrt{nP} = \sigma\sqrt{n}\) in an \(n\)-dimensional space\(^4\). The source coding theorem addresses the problem of optimally covering the area of the sphere using the least possible number of \(2^{nR}\) codeword spheres, as depicted in Figure 2.3. Each of these codeword spheres contains the

\(^4\) The volume of an \(n\)-dimensional sphere of radius \(\rho\) is \(V = \frac{\pi^{n/2}}{\Gamma((n/2)+1)}\rho^n\), where \(\Gamma(\cdot)\) is the gamma function.
source sequences that are located within a distance $\sqrt{nD}$ with a codeword, where $D = E\left[(x - \hat{x})^2\right]$ and $0 \leq D \leq \sigma^2$. Therefore, the least possible number of codewords is found by dividing the volume of the outer sphere by the volume of the inner sphere, i.e., $2^{nR} = \left(\sqrt{\sigma^2 / D}\right)^n$.

![Figure 2.3: The sphere covering argument: Geometric illustration of the source coding theorem. The outer sphere represents the space occupied by source sequences (with high probability), while the inner spheres signify the space occupied by the codewords of the employed source code.](image)

This geometric argument, which is known as sphere covering, is the reverse of the sphere packing argument used to describe the Shannon channel coding theorem [26, 27]. This implies that good channel codes can actually be used to perform source coding. This implication is realizable based on the Slepian-Wolf and Wyner-Ziv theoretical findings, as detailed in Sections 2.5 and 2.6.

Source coding is typically realized by quantization. Essentially, the encoding function maps the source samples to a finite set of values, called quantization indices, which correspond to assignment regions or quantization cells. At the decoder, the source samples are reconstructed as the center-of-mass (a.k.a. the centroid) of the associated cell.

In order to determine good quantizers, an effective yet simple iterative algorithm is constructed. In detail, given a set of reconstruction points (i.e., a decoding function), the algorithm finds the optimal set of reconstruction regions, or the encoding function. These regions are the nearest-neighbor regions with respect to the employed distortion metric. Subsequently, the algorithm finds the optimal reconstruction points for these regions, and repeats the iteration for this new set of reconstruction points. Notice that, after each iteration of the algorithm, the expected distortion is decreased, and thus the algorithm will converge to a
Chapter 2

local minimum value of the distortion. This algorithm is known as the Lloyd-Max algorithm [35, 36].

2.4 SUCCESSIVE REFINEMENT OF INFORMATION

In the previous section, we described fixed-rate, alias non-progressive, source coding, in which a specific amount of information rate was transmitted to optimally describe the source samples with a certain distortion. In several applications, though, it is desirable to describe a source with an initial distortion and afterwards choose to further reconstruct it more accurately. This type of coding is referred to as progressive (or quality scalable) coding. To achieve optimality with progressive coding, any supplement to the original information rate should correspond to the rate-distortion performance obtained with solid source coding, that is, as if the total information was transmitted at once.

In essence, the problem of successive refinement of information, the information theoretic foundation of which has been extensively studied by Equitz and Cover [37], consists of first approximating data using a few bits of information, then progressively improving the approximation as more and more information is supplied. The critical goal is to achieve rate-distortion optimality at each progressive stage.

![Figure 2.4: Illustration of the successively refined coding scheme.](image)

2.4.1 Formal Definition of Successive Refinement of Information

For instance, consider a two-stage successive refinement of a sequence of random variables \( X^n \equiv X_1, X_2, \ldots, X_n \) that achieves optimality at each stage. A schema of the assumed successively refined coding setting is depicted in Figure 2.4. In detail, at the coarse stage, the sequence \( X^n \) is represented using a rate of \( nR_1 \) bits incurring distortion \( D_1 \). Subsequently, at the refinement stage, a rate supplement of \( n(R_2 - R_1) \) bits is transmitted such that the refinement-stage decoder reconstructs...
the sequence with distortion $D_2$, where $D_2 \leq D_1$. In this setting, the sequence $X^n$ is said to be successively refined if the following relations are satisfied:

$$R_1 = R(D_1) \quad \text{and} \quad R_2 = R(D_2)$$

(2.27)

where $R(D)$ is the rate-distortion function for the sequence $X^n$, as described in Section 2.3.

In this context, a strict information theoretic formalism of successively refined coding, as given by Equitz and Cover [37], is the following: Successive refinement from distortion $D_1$, to distortion $D_2$, where $D_2 \leq D_1$, is achievable if there exist two encoding functions, namely,

$$f_{1,n} : \mathcal{A}_X^n \to \{1, 2, \ldots, 2^n R_1\},$$

$$f_{2,n} : \mathcal{A}_X^n \to \{1, 2, \ldots, 2^n (R_2 - R_1)\},$$

(2.28)

and two decoding functions, that is,

$$g_{1,n} : \{1, 2, \ldots, 2^n R_1\} \to \mathcal{A}_{X_1^n},$$

$$g_{2,n} : \{1, 2, \ldots, 2^n R_1\} \times \{1, 2, \ldots, 2^n (R_2 - R_1)\} \to \mathcal{A}_{X_2^n},$$

(2.29)

such that for $\hat{X}_1^n = g_{1,n}(f_{1,n}(X^n))$ and for $\hat{X}_2^n = g_{2,n}(f_{1,n}(X^n), f_{2,n}(X^n))$, we have

$$\limsup_{n \to \infty} E\left[ d\left( X^n, \hat{X}_1^n \right) \right] \leq D(R_1),$$

(2.30)

and

$$\limsup_{n \to \infty} E\left[ d\left( X^n, \hat{X}_2^n \right) \right] \leq D(R_2),$$

(2.31)

where $D(R)$ is the distortion-rate function for the sequence $X^n$. If the aforementioned definition of successive refinement from distortion $D_1$, to distortion $D_2$, can be generalized to every $D_2 \leq D_1$, then the scheme is called successively refined in general.

### 2.4.2 Conditions for Achievability

In principle, it is difficult to achieve optimality with progressive coding, as optimal rates, i.e., rates associated with distortions on the rate-distortion curve, are not always successive refinements of one another. Equitz and Cover [37] showed that optimality could not be achieved in simple progressive compression of a single Gaussian random variable $X \sim \mathcal{N}(0,1)$ using the MSE distortion metric. Nevertheless, they showed that when assuming a sequence $X^n$ of i.i.d. random variables successive refinability is asymptotically feasible (i.e., when $n \to \infty$).

In the above two-stage coding scheme, shown in Figure 2.4, Equitz and Cover
[37] proved that successive refinement from a coarse level $\hat{X}_1^n$ (with distortion $D_1$) to a finer level $\hat{X}_2^n$ (with distortion $D_2$) can be achieved if and only if the conditional pmfs $p_{\hat{X}_i|X}(\hat{x}_i|x)$ and $p_{\hat{X}_i|X^n}(\hat{x}_i^n|x)$ that achieve the non-progressive rate-distortion bound can form a Markov chain $\hat{X}_1 \leftrightarrow \hat{X}_2 \leftrightarrow X$, namely, the joint pmf can be written as

$$p_{\hat{x}_1,\hat{x}_2,x}(\hat{x}_1,\hat{x}_2,x) = p_{x}(x)p_{\hat{x}_1|X}(\hat{x}_1|x)p_{\hat{x}_2|\hat{x}_1}(\hat{x}_2|\hat{x}_1)$$

(2.32)

In particular, successive refinement is verified to be feasible for all finite alphabet sources with Hamming distortion, for Gaussian sources with squared-error, and for Laplacian sources with absolute-error [37].

### 2.4.3 Practical Progressive Coding

Embedded quantization constitutes the typical means to practically realize progressive source coding. The basic idea in embedded quantization is that the quantization cells of higher rate quantizers are embedded within the quantization cells of lower rate quantizers. Equivalently, the quantization cells of lower rate quantizers are partitioned to yield the quantization cells of higher rate quantizers. In this setup, a particularly important set of quantizers is the family of embedded deadzone quantizers. Notice that, although it is sub-optimal in rate-distortion sense, embedded double deadzone scalar quantization, also known as Successive Approximation Quantization (SAQ), is often employed in embedded image coding due to its easy practical implementation [38].

Essentially, though typically coming with a performance loss compared to non-progressive coding, progressive (scalable) image and video coding [38-40] is nowadays considered as the most viable technical solution for multimedia providers to support the diverse requirements of the numerous devices and channels. Indeed, scalable coding can generate one single bit-stream that can be used to accommodate the quality, resolution and/or frame-rate of the transmitted encoded video stream to the fluctuating channel bandwidth and to the needs of the end-user device, without requiring transcoding or re-encoding for each application.

We now turn to distributed data compression. The information theoretic foundations of the DSC principles comprise the Slepian-Wolf [9], and Wyner-Ziv [10] theorems, which have recently been extended to the successively refined scenario by Steinberg and Merhav [41].
2.5 DISTRIBUTED ENCODING OF CORRELATED SOURCES

Consider the compression of two correlated, discrete, i.i.d. random sources $X$ and $Y$. In traditional (predictive) coding, both the encoder and the decoder have access to the statistical dependencies between the sources. According to Shannon’s source coding theory [28, 42], the achievable lower rate bound for lossless compression is given by the joint entropy, $H(X,Y)$. Adhering to an unorthodox coding perspective, a distributed coding scenario considers the sources to be independently encoded and jointly decoded, exploiting the inter-source correlation at the decoder side.

![Figure 2.5: Distributed compression of two correlated, i.i.d., discrete random sequences, $X$ and $Y$.](image)

**2.5.1 The Slepian-Wolf Coding Theorem**

Figure 2.5 illustrates the concept, in which sources $X$ and $Y$ are encoded by separate encoders at rate $R_X$ and $R_Y$, respectively, but jointly decoded by a set of linked decoders, producing the reconstructed signals $\hat{X}$ and $\hat{Y}$. The distributed source coding scenario was first studied by Slepian and Wolf [9].

When the compression is lossless, the achievable rate region for decoding $X$ and $Y$ with an arbitrarily small error probability, is given by the Slepian-Wolf theorem [9]:

\[
R_X \geq H(X|Y) \\
R_Y \geq H(Y|X) \\
R_X + R_Y \geq H(X,Y),
\]

where $H(X|Y)$ is the conditional entropy of $X$ given $Y$ and $H(Y|X)$ is the conditional entropy of $Y$ given $X$. The achievable rate region for the Slepian-Wolf (SW) coding scenario is schematically depicted in Figure 2.6.
The inequalities in (2.33) reveal that even when correlated sources are coded independently, a total rate equal to the joint entropy suffices to achieve lossless compression. Hence, according to information theory, lossless distributed encoding of i.i.d. correlated sources does not have any compression efficiency loss compared to joint encoding. It is important to point out that Slepian and Wolf [9] proved the achievability of their distributed compression scheme for i.i.d. correlated sources. However, their result has been extended to any arbitrary correlated sources that satisfy the asymptotic equipartition property, e.g., the case of any jointly ergodic source [43].

2.5.2 Achievability of Slepian-Wolf Coding: The Random Binning Argument

To explain how Slepian-Wolf coding works, we hereby summarize the non-constructive coding argument of Slepian and Wolf [9]. In brief, consider the coding case described by the point A in Figure 2.6, which corresponds to the achievable rates $R_Y = H(Y)$ and $R_X = H(X | Y)$. According to the AEP, which is discussed in Section 2.2.1, the $Y$-encoder can asymptotically construct a code, which requires $nH(Y)$ bits to efficiently encode $Y_n$, so that the decoder can reconstruct $Y_n$ with arbitrarily low probability of error.

At this moment, the joint decoder can use the decoded $Y_n$ sequence as side information to Slepian-Wolf decode a correlated $X_n$ sequence using $nH(X | Y)$ bits. Notice that, based on the definition of jointly typical sets\(^5\), associated with

\(^5\) Similar to the definitions of typicality and distortion typicality (see Sections 2.2.1 and 2.3.3), the set $\mathcal{A}^{(\delta)}_{n}$ of jointly typical sequences $X_n$, $Y_n$ with regard to the joint distribution $p_{XY}(x,y)$ contains the sequences with empirical entropies $\delta$-close to the true entropies [26].
every $Y^n$ sequence, there exists a typical set of $X^n$ sequences that are jointly typical with the given $Y^n$, as illustrated in Figure 2.7. In traditional coding, since the $X$-encoder knows $Y^n$, it can send the index of the $X^n$ sequence within this typical fan. In this case, the decoder, which also knows $Y^n$, can then construct this typical set, and hence reconstruct $X^n$.

In Slepian-Wolf coding, however, the $X$-encoder does not know the sequence $Y^n$, realized only at the separate $Y$-encoder and decoded at the decoder. The achievability of the Slepian-Wolf theorem is proved using a random binning argument, which stems from the concept of hash functions [9]. In detail, Slepian and Wolf thought of the following random binning code. During the code construction, all possible $X^n$ sequences of the $X$-source are randomly distributed into $2^{nR_X}$ bins, as sketched in Figure 2.8. This random procedure can be seen as first placing down $2^{nR_X}$ bins and then randomly throwing the $X^n$ sequences into the bins.

Figure 2.7: Depiction of jointly typical sets [26].

Figure 2.8: The Slepian-Wolf random binning argument. Randomly distribute the $X^n$ sequences of the source alphabet into $2^{nR_X}$ bins. Asymptotically, the average number of $X^n$ sequences in each bin is $|A_{X^n}|/2^{nR_X}$. 

29
Thereafter, this random code is revealed to both the X-encoder and the joint decoder. To encode the \( X^n \) sequence generated by the X-source, the X-encoder finds the bin to which the generated sequence belongs, and sends the corresponding index, i.e., \( \{1, 2, ..., 2^{nR_X}\} \), to the joint decoder. The decoder receives the index of the bin and searches for the sequence inside the bin which is jointly typical with the available side information sequence \( Y^n \). If there is one and only one jointly typical \( X^n \) sequence inside the bin corresponding to the received index, this sequence is declared to be the decoded \( \hat{X}^n \) sequence; otherwise, an error is declared.

To analyze the probability of error for the above-described Slepian-Wolf code, we distinguish between the following two cases:

First, if the encoded \( X^n \) sequence is a jointly non-typical sequence, then the decoder will decode a typical sequence into the bin corresponding to the received index. In this case, there will always be a decoding error. Nevertheless, according to the AEP (see Section 2.2.1), the probability that a non-typical sequence is generated becomes asymptotically very small, that is, when \( n \to \infty \) then \( P[X^n \notin \mathcal{A}_b^{(n)}] \to \delta \).

Second, if the encoded \( X^n \) sequence is jointly typical with the side information sequence, then the bin corresponding to this source sequence will contain at least one typical sequence (that is, the source sequence itself). In this case, there will be an error only if there is more than one typical sequence in this bin. However, anchored in the hash functions theory it can be proven that, if the number of bins is larger than the number of typical sequences, then the probability of more than one typical sequence in a bin is asymptotically very small [9, 26]. Hence, the probability that a generated typical sequence will result in an error becomes asymptotically very small. Indeed, as proved by Slepian and Wolf, since there are asymptotically \( 2^{nH(X|Y)} \) jointly typical sequences generated by the X-source, then a rate of \( R_X > H(X|Y) \), is asymptotically adequate to ensure a very small probability of decoding error [9].

### 2.5.3 Practical Slepian-Wolf Coding

Since the fundamental Slepian-Wolf code design is based on random binning, the employed code generation is asymptotic and non-constructive. First attempts towards practical Slepian-Wolf coding target to achieve the corner points of the achievable rate region. In this so called asymmetric scenario, one source is compressed to its entropy and used as side information at the decoder in order to code the other source. In contrast to predictive coding, the aim here is to code \( X \) at a rate that approaches \( H(X|Y) \) based on the correlation statistics and independently of the specific realization of the side information at the encoder.

Functional Slepian-Wolf coding is based on algebraic binning [44] constructed
using channel coding. Two principal directions for realizing an algebraic binning approach have been reported in the literature. The first method consists of a syndrome-based scheme, rooted in the landmark paper of Wyner [45]. The second method is known as the parity-based scheme. Both the aforementioned methods exploit linear channel codes for algebraic binning, but they differ in the way codewords are distributed into the Slepian-Wolf bins.

To further describe these two methods, we define a convenient model for the correlation between the encoded sources. In particular, let the correlation between the source \( X \) and the side information \( Y \) be described by a communication channel, alias a virtual correlation channel, that is, \( X = Y + N \), where the i.i.d. random variable \( N \) signifies the correlation channel noise.

### 2.5.3.1 The Syndrome approach

In [45], Wyner first pointed out the strong relation between random binning and channel coding, implying the use of linear channel codes as a practical solution for Slepian-Wolf coding. Wyner’s scheme partitions the source alphabet into disjoint sets, similar to the cosets of a linear channel code tailored to a particular correlation model, compressing the source into syndromes.

Specifically, assume binary sources \( X \) and \( Y \) and a correlation noise \( N \) given by \( N \sim \mathcal{B}(p_c) \). Namely, the correlation channel is the binary symmetric channel (BSC) with crossover probability \( p_c \). The pmf of the BSC channel is given by

\[
p(x|y) = \begin{cases} (1-p_c)\delta(x) + p_c\delta(x-1), & y = 0 \\ p_c\delta(x) + (1-p_c)\delta(x-1), & y = 1 \end{cases}
\]

where \( \delta(x) \) is the Dirac delta function.

To compress the binary \( n \)-tuple \( x \), the Wyner’s syndrome-based encoder employs an \((n,k)\) linear channel code \( \mathcal{C} \) constructed by the generator matrix \( G_{k \times n} = \begin{bmatrix} I_k & P_{k \times (n-k)} \end{bmatrix} \)\(^6\), where \( I \) and \( P \) represent the unit and the parity matrix, respectively. The corresponding \((n-k) \times n\) parity-check matrix of \( \mathcal{C} \) is \( H_{(n-k) \times n} = \begin{bmatrix} P_{k \times (n-k)}^T & I_{n-k} \end{bmatrix} \). The encoder forms the syndrome \((n-k)\)-tuple as \( s = xH^T \), and transmits it to the decoder. Then, the decoder applies a decoding function on the received syndrome and the side information to derive the error vector \( e \). Finally, the encoded source sequence is reconstructed as \( \hat{x} = e \oplus y \), where \( \oplus \) is the exclusive-OR (XOR) operator [45].

Notice that using this scheme, the achievable compression ratio is \((n-k)/n\). Moreover, recall that according to the Slepian-Wolf bound, \( R_X > H(X|Y) \). Therefore, the rate of the employed linear code \( \mathcal{C} \) must fulfill the following

\(^6\) To simplify the presentation, the linear code is assumed systematic.
inequality:

\[ n - k > nH(X|Y) \]

\[ \Rightarrow \frac{k}{n} > 1 - H(X|Y) , \]  

(2.35)

Surprisingly, Wyner’s methodology was only recently used by [46] for practical Slepian-Wolf code design based on conventional channel codes, like block and trellis codes. Alternatively, Liveris et al. [47] engineered a syndrome-based code design rooted in the state-of-the-art low-density parity-check (LDPC) codes, that achieves a compression performance very close to the Slepian-Wolf limit. Furthermore, Varodayan et al. [48] developed a rate-adaptive LDPC syndrome-based code that achieves various puncturing rates, while maintaining very good performance.

A short introduction to linear block codes, and especially to LDPC codes, is given in Appendix A. Furthermore, due to their vital part in this dissertation, the rate-adaptive LDPC codes of Varodayan et al. [48] are also described in Appendix A.

2.5.3.2 The Parity approach

In the parity-driven coding approach, parity-check bits, rather than syndrome bits, of a systematic channel code are employed to index the Slepian-Wolf bins. Specifically, in order to encode the binary n-tuple \( x \), a parity-based Slepian-Wolf encoder deploys an \( (n+r, n) \) systematic channel code \( C'' \) defined by the generator matrix \( G'_{n \times (n+r)} \). The encoder develops a parity bits \( r \)-tuple \( p = xP' \), which constitutes the compressed information, and sends it to the decoder. Thereafter, the decoder produces an \( (n+r) \)-tuple \( g = [y_{1:n}\ p] \) by attaching the side information \( n \)-tuple \( y_{1:n} \) to the received parity bits \( r \)-tuple \( p \). By decoding \( g \) on the channel code \( C'' \), the designed parity-based Slepian-Wolf decoder yields the decoded codeword \( \hat{c} = \hat{x}G'_{n \times (n+r)} \). The systematic part of the latter is extracted, and constitutes the reconstruction \( \hat{x} \) of the encoded source array [49].

Practical Slepian-Wolf codes following the parity-based approach have been engineered anchored in state-of-the-art capacity approaching binary linear codes, such as Turbo [12, 50, 51] and LDPC [49] codes. These code designs have shown great performance, namely, very close to the Slepian-Wolf bound. Moreover, as it will be explained below, these designs offer inherited robustness against transmission errors, and allow for a range of rates based on modified puncturing patterns.

Comparing the syndrome- with the parity-based approach, one observes that in case \( H^T = P' \), that is, if the above-explained parity-check matrix of \( C' \) is equal to the parity matrix of \( C'' \), then the compression performance of the two approaches is
exactly the same [49].

However, notice that, in order to compress an \( n \)-length source to a \( k \)-tuple Slepian-Wolf code, the syndrome-based approach is structured on an \( (n, k) \) linear channel code \( C' \), which has an \( n \)-length codeword. Conversely, the parity-based approach deploys an \( (n + r, n) \) linear systematic channel code \( C'' \), thereby featuring a codeword of size \( (2n - k) \) bits. This means that, although both methods are equivalent in terms of compression performance, the parity-based approach suffers additional computational complexity because of a longer codeword length [49].

Notice also that for distributed compression under a noiseless transmission scenario the syndrome-based Slepian-Wolf scheme is optimal, as it can achieve the theoretic bound with the shortest channel codeword length. Nonetheless, in order to address distributed compression in a noisy transmission scenario the parity-based Slepian-Wolf scheme is preferable, as it can derive a distributed joint-source channel coding (DJSCC) scheme \(^7\) [52]. Specifically, let the capacity of the communication channel be given by \( C \leq 1 \), and assume that the transmitted parity bits of the parity-based Slepian-Wolf code are increased to \( R_x > H(X | Y) / C \). According to the DJSCC theory, the added parity bits redundancy can be employed to protect against the errors occurred in the communication medium.

One notices that the abovementioned practical Slepian-Wolf coding schemes target to achieve the corner points of the achievable rate region, that is, points A or B in Figure 2.6. In this so-called asymmetric scenario, one source is compressed to its entropy and used as side information at the decoder in order to code the other source. For the symmetric case, i.e., comprising the intermediate points between A and B, e.g., point C in Figure 2.6, constructive code designs based on time sharing or channel code partition techniques have been proposed in the relevant literature, see for instance [50, 53-55] and [56].

In general, it can be remarked that in case the joint statistics between the correlated sources can be modeled using a virtual correlation channel, then a capacity approaching channel code for this channel can be interpreted as a random coset code approaching the Slepian-Wolf bound.

\(^7\) The syndrome-based Slepian–Wolf bits can only compress but not protect against communication channel errors. Therefore, the syndrome-based approach requires separate source and channel coding designs. According to Shannon’ separation theorem [26] separate designs are asymptotically optimal. In practice, however, good designs are employing DJSCC [49].
2.6 Rate-Distortion with Side Information

The alluring compression scenario considered by Slepian and Wolf, stimulated Wyner and Ziv to study a particular asymmetric Slepian-Wolf coding case, where $X$ is coded in a lossy manner, given that a correlated source $Y$, referred to as the side information, is available only at the decoder [10]. In the literature, this case is known as Wyner-Ziv coding or lossy compression with decoder side information.

![Virtual Correlation Channel](image)

**Figure 2.9:** Lossy compression with side information at the decoder (the Wyner-Ziv problem).

### 2.6.1 The Wyner-Ziv Coding Theorem

Figure 2.9 shows the general arrangement of Wyner-Ziv coding. Let $X$ and $Y$ be two statistically dependent i.i.d. random sequences, where $X$ is independently encoded and jointly decoded, using $Y$ as side information, to form a reconstructed sequence $\hat{X}$, yielding an expected distortion $D = E\left[d\left(X, \hat{X}\right)\right]$. The formal definition of the Wyner-Ziv coding problem is as follows [10]:

For a specific distortion metric $d(X^n, \hat{X}^n)$, i.e., $d: \mathcal{A}_n \times \mathcal{A}_n^* \rightarrow \mathbb{R}^+$, a Wyner-Ziv code $(R_{XW}^Z, D)$ is defined by [10] an encoding function

$$f_n^{WZ}: \mathcal{A}_n \rightarrow \left\{1, 2, \ldots, 2^{nR_{XW}^Z}\right\},$$

and a decoding with side information function

$$g_n^{WZ}: \mathcal{A}_n \times \left\{1, 2, \ldots, 2^{nR_{XW}^Z}\right\} \rightarrow \mathcal{A}_n,$$

where $\mathcal{A}_n$ is the alphabet of the random variable corresponding to the side information. The distortion corresponding Wyner-Ziv $(R_{XW}^Z, D)$ code is given by

$$D = E\left[d\left(X^n, g_n^{WZ}\left(Y^n, f_n^{WZ}\left(X^n\right)\right)\right)\right],$$

where $\hat{X}^n = g_n^{WZ}\left(Y^n, f_n^{WZ}\left(X^n\right)\right)$ is the reconstructed sequence.

Derived by the Wyner-Ziv theorem [10], the rate-distortion function with decoder
side information is given by

$$R_{X|Y}^{WZ} (D) = \inf_{f(U|X)} \{I(X;U) - I(Y;U)\},$$

(2.39)

where the infimum is taken over all reconstruction functions $\varphi: \mathcal{A}_U \times \mathcal{A}_Y \rightarrow \mathcal{A}_X$ and conditional probability density functions (pdfs) $f(u|x)$ such that,

$$\prod f(x,y)f(u|x)d(x,\varphi(y,u))dxdu \leq D.$$  

(2.40)

Note that $U$ is an auxiliary random variable satisfying the following Markov chains [10]:

$$U \leftrightarrow X \leftrightarrow Y,$$

(2.41)

$$X \leftrightarrow (U,Y) \leftrightarrow \hat{X}.$$  

(2.42)

The first Markov chain, that is, Eq. (2.41) above, indicates that, in Wyner-Ziv coding, the selection of the auxiliary $U$ codebook is independent of the side information $Y$. Furthermore, the second Markov chain, given by Eq. (2.42), designates that the reconstruction function $\varphi(y,u)$ in Eq. (2.39) is independent of the source signal $X$.

Recall that in case the side information is also available to the encoder, the predictive coding rate-distortion function is given by

$$R_{X|Y} (D) = \inf_{f(x,y)} \{I(X;\hat{X}) - I(Y;\hat{X})\},$$

(2.43)

where the minimization is over all conditional pdfs $f(\hat{x}|x,y)$ for which the joint pdf satisfies the distortion constraint, i.e.,

$$\prod f(x,y)f(\hat{x}|x,y)d(x,\hat{x})dxdyd\hat{x} \leq D.$$  

(2.44)

According to the theoretic proof derived by Wyner and Ziv [10], a rate loss compared to traditional predictive coding is sustained when the encoder does not have access to the side information, namely,

$$R_{X|Y}^{WZ} (D) - R_{X|Y} (D) \geq 0.$$  

(2.45)

However, Wyner and Ziv [10] further demonstrated that equality in Eq. (2.45) holds for the quadratic Gaussian case, i.e., the case where $X$ and $Y$ are jointly Gaussian and a mean-square distortion metric $d(x,\hat{x})$ is used. Later, Pradhan et al. [57] generalized the Wyner-Ziv equality to include sources defined by the sum of arbitrarily distributed side information $Y$ and independent Gaussian noise $N$, i.e. $X = Y + N$. What is more, assuming generic source statistics, Zamir [58] proved that the rate loss, due to the exploitation of the side information at the decoder side only, is upper bounded by 0.5 bits per sample, namely,
2.6.2 Achievability of Wyner-Ziv Coding

To prove the achievability of their theorem, Wyner and Ziv constructed the subsequent asymptotic code [10, 26]: They set the pdf \( f_{UX}(u|x) \) and the reconstruction function \( \phi(y,u) \). Based on \( f_U(u)=f_{UX}(u|x)f_X(x) \) they generated \( U^n(c) \), \( c \in \{1,2,\ldots,2^{nR_1}\} \), source codewords, where \( R_1=I(X;U)+\delta \). Furthermore, they randomly distributed the generated \( U^n(c) \) source codewords into \( 2^{nR_2} \) bins, denoted by \( \text{Bin}(s) \), \( s \in \{1,2,\ldots,2^{nR_2}\} \), where they considered \( R_2=I(X;U)-I(Y;U)+5\delta \). Observe that asymptotically there will be \( 2^{n(R_2-R_1)} = 2^{n[I(Y;U)-4\delta]} \) source codewords \( U^n(c) \) in every bin \( \text{Bin}(s) \). This generated Wyner-Ziv code is thereafter revealed to the encoder and the decoder.

During encoding, the encoder receives a source sequence \( X^n \) and looks for a \( U^n(c) \) source codeword, which is jointly typical with the received sequence, i.e., \( (X^n,U^n(c)) \in \mathcal{A}_6^{(n)} \). If the encoder finds one typical source codeword then it corresponds the sequence to it. Otherwise, if there are more than one typical source codewords, then the encoder selects the smallest one. If there is none such typical codeword then the encoder selects \( U^n(1) \). Next, the encoder finds the index \( s \in \{1,2,\ldots,2^{nR_2}\} \) of the bin \( \text{Bin}(s) \) in which the selected codeword belongs to and transmits it to the decoder.

During decoding, the decoder receives the transmitted index \( s \) and decodes the encoded source codeword \( U^n(c) \) using joint typicality. Namely, the decoder searches in the bin \( \text{Bin}(s) \) indexed by the received index \( s \), in order to find a \( U^n(c) \) codeword, which is jointly typical with the available side information sequence \( Y^n \), i.e., \( (Y^n,U^n(c)) \in \mathcal{A}_6^{(n)} \). If the decoder finds one and only one jointly typical codeword in the bin, then it reconstructs the source sequence as \( \hat{X}^n = \phi(Y^n,U^n(c)) \). Conversely, if the decoder cannot find a typical codeword, or else, if it finds more than one, then it decodes the source as an arbitrary reconstruction sequence \( \hat{X}^n \).

Next, we analyze the probability of error of the aforementioned code; the following events are possible [10, 26]:

1. The encoder source sequence \( X^n \) is not jointly typical with the side information sequence \( Y^n \), which is available at the decoder, that is, \( (X^n,Y^n) \notin \mathcal{A}_6^{(n)} \). However, based on the AEP (see Section 2.2.1), this probability is asymptotically minor.
2. Although we have \( (X^n,Y^n) \in \mathcal{A}_6^{(n)} \), the selected source codeword \( U^n(c) \) is not distortion typical with the encoded sequence \( X^n \), i.e.,

\[
0 \leq R_{WX}^{WZ}(D) - R_X^Y(D) \leq \frac{1}{2}.
\]
According to the rate-distortion theorem (see Section 2.3.3) this probability is asymptotically negligible due to the fact that $R_i > I(X;U)$. 

3. We have $(X^n, Y^n) \in \mathcal{A}_6^{(n)}$ and $(X^n, U^n(c)) \in \mathcal{A}_6^{(n)}$, but $(Y^n, U^n(c)) \in \mathcal{A}_6^{(n)}$, namely the source codeword is not typical with the side information sequence. This probability is also asymptotically negligible based on the Markov chain in Eq. (2.41). For further information, the interested reader is referred to [26].

4. All the previous cases are satisfied, but there is another source codeword $U^n(c') \neq U^n(c)$, inside the selected bin $Bin(s)$, which is jointly typical with the side information sequence $Y^n$. Yet, based on the AEP, the total number of randomly selected source codewords that are jointly typical with $Y^n$ is approximately $2^{n[I(U;Y) - 3\delta]}$. Recall that there are $2^{n(R_1 - R_2)} = 2^{n[I(Y;U) - 4\delta]}$ source codewords $U^n(c)$ in every bin $Bin(s)$. As a consequence the probability of having more than one jointly typical codeword in the received bin is bounded by

$$P_{error} \leq 2^{n(R_1 - R_2) + n[I(U;Y) - 3\delta]},$$

which asymptotically goes to zero.

If $(X^n, Y^n) \in \mathcal{A}_6^{(n)}$, $(X^n, U^n(c)) \in \mathcal{A}_6^{(n)}$ and the source codeword is correctly decoded then it can be deduced that $(X^n, X^n, U^n(c)) \in \mathcal{A}_6^{(n)}$, that is, the empirical joint pdf $f_{UXX}(u,x,y)$ is close to the assumed distribution which achieves the target distortion constraint $D$. Hence, the aforementioned analysis shows that, with high probability, the decoder will reconstruct the source sequence $X^n$ to a reconstruction sequence $\hat{X}^n$, such that the distortion between $X^n$ and $\hat{X}^n$ is close to $D$.

Based on the above proof, one can point out the difference between Wyner-Ziv coding and traditional rate-distortion theory. Specifically, instead of transmitting the index of the source codeword that is jointly typical with the source sequence, Wyner-Ziv coding randomly distributes the source codewords into Slepian-Wolf bins and then sends the bin’s index. In this scheme, if the conditions for the achievability of Wyner-Ziv coding are met, then, at the decoder, the side information enables a correct identification of the encoded codeword in the bin.

Using an illustrative geometric parallelism, depicted in Figure 2.10, Wyner-Ziv coding can be seen as sphere covering, for source coding, followed by sphere packing, for Slepian-Wolf coding. In Figure 2.10, the outer sphere represents the space occupied by all source sequences, while the small inner spheres signify the space occupied by the codewords of the employed source code. Analogously, the large inner sphere corresponds to the Slepian-Wolf bins, which randomly group the
source codeword spheres. Hence, achieving efficient Wyner-Ziv coding requires a two-fold approach. First, the space of the total source sequences needs to effectively be covered by the minimum possible number of codeword spheres, which means that a strong source code is required. Second, the maximum possible number of codeword spheres need to be packed into the Slepian-Wolf bins, thereby diminishing the total compression rate. The latter means that an efficient Slepian-Wolf code is also required to perform effective Wyner-Ziv coding. Lastly, as explained in Section 2.5.3, this geometric example also highlights the link between Slepian-Wolf and channel coding, as the means to interpret a sphere packing argument.

\[ \mathcal{A}_n \]

**Figure 2.10: Geometric interpretation of the Wyner-Ziv coding theorem.**

### 2.6.3 Practical Wyner-Ziv Coding

In essence, practical Wyner-Ziv coding combines quantization followed by Slepian-Wolf coding of the quantization indices. In Wyner-Ziv coding, source coding is needed to quantize \( X \) to the \( U \) codebook. Recall that, in Wyner-Ziv coding, quantization is performed independently of the realization of the side information. Subsequently, Slepian-Wolf coding is employed to exploit the remaining correlation in the quantized version of \( X \) and the side information \( Y \), thereby reducing the required rate for compression.

Consequently, Wyner-Ziv coding is a joint source-channel coding problem. To operate closely to the Wyner-Ziv bound, one needs to employ both source codes, e.g., trellis coded quantization (TCQ), that minimize the source coding loss and sophisticated channel codes, e.g., Turbo and LDPC codes, that can approach the Slepian-Wolf limit. Except for channel decoding, the side information is also used at
the decoder to perform source reconstruction. In this way, the side information diminishes the imposed distortion of the reconstructed source \( \hat{X} \).

In more detail, initial practical Wyner-Ziv code designs, focused on finding good nested codes among lattice and trellis-based codes for the quadratic Gaussian case. In [44] Zamir et al. introduced nested lattice codes, demonstrating their optimality for (prohibitively) large dimensions. Motivated by the latter theoretical scheme, Servetto [59] proposed specific nested lattice constructions, based on similar sub-lattices for the high correlation case. Recent results have shown that trellis-based nested codes can realize high-dimensional nested lattice codes. In distributed source coding using syndromes (DISCUS) [46], scalar quantization or TCQ, for source coding combined with scalar coset codes or trellis-based coset codes, for channel coding, were employed to realize Wyner-Ziv codes.

However, as dimensionality increases, lattice source codes approach much faster the source coding limit than lattice channel codes approach the capacity. Hence, the need for outstanding channel codes, i.e., codes with higher dimensionality compared to source codes, is highlighted in order to approach the Wyner-Ziv bound. This observation has induced the second wave of Wyner-Ziv code design which is based on nested lattice codes followed by binning, entitled Slepian-Wolf coded nested quantization (SWC-NQ) [60]. Under high rate assumptions, asymptotic performance bounds of SWC-NQ were derived in [60], where it was shown that ideal Slepian-Wolf coded one-/two-dimensional (1-D/2-D) nested lattice quantization performs 1.53/1.36dB worse than the Wyner-Ziv bound function (with probability almost one).

The third practical approach to Wyner-Ziv coding considers non-nested quantization followed by efficient binning, realized by a high-dimensional channel code. In this approach, the performance loss with respect to the Wyner-Ziv bound is only due to quantization without knowledge of side information at the encoder. Considering ideal Slepian-Wolf coding and applying high rate assumptions, [12, 61], have shown that scalar quantization coupled with ideal Slepian-Wolf coding leads a 1.53dB gap from the Wyner-Ziv bound. Notice that the aforementioned gap is the same as the one incurred by entropy-constrained scalar quantization in the non-distributed case [61] (see also Section 3.2.5 in [38]). Other constructions in the literature propose turbo-trellis Wyner-Ziv codes [15], in which TCQ is concatenated with a turbo channel code. In [16] Yang et al. have shown that at high rates, TCQ with ideal Slepian-Wolf coding performs 0.2dB away from the Wyner-Ziv limit (with probability almost one).
2.7 Successively Refined Wyner-Ziv Coding

Instead of applying Wyner-Ziv coding in a straightforward manner, successively refined Wyner–Ziv coding encodes a source in multiple refinement stages, with different side information and distortion conditions at each stage.

2.7.1 Definitions and Conditions for Achievability

Explicitly, consider the coding scheme illustrated in Figure 2.11, in which a source $X$ is successively coded by a coarse and a refinement stage. Also, let $R_1$, $\Delta R = R_2 - R_1$, $D_1$, $D_2$, and $Y_1$, $Y_2$ be the rates, the distortion levels, and the side information of the coarse and the refinement stages, respectively.

Figure 2.11: The setting of successive refinement with side information at the decoder.

Under the condition that the side information at the different stages forms a Markov chain, that is, $X \leftrightarrow Y_2 \leftrightarrow Y_1$, Steinberg and Merhav [41] managed to characterize the set of all achievable quadruples, i.e., $(R_1, \Delta R = R_2 - R_1, D_1, D_2)$. According to their findings, when $D_2 \geq D_1$, the successively refined Wyner-Ziv rate-distortion function $R_{X|Y_1,Y_2}^{SRWZ}(D_1,D_2)$ is bounded by

$$R_{X|Y_2}^{WZ}(D_2) \leq R_{X|Y_1,Y_2}^{SRWZ}(D_2) \leq R_{X|Y_1}^{WZ}(D_2),$$

(2.48)

where $R_{X|Y_1}^{WZ}(D_2)$, $R_{X|Y_2}^{WZ}(D_2)$ are the rate-distortion functions of non-progressive Wyner–Ziv coding with side information $Y_1$ and $Y_2$, respectively.

When the side information is identical for all the refinement stages, Steinberg and Merhav [41] proved that Wyner-Ziv scenarios involving doubly symmetric binary or quadratic Gaussian sources are successively refinable. This means that, in the aforementioned scenarios, equality in Eq. (2.48) can be achieved, namely, $R_{X|Y_1}^{WZ}(D_2) = R_{X|Y_1,Y_2}^{SRWZ}(D_1,D_2)$. Moreover, Cheng and Xiong [17] broadened the Wyner-Ziv successive refinability property to sources $X = Y + N$ defined by the sum of independent Gaussian noise $N$ and arbitrarily distributed side information $Y$. 

40
2.7.2 Layered Wyner-Ziv Coding

Practical layered (scalable) Wyner-Ziv code design, with identical side information at all levels, was presented by Cheng and Xiong in [17]. The schema of a layer Wyner-Ziv coding system, for generic sources, is depicted in Figure 2.12. The encoder performs quantization of the incoming source information \( n \)-tuple \( \mathbf{x}^n = x_1, x_2, \ldots, x_n \)

\(^8\)

and then the quantized symbols \( \mathbf{q}^n = q_1, q_2, \ldots, q_n \) are divided into a number of \( L \) bit-planes, namely, \( \mathbf{b}_1^n, \mathbf{b}_2^n, \ldots, \mathbf{b}_L^n \), where \( \mathbf{b}_i^n = b_{1i}, b_{2i}, \ldots, b_{ni} \) and \( b_{ji} = \{0,1\} \). Using a specific order, the source bit-planes are separately passed to a syndrome- or parity-based Slepian-Wolf encoder, which performs structured binning using channel codes. Bear in mind that, in order to operate close to the Wyner-Ziv bound, capacity achieving channel codes, like Turbo and LDPC, are used to realize Slepian-Wolf coding. Particularly, in the approach of [17] LDPC codes are deployed as the latter have shown higher performance compared to Turbo codes – see for example [47] for further details.

At the decoder, for every source bit-plane, the correlation statistics between the source and the side information are interpreted to soft estimates, i.e., log-likelihood ratios (LLRs). These LLRs, which provide \textit{a priori} information about the probability of each bit to be 0 or 1, are passed to the soft channel decoder (e.g., to the variable nodes of the LDPC decoder, in case an LDPC-based Slepian-Wolf code is considered). Then, an iterative soft decoding algorithm is executed to decode the source bit-planes. More specifics on the message passing algorithm [62-64], for iterative LDPC decoding, are given in Appendix A.

For optimal layered coding, during the formulation of the LLRs, information given by the side information and the already decoded source bit-planes is taken into

\(^8\) From here onwards, tuples are denoted using vector representations in order to accommodate more complicated notations without loss of comprehension.
account. Specifically, let the $b^l_i$ be a bit of the $l^{th}$ bit-plane of the source and $b^1_i, \ldots, b^{l-1}_i$ be the already decoded bits in the previous $l-1$ bit-planes. Then the estimated LLR is given by

$$\text{LLR} = \log \frac{P(b^l_i = 0 | y_i, b^1_i, \ldots, b^{l-1}_i)}{P(b^l_i = 1 | y_i, b^1_i, \ldots, b^{l-1}_i)} = \log \frac{p(b^1_i, \ldots, b^{l-1}_i, b^l_i = 0 | y_i)}{p(b^1_i, \ldots, b^{l-1}_i, b^l_i = 1 | y_i)}, \quad (2.49)$$

where the equality in Eq. (2.49) is derived by the definition of conditional pdfs, i.e.,

$$p(b^l_i | y_i, b^1_i, \ldots, b^{l-1}_i) = \frac{p(b^1_i, \ldots, b^{l-1}_i, b^l_i | y_i)}{p(b^1_i, \ldots, b^{l-1}_i | y_i)}.$$  

Hence, in Eq. (2.49) the nominator and the denominator are calculated by integrating the conditional pdf of the correlation channel, i.e., $f_{X|Y}(x|y)$, over the quantization bin indexed by $b^1_i, \ldots, b^{l-1}_i, b^l_i$. We highlight that because of the previously explained chained approach to construct the LLRs, the total layered Wyner-Ziv rate conforms to the chain rule of entropies [26, 27], that is,

$$H(Q_L(X)|Y) = H(b^n_1, b^n_2, \ldots, b^n_L | Y) = \sum_{i=1}^{L} H(b^n_i | Y, b^n_1, \ldots, b^n_{i-1}), \quad (2.50)$$

where $Q_L(X)$ represents quantization of the source with $2^L$ levels. This means that, ideally, there is no performance loss between non-layered and layered Wyner-Ziv coding.

Once all the $L$ source bit-planes are decoded, they are combined to form the decoded quantization indices of the source. These indices are used to reconstruct (a.k.a., inverse quantize) the source with the help of the side information and the correlation statistics. In particular, if the MSE distortion metric is employed, the optimal reconstruction of a source sample $x_i$ is obtained as the centroid of the random variable $X$ given the corresponding side information sample $y_i$ and the decoded quantization index $q_l$ [17]. Namely,

$$E[x_i | y_i, q_l] = \frac{\int_{q_l}^{q_H} x_i f_{X|Y}(x_i | y_i) dx_i}{\int_{q_l}^{q_H} f_{X|Y}(x_i | y_i) dx_i}, \quad (2.51)$$

where $q_L, q_H$ denotes the lower and upper bounds of the quantization bin $q_l$.

For a jointly Gaussian source, the layered code of [17], has been shown to perform 1.29 to 3.45dB away from the Wyner-Ziv bound for a wide range of rates. Furthermore, when the side information $Y$ is Laplacian and the source is given by $X = Y + N$ where $N$ is a side-information-independent Gaussian noise component, the layered code of [17] obtains rate-distortion performance 1.33 to 3.90dB away from the Wyner–Ziv bound for a wide range of rates.
Chapter 3
CONVENTIONAL VS. DISTRIBUTED VIDEO CODING

3.1 INTRODUCTION

The fundamental information theoretic findings of Slepian and Wolf [9] and Wyner and Ziv [10] provide bounds for the compression performance of DSC systems. One of the applications of Wyner-Ziv coding that has received a substantial amount of research attention is distributed video coding (DVC), also known as Wyner-Ziv video coding.

Traditional video coding technology is developed following the principles of conventional source compression and rate-distortion theory. In particular, traditional video coding architectures are engineered based on a hybrid video coding paradigm which employs complex motion prediction, rate-distortion optimized mode decision and rate control, block-based transform and entropy coding at the encoder. This conventional paradigm successfully exploits the source redundancies at the encoder, leading to complex video encoders versus light decoders. As mentioned in Chapter 1, this complexity distribution of conventional video codecs is primarily driven by a down-link transmission model, in which a video signal is encoded by a powerful encoder and decoded by several light decoders.

However, the Slepian-Wolf and Wyner-Ziv theorems suggest that efficient coding systems, in which the inter-source correlation is exploited at the decoder instead of at the encoder, are practicable. Hence, as explained in Chapter 1, compared to conventional video coding architectures, DVC provides low-complexity encoding systems. Furthermore, since it is based on channel coding principles [12, 15, 16], DVC inherently offers error resilience, diminishing the effects of error propagation and drift [12]. Moreover, layered Wyner-Ziv coding [17] offers scalable video coding.

This chapter provides an inclusive summary of the most significant achievements
in both conventional and distributed video coding. This chapter also mentions the contributions in the DVC literature made by the Interdisciplinary Institute for BroadBand Technology (IBBT) DVC research group (of which I am part).

In the beginning, a brief outline of conventional video coding standards and architectures is given in Section 3.2. In the next part of this chapter, that is, in Section 3.3, emphasis is put on distributed video compression. The two pioneering approaches in the DVC research field, namely, the DVC architectures developed at Berkeley and Stanford University, are described in Section 3.3.1 and Section 3.3.2, respectively. A key extension of the latter system has been developed in the context of the DISCOVER project. The ensuing coding system, which is a state-of-the-art reference in the DVC literature, is detailed in Section 3.3.3. Finally, Section 3.3.4 refers to the developments introduced by the IBBT DVC research group and highlights the main contributions of this dissertation in improving over the DISCOVER system.

### 3.2 Conventional Video Coding

The most fruitful video coding schemes proposed in the past decades [3-7], are the result of standardization efforts by the Telecommunication Standardization Sector (ITU-T) of the International Telecommunication Union (ITU), or by the International Organization for Standardization (ISO) in tandem with the International Electrotechnical Commission (IEC).

Specimens of video coding standards include H.261 [5] and its descendant H.263 [6], which have been extensively employed in video teleconferencing applications. Furthermore, MPEG-2 [3] has been used for digital broadcast and as a storage format on Digital Versatile Discs (DVD). Subsequent standardization efforts have led to the development of the MPEG-4 Visual [4], and the H.264/Advanced Video Coding (AVC) [7] standard. Although H.263 is still required by the 3rd Generation Partnership Project (3GPP) Release 9 Packet Switched Streaming Service as mandatory codec, H.264/AVC is considered today's codec for mobile video. As regards 3GPP, H.264/AVC is optional, but practically all new smart phones support both encoding and decoding of H.264/AVC baseline profile.

Though H.264/AVC delivers state-of-the-art compression performance for single layer video coding, it does not support scalability, an essential feature for on-the-fly bit-rate adaptation. Scalable video techniques are defined by the Scalable Video Coding (SVC) extension of the H.264/AVC video compression standard, defined in Annex G of the standard. H.264/SVC [40, 65] regulates the encoding of a high-quality video bit-stream that also contains one or more subset bit-streams. A subset
bit-stream is derived by dropping packets from the high-quality video bit-stream, thereby reducing the required bit-rate, decoding complexity and reconstruction quality. A subset bit-stream can represent lower spatial resolution (corresponding to resolution scalability), lower temporal resolution (a.k.a., temporal scalability), or lower quality (i.e., quality scalability) video.

Moreover, the H.264 Multiview Video Coding (MVC) amendment of the H.264/AVC video compression standard enables stereo 3-D coding, which is backward compatible to single-view H.264/AVC. H.264/MVC [66] exploits the similarities between two or more views of the same scene, enabling both temporal and inter-view prediction.

Presently under joint development by the ISO/IEC Moving Picture Experts Group (MPEG) and by the ITU-T Video Coding Experts Group (VCEG), High Efficiency Video Coding (HEVC) is an emerging video compression standard, which promises to become the replacement of the H.264/AVC standard. HEVC aims to essentially improve compression performance with respect to the high profile of H.264/AVC, at the expense of a moderate increase in encoding computational complexity. HEVC mainly targets next-generation HDTV displays and video content with frame rates and display resolutions ranging from QVGA up to 1080p and Ultra HDTV.

The aforementioned video compression schemes share similar architectural features, combining motion-compensated prediction with a block-based discrete cosine transform (DCT), scalar quantization and Huffman or arithmetic entropy coding. In the following, we concisely describe these main functional components of contemporary video coding systems. First, in Section 3.2.1, an overview of the traditional hybrid video coding architecture employed by all current ITU-T and ISO/IEC video coding standards is given. Then, in Section 3.2.2, a brief review of the key features of the H.264/AVC standard is provided.

### 3.2.1 Motion-Compensated Predictive Video Coding Architecture

The general setup of a motion-compensated predictive video encoder is depicted in Figure 3.1. In this architecture, each coded frame is represented in block-shaped units of luma and associated chroma samples called macroblocks (MBs). The encoding of each frame proceeds on a macroblock-by-macroblock basis.

To encode a video sequence, the incoming video frames are divided in groups of pictures (GOPs), e.g., groups of 16 frames. The MBs in the first frame of each GOP are all intra-coded, which means that they are coded without any reference to other frames in the sequence. When encoding these MBs, switches $S_1$, $S_2$ and $S_3$ in Figure 3.1 are left open. The MBs are divided into blocks of pixels, which are thereafter independently transformed using a block-based transform, i.e., typically
the DCT. The transform coefficients are subsequently quantized by dividing each coefficient by a quantization step $\Delta$ and rounding the result down to an integer value. The quantization step size is often frequency dependent, namely, the higher-frequency information is more coarsely represented. In the last step, the quantized coefficients are properly binarized and entropy coded.

Starting from the quantized coefficients, each MB is reconstructed by performing the inverse quantization and transform steps. The resulting MBs form the reconstructed frame, which is stored in the frame buffer for use as a reference in motion-compensated prediction of subsequent frame MBs. In this way, the reference frame used for motion compensation at the encoder-side is exactly the same as the one used at the decoder-side, thereby preventing discrepancies between the information used in the prediction by the encoder and the decoder, i.e., the so-called drift. Because the reconstructed frame is used as a reference for future motion-compensated prediction, such an encoder is referred to as a closed-loop video encoder.

The MBs in the remaining frames in the GOP are inter-coded. In this case, switches $S_1$, $S_2$ and $S_3$ in Figure 3.1 are closed. At first, motion estimation (ME) is performed according to a block-based motion model. The process finds a corresponding block in a reference frame that closely matches each motion estimated block. The reference frame is a previously encoded frame from the sequence and may be before or after the current frame in display order. Typically, only equal size blocks in a limited area (search area) in the reference frame centered on the current block position are considered, and the block within the search area

![Figure 3.1: The classical hybrid video coding architecture.](image)
that minimizes a matching criterion is chosen as the *best matching block*. The resulting motion vector is encoded using motion vector prediction from the motion vector(s) of previously encoded block(s) in a causal neighborhood, followed by entropy coding of the prediction errors. The MB’s motion-compensated prediction, generated by *motion compensation* (MC) from the reference frame using the derived motion vector, is subsequently subtracted from the original MB. This results in a prediction-error, which is coded in a similar way as an intra-coded MB. Starting from the quantized transform coefficients generated by this process, the MB’s prediction error is reconstructed and added to the MB’s motion-compensated prediction. The resulting decoded MB is stored in the frame buffer as a part of the next reference frame.

There are many variations on the basic coding processes described above, depending on the video coding standard. For the particular specifications of each video coding standard the reader is referred to the standards [3-7].

### 3.2.2 Video Coding with H.264/AVC

In H.264/AVC [7, 8], for each picture of a video sequence, each luma component is divided in blocks of 16×16 pixels and, in case of video in 4:2:0 sampling format, the two chroma components are each divided in blocks of 8×8 samples. The MBs of a picture can be organized in *slices*, each of which can be processed independently of other slices in a picture. The H.264/AVC Main profile supports three slice coding types, namely, the I-slice, the P-slice and the B-slice.

For *I-slices*, all MBs are encoded using intra-frame predictive coding with several directional spatial intra prediction modes. For the luma component, intra prediction is either applied to 4×4, 8×8 or 16×16 blocks, while for the chroma components it is applied on a MB basis. Transform coding in H.264/AVC is performed using a 4×4 or 8×8 sample integer transform [67], which is an integer approximation of the DCT. A detailed description of this transform and the supported directional intra-prediction modes can be found in [33].

Slices that use motion-compensated temporal prediction are categorized as *P*- and *B-slices*. Essentially, H.264/AVC enables motion estimation and compensation using any previously encoded frame as a reference. Two reference frame lists, denoted as *list 0* and *list 1*, are used to store the set of available reference frames. Updates of these lists are signaled in the bit-stream. Moreover, H.264/AVC takes into account that objects may move by a fractional number of pixels between frames and enables up to quarter-pel motion estimation accuracy. Every motion vector resulting from ME has an associated reference frame index, which indicates the position of the employed reference frame in these lists. Moreover, as moving objects
in a video scene rarely follow precise $16 \times 16$-pixel boundaries, it may be more efficient to use a **variable block size** for motion estimation and compensation. This is supported in H.264/AVC for P- and B-slices, where each MB can be partitioned in blocks and sub-blocks as illustrated in Figure 3.2.

![Figure 3.2: Partitioning of MBs and sub-MBs for motion-compensated prediction in the H.264/AVC standard.](image)

The MBs of a *P-slice* can be coded using (i) intra-frame predictive coding, (ii) inter-frame predictive coding with one reference frame chosen from *list 0* or (iii) a skipped mode. For P-slices, each block (and sub-block) can use a different reference frame chosen from *list 0*. The MBs of a *B-slice* can be encoded using (i) intra-frame predictive coding, (iii) predictive coding from one reference frame chosen from *list 0*, (iii) predictive coding from one reference frame chosen from *list 1*, (iv) bi-directional predictive coding from two reference frames, one from *list 0* and the other from *list 1*, (v) a bi-directionally direct prediction mode or (vi) a skipped mode. For the special direct prediction and skipped modes in P- and B-slices, data such as motion vectors and reference indexes are simply derived from previously transmitted information.

Furthermore, H.264/AVC uses uniform reconstruction quantizers. Fifty two (52) quantization step sizes can be selected for each MB by the *quantization parameter (QP)*. The scaling operations for the quantization step sizes are arranged with logarithmic increments, such that an increment of QP by 6 corresponds to approximately doubling the quantization step size [8, 33].

Also, in order to reduce blocking artifacts, which are typical in block-based coding, an *adaptive deblocking filter* is specified [8, 33, 68]. This filter operates within the motion-compensated prediction loop. By suppressing the blocking artifacts, the subjective quality at the decoder side is improved and a better reference for motion-compensated prediction is generated.

Notice that H.264/AVC supports two entropy coding methods: (i) context-based adaptive variable-length coding (CAVLC) for residual transform data combined with exp-Golomb codes for other variable-length coded units and (ii) context-based...
adaptive arithmetic coding (CABAC). Both CAVLC and CABAC employ the concept of context-based coding, which means that the probability model used by the entropy coder depends upon the context of the encoded symbol, which is typically determined based upon neighboring information. In general, CABAC delivers 10% to 15% of bit rate savings compared to CAVLC. Details on these entropy coders can be found in [33]. For more information on the data format, supported syntax and decoding process of the H.264/AVC standard the interested reader is referred to [8, 33].

In conclusion, compared to previous video coding standards, H.264/AVC provides increased flexibility on a MB, frame and sequence level. As a matter of fact, H.264/AVC supports multi-hypothesis block-based ME with variable block sizes, multiple reference frames and up to quarter-pel accuracy. Moreover, in H.264/AVC, the coding and display orders are completely decoupled. Also, any frame can be marked as reference frame for motion-compensated prediction of following frames. All these features enable H.264/AVC to achieve better compression efficiency than prior video coding standards. Explicitly, in comparison with MPEG-2 [3], H.264/AVC typically provides an improvement of the coding efficiency by a factor of two. Furthermore, as shown in [33], H.264/AVC outperforms MPEG-4 Visual in terms of compression performance.

### 3.3 Distributed Video Coding

The development and subsequent commercialization of wireless lightweight multimedia technology, which addresses applications like wireless visual sensor networks and wireless capsule endoscopy, has raised a major interest in low-cost encoding solutions for video. At the same time, advances in channel coding theory provided capacity-approaching channel codes, a necessary component to construct efficient DVC systems. The symbiosis of these two elements constituted a prolific environment for practical DVC solutions, attracting the interest of both industry and the academic world.

Most of the research efforts dedicated to DVC focus on the development of efficient mono-view codecs, namely, low-cost systems employed to compress a video sequence captured by a single camera. To apply the Slepian-Wolf and Wyner-Ziv coding principles in such a mono-view coding paradigm, the video sequence is split in two correlated sources of information. One source is coded using conventional intra-frame video coding techniques, and then used at the decoder to generate a side information signal, commonly denoted by $Y$. This side information signal is subsequently employed to Wyner-Ziv code the other source of information,
which is typically signified by $X$.

The PRISM architecture [11] was among the earliest practical DVC implementations (see Section 3.3.1). Almost at the same time, an alternative Wyner-Ziv video coding architecture was designed at Stanford University (see Section 3.3.2). This architecture was further improved upon in the VISNET I European Union project, resulting in the DISCOVER [24] codec (see Section 3.3.3). The DISCOVER codec delivers state-of-the-art compression performance and is a well-established reference in DVC. In the previous years, DVC has attracted the attention of the scientific community in video coding, and as a consequence, several alternatives or improvements to the initial architectures have been proposed. Particularly, IBBT’s research group on DVC – of which I am part – has profoundly contributed to the related literature with important publications. A summary of these contributions is included in Section 3.3.4, while the main contributions of this dissertation are thoroughly presented in the following chapters.

3.3.1 The PRISM Architecture

In the late 90’s, Pradhan and Ramchandran introduced DISCUS [46, 69], one of the first constructive and efficient code designs to realize Wyner-Ziv coding – see Section 2.6.3 for more details. Subsequently, from a theoretical point of view, Ishwar et al. [70] examined the performance of a Wyner-Ziv video coding scheme that would perform motion-search at the decoder. Assuming an additive Gaussian noise model, i.e., $X = Y + N$, between the source to be coded and the side information, and considering the MSE distortion metric, Ishwar et al. proved that such a Wyner-Ziv-based system could be developed with a small loss in performance with respect to a conventional predictive video coding. Furthermore, it was shown that, asymptotically, i.e., in the limit of large block of pixels sizes, the

Figure 3.3: Block diagram of the PRISM [23] encoder.
The aforementioned performance loss compared to conventional predictive video coding vanishes. The work of Ishwar et al. [70] formed the theoretical basis to create a pioneering distributed video coding framework entitled Power-efficient, Robust, High compression, Syndrome-based Multimedia coding (PRISM) [11, 23].

Figure 3.4: Bit plane view of a block of sixty-four coefficients. Bit planes are arranged in increasing order of significance, with 0 corresponding to the least significant bit [23].

The encoding process of PRISM is depicted in Figure 3.3. The system first divides each incoming video frame into non-overlapping blocks of 8×8 pixels. The samples within each block are spatially de-correlated using the DCT. The derived transformed coefficients are scanned in zig-zag order, and arranged in a 1-D array. Then, the arranged coefficients are quantized with a quantization step size chosen based on the targeted reconstruction quality. The quantization method of the H.263 standard [71] is employed. The 64 quantized coefficients are split into bit-planes, which are next divided in the following three categories (see Figure 3.4):

- The first category includes a number of most significant bits per quantized coefficient (shown in white color in Figure 3.4). These bits are decoded using the side information and the encoded Slepian-Wolf syndrome information at the decoder.
- The second category includes a number of top-most least significant bits per quantized coefficient (shown in gray color in Figure 3.4), which are encoded using a syndrome-based Slepian-Wolf code, as explained in Section 2.5.3.1. In the implementation of [23], the linear error correction code chosen to implement structured Slepian-Wolf binning was the Bose-Chaudhuri-Hocquenghem (BCH) code. The reason was that, at a small codeword length (a length of 64 was considered), a simple BCH block code works better than sophisticated channel codes, e.g., LDPC and Turbo codes.
- The third category is formed by the remaining (i.e., bottom-most) least
significant bits per quantized coefficient (shown in black color in Figure 3.4).

These bits are entropy coded with a Huffman or an arithmetic entropy codec.

Note that by using such a structured binary representation of the quantized coefficients, the bits falling in the second and the third category are actually forming the cosets based on which the coefficients’ bits falling in the first category are coded. Namely, the encoded least-significant bits per coefficient (i.e., the ones in grey and black color in Figure 3.4) constitute the Slepian-Wolf syndrome information for the most-significant bits (i.e., the ones in white color) of the coefficient – for more details we refer to [23].

In this fashion, the PRISM system defines a multi-level coding scheme to code the different bits of the quantized coefficients in the block. The pattern according to which the bits are coded is determined by a classification scheme. This scheme classifies each block in 16 different coding modes (or classes) based on its estimated correlation with the side information. In particular, classification is performed at the encoder based on the squared error difference between the block to be coded and the co-located block in the previous frame. On the one extreme, if the block to be coded is very much alike its co-located previous frame block, then the block is signaled as skip, and the decoder reconstructs it as the co-located block in the decoded previous frame. On the other extreme, if there is very little correlation, then the bit-planes of the quantized coefficients in the block are all intra coded (i.e., all the bit-planes are classified to the third category). The remaining coding classes correspond to different Wyner-Ziv (WZ) coding modes. Each such WZ mode assigns a different number of bits per coefficient in each of the aforementioned categories based on the estimated statistics of the correlation noise.

Lastly, a 16-bit cyclic redundancy check (CRC) checksum is calculated per block and communicated to the decoder. This checksum serves as a signature of the quantized codeword sequence and enables the decoder to recognize when decoding of the block was successful.

![Figure 3.5: Block diagram of the PRISM [23] decoder.](image-url)
Per WZ coded block, the PRISM decoder (see Figure 3.5) performs half-pel motion search in already decoded frames, obtaining candidate predictor blocks that can act as side information to decode the Slepian-Wolf coded information of the block [23]. Each of the generated candidate side information predictors is used to perform soft-decision syndrome decoding based on the received syndrome information. To obtain this syndrome information, firstly, the bits that were entropy coded (the black colored bits in Figure 3.4) are recovered by the entropy decoder. Then, secondly, in case there is a syndrome-based Slepian-Wolf coded bit plane (represented in gray color in Figure 3.4), then this received syndrome bits together with the entropy decoded bits specify the coset in which Slepian-Wolf decoding must be performed. Per candidate predictor block, if the decoded information matches the transmitted CRC checksum, then Slepian-Wolf decoding is declared successful and the decoder proceeds with the next WZ block. Otherwise, the next best predictor is chosen and Slepian-Wolf decoding is repeated.

Once the quantization indices of the coefficients in the block are decoded, then they are used along with the identified side information block to reconstruct the transform coefficients of the block. In [23], the mean squared estimate is employed in the reconstruction module. Finally, the reconstructed transform coefficients of the block are inverse discrete cosine transformed (IDCT) to obtain the decoded block in the pixel-domain.

In essence, PRISM was the first practical approach to prove the applicability of DSC principles in video coding. As any DSC-based video codec, the PRISM codec has inbuilt robustness to channel losses, enables flexible distribution of computational complexity between the encoder and the decoder, imposes a low end-to-end delay \(^9\), and delivers good compression performance. Specifically, the compression performance of PRISM lies in between the performance of H.263+ Intra- and Inter [71] coding – see [23] for a detailed evaluation of the PRISM system. On the other hand, PRISM performs repeated decoding and motion estimation which, depending of the number of candidate predictors and the WZ coded blocks per frame, can increase the complexity of the decoder to a prohibitive level.

\(^9\) The PRISM system does not deploy a feedback channel from the decoder to the encoder, as typically featured in other DVC systems.
Chapter 3

Figure 3.6: Block diagram of the distributed video codec of Aaron et al. \[12, 72-74\]. If the modules represented by the dotted rectangles are not included in the architecture, the resulting system operates in the pixel domain. Otherwise, the block diagram sketches the transform-domain Wyner-Ziv video coding architecture.
3.3.2 The Stanford Architecture

In contrast to the PRISM codec, which follows a block-based coding approach, an alternative DVC architecture, that implemented Wyner-Ziv coding on a frame-based approach, was developed at Stanford University by Aaron et al. [12, 72-75]. According to this innovative architecture, of which the block diagram is depicted in Figure 3.6, the frames of the incoming video sequence are grouped in GOPs. The frames in each GOP are subsequently partitioned into key and Wyner-Ziv (WZ) frames. The key frames, denoted by $I$ in Figure 3.6, are intra coded using a conventional intra-frame video codec, such as, the Motion JPEG$^{10}$, the H.263+ Intra or the H.264/AVC Intra codec. The remaining frames in each GOP, called WZ frames and denoted by $X$ in Figure 3.6, are independently encoded using Wyner-Ziv coding principles. Depending on whether these WZ frames are coded in the pixel or in the transform domain, two coding schemes can be distinguished.

Initially, a pixel-domain Wyner-Ziv (PDWZ) architecture, was introduced in [72]. The values of the pixels in each WZ frame are uniformly quantized and the quantization indices are split into bit-planes. One after the other, these bit-planes are fed to a rate compatible punctured Turbo [77] encoder which implements parity-based Slepian-Wolf encoding, as explained in Section 2.5.3.2. Given the source bit-planes, parity Slepian-Wolf bits are produced and stored in a buffer.

At the decoder, the encoded key frames are intra decoded and stored in a reference frame buffer. Next, side information for each WZ frame is generated by temporal interpolation of already decoded key and/or WZ frames in a hierarchical bidirectional prediction structure. In [72] the employed interpolation strategy was simply averaging the pixel values at the same location from the two future and past reference frames referred to average interpolation. In [73], though, a block-based motion-compensated interpolation (MCI) technique based on symmetric motion vectors (SMV) was implemented. SMV-based interpolation assumes that motion vectors between a WZ frame and the past and the future reference frame are each other's opposites. For every block in the past frame $n-1$, the block in future frame $n+1$ that minimizes some distortion metric, e.g., the sum-of-absolute differences (SAD) or the MSE, is found and its corresponding motion vector is calculated. Then, the motion vector is halved and used to compensate frame $n-1$ to frame $n$ (i.e., forward compensation), frame $n+1$ to frame $n$ (i.e., backward compensation).

---

$^{10}$ Motion JPEG is based on the JPEG coding standard [76] and includes a file format that can handle multiple JPEG images. Unlike the Motion JPEG 2000 standard [38, 39], no standard specification has been defined for Motion JPEG, and hence only proprietary solutions are available (e.g. support in Microsoft AVI files, Apple Quicktime format or the RFC 2435 spec that describes how Motion JPEG can be supported by an RTP stream).
or the average of the two forward and backward compensated frames. The motion-compensated frame \( n \) serves as side information \( Y \) of the encoded WZ frame \( X \).

After generating side information, punctured Turbo parity bits are sent to the decoder. These bits are used to decode each Slepian-Wolf coded source bit-plane, given the side information and the previously decoded bit-planes (if any) of the WZ frame. Depending on the amount of transmitted parity bits and the statistical modeling of the correlation channel, the turbo-like Slepian-Wolf decoder outputs a degree of confidence whether an encoded bit was 1 or 0, i.e., the so-called soft output. Since Turbo decoding is an iterative process, the output probabilities are fed back to the decoder input (the so-called soft-input). After the execution of a given number of iterations and if the decoder cannot decode the bit-plane up to a certain degree of trust, it requests additional parity bits from the encoder buffer via a feedback channel. The request and decode process is repeated until an acceptable probability of error is guaranteed. It is important to note that, conversely to PRISM, in this DVC scheme the rate is determined by the decoder using a feedback channel.

Upon Slepian-Wolf decoding of the encoded bit-planes per frame, the corresponding quantization indices are send to the reconstruction module. The latter performs minimum MSE (MMSE) source reconstruction, \( E[x_i|y_i, q_i] \), based on the side information value \( y_i \) and the decoded quantization index value \( q_i \). According to the employed reconstruction function, if the side information value \( y_i \) in a pixel position lies inside the quantization bin indexed by \( q_i \), then the reconstructed pixel will take a value close to the side information value. However, in case the side information value lies outside the decoded quantization bin, the reconstructed pixel value is given by the closest (to the side information) value belonging to the decoded quantization bin. One notices that the employed reconstruction function confines the reconstruction error to a maximum value determined by the considered quantization interval. Apart from bounding the maximum reconstruction error, this function also improves the visual quality of the decoded video frames as extreme errors, which incur annoying visual artifacts, are diminished.

A similar DVC scheme can be applied in the transform domain giving rise to transform domain Wyner-Ziv (TDWZ) video coding solutions. In particular, in the system of [12, 75], a block-wise DCT is applied to the WZ frames before Wyner-Ziv encoding, as shown in Figure 3.6. In general, transform domain DVC encoders exhibit increased complexity with respect to their pixel domain counterparts, but the overall compression efficiency of the system increases.

In the transform domain system of [12, 75], the WZ frames, \( X \), are divided into non-overlapping spatial blocks of size 4\( \times \)4, and each block undergoes a 4\( \times \)4 DCT. The derived DCT coefficients are grouped into 16 coefficient bands. Employing a
set of predefined quantization matrices (QMs), each coefficient band, \( \beta = \{0,1,\ldots,15\} \), is independently quantized with \( 2^{L_\beta} \) levels. In [75] the QMs 1-7 shown in Figure 3.7 were introduced. The derived quantization indices are subsequently split into bit-planes, resulting into \( L_\beta \) bit-planes per coefficient band. Similarly to the pixel domain architecture, each bit-plane of each DCT band is sequentially passed to a rate compatible punctured Turbo [77] Slepian-Wolf encoder, and the produced parity bits are stored in a bit-plane buffer. The derived Slepian-Wolf parity bits per bit-plane of each DCT band are transmitted in portions upon decoder’s request using a feedback channel.

![Figure 3.7](image-url)

*Figure 3.7: Quantization matrices employed in transform domain Wyner-Ziv video coding architectures [24, 75]. Each DCT band is quantized with \( 2^{L_\beta} \) levels given by the index in the corresponding band in each QM.*

As in the pixel-domain architecture, the transform-domain Wyner-Ziv (TDWZ) video decoder first decodes the key frames and stores them in a frame buffer for referencing. Subsequently, a motion-compensated interpolation (MCI) or extrapolation (MCE) [12] method is employed to generate a motion-compensated prediction of the WZ frame at the decoder. The created motion-compensated prediction frame is transformed using the DCT, serving as side information to the Wyner-Ziv decoder. The obtained side information is then converted to soft-input information according to a correlation channel estimation approach. Similar to the pixel domain, the obtained soft-input information is used to Slepian-Wolf decode the bit-planes of the quantized coefficients per coded DCT band\(^{11}\). After Slepian-Wolf decoding, the derived bit-planes are grouped into quantization indices and

\(^{11}\) The coded DCT bands are those for which the number of quantization levels in a QM in Figure 3.7 is non-zero. The rest of the bands, which are assigned a zero index in a QM, are called non-coded. As no Wyner-Ziv information is encoded for these DCT bands, the decoded coefficients are given by the coefficients of the side information.
MMSE reconstruction is carried out to obtain the decoded DCT coefficients. Thereafter, the decoded coefficients are assembled together with the coefficients of the side information frame at the positions of the non-coded DCT bands, yielding the decoded WZ frame in the transform domain. Finally, conversely to the pixel-domain architecture, the inverse DCT (IDCT) is performed and the decoded WZ frame is displayed and stored in the reference frame buffer.

Analogously to the PRISM codec, the DVC architecture designed at Stanford University performs low complexity encoding and shifts the computationally heavy operations to the decoder. Moreover, similar to PRISM, the codec provides inherent robustness to communication channel errors, since encoding is performed using Turbo codes. Furthermore, as the WZ frames are independently encoded but jointly decoded, that is, there is no inter prediction loop at the encoder, the DVC scheme of Aaron et al. avoids error propagation and drift. In contrast to PRISM, however, the Stanford DVC scheme requires a reversed (a.k.a., feedback) channel to perform optimal rate control at the decoder. Such a feedback can introduce a structural latency and renders the system unsuitable for certain applications, e.g., storage, etc. Regarding the compression capacity of the system, experimental results given in [12] demonstrate that the Stanford TDWZ architecture operates better than the H.263+ Intra, but worse than the H.263+ Inter conventional video codec [71].

3.3.3 The DISCOVER Codec

The abovementioned Wyner-Ziv video coding architectures are generally recognized to have laid the foundations of distributed video coding. Starting from these architectures, extensive research activities have been conducted during the last few years to further improve the compression performance and to develop more appealing coding features. Due to its attractive characteristics and its good compression results, most of these research activities concentrate on the TDWZ architecture developed at Stanford University (see Section 3.3.2 above). One of the most acknowledged extensions of the Stanford TDWZ architecture has been developed in the framework of the VISNET 1 European Union funded project, which resulted in the DISCOVER [24] codec. Essentially, the DISCOVER codec is considered a well-established reference in DVC, delivering state-of-the-art compression performance. In addition, the codec serves as an reliable benchmark, regarding the fact that its executables have been made available online [24] and that it is well-documented and supported by several scientific publications.

12 Detailed information concerning the project participants, duration, achievements and executables of the DISCOVER codec can be found on the project’s web-site: www.discoverdvc.org
The DISCOVER codec advances upon the TDWZ architecture of Aaron et al. in several ways, as explained next. At the encoder, in the quantization module, the DISCOVER group added one more quantization matrix, namely, QM 8 in Figure 3.7, in order to achieve a higher rate range [78]. Furthermore, a uniform quantizer has been employed for the DC coefficients but a double-deadzone one has been used for the AC coefficients, resulting in slightly better coding performance [78].

Regarding Slepian-Wolf coding, instead of employing rate-compatible punctured Turbo codes as in the Stanford codec, DISCOVER incorporated the rate-compatible low-density parity-check accumulate (LDPCA) codes of Varodayan et al. [48]. These specifically designed rate-compatible DSC codes are using LDPC codes as basis and they have shown to outperform rate-compatible punctured Turbo codes [24]. Due to their employment in this dissertation as well, the LDPCA codes of [48] are explained in Appendix A.

An additional modification introduced in the DISCOVER codec carries out an initial rate estimation at the encoder in order to confine the usage of the feedback channel [79]. In particular, the DISCOVER encoder performs a coarse estimation of the required encoding rate per encoded bit-plane of each DCT coefficient band. If the decoder cannot decode the bit-plane using the transmitted amount of Slepian-Wolf bits then it requests a complementary chunk of bits with the feedback channel. In this fashion, the total number of feedback channel requests from the decoder to the encoder are reduced, thereby diminishing the associated latency.

At the decoder, the DISCOVER codec features an improved side information generation tool compared to the one developed by the Stanford DVC group [12]. As in the latter, the MCI method of DISCOVER creates a motion-compensated prediction of the encoded WZ frame using a past and a future reference frame in a hierarchical bidirectional prediction structure. However, the MCI method of DISCOVER encompasses advanced extensions comprising block-based bidirectional motion estimation and compensation [80, 81], sub-pixel motion refinement [82], and spatial motion vector smoothing [80, 81]. More information on the MCI method of DISCOVER is given in Appendix B, where we also explain bidirectional overlapped block motion compensation, a novel more accurate compensation method developed in the context of this dissertation.

An alternative research direction during the development of the DISCOVER codec concentrated on the accurate modeling and estimation of the correlation statistics between the original frame, which is only present at the encoder-side, and the side information, which is only produced at the decoder-side. This correlation is typically expressed in terms of a virtual correlation channel between the source to be encoded and the side information. At a DVC decoder, the correlation statistics are
used as soft-input information to the soft Slepian-Wolf (i.e., Turbo or LDPC) decoder and to the reconstruction module. Similar to prior art [12, 73-75, 78] the DISCOVER codec employs a Laplacian pdf to model the correlation channel. In initial DVC solutions, the correlation channel distribution parameters were estimated using an offline training phase [12] or extrapolated from previously decoded information at the decoder [74]. Conversely, in DISCOVER [24], the parameters of the correlation noise are estimated online at the decoder considering that the correlation noise may vary temporally and/or spatially. In particular, the partners of the VISNET I project [83-86] assumed the correlation noise as side-information-independent and stationary at different levels including sequence-, frame-, block-, and pixel-level in the pixel domain architecture, as well as, band-per-sequence-, band-per-frame-, and coefficient-level in the transform domain architecture. Experimental evaluation of the DISCOVER correlation channel estimation method has shown systematic gains over previous techniques [86]. More information on the DISCOVER correlation channel estimation methods can be found in Chapter 5, where one of the main contributions of this dissertation, namely, the concept of side-information-dependent correlation channel modeling in DSC and its application in DVC, is detailed.

Another feature of the DISCOVER codec includes the application of layered Wyner-Ziv principles (see Section 2.7.2) in the TDWZ video coding architecture. Specifically, following the design of [17], the soft-input information to the Slepian-Wolf decoder of DISCOVER is calculated based on the correlation model, the value of the side information and the values of the previously decoded source bit-planes (if any). In this way, DISCOVER enables quality-layered coding of the WZ frames.

After Slepian-Wolf decoding, DISCOVER performs optimal MMSE reconstruction to obtain the decoded coefficient values. In this context, an additional contribution of the DISCOVER group was the calculation of the reconstruction value using closed-form expressions derived in [87] for a Laplacian correlation model. More information on the topic can be found in [87].

As regards its compression performance, DISCOVER has shown to systematically outperform alternative TDWZ architectures in the literature, e.g., [12, 75]. In fact, as mentioned above, the DISCOVER codec is considered the state-of-the-art TDWZ video coding reference in the literature. Moreover, the rate-distortion performance of DISCOVER has been compared with the performance of traditional video coding solutions, configured to exhibit low encoding complexity [88]. In this respect, low-cost encoding configurations of H.263+ [71] and H.264/AVC [8] have been used in the literature to benchmark the performance of DISCOVER [88], as detailed next:
H.263+ Intra coding exploits the spatial but not the temporal redundancy in video. Although nowadays H.263+ has been widely replaced by the H.264/AVC standard, initial DVC codecs [12, 23] were using the former as benchmark, mainly due to its significantly lower encoding complexity compared to H.264/AVC.

H.264/AVC Intra coding encodes the frames in a video sequence independently and exploits only the intra-frame (i.e., spatial) redundancies. H.264/AVC Intra is considered one of the most efficient intra frame coding schemes, its high compression capacity being attributed to the employment of advanced entropy codecs and intra coding modes.

H.264/AVC Inter without motion estimation (H.264/AVC No Motion) exploits the spatial correlation in each video frame and enables partial temporal redundancy removal by using simple differential coding principles. H.264/AVC No Motion reduces encoding complexity by impeding motion estimation at the encoder-side. In general, H.264/AVC No Motion outperforms H.264/AVC Intra since it exploits more redundancies. In extreme motion content, however, when temporal correlation between neighboring video frames is limited, H.264/AVC Intra may exhibit higher compression efficiency. When compared against full H.264/AVC Inter coding (with full motion search at the encoder), H.264/AVC No Motion exhibits reduced compression capability as motion estimation is not carried out.

For the used test video sequences as well as many others, the DISCOVER codec operating at GOP 2 can always outperform H.263+ Intra. Similarly, DISCOVER always performs better or equal to H.264/AVC Intra. What is more, DISCOVER’s performance is occasionally above the performance of H.264/AVC No Motion. This is due to the fact that DISCOVER applies motion estimation at the decoder while the H.264/AVC No Motion codec is not performing any at the encoder. However, apart from test sequences which contain regular and easy-to-capture motion content, DISCOVER’s RD performance drops severely when long GOP sizes are considered. This is caused by the deficiency of the DISCOVER’s motion interpolation technique to produce good quality side information when the GOP size increases, that is, when reference frames are far apart.

Note that, as it exhibits very low encoding complexity, DISCOVER achieves a much lower compression performance compared to the one delivered by H.264/AVC inter frame coding with full motion estimation at the encoder-side. At this point, it is important to mention that, conversely to the aforementioned academic and asymptotic analysis of Ishwar et al. [70] (see Section 3.3.1) practical...
DVC systems still cannot (and may not in the future) reach the performance of their conventional inter-frame video coding equivalents. This is because of several reasons. First, compared to traditional predictive video coding, which is built on conventional source coding principles, DVC is founded on the basics of DSC (see Chapter 2). Therefore, in contrast to classical video coding schemes, DVC systems suffer from the inherent loss of Wyner-Ziv coding, which is attributed to the absence of side information at the encoder-side (see Section 2.6.1). Second, in DVC systems Slepian-Wolf compression is realized using channel codes, which constitute the counterpart of entropy coding in traditional codecs. As mentioned in Section 2.5, Slepian-Wolf coding, using a capacity achieving code for a given correlation channel, does not incur a rate loss with respect to contemporary entropy coding techniques (e.g., arithmetic coding). However, in general, natural sources, like video or image data, exhibit complex correlation statistics which cannot easily be captured by a common correlation channel model. This typically affects the performance of Slepian-Wolf based channel coding, thereby rendering it less effective than entropy coding. Third, in essence, traditional video codecs exploit the available temporal correlation in the video data by producing high quality motion-compensated prediction at the encoder, while having access to the original frame. In contrast, DVC schemes generate temporal prediction, i.e., side information, at the decoder without access to the original frame to be encoded. As a consequence, DVC systems bear an additional performance loss compared to traditional video coding that is attributed to their deficiency to generate efficient motion-compensated predictions [25, 89].

3.3.4 Improvements over the DISCOVER Codec

The DVC research group that I am part of is staffed by the Electronics and Informatics Department at the Vrije Universiteit Brussel and by the Multimedia Laboratory at the Ghent University, operating under the umbrella of the Interdisciplinary Institute for BroadBand Technology (IBBT). IBBT’s research group has acquired in-depth expertise and made significant contributions in the international literature in the field of distributed video coding and its applications.

The Stanford TDWZ video coding architecture with the DISCOVER improvements has independently been implemented by both parties of the DVC group in order to serve as a dependable state-of-the-art basis for the conducted research. Although each party has put more emphasis on specific problems and research directions, the proposed solutions and methods are in general a result of collaborative work.

To augment the efficiency of the TDWZ architecture, IBBT has developed DVC
codecs including several coding modes, that is, Wyner-Ziv, intra and skip. This is motivated by traditional state-of-the-art video codecs, e.g., the H.264/AVC [8] coding standard, which employ a rich set of intra- and inter-prediction modes, as well as, advanced rate-distortion-driven mode selection mechanisms. Such a mode decision process allows for adapting to possibly varying characteristics in a video sequence, thereby increasing the coding performance. In the TDWZ architectures proposed by IBBT, mode decision is performed in a bit-plane [90, 91], or block-based [92, 93] manner depending on whether the coded information is partitioned in bit-planes or blocks, respectively. Note that the proposed mode decision mechanisms have been performed in a rate-distortion driven strategy, based on closed-formed expressions for the rate and the distortion for a Laplacian pdf. It is also important to mention that, apart from the approach of [93], the mode selection process in [90-92], is carried out at the decoder, hence remarkably boosting the compression capability of pure Wyner-Ziv video coding without jeopardizing the low complexity encoding characteristics of the DVC framework.

The IBBT DVC research group has also made a significant contribution towards constraining the use of the feedback channel in distributed video coding [94, 95]. Anchored in the Stanford TDWZ codec, most Wyner-Ziv video coding systems proposed in the literature utilize a feedback channel from the decoder to the encoder to optimally allocate the rate. Although the use of such a feedback channel guarantees the best performance, it also imposes specific limitations. First, certain applications, e.g., storage, cannot be straightforwardly addressed as the encoding rate is only determined by the decoder. Second, frequent use of the feedback channel may introduce non-negligible end-to-end delays associated with the forward and backward communication channel. To tackle these limitations, a number of feedback-channel-free systems have been presented [23, 96, 97]. In these systems, it is the encoder that determines the rate, denying the decoder the right to issue any requests for bits. Noticing that a limited feedback may be supported in many video streaming scenarios, Slowack et al. [94, 95] introduced a method for constraining the number of feedback requests to a fixed maximum number per WZ frame. The technique of [94, 95] initially estimates the WZ rate at the decoder based on information obtained from previously decoded frames, and thereafter updates the feedback channel requests based on the estimated difference between the already spend and the true WZ rate. The experimental results provided in [94, 95] show for example that, for a GOP of size 4, the rate penalty is less than 5% when only 5 requests are allowed per WZ frame. Moreover, because of the improved performance delivered by the bit-plane-based mode decision architecture of [90, 91], the system is still able to perform better than or similar to the DISCOVER [24]
codec even when up to 2 requests per WZ frame are issued. The effectiveness of the approach is justified by estimating end-to-end delays and encoder buffer requirements, showing that Wyner-Ziv video coding systems with constrained feedback can be a viable and key solution for lightweight video streaming applications.

Low coding delay is an important prerequisite for several applications in which high delay would obscure their proper functionality. In general, generating the side information is arguably the most important task in DVC. The side information generation methods used in most works in the literature, e.g., [24], rely on MCI, which deploys a bidirectional prediction structure. Although side information creation techniques based on bidirectional prediction provide the finest DVC performance, they introduce structural latency that has to be absorbed by the codec. Depending on the application scenario, this delay may or may not be acceptable. To overcome this drawback, the IBBT DVC research group has also investigated low-delay DVC architectures that consider unidirectional, i.e., low-delay, side information generation methods. Motion-compensated extrapolation (MCE) is an alternative method that eliminates the delay introduced by interpolation techniques at the cost of prediction accuracy [12, 25, 98]. In this context, Skörupu et al. [93] have proposed a DVC system based on MCE that allows for efficient low-delay video coding with low complexity at the encoder. The presented extrapolation technique first estimates the motion field between the two most recently decoded frames using the Lucas-Kanade algorithm [99]. The obtained motion field is then extrapolated to the current frame using an extrapolation grid. The experimental results presented in [93] have proven that the presented method outperforms alternative techniques like [98]. The MCE technique proposed in [93] is incorporated into an efficient DVC architecture featuring hybrid block-frequency Wyner-Ziv coding as well as mode decision. The designed architecture is shown to deliver comparable or superior rate-distortion performance compared to alternative state-of-the-art codecs, e.g., [24].

Another issue of contemporary DVC architectures, which has been studied by IBBT’s DVC group, is the mismatch between the intra and WZ frame quantization processes [100]. In fact, quantization in intra-frame coding like H.264/AVC [8] is rather dissimilar to its counterpart in Wyner-Ziv coding. First, the nature and range of the values that are provided as inputs to the quantizer are different. In H.264/AVC Intra, for instance, the encoder first generates a spatial prediction of the current block of pixels, and then transforms and quantizes the residual between the current block and its prediction. In Wyner-Ziv coding, though, pixel values are directly transformed and quantized, without generating a prediction. Second, both
Conventional vs. Distributed Video Coding

quantization approaches normally feature different bin widths, and properties. Third, at the decoder, the reverse quantization steps are different as well; namely, optimal MMSE reconstruction is performed in WZ decoding which is not the case in intra-frame decoding. As shown in [100], due to this mismatch, Slepian-Wolf rate is spent even for spatial regions that are accurately approximated by the side information. To solve this issue, Slowack et al. [100] proposed side information generation based on MCI with selective unidirectional motion compensation from temporally adjacent WZ frames. For more details we refer to [100].

To improve the performance of MCI-based Wyner-Ziv video coding architectures, our group has also focused on improving the MCI method of DISCOVER [24, 80, 81]. In the context of this PhD research, we have replaced the bidirectional motion compensation method of [80, 81] with bidirectional overlapped block motion compensation [101]. This has caused the energy of the prediction error to decrease, thus increasing the quality of MCI-generated side information and in turn improving the compression performance [102]. More information on the aforementioned method is given in Appendix B.

A common characteristic of the abovementioned IBBT contributions on DVC – except for [93] – is that they employ an MCI-based algorithm to produce side information at the decoder, as explained in Appendix B. However, MCI techniques [24, 80, 81], which perform blind motion estimation at the decoder based on an assumed linear motion model, fail to deliver adequate prediction quality in sequences with irregular motion content [25, 89]. Within the DVC research group of IBBT, one of the main focuses of this dissertation is to design appropriate side information generation methods and associated DVC architectures that can successfully tackle the aforementioned shortcoming. In this direction, we have proposed overlapped block motion estimation and compensation (OBMEC) [103], a powerful motion estimation and compensation technique enabling accurate capturing of motion using a rough description of the original frame present at the decoder-side. The OBMEC method and its applications in (i) a hash-based feedback-channel-free DVC system [103-105], (ii) a hash-based Wyner-Ziv video coding architecture [106-110] and (iii) a successively refined Wyner-Ziv video codec [102] are presented in Chapter 4. It is important to highlight that the research conducted in this dissertation mainly concentrates on hash-based DVC architectures as they have shown to be advantageous over their MCI-based counterparts, especially when video content with irregular motion is coded. In this respect, a novel efficient hash-based DVC architecture with an improved overlapped block motion estimation and compensation with sub-sampled matching (OBMEC/SSM) technique [111] is also presented in Chapter 4. Moreover, in order to design an effective DVC
system for demanding applications like wireless capsule endoscopy [112], OBMEC is modified as explained in Chapter 6 so as to adapt to the spatial discrepancies in temporal correlation in a frame.

Furthermore, the DVC research group of IBBT has conducted work in the field of correlation channel modeling and estimation. In particular, IBBT has proposed novel correlation models in the state-of-the-art MCI-based DVC architecture, with ability to adapt to varying content and coding parameters [113, 114]. The parameters of the correlation noise in the transform-domain are estimated by exploiting the pixel-domain correlation channel estimate and the spatial correlation of the noise signal. In addition, anchored in the research resulted in this dissertation, IBBT’s DVC group was the first to bring the novel concept of side-information-dependent (SID) correlation channel modeling in DVC [104, 111, 115-117]. Side-information-dependent (SID) modeling, a novel view of the correlation channel which implements asymmetric channel modeling, improves upon the modeling accuracy of existing symmetric approaches. Further specifics on the SID channel model, its theoretical evaluation and its experimental assessment are provided in Chapter 5.
Chapter 4
OVERLAPPED BLOCK MOTION ESTIMATION AND COMPENSATION AT THE DECODER

4.1 INTRODUCTION

The quality of the side information largely affects the compression performance of a Wyner-Ziv (video) codec in a twofold way. First, the higher the correlation between the side information $Y$ and the source $X$ to be encoded, the lower is the Slepian-Wolf rate necessitated for successful decoding. That is, on a theoretical level, the higher the quality of the side information, the better the performance of sphere packing (see Section 2.6.2). Second, the better the side information predicts the source, the higher the Wyner-Ziv reconstruction quality becomes. Recall from Section 2.6.1 that Wyner-Ziv reconstruction uses a function $\hat{\phi}: \mathcal{A}_Y \times \mathcal{A}_U \rightarrow \mathcal{A}_X$, where $U$ is an auxiliary random variable, representing the sphere covering argument (see Section 2.6.2). This chapter addresses the challenging problem of side information generation, in terms of effective motion-compensated prediction at the decoder, for the design of efficient Wyner-Ziv video coding schemes.

4.1.1 Related Work on Side Information Generation

Traditional video codecs, e.g., H.264/AVC [8], exploit the available temporal correlation by producing high quality motion-compensated prediction at the encoder. Conversely, common DVC schemes [12, 24] employ motion-compensated interpolation (MCI) to generate temporal prediction (i.e., side information) at the decoder. However, MCI performs blind motion estimation (i.e., without access to the original frame to be coded) by assuming a linear motion model which falls short in capturing reality in case of irregular motion [25]. Additionally, traditional video coding systems can effectively exploit temporal correlations in large GOP, thereby vastly increasing the compression efficiency. In DVC, coding large GOPs comes
with the profit of reduced encoding complexity, since more frames are coded under the Wyner-Ziv paradigm. Alas, the compression performance of state-of-the-art MCI-based DVC architectures [24] deteriorates when the GOP size increases [24]. To increase the quality of the created side-information, alternative DVC schemes employ hash-based motion estimation [118-120], where appropriate information per WZ frame, referred to as hash, is transmitted to the decoder to support motion prediction. Moreover, driven by the principles of successively refined WZ coding [41], joint decoding and side information refinement [121-123] has been considered.

4.1.1.1 Motion-Compensated Interpolation

Initially, Aaron et al. [72] proposed a simple average interpolation scheme in which the pixel values at the same location from the two future and past reference frames were averaged. However, when no attempt is made to model the true motion of pixels or blocks of pixels, the generated side information will be a rough estimate of the original frame, especially in medium or high motion video sequences. Therefore, assuming a linear motion model [73], they introduced a block-based MCI technique using symmetric motion vectors (see also Section 3.3.2). An improved MCI method, which was incorporated in the state-of-the-art DISCOVER [24] codec, performed MCI comprising block-based bidirectional motion estimation and compensation [80, 81], sub-pixel motion refinement [82], and spatial motion vector smoothing [80, 81]. For further details on the MCI method of the DISCOVER codec the reader is referred to Section 3.3.3 and to Appendix B.

Later, a variable block-size MCI technique based on a block-adaptive matching algorithm driven by the local motion activity in the reference frames was proposed by Argyropoulos et al. in [124]. Variable block-size MCI was also independently proposed by Huang and Forchhammer in [125]. In the latter framework, an overlapped block motion compensation based approach was included to improve the compensation.

4.1.1.2 Hash-Based Motion Estimation

A first example of a DVC scheme encoding auxiliary (hash) information to assist side information generation was the PRISM system [11, 23], where a CRC of the quantized blocks was used to aid in determining the motion at the decoder (see Section 3.3.1). In [118], Aaron et al. proposed a hash code consisting of a coarsely sub-sampled and quantized version of each block in a WZ frame. The encoder performed a block-based decision whether to transmit the hash. For the blocks for which a hash code was sent, hash-based motion estimation was carried out at the decoder, while for the rest of the blocks, the co-located block in the previous reconstructed frame was used as side information. In [126] several hash generation
approaches – either in the pixel or in the transform domain – were investigated. It was shown that hash information formed by a quantized selection of low frequency DCT bands per block was outperforming the other methods [126].

In [120], so-called low quality references, obtained by means of very low bit-rate H.264/AVC coding, served as hash information. Zero-motion-vector (no motion) inter prediction was enabled in H.264/AVC to create these references, introducing temporal dependency between Wyner-Ziv hash frames in larger GOPs. In [127] the encoder detected blocks with motion content based on the difference with the previous frame and sent a hash to the decoder. The hash consisted of a number of quantized DCT coefficient bands. To reduce the hash rate, analogous to [120], temporal prediction of the hash with respect to the previous hash sent to the decoder and entropy coding were deployed. For the purpose of signaling, the hash-block binary maps were encoded with run-length encoding and entropy coding. At the decoder, if a DCT-based hash was sent, hash-based motion estimation was performed; otherwise, MCI was used. The hash quantized DCT coefficients were also multiplexed with the corresponding coefficients of the side information frame to further improve the quality of the prediction. In [128], a hash construction similar to [127] has been considered at the encoder. However, at the decoder, several side information techniques, namely, average interpolation, MCI and hash-based motion interpolation, were combined using genetic algorithms.

Extending over their previous approach i.e., [127], Ascenso et al. [119] proposed to perform a block-based selection at the encoder, based on the current frame to be coded and its future and past frames in hierarchical order. Blocks for which MCI was foreseen to fail were low quality H.264/AVC Intra encoded and transmitted to the decoder to assist MCI. The residual frame, given by the difference between all reconstructed intra coded blocks (for the hash blocks) or the central luminance value (for non-hash blocks) and the corresponding blocks in the WZ frame, was formed and Wyner-Ziv encoded.

**4.1.1.3 Joint Decoding and Side Information Generation**

Initial works proposed successive side information refinement in pixel-domain Wyner-Ziv (PDWZ) architectures [122, 129, 130]. More intricate, [131] proposed a multistage side information refinement technique by performing Wyner-Ziv coding of a frame in several spatial resolution layers. Every decoded lower level was used to refine the motion estimation process, thereby yielding improved side information for the next higher level. However, since the compression performance of PDWZ video coding is rather low, state-of-the-art DVC solutions operate in the transform domain.
Regarding TDWZ systems, successive refinement of the side information through unsupervised learning of the motion vectors combined with LDPC decoding was proposed in [123]. A motion estimation refinement stage based on the decoded DC components of the WZ frames was presented in [132]. Both in [123] and [132] refinement was carried out in the transform domain, thereby requiring an overcomplete DCT representation of the reference frames.

Alternatively, [133] decoded and reconstructed all DCT bands of the WZ frame but added a post-processing procedure, including motion vector refinement and optimal reference frame selection, to update the side information, prior to the final coefficient reconstruction. Repeated side information generation after the reconstruction of every DCT coefficient band was put forward in [121]. However, the approach in [121] only employed the already available side information frame as a reference frame to further refine the side information for decoding the next band. An approach similar to the one in [121] has been proposed in [134], in which reference information for side information refinement was extracted from the previous and the next reference frames (in a hierarchical bidirectional structure).

4.1.2 Contributions

Advancing over the abovementioned state-of-the-art solutions, this chapter proposes innovative effective side information generation techniques that improve the Wyner-Ziv video coding performance. To this purpose, overlapped block motion estimation and compensation (OBMEC), a novel scheme that performs multi-hypothesis pixel-based motion-compensated prediction at the decoder, is proposed. By not limiting the motion information during motion compensation to just one motion vector for an entire block as in contemporary techniques (see above), OBMEC mitigates the prediction error at a pixel level, thereby boosting the RD performance. The innovative conception of performing OBMEC at the decoder has resulted in several important publications in the related international literature [102-111].

Incorporating the OBMEC technique in several DVC systems leads to the following contributions:

- Firstly, by communicating hash information consisting of a number of most significant bit-planes (MSBs) to the decoder, the proposed OBMEC technique triggers the design of a novel hash-based codec featuring very low encoding complexity and operating without a feedback channel. Although lacking a transform-domain Wyner-Ziv codec with a feedback channel, the codec delivers reasonable performance, systematically surpassing the performance of the best pixel-domain feedback-channel-based DVC system in the literature,
Overlapped Block Motion Estimation and Compensation at the Decoder

i.e., [85].

- Secondly, by adding a Wyner-Ziv coding layer, operating in the transform domain and employing a feedback channel, to the previous architecture, a novel hash-based Wyner-Ziv video codec is designed, which yields state-of-the-art compression performance. Experimental results show that the proposed system significantly advances over the basic architecture and yields improved RD performance with respect to the state-of-the-art DISCOVER codec.

- Thirdly, to further boost the DVC compression performance, an innovative efficient hash-based DVC architecture is proposed. Compared to other solutions in the literature, e.g., [118, 119, 127, 128], the presented architecture comprises a novel approach to form and compress a hash per WZ frame, which imposes a negligible complexity and memory overhead at the encoder. At the decoder of the proposed system, high-quality side information is produced by an original overlapped block motion estimation and compensation with sub-sampled matching (OBMEC/SSM) technique. OBMEC/SSM improves over our bit-plane-based OBMEC, by employing a different matching strategy and predictors that improve the prediction quality and reduce the computational complexity.

- Fourthly, this work introduces a novel OBMEC-based algorithm, which enables side information refinement after decoding of low frequency DCT coefficients of the WZ frames. Initial side information is generated by means of an advanced MCI technique sharing key features with [24, 80, 81]. However, contrary to bidirectional motion compensation employed in [80, 81], the proposed approach performs bidirectional OBMC, causing the energy of the prediction error to decrease (see Appendix B). Contrary to [123, 132], the proposed OBMEC-based refinement approach is performed in the pixel-domain. Also, conversely to [121, 134], in which side information refinement is carried out after decoding of each DCT band, OBMEC enables a significant quality improvement of the side information by exclusively refining after decoding the lowest frequency (DC) coefficient band. The experimental results report that the in-house MCI-based TDWZ codec delivers state-of-the-art DVC performance, outperforming the DISCOVER codec [24]. In addition, when performing OBMEC-based side information refinement, the proposed codec achieves significant improvements over reference codecs.

This chapter is organized as follows. Section 4.2 elaborates on the basic pixel-domain feedback-channel-free architecture and presents our novel OBMEC method, which uses the available hash information, i.e., a number of MSBs of the WZ frames at the decoder. The novel extension of this architecture, by the addition of a
Chapter 4

transform-domain Wyner-Ziv layer, is discussed in Section 4.3. Section 4.4 expands on our additional novel hash-based DVC architecture, which includes new components, i.e., the new hash codec and the novel OBMEC/SSM method. Section 4.5 describes the proposed OBMEC-based side information refinement technique. Section 4.6 reports the experimental evaluation of the proposed side information creation methods based on the OBMEC and OBMEC/SSM techniques. This section also presents the assessment of the proposed coding systems. Finally, Section 4.7 draws the conclusions of this chapter.

4.2 BIT-PLANE-BASED OBMEC

The aim of this section is to describe overlapped block motion estimation and compensation (OBMEC), a novel powerful technique which enables the generation of an accurate prediction of a frame using hash information, i.e., a coarse description of the frame, at the decoder. To this purpose, we propose a basic low-cost encoding architecture, which losslessly codes a coarsely quantized version of the luma components of an original frame and enables the prediction of the remaining information at the decoder by exploiting temporal correlation with OBMEC.

4.2.1 The Spatial-Domain Unidirectional DVC Codec

The encoding syntax of the proposed spatial-domain unidirectional DVC (SDUDVC) codec is shown in Figure 4.1. The input video sequence is organized into GOPs, and is decomposed into key frames, i.e., the first frame in each GOP, and WZ frames, i.e., the remaining frames in each GOP.

Figure 4.1: Block diagram of the SDUDVC codec. This architecture serves as a basis for the design of our hash-based transform-domain Wyner-Ziv codec.

In our implementation, the key frames $I$ are encoded using a conventional intra-frame codec, e.g., the H.263+ Intra or the H.264/AVC Intra codec. The luminance component of the WZ frames $X$ is subjected to coarse uniform scalar quantization
with a quantization step size $2^{M-b}$, where $M$ denotes the bit-depth of the original samples and $b \in \mathbb{Z}_+$ indicates the number of coded bit-planes. This operation yields the proposed hash information per WZ frame, namely, the quantized frame $\tilde{X}$, which contains quantization indices $Q_{\text{hash}}(X) = \left\lfloor X/2^{M-b} \right\rfloor$ in the range $[0, 2^b - 1]$. Visual examples of the proposed hash frames (for $b = 1$) are depicted in Figure 4.2.

Each quantized (i.e., hash) frame $\tilde{X}$ is then compressed. Specifically, the retained $b$ luma bit-planes of the first quantized frame $\tilde{X}$ in the GOP are entropy coded (i.e., intra). For the remaining quantized WZ frames in the GOP the retained $b$ luma bit-planes are entropy coded in a differential bit-plane-by-bit-plane manner. Namely, each significant luma bit-plane $b_i$ of a quantized WZ frame $\tilde{X}_n$, denoted by $\tilde{X}_n^{b_i}$, is subtracted (i.e., using an exclusive OR operation) from the corresponding bit-plane of the previously encoded quantized frame, denoted by $\tilde{X}_n^{b_{i-1}}$, and the prediction error is entropy encoded.

We note the use of simple differential coding and entropy coding instead of Slepian-Wolf codes to efficiently encode the $b$ most significant luma bit-planes of the WZ frames. This differentiates the presented low-complex scheme from classical Wyner-Ziv systems equipped with Slepian-Wolf coding, such as [12, 23, 24]. Notice, however, that temporal prediction of a coarsely quantized version of the WZ frame has been also proposed in the hash-based coding systems presented in [120, 127]. Our approach is motivated by the findings in [135], where it has been shown that in an unidirectional DVC system, channel coding is deficient compared to simple entropy coding of the bit-planes, unless one performs motion estimation at the encoder, which is basically undesired. In this regard, we remark that DSC without Slepian-Wolf compression has been theoretically studied by Yang and Xiong in [136], where Wyner-Ziv coding based on entropy-coded quantization was proposed as a viable solution for short codeword lengths. In general, the presented SDUDVC codec encodes the WZ frames without knowledge of the side information, which is only produced at the decoder. That is, it follows the major principle and meets the application target of DVC.

Rate control is performed by manipulating $b$ and the quantization parameter in the WZ and key frames, respectively. To achieve fair compression efficiency, we use $b \leq 2$.

---

13 According to the Slepian-Wolf random binning argument, an infinite code length (which is unrealistic) is needed to achieve error-free decoding of the quantized source. As a result, hands-on Slepian-Wolf codes suffer from distortion introduced by decoding errors, especially at short code lengths. To obtain the best RD performance at short code length, one needs to find the best tradeoff between rate and distortion due to decoding errors. Yang and Xiong [136] replaced Slepian-Wolf coding with entropy coding, which guarantees perfect decoding.
At the decoder, the missing WZ frame \((M - b)\) bit-planes are decoded using our novel OBMEC method, which is the key feature of the proposed scheme, as explained next.

### 4.2.2 Bit-Plane Overlapped Block Motion Estimation

Motion estimation relies on the observation that pixel intensities are related to the intensities of pixels in previous frames due to the motion of objects through a scene. Both in conventional \([8]\) and in distributed video codecs, e.g., \([24, 25, 80, 81, 124]\), motion estimation is typically performed on a block basis where the motion of all pixels within the same block is assumed constant and matching blocks are selected by minimizing a distortion metric, usually SAD or MSE.

Since motion estimation is performed at the decoder using low quality (hash) information for the original frame samples [see for example Figure 4.2(b) and (d)], the work in this dissertation proposes overlapped block motion estimation (OBME) in order to reduce the motion uncertainty at a pixel level.

Unlike alternative side-information generation approaches, e.g., \([24, 25, 80, 81, 124]\), which limit the motion information during motion compensation to one (bidirectional) motion vector for an entire block, the proposed technique enables
**Overlapped Block Motion Estimation and Compensation at the Decoder**

**multi-hypothesis pixel-based prediction.** In this fashion, OBME offers an improved approximation of the motion at an individual pixel level, thereby decreasing the total energy of the prediction error and rendering it more uniform [101]. Moreover, OBME reduces blocking artifacts, which vastly increases the subjective quality compared to state-of-the-art DVC schemes [24]. It is also worth mentioning that in traditional video coding, e.g., [8], increasing the density of the motion field leads to a significant increase in the bit-rate required to code the motion vectors. In the proposed scheme, though, OBME is performed at the decoder and the aforementioned drawback vanishes.

![Figure 4.3: The supported hierarchical bidirectional motion prediction structure for a GOP of size 8. Black and grey rectangles represent key and WZ frames, respectively. Arrows signify the prediction direction; they originate from the decoded hash (or the partially decoded WZ frame in case of side information refinement), and end at each reference frame. Predictions of the same temporal level are indicated by identical line patterns.](image)

OBME is performed in a hierarchical bidirectional prediction structure similar to that used in state-of-the-art MCI-based DVC systems [12, 24], as shown in Figure 4.3. Using the reconstruction of two previously encoded WZ and/or key-frames as past and future reference frames, the decoder performs OBME using the hash information, i.e. the available $b$ most significant luma bit-planes of the current WZ frame.

To describe the subsequent steps of the proposed method, let us adopt the following notation. Specifically, let $X$, $Y$ and $R_k$, $k = \{0,1\}$, denote the WZ, the side information and the reference frames, respectively, and let $\tilde{X}$, $\tilde{R}_k$ be the frames containing the quantization indices determined by the $b$ most significant bit-planes in the WZ and the reference frames, respectively. Also, denote by $X_m$, $R_{x,m}$, $\tilde{X}_m$, $\tilde{R}_{x,m}$ blocks of size $B \times B$ samples with top-left coordinates $m$ in $X$, $R_k$, $\tilde{X}$.
and \( \tilde{R}_k \) respectively. Additionally, let \( X_m(s) \) denote the luma sample at position \( s = (i, j) \) in the block \( X_m \).

Figure 4.4 contains a graphical representation of the OBME algorithm. The hash frame \( \tilde{X} \) is divided into overlapping spatial blocks, \( \tilde{X}_{u}(s) \), with top-left coordinates

\[
u = (\varepsilon \cdot u_1, \varepsilon \cdot u_2), \quad 0 \leq u_1 < \left\lfloor \frac{H}{\varepsilon} \right\rfloor, \quad 0 \leq u_2 < \left\lfloor \frac{W}{\varepsilon} \right\rfloor,
\]

(4.1)

where \( \varepsilon \in \mathbb{Z}_+ \), \( 1 \leq \varepsilon < B \) is the overlapping step, \( \left\lfloor \cdot \right\rfloor \) is the floor function and \( H \), \( W \) are the frame height and width, respectively. We remark that the value of the overlapping step \( \varepsilon \) controls the complexity-prediction performance tradeoff of the presented OBME method. In particular, a small overlapping step \( \varepsilon \) yields a large number of overlapping blocks and in turn an accurate estimation of the true motion field. Increasing the value of \( \varepsilon \) reduces the computational complexity of the OBME technique. However, at the same time, a large \( \varepsilon \) reduces the prediction quality as the number of overlapping blocks diminishes.

For each block \( X_u \), the best matching block within a specified search range \( \rho \), is found in each of the reference frames \( R_k \). Since motion estimation is performed on the \( b \) MSBs of the WZ frame, traditional error metrics, e.g., the SAD or the MSE, are imprecise matching criteria. Instead, the proposed matching criterion [103] maximizes the complement, i.e., \( 1 - \text{PER} \), of the so-called pixel error ratio (PER) calculated on the available \( b \) MSBs between the current block \( \tilde{X}_u \) in the WZ frame and a block \( \tilde{R}_{k,u-v} \), \( v = (v_1, v_2), \ -\rho < v_1, v_2 \leq \rho \) in a reference frame. The measure \( 1 - \text{PER} \) is defined as the number of quantized indices in \( \tilde{R}_{k,u-v} \) which are identical to those of the co-located quantization indices in \( \tilde{X}_u \), divided by the total number of samples in the block. The corresponding motion vector is thus found as

\[
v_k = \arg \max_{v_i} \frac{1}{B^2} \sum_s \delta\left[ \tilde{X}_u(s) - \tilde{R}_{k,u-v}(s) \right],
\]

(4.2)

where \( \delta[\cdot] \) is the Kronecker delta function and \( s \) scans all the samples in the block. Eq. (4.2) can be equivalently expressed as

\[
v_k = \arg \max_{v_i} \Pr[\tilde{X}_u | \tilde{R}_{k,u-v}].
\]

(4.3)

According to (4.3), identifying the motion vectors by using \( 1 - \text{PER} \) as matching criterion is equivalent to maximizing the correlation \( \Pr[\tilde{X}_u | \tilde{R}_{k,u-v}] \) between the available \( b \) most significant bit-planes \( \tilde{X}_u \) and \( \tilde{R}_{k,u-v} \) of the blocks in the WZ and reference frames. The best match in each reference frame, i.e., the block which satisfies (4.3), yields the best temporal predictor, denoted as \( \Psi_{k,u} \) for the considered WZ block \( X_u \). Also, \( 1 - \text{PER} \) is used as an estimate of \( \Pr[X_u | \Psi_{k,u}] \).
Figure 4.4: Graphical representation of the motion estimation algorithm. All the overlapping $B \times B$ blocks, that contain the current motion-compensated pixel position in the hash frame $\tilde{X}$, are designated $\tilde{X}_{u'}$, where $u' = (u'_1, u'_2)$ are the top-left coordinates of the block. For every block $\tilde{X}_{u'}$, the best matching block in $\tilde{R}_{k=0}, \tilde{R}_{k=1}$ is found, yielding the motion vectors $v^j_{k=0}, v^j_{k=1}$. The co-located pixels in the estimated blocks $\tilde{R}_{k=0,u'-v^j_{k=0}}, \tilde{R}_{k=1,u'-v^j_{k=1}}$ serve as temporal predictors for the current motion-compensated pixel position.
that is, of the correlation between the original pixel values and the side information. Thus, for each overlapping block, OBME provides two temporal prediction blocks, i.e. one for each reference frame $R_k, 0 \leq k \leq 1$, and the associated correlation estimates.

### 4.2.3 Probabilistic Motion Compensation

After the execution of the OBME, each pixel $X(s) = (i, j)$ in the WZ frame belongs to a number of overlapping blocks $X_{u_c}$ with $u_c = (u_{c,1}, u_{c,2})$, $c = \{1, 2, ..., C\}$. For each of these blocks, OBME has identified a temporal predictor block $\Psi_{k, u_c}$ in each reference frame. This means that, each pixel $X(s)$ in the WZ frame is linked to a number of corresponding *candidate predictors* $\psi_{k, u_c}$ in the blocks $\Psi_{k, u_c}$. Depending of the considered block size, the overlapping step and the position of a pixel in the motion-compensated frame, the maximum number of predictors per pixel is $(B/\varepsilon)^2$ per reference frame.

Out of these candidate predictors, only the ones for which the $b$ MSBs are identical to the ones of the motion-compensated pixel WZ pixel, that is, $Q_{\text{hash}}(\psi_{k, u_c}) = Q_{\text{hash}}(X(s)) = \hat{X}(s)$, are retained and referred to as *valid predictors*. The remaining invalid predictors are discarded. Additionally, each valid $\psi_{k, u_c}$ has a certain “degree of trust” associated to it, given by $\Pr[ X(u_c) \mid \Psi_{k, u_c} ]$, i.e., the $1 -$ PER of the corresponding best matching block $\Psi_{k, u_c}$ to which the candidate predictor belongs. Subsequently, one uses the hash information and the retained predictors to produce high quality side information as

$$Y(s) = 2^{M-b} \cdot \hat{X}(s) + Y_{M-b}(s), \quad (4.4)$$

where $Y_{M-b}(s)$ denotes the $M-b$ bit-planes in the side information $Y(s)$. $Y_{M-b}(s)$ is determined as the center of mass of its valid residual predictor values, i.e.,

$$\psi_{k, u_c, M-b} = \psi_{k, u_c} - 2^{M-b} \cdot \hat{X}(s), \quad (4.5)$$

as

$$Y_{M-b}(s) = \sum_k \sum_{u_c} \psi_{k, u_c, M-b} \frac{\Pr[X(u_c) \mid \Psi_{k, u_c}]}{\sum_k \sum_{u_c} \Pr[X(u_c) \mid \Psi_{k, u_c}]} \quad (4.6)$$

In the SDUDVC codec, the side information frame given by Eq. (4.4) serves as a reconstruction of the source at the decoder, that is, $\hat{X} = Y$. However, in the extension of the codec given in Section 4.3, $Y_{M-b}$ serves as side information to the added Wyner-Ziv layer, which further encodes the $M-b$ bit-planes of the WZ frame in the transform domain.
Overlapped Block Motion Estimation and Compensation at the Decoder

Figure 4.5: Block diagram of the proposed hash-based transform-domain DVC architecture. The SDUDVC codec serves as a basis, compressing the hash information per WZ frame.
We highlight that OBMEC uses the available $b$ most significant luma bit-planes of the WZ frame to perform the estimation of the chroma components as well. In this context, for all the bit-planes of the chroma components of the WZ frames one employs the weights and predictor locations as determined for the luma component at the decoder, while the corresponding predictor values are taken from the chroma components of the reference frames, taking into account the chroma subsampling.

We note that after OBME, there may be pixels without valid predictors. These pixels were reconstructed by simply taking the mean of the surrounding reconstructed pixels. In case of multiple missing pixel values, this process is executed recursively.

### 4.3 Codec Extension To The Transform Domain

In this section, we extend over the basic architecture of Section 4.2.1 by adding a transform-domain Wyner-Ziv layer operating with a feedback channel as in state-of-the-art distributed video codecs, e.g., [12, 24, 119]. Transform-domain DVC systems exhibit increased encoding complexity with respect to pixel-domain codecs, but the overall compression efficiency increases. Furthermore, using a Wyner-Ziv layer with a feedback channel for optimal rate control boosts the RD performance.

The block diagram of the proposed transform-domain feedback-channel-based DVC (TDFDVC) system is given in Figure 4.5. Similar to the codec in Section 4.2.1, the input video is organized into GOPs, and is decomposed into key and WZ frames. The key frames are encoded using the H.264/AVC Main profile Intra frame codec, while the WZ frames are encoded in two parts, a hash layer and a WZ layer. Similar to the SDUDVC system, the hash information consists of $b$ most significant luma bit-planes of the WZ frame [see Figure 4.2(b) and (d)], which are coded as in Section 4.2.1.

In the second stage, for each WZ frame, the residual information between the original samples and the hash values, that is, $X_{M-b} = X - Q_{hash}^{-1}(\hat{X})$, is formed. This approach exploits the fact that a part of the WZ frame, i.e., the hash, has already been encoded. In order to obtain residual values greater or equal to zero, the reconstruction of the hash quantization is performed at the lower bound of the uncertainty interval, i.e., $X_{M-b} = X - \left\lfloor X/2^{M-b} \right\rfloor \cdot 2^{M-b}$. The obtained residual information is Wyner-Ziv encoded in the transform-domain forming the Wyner-Ziv layer of the codec. The implementation of the Wyner-Ziv codec is based on the Stanford architecture [12], detailed in Section 3.3.2. Specifically, the residual frame
values first undergo the $4 \times 4$ integer approximation of DCT\(^{14}\) [8]. The DCT coefficients are then grouped together into bands $\beta$ which are independently quantized with $2^{L\beta}$ levels. A uniform and a double-deadzone scalar quantizer are employed for the DC and the AC bands, respectively. Inspired by [24, 75], a set of predefined quantization matrices (QMs) is used for the transformed residual information. The selected QMs have been trained on an experimental set of several video sequences (excluding the sequences reported in the experimental results) based on the strategy presented in [75]. The employed QMs yield the best compression performance, enabling a fair comparison in a similar rate region with the state-of-the-art [24]. After quantization, in both architectures, the quantized symbols are converted to binary codewords and fed to the LDPCA encoder. The LDPCA encoder performs rate-adaptive Slepian-Wolf coding, as mentioned in Section A.3 in Appendix A.

At the decoder, the key frames are H.264/AVC Intra decoded and stored in a reference frame buffer. The hash is decoded by inverting the tasks applied at the encoder, i.e., entropy decoding and inverse prediction, and the obtained bit-planes $\hat{X}$ are stored. Next, OBMEC is applied to estimate the $M - b$ bit-planes $Y_{M-b}$ in the side-information, as detailed in Sections 4.2.2 and 4.2.3.

The residual side information frame $Y_{M-b}$ is subsequently DCT transformed, forming the side information for the Wyner-Ziv layer. After LDPCA decoding all the bit-planes of the Wyner-Ziv coded bands, optimal MMSE reconstruction [87] and inverse DCT are carried out providing the residual reconstructed frame $\hat{X}_{M-b}$, which is added back to the stored hash information, yielding the reconstructed WZ frame $\hat{X}$.

### 4.4 Hash-Based DVC Using OBMEC/SSM

By encoding a strong hash code per WZ frame, i.e., the $b$ MSBs of the frame, the TDFDVVC system is particularly advantageous when irregular motion content is coded. However, in low motion sequences, the motion can be easily predicted at the decoder and therefore, a lower hash rate would favor the compression performance. To this end, this section presents an innovative hash-based DVC (HDVC) architecture (see Figure 4.6), which reduces the required hash rate while delivering significantly improved compression efficiency over contemporary DVC systems.

\(^{14}\) Alternative decorrelating transforms, typically used in image and video coding, e.g., the wavelet transform, could also be employed in the proposed scheme. Yet, in the proposed transform-domain codecs, the DCT is chosen instead of the wavelet transform in order to comply with the state-of-the-art in conventional [8] and distributed [24] video coding.
Figure 4.6: Block diagram of the proposed hash-based DVC (HDVC) scheme.
– including our SDUDVC and TDFDVC systems in Sections 4.2.1 and 4.3. Furthermore, the proposed HDVC system involves very low computational complexity and memory usage at the encoder.

The key components of the proposed HDVC codec, are a new technique to form and compress a hash per WZ frame and a novel extension of OBMEC, referred to as OBMEC/SSM, which is specifically tailored to the employed hash method.

At the encoder, similar to our SDUDVC and TDFDVC codecs, the input video sequence is organized into GOPs and is decomposed into key frames, i.e., the first frame in each GOP, and WZ frames. The key frames, denoted by $I$, are encoded using H.264/AVC Intra frame coding, adhering to the Main profile and employing context-based adaptive binary arithmetic coding (CABAC) and rate-distortion optimization (RDO), as configured in [24]. For each WZ frame, a novel hash is sent to aid side information creation at the decoder, as detailed in Section 4.4.1. In addition to the coded hash, a WZ bit-stream is formed for each WZ frame, based on the transform-domain Wyner-Ziv (TDWZ) architecture [12].

Conversely to other hash-driven DVC schemes, e.g., [119], and our prior TDFDVC [103, 107-109] system in Section 4.3, the Wyner-Ziv encoder is chosen to encode the original WZ frame rather than its difference with the hash. In this way, the hash rate is an overhead, which is only used to assist motion estimation at the decoder. However, this approach comes with the advantage that the codec can be straightforwardly extended to the distributed joint source-channel coding (DJSCC) case, thus offering error resilience for the entire WZ frames’ waveform [12, 21].

Analogous to [24], at the encoder, the WZ frame’s pixel values are transformed using the $4 \times 4$ separable integer transform as in H.264/AVC [8], which has properties similar to the DCT. Using a set of predefined quantization matrices (QMs) – see Figure 3.7 – each DCT band, $\beta$, is independently quantized with $2^{L_{\beta}}$ levels. As in our TDFDVC and in [24], a uniform and a double-deadzone scalar quantizer are used for the DC and the AC DCT bands, respectively. After quantization, the quantization indices are converted into binary codewords and fed to the LDPCA encoder. The derived syndrome bits per codeword are stored in a buffer and a feedback channel is used to allow optimal rate control [12].

At this moment, we note that the choice of LDPCA codes to realize Slepian-Wolf coding is twofold. First, LDPCA codes have been shown to outperform Turbo codes [48] in Slepian-Wolf compression and they have been included in the state-of-the-art DISCOVER [24] codec. Second, as will be detailed in Section 5.3, LDPCA codes can be successfully applied to deal with the asymmetric nature of the novel SID correlation channel modeling scheme proposed in this dissertation. We refer to Chapter 5 for further specifics.
At the decoder, the intra frames are H.264/AVC Intra decoded and stored in a reference frame buffer. The hash information is decoded by inverting the tasks applied at the encoder. Next, the new OBMEC/SSM technique is used to generate a motion-compensated prediction of the WZ frame based on the received hash and reference frames, as detailed in Section 4.4.2. Subsequently, the produced motion-compensated frame is DCT transformed, forming the side information for the Wyner-Ziv codec.

Thereafter, per coded DCT band $\beta$ of the WZ frame, the decoder performs the proposed novel online SID correlation channel (CC) estimation algorithm, which is described in Section 5.4.2. After Slepian-Wolf decoding of all the bit-planes of the coded bands, MMSE reconstruction [87] and inverse DCT are performed, producing the reconstructed WZ frame $\hat{X}$.

Figure 4.7: Snapshots of hash frames used in the proposed HDVC relative to the hash frames in the SDUDVC and TDFDVC systems: (a), (c) The MSB of the original frames, i.e., the hash information used in the proposed SDUDVC and TDFDVC codecs; (b), (d) The MSB of the dyadically sub-sampled original frames, i.e., the hash used in the proposed HDVC system.

4.4.1 Hash Information Coding in HDVC

The proposed hash information, $\hat{X}$, consists of the MSB of the dyadically sub-sampled luma component of the original WZ frame $X$. Examples of such hash
Overlapped Block Motion Estimation and Compensation at the Decoder

frames, in relation to the corresponding hash frames in the SDUDVC and TDFDVC systems, are depicted in Figure 4.7.

To encode this information, each binary value $\tilde{X}(s)$ at position $s=(i,j)$ in the hash is first spatially predicted. The employed prediction scheme is essentially a low-complexity binary equivalent of the well-known edge-adaptive JPEG-LS predictor [137]. More specifically, each binary value $\tilde{X}(s)$ is predicted by a Boolean function $\tilde{X}'(s)=(a+b)\cdot\overline{c}+a\cdot b$, with $a$, $b$ and $c$ respectively denoting the left, top and top-left neighboring binary values in the hash, as shown in Figure 4.8. After the prediction stage, each prediction error $\tilde{X}''(s)=\tilde{X}(s)\oplus \tilde{X}'(s)$ is directly calculated using a single exclusive-or operation between the predictor $\tilde{X}'(s)$ and the predicted value $\tilde{X}(s)$. Finally, each binary symbol $\tilde{X}''(s)$ is coded using multiplication-free context-based binary arithmetic coding employing one of eight different probability models. The probability model is selected based on the neighboring local gradients $b-c$, $c-a$ and $d-b$ in the original hash $\tilde{X}$, with $d$ denoting the top-right neighbor of the predicted value $\tilde{X}(s)$ (see Figure 4.8).

![Figure 4.8: Overview of the hash formation and spatial prediction processes. On the left, gray circles with solid lines denote the sub-sampled original pixel values. On the right, dashed circles signify the MSB of the sub-sampled pixel values.](image)

Notice that the proposed hash formation and coding processes are designed in order to impose a limited complexity and memory usage overhead at the encoder. Firstly, conversely to our previous architectures [103, 107-110] in Sections 4.2 and 4.3, the hash is formed based on the sub-sampled pixel values, requiring only $\frac{1}{4}$ of the samples to be further processed. Secondly, the spatial prediction process, which forms the heart of the hash encoder, can be implemented using simple binary arithmetic, making it ideal for hardware implementation. Thirdly, in contrast to alternative solutions, e.g., [118, 119, 127, 128], the proposed technique does not perform any block-based decisions on the transmission of hash information at the encoder side. Hence, it is not burdened by the computationally expensive block-based comparisons required for such mode decision, nor does it require storing reference information from temporally adjacent frames. Fourthly, it should also be
emphasized that, contrary to our previous work [103] in Section 4.2.1, and other approaches in the literature [127], no temporal prediction is exploited by the proposed hash encoder, thereby preventing error propagation between the hash data of consecutive WZ frames.

\[ \begin{align*}
X & = 1, 0 \\
\pi & = 0, 1 \\
\kappa & = 1, 0 \\
\gamma & = 0, 1 \\
\lambda & = 0, 0 \\
\delta & = 0, 0 \\
\end{align*} \]

Figure 4.9: Creation of the sub-sampled reference frames in the OBME process.

4.4.2 Overlapped Block Motion Estimation and Compensation with Sub-Sampled Matching

In this section, we describe overlapped block motion estimation and compensation with sub-sampled matching (OBMEC/SSM), a powerful technique that generates accurate side information at the decoder based on the proposed hash. Compared to our previous bit-plane-based OBMEC [103] technique in Section 4.2, OBMEC/SSM not only operates on a different hash but also incorporates novel tools to further improve the prediction quality and reduce the computational complexity, as explained next.

4.4.2.1 Overlapped Block Motion Estimation with Sub-Sampled Reference Frames

OBMEC/SSM is performed in a hierarchical bidirectional prediction structure similar to prior DVC systems [12, 24, 107-109] (see Figure 4.3). Using two previously decoded WZ and/or key-frames as past and future reference frames, the decoder performs OBME based on the proposed hash. Specifically, let \( R_k, k \in \{0,1\} \) denote the reference frames and let \( \tilde{X} \) denote the decoded hash information. Also, let \( \tilde{R}_k, k \in \{0,1\} \) denote the MSB of the luma component of \( R_k, k \in \{0,1\} \).

Unlike our prior bit-plane-based OBMEC method in Section 4.2.2, due to the lower resolution of the hash frame \( \tilde{X} \), the values in each binary frame \( \tilde{R}_k \) are reorganized into 4 sub-sampled reference frames, i.e., \( \tilde{R}_k^{p,q}, p,q \in \{0,1\} \), by separating the values at even and odd positions in \( \tilde{R}_k \) as \( \tilde{R}_k^{p,q}(s) = \tilde{R}_k(2s+(p,q)) \) (see Figure 4.9). In this way, the newly formed binary reference frames \( \tilde{R}_k^{p,q} \) have the same resolution as the hash frame \( \tilde{X} \), hence facilitating the execution of
Overlapped Block Motion Estimation and Compensation at the Decoder

OBMEC/SSM. Next, based on $\tilde{X}$ and $\tilde{R}_k^{p,q}$, down-scaled motion vectors between the WZ frame and the reference frames are found by OBMEC/SSM. Recall that OBMEC/SSM derives more than one motion vector per pixel, thereby decreasing the energy of the prediction error compared to other methods, e.g., [24]. Also, blocking artifacts are drastically reduced, thus increasing the subjective quality of the decoded frame.

![Figure 4.10: The OBMEC/SSM method. In this example, \( \varepsilon = B/2 \). The considered overlapping blocks are also given on the right. The top-left pixel of each up-most, middle, and down-most block is specified by a circle, a triangle and a square, respectively. The left-most, middle and right-most blocks are signified by a dotted, a continuous and a dot-dashed line, respectively.](image)

The OBMEC/SSM process, a schema of which is given in Figure 4.10, proceeds as follows. Using an overlap step size $\varepsilon$, the hash frame is divided into overlapping blocks $\tilde{X}_u$ of size $B \times B$ samples, with top-left coordinates $u = (u_1, u_2)$. For each overlapping block $\tilde{X}_u$, the best matching block within a specified search range $\rho$ is found in one of the sub-sampled reference frames $\tilde{R}_k^{p,q}$. Analogously to our prior OBMEC in Section 4.2, the proposed matching criterion maximizes the number of binary values in the hash block $\tilde{X}_u$ that are identical to the co-located binary values in the best reference block $\tilde{R}_k^{p,q}$, where $v = (v_1, v_2)$ is the associated motion vector. Therefore, for each overlapping block $\tilde{X}_u$ in the hash frame, OBMEC/SSM has identified the motion vector $v$ and the indices $k, q, p$, which define the best reference block $\tilde{R}_k^{p,q}$. In addition, OBME retains the corresponding matching strength $w_u$ for the overlapping block $\tilde{X}_u$, which will be used in the compensation process. The matching strength $w_u$ is defined as the number of binary values in $\tilde{X}_u$ that are identical to the co-located binary values in the best reference block $\tilde{R}_k^{p,q}$ divided by $B^2$, i.e., the total number of samples in a block.

We remark that, unlike OBMEC in Section 4.2, due to the nature of the new hash, the matching process is carried out with binary comparisons only, thereby vastly diminishing the associated complexity. Furthermore, for the same number of
predictors per pixel and considering the same search space for each predictor, OBMEC/SSM requires one fourth of operations for each block comparison with respect to the OBMEC technique in Section 4.2. This is another factor contributing to the lower computational complexity of OBMEC/SSM as opposed to that of OBMEC.

4.4.2.2 The Compensation Method in OBMEC/SSM

To generate side information, the motion vectors derived by OBMEC/SSM are first up-sampled. By construction, there is a direct correspondence between every block $X_u$ (with top-left coordinates $u$ and size $B \times B$ pixels) in the hash frame and the block $Y_{2u}$ (with top-left coordinates $2u = (2u_1, 2u_2)$ and size $2B \times 2B$ pixels) covering the side information frame $Y$. As a result, motion vectors for the overlapping blocks covering the side information frame can simply be derived by appropriate scaling of the motion vectors associated to the overlapping blocks covering the sub-sampled hash. In detail, based on $v$ and $k, q, p$ found by OBMEC/SSM for the hash block $X_u$, an equivalent motion vector, i.e., $v' = 2v + (p, q)$, to the original reference frame $R_k$ can be derived for the side information block $Y_{2u}$. In this way, for each overlapping block $Y_{2u}$ in the side information frame, a temporal predictor block, denoted by $\Psi_{k, 2u}$, is determined in the reference frame $R_k$.

After scaling the motion vectors, side information is generated by multi-hypothesis pixel-based compensation. Specifically, similar to Section 4.2.3, each pixel in the side information frame belongs to a number of overlapping blocks $Y_{2u}$, $c = 1 \ldots C$. This means that, each pixel in the side information frame is linked to a number of predictors $\Psi_{k, 2u}$, $c = 1 \ldots C$, being the co-located pixel values in the blocks $\Psi_{k, 2u}$. Each side information pixel value is then derived by properly combining its predictors.

In contrast to our prior OBMEC [103] method in Section 4.2, the side information frame is determined separately for the even positions in the WZ frame and for the other positions. In detail, for the even pixel positions in the side information frame, the MSB of the original frame $X$ was transmitted in the hash $X$. This binary value is used to determine the weight of the predictor during compensation. Specifically, if the binary value in the hash agrees with the MSB of a predictor $\Psi_{k, 2u}$, then the predictor is said to be verified and its weight is equal to the associated matching strength $w_{eq}$. Otherwise, the predictor is categorized as unverified and its weight is empirically set to the lowest value, that is, $1/B^2$. For the other pixels in the side information frame, for which hash information is unavailable, simple averaging of their corresponding predictors is applied to derive the side information values.

We observe that after motion estimation in the OBMEC [103] technique in Section
4.2, there could be pixels without having any predictor. These pixels required an extra reconstruction phase. In OBMEC/SSM, however, all pixels are assigned a set of predictors, as explained above, thereby yielding superior pixel value reconstruction.

Lastly, we note that, similar to OBMEC in Section 4.2, OBMEC/SSM also uses the motion vectors generated by OBME to produce the chroma components of the WZ frame at the decoder, generating candidate predictors based on the chroma components of the reference frames. The weights derived for the even positions in the luma component are employed in the weighted averaging of the predictors.

4.5 OBMEC-BASED SIDE INFORMATION REFINEMENT

In Sections 4.2, 4.3 and 4.4, we presented the employment of OBMEC (or OBMEC/SSM in Section 4.4) in hash-based DVC systems. In this section, we show another way to deploy OBMEC at the decoder, yielding an efficient side information refinement technique. The proposed technique is implemented in our reference MCI-based TDWZ architecture [12] with the DISCOVER [24] modifications. We highlight that the in-house MCI-based TDWZ codec, developed in the context of this dissertation, improves over prior art [12, 24] by incorporating a novel MCI technique with overlapped block motion compensation (OBMC). Further information is given in Appendix B.

Although OBMEC-based side information refinement is codec-independent, the purpose of introducing the method in this architecture is twofold. First, in this way, except for a hash-based DVC system, we demonstrate the application of OBMEC in the most common DVC architecture [12, 24]. Second, in this fashion, we assess the performance of our own MCI-based TDWZ [102] codec. This in-house codec is later employed as a benchmark, to show the developments brought by our novel hash-based system in the specialized application of wireless capsule endoscopy.\footnote{The DISCOVER codec’s executable supports only specific frame resolutions, namely, CIF and QCIF, and therefore, it is not compatible with the resolution of the frames captured by a wireless capsule endoscope.}

The focus of this section lies on the design of a novel side information refinement algorithm. Contrary to other methods, e.g., [123, 132], the refinement is performed in the pixel-domain, alleviating the need for an overcomplete DCT transform and the corresponding computational load, which would be significant. Moreover, the proposed technique facilitates multi-hypothesis pixel-based prediction, which contrasts [121, 123, 132, 134]. Additionally, unlike [121, 134], in which side
information refinement is repeatedly carried out after decoding each DCT band, OBMEC enables a significant quality improvement of the side information by performing a *single refinement step*, that is, after decoding the DC coefficient band. In this way, the structural delay caused by repeated motion estimation at the decoder is vastly diminished.

![Diagram](image)

**Figure 4.11**: The presented MCI-based DVC architecture featuring the proposed DC-driven OBMEC-based side information refinement technique.

### 4.5.1 Codec Architecture

The block diagram of the proposed TDWZ codec with OBMEC-based side information refinement is depicted in Figure 4.11. Similar to [12, 24] and our hash-based codecs in Sections 4.2-4.4, at the proposed DVC encoder, the video frame sequence is partitioned into key and WZ frames. The key frames, $I$, are intra coded using the H.264/AVC Main profile intra frame codec. The WZ frames, $X$, are coded according to the process described in [12, 24]; namely, they undergo a $4 \times 4$ integer DCT followed by quantization, and the derived quantization indices are split into bit-planes, which are coded with an LDPCA encoder. The produced syndrome bits are stored in a bit-plane buffer and transmitted in portions upon the decoder’s request based on a feedback channel.

At the decoder, the encoded key frames are H.264/AVC intra decoded and stored in a frame buffer. Next, a new MCI technique with OBMC, detailed in Appendix B, generates the initial side information, $Y_{MCI}$, providing state-of-the-art MCI-based DVC performance. The created side information is DCT transformed and converted to soft-input information, in order to LDPCA decode the bit-planes of the WZ DC
coefficient band. To obtain the required soft-input information, our novel online SID correlation channel estimation method is performed at the decoder, as detailed in Section 5.4.2. After LDPCA decoding, the derived bit-planes are grouped into quantization indices and MMSE [87] reconstruction is carried out to obtain the decoded DC band coefficients. Then, the DCT transformed side information is updated with the decoded DC band coefficients and inverse DCT is performed. This operation yields the partially decoded WZ frame.

Thereafter, the proposed side information refinement module performs bidirectional OBMEC using the partially decoded WZ frame and reference frames from the buffer, as detailed in Section 4.5.2. In the following, this algorithm is referred to as the \textit{DC-OBMEC side information refinement} method. This operation yields an improved side information, denoted by $Y_{\text{DC-OBMEC}}$, which is used to LDPCA decode and MMSE reconstruct the remaining DCT coefficient bands of the WZ frame. Again, online correlation channel estimation is performed using our novel SID algorithm, presented in Chapter 5. Finally, after decoding all the DCT coefficient bands, the inverse DCT is performed and the decoded WZ frame is displayed and stored in the reference frame buffer.

\textbf{4.5.2 Side Information Refinement Technique}

This section details the proposed algorithm, which improves the quality of the side information after decoding the DC coefficient band of a WZ frame. The basic concept of the proposed algorithm is that after decoding the DC coefficient band, the decoder has an approximation of the WZ frame at its disposal; actually, it is aware of the low frequency component of the WZ frame. This critical information can be used to significantly improve the quality of the initial side information, which is in turn used to more efficiently decode the remaining high frequency coefficients.

The proposed side information refinement technique uses the available partially decoded WZ frame to perform motion estimation employing the previous, $R_0$, and the next, $R_1$, reference frame in a hierarchical bidirectional prediction structure (see Figure 4.3). In contrast to the MCI method described in Appendix B, in which motion estimation is blindly performed using symmetric motion vectors, the proposed refinement technique enables asymmetric bidirectional motion search. Moreover, in the proposed technique, the motion estimated blocks are considered overlapping so as to decrease the energy of the prediction error, while simultaneously reducing blocking artifacts.

Specifically, let $\tilde{X}$ be the partially decoded WZ frame after DC coefficient band decoding and reconstruction. Similar to the bit-plane-based OBME [103] method in Section 4.2.2, the WZ frame is divided into overlapping spatial blocks, of $B \times B$
pixels with an overlapping step \( \varepsilon \). For each overlapping block \( \tilde{X}_u \) in \( \tilde{X} \), the best matching block within a search range of \( \rho \) pixels, is found in each reference frame.

Notice that motion estimation is performed using the entire waveform of the partially decoded frame \( \tilde{X} \) and therefore, block matching is applied on the entire reference frames’ pixel values and not on their \( b \) MSBs (as in the bit-plane-based OBMEC in Section 4.2.2), or on the binary sub-sampled reference frames (as in OBMEC/SSM in Section 4.4.2). Furthermore, unlike the bit-plane-based OBMEC and OBMEC/SSM methods, in this case the employed block matching criterion minimizes the SAD metric. That is, the derived motion vector \( v_k \) per reference frame \( R_k, k = \{0,1\} \) is

\[
v_k = \arg \min_{v_k} \sum_s |\tilde{X}_u (s) - R_{k,u-v} (s)|,
\]

where, \( \tilde{X}_u (s) \), \( R_{k,u-v} (s) \) denote the sample values at position \( s = (i,j) \), \( 0 < i, j \leq B \), in the blocks \( \tilde{X}_u \), \( R_{k,u-v} \) in \( \tilde{X} \) and \( R_{k,u-v}, k = \{0,1\} \), respectively.

Next, per overlapping block \( \tilde{X}_u \), the proposed technique selects the best prediction direction. Namely, for every overlapping block, the algorithm retains only the motion vector (forward or backward) that provides the minimum SAD value. If both predictors have the same SAD value, then the motion vector corresponding to the minimum displacement is selected. In this way, outliers, i.e., candidate motion vectors with low reliability, are sorted out, thereby improving the consistency of the derived motion field and in turn the quality of the produced side information.

Equivalent to Sections 4.2.3 and 4.4.2.2, since the motion estimated blocks in the partially decoded WZ frame \( \tilde{X} \) are overlapping, each pixel \( \tilde{X} (s) \) in the partially decoded WZ frame belongs to several overlapping blocks \( \tilde{X}_{u_c}, c = \{1,2,...,C\} \). For each overlapping block, DC-OBMEC has identified a temporal predictor in a reference frame, denoted by \( \Psi_{k,u_c} \). Therefore, each pixel \( \tilde{X} (s) \) in the partially decoded WZ frame has a number of pixel predictors \( \psi_{k,u_c} \) from the two reference frames. This information is taken into account during compensation. In particular, the estimated value of a pixel in the refined side information frame, denoted by \( Y_{DC-OBME} (s) \), is determined as the mean value of the predictor pixel values determined by DC-OBMEC, that is,

\[
Y_{DC-OBME} (s) = \frac{1}{C_{k,u_c}} \sum_{u_c} \psi_{k,u_c},
\]

where \( C_{k,u_c} \) is the number of predictors for the refined side information pixel \( Y_{DC-OBME} (s) \).
Table 4-I: Comparison of the characteristics of the bit-plane-based OBMEC method (Section 4.2), the OBMEC/SSM technique (Section 4.4) and the DC-OBMEC side information refinement scheme (Section 4.5).

<table>
<thead>
<tr>
<th></th>
<th>Bit-plane-based OBMEC</th>
<th>OBMEC/SSM</th>
<th>DC-OBMEC</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Method</strong></td>
<td>Hash-based side information generation technique.</td>
<td>Hash-based side information generation technique.</td>
<td>Side information refinement technique.</td>
</tr>
<tr>
<td><strong>Motion search</strong></td>
<td>The $b \leq 2$ MSBs of the WZ frame (luma component).</td>
<td>The MSB of sub-sampled WZ frame (luma component).</td>
<td>The partially decoded WZ frame (luma component).</td>
</tr>
<tr>
<td><strong>(based on)</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Matching criterion</strong></td>
<td>Maximizes the 1-PER metric.</td>
<td>Minimizes the hamming distance.</td>
<td>Minimizes the SAD metric.</td>
</tr>
<tr>
<td><strong>Reference</strong></td>
<td>The $b \leq 2$ MSBs of the reference frames.</td>
<td>Binary sub-sampled reference frames.</td>
<td>Pixel values of reference frames.</td>
</tr>
<tr>
<td><strong>information for</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>motion search</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>method</strong></td>
<td>Invalid predictors: Sort out.</td>
<td>Other positions: Average of predictors.</td>
<td></td>
</tr>
<tr>
<td><strong>Output</strong></td>
<td>The $M - b$ least significant bit-planes of the side information frame.</td>
<td>The side information frame.</td>
<td>The refined side information frame.</td>
</tr>
<tr>
<td><strong>Additional</strong></td>
<td>Pixels without predictors – average of surrounding pixels.</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td><strong>concealment</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
At this point, it is useful to compare the features of the proposed novel side information generation techniques, engineered based on the innovative concept of OBMEC at the decoder. For a concise yet instructive comparison, the main characteristics of the proposed bit-plane-based OBMEC, OBMEC/SSM and DC-OBMEC techniques are tabulated in Table 4-1.

4.6 EXPERIMENTAL RESULTS

In this section, we experimentally evaluate the novel DVC solutions proposed in this chapter. Our experimental investigations are grouped in four categories. First, we examine the capacity of the bit-plane-based OBMEC method to predict a WZ frame at the decoder starting from a partial knowledge of it. Next, we assess the compression performance offered by the proposed hash-based DVC systems that employ the bit-plane-based OBMEC technique, namely, the SDUDVC codec in Section 4.2.1 and the TDFDVC codec in Section 4.3. Subsequently, we evaluate the compression performance of the proposed HDVC codec that employs our novel hash, described in Section 4.4.1, and our new OBMEC/SSM technique, detailed in Section 4.4.2. Last, we study the performance of the TDWZ codec, proposed in Section 4.5, which incorporates our novel MCI and DC-OBMEC methods.

4.6.1 Evaluation of Bit-Plane-Based OBMEC

To evaluate the performance of the proposed bit-plane-based OBMEC method to perform motion-compensated prediction at the decoder, we attempt a comparison against the most common approach to generate side information in DVC systems, that is, MCI. In Figure 4.12, we depict the per-WZ-frame quality of the side information, given in terms of the PSNR with the original frame, as produced by the in-house state-of-the-art MCI technique, detailed in Appendix B, and the bit-plane-based OBMEC in Section 4.2. The MCI algorithm was configured as in [81], i.e., a block size of 16 pixels was used, forward motion search was carried out at integer-pel accuracy within a search range of ±32 pixels and bidirectional motion estimation was executed at half-pel accuracy within a search range of ±8 pixels (see Appendix B and [81]). The OBMEC module employed a block-size of $B = 16$ pixels, a step size of $\varepsilon = 2$ and a search range of $\rho = \pm 16$ pixels and motion estimation was executed at integer-pel accuracy on the first MSB, i.e., $b = 1$. In both the evaluated techniques, the key frames in the sequence were coded using H.264/AVC Intra in Main Profile using QP=25. A GOP of size 2 was used so as to guarantee that in both methods the reference frames’ quality is identical. The experimental results, in Figure 4.12, confirm the superior capacity of the proposed OBMEC method over the
state-of-the-art MCI [102] approach in generating accurate side information in DVC. Specifically, in Carphone, a sequence with a medium amount of motion activity OBMEC brings a side information quality improvement of 0.95dB. Yet, in Soccer, a sequence with highly irregular motion content, OBMEC surpasses MCI by 4.55dB on average. These results promote the conception of hash-based methods based on OBMEC as a means to design efficient DVC systems. Note that, unlike MCI-based systems, hash-based schemes consume a part of the bit rate to communicate the hash code from the encoder to the decoder. In our example in Figure 4.12, this rate is 48.4kbps and 47.6kbps, for Carphone and Soccer, respectively. In the subsequent sections, we will compare our hash-based DVC systems against the state-of-the-art in MCI-based DVC and we will evaluate the impact of the hash on the performance.

![Figure 4.12: Quality of motion-compensated prediction per WZ frame, as obtained with MCI and OBMEC (b = 1). The key frames in the sequence were coded using H.264/AVC Intra in Main Profile using QP=25; (a) Carphone GOP2, and (b) Soccer GOP2.](image)
A visual evaluation of the performance of the bit-plane-based OBMEC module is given in Figure 4.13. The illustrated snapshots demonstrate that, even in large GOPs, OBMEC can successfully exploit the present temporal correlation to accurately reconstruct the remainder luma bit-planes and the entire chroma components starting from the most significant luma information. Furthermore, we notice that, due to employing OBMEC, the reconstructed video does not suffer from blocking artifacts.

Figure 4.13: Visual comparison for the Salesman (QCIF, 30Hz) sequence coded at 360Kbps, GOP 8, between (a) the hash information available at the decoder (2bpp), (b) the reconstructed video frame by OBMEC and (c) the original frame. OBMEC introduces only a few hardly-noticeable motion artifacts in the areas with irregular motion (fingers and mouth of Salesman).

4.6.2 Compression Results of the Hash-Based Architectures with bit-plane-based OBMEC

In this section, we assess the performance of the presented hash-based DVC architectures equipped with the proposed bit-plane-based OBMEC method. In the beginning, we evaluate the performance of the pixel-domain architecture, i.e., the SDUDVC codec, against relevant references from the literature. Subsequently, we appraise the compression efficiency of its transform-domain Wyner-Ziv extension, i.e., the TDFDVC codec.
4.6.2.1 SDUDVC Compression Results

Figure 4.14 illustrates the coding results obtained with the proposed SDUDVC codec in comparison to H.263+ Intra and Inter for the Salesman and Paris QCIF video sequences, considering different GOP sizes, i.e., 2, 4 and 8. We underline that in the DVC related literature, the H.263+ codec has been considered as the benchmark to evaluate pixel-domain DVC systems [72-75, 80, 84, 118], as well as initial transform-domain solutions [11, 12, 23, 81].

The H.263+ results were obtained with the TMN 3.2.0 software operating in Baseline Profile. Regarding the proposed codec, the key frames were encoded using the H.263+ Intra codec while, from the WZ frames, the MSB was extracted and differentially encoded. At the decoder, the OBMEC module was set up as mentioned in Section 4.6.1.

![Graphs](image-url)  

\(\text{Figure 4.14: Compression performance comparison of the proposed SDUDVC codec with the H.263+ codec for different GOP sizes; (a) Salesman QCIF and (b) Paris QCIF, at 15Hz. Only the luma component is encoded.}\)
The results show that the proposed SDUDVC codec outperforms H.263+ Intra bringing a performance improvement, which increases with the GOP size. In particular, we observe a PSNR gain of up to 6dB in Paris and 6.5dB in Salesman, at a GOP of size 8. These results demonstrate that the available temporal correlation in both Salesman and Paris can effectively be captured by our OBMEC method even in large GOPs. However, SDUDVC falls behind H.263+ Inter coding by 0.7dB in Paris and 0.5dB in Salesman, at GOP 2. This lack in performance with respect to traditional inter-frame coding, which grows larger with the GOP size and the rate, is well-known. Recall from Section 2.6.1 that Wyner-Ziv coding suffers a performance loss with respect to traditional predictive coding, due to the absence of side information at the encoder.

Figure 4.15: Compression performance of the proposed SDUDVC, the IST-PDWZ [85] and the H.263+ codecs for GOP4; the luma components from the first 101 frames of (a) Coastguard and (b) Foreman, QCIF, both at 30Hz were encoded.
In the second set of our experiments, the proposed SDUDVC is compared against the best performing feedback-channel-based pixel-domain DVC system in the literature, i.e., the IST-PDWZ codec in [85]. The latter embodies similar compression features and settings to those of DISCOVER, but operates in the pixel domain. The IST-PDWZ results were reproduced from [85], while the settings of the H.263+ and our codec were kept unchanged from the previous experiment. Comparisons were made for the first 101 frames of the Coastguard and Foreman sequences at 30Hz and considering a GOP size of 4. The results, given in Figure 4.15, show that apart from low rates where both systems exhibit similar RD performance, the proposed codec clearly outperforms the IST-PDWZ codec, introducing a considerable gain of up to approximately 1dB. At the same time, the proposed scheme retains the important advantage of being liberated from the use of a feedback channel.

4.6.2.2 Compression Results of the TDFDVC System

Continuing our experimentation, the proposed hash-based TDFDVC codec is compared against a collection of state-of-the-art distributed and traditional video codecs, namely, DISCOVER [24], the SDUDVC codec evaluated above, and H.264/AVC Intra and No Motion [8]. Comparative tests were carried out on the complete Foreman, Soccer, Carphone and Silent test video sequences, at QCIF resolution and at a frame rate of 15Hz. These sequences contain a variety of object and camera motion characteristics. Specifically, Silent contains a low amount of motion and includes a constant background. Carphone and Foreman are characterized by a reasonable amount of motion activity, whereas Soccer contains fast motion and complex camera movements. All sequences were coded using a GOP of size 2, 4 and 8 frames.

The configuration of the OBMEC component was kept unchanged with respect to our prior experiments (i.e., \( \varepsilon = 2 \), \( B = 16 \) and \( \rho = \pm 16 \)). However, the number of MSBs composing the hash was put to \( b = 2 \). The quantization parameters (QPs) of the H.264/AVC Intra coded key frames were properly tuned as to minimize the quality fluctuation over the entire sequence – as in [24]. Although chroma (i.e., YUV) encoding is supported, results are presented only for the luma (Y) component to allow a meaningful comparison with prior art [24].

Figure 4.16 through Figure 4.19 illustrate the RD behavior of the proposed hash-based TDFDVC system\(^{16}\). Initially, TDFDVC is compared against the basic pixel-domain architecture, namely, the SDUDVC scheme. The SDUDVC scheme retained

\(^{16}\) For a concise presentation of the experimental results, the figures also include the performance of the proposed HDVC scheme, which will be commented in the next section.
in the experiments was configured using a single bit-plane hash, and employed H.264/AVC Intra to code the key frames. Although lacking a transform-domain Wyner-Ziv layer and avoiding the use of a feedback channel, SDUDVC delivers decent RD performance, especially at low and medium rates, at sequences with low and medium motion activity and at short GOPs. Yet, the results, in particular the ones depicted in Figure 4.16 and Figure 4.17, clearly show the dramatic increase in compression efficiency by adding a transform domain Wyner-Ziv layer. Moreover, due to the additional Wyner-Ziv layer, TDFDVC can achieve a wider range of rates compared to SDUDVC. In essence, the proposed TDFDVC codec substantially advances over SDUDVC, yielding gains of up to 29.58% and 43.85% in terms of Bjøntegaard Delta (BD) [138] rate reduction for Foreman and Soccer, GOP8, respectively.

Furthermore, the coding performance of the proposed codec is assessed against the DISCOVER [24] codec. In Section 3.3.3, we have mentioned that DISCOVER represents the state-of-the-art in transform-domain DVC and constitutes an established reference in the DVC related literature. We remark that DISCOVER is an MCI-based DVC system, encompassing a state-of-the-art MCI technique to generate side information [24, 80-82]. The experimental results, depicted in Figure 4.16 to Figure 4.19, show that in sequences containing medium to high motion activity, i.e., Foreman (GOP4 and GOP8), Soccer and Carphone (GOP4 and GOP8), the proposed TDFDVC codec outperforms DISCOVER. The reported performance gains increase with the amount of motion in the sequence and the length of the considered GOP. Specifically, in Carphone and Foreman, GOP8, TDFDVC introduces respective BD rate savings of 3.43% and 15.10% over DISCOVER. The largest compression gains, which are obtained on the Soccer sequence at a GOP of 8, are of up to 24.55% in BD rate savings. These results, which are in line with the ones in Figure 4.12, demonstrate the superior motion capturing power of the proposed hash-based OBMEC compared to the MCI method of DISCOVER, in particular when the GOP size increases, which undermines the linear motion assumption of MCI.

On the other hand, the results in Figure 4.19 show that the TDFDVC codec falls behind DISCOVER in Silent. The observed performance loss reduces with the GOP, translating to a Bjøntegaard [138] rate penalty of 8.20% in Silent GOP8. This performance deficiency is due to the fact that Silent is a sequence with low motion characteristics, in which the MCI method succeeds in accurately capturing the motion. Conversely, as explained in Section 4.6.1, the hash-based nature of the presented TDFDVC system inherently causes a rate increase because the hash must be conveyed to the decoder. Such supplementary rate is not present in MCI-based
Figure 4.16: Compression performance of the proposed hash-based codecs in the Foreman QCIF sequence at 15Hz; (a) GOP2, (b) GOP4, and (c) GOP8.
Figure 4.17: Compression performance of the proposed hash-based codecs in the Soccer QCIF sequence at 15Hz; (a) GOP2, (b) GOP4, and (c) GOP8.
Overlapped Block Motion Estimation and Compensation at the Decoder

Figure 4.18: Compression performance of the proposed hash-based codecs in the Carphone QCIF sequence at 15Hz; (a) GOP2, (b) GOP4, and (c) GOP8.
Figure 4.19: Compression performance of the proposed hash-based codecs in the Silent QCIF sequence at 15Hz; (a) GOP2, (b) GOP4, and (c) GOP8.
systems, which do not require a hash to be sent in order to perform motion estimation.

In general, the experimental results in Figure 4.16 to Figure 4.19 demonstrate that, by using the hash, the proposed codec manages to raise the decoded quality, to such an extent that its RD performance surpasses the performance of the DISCOVER codec when the sequences contain medium to high motion, i.e., Carphone, Foreman and Soccer (in the order of increasing motion complexity). Nevertheless, in sequences with low-complex motion, e.g., Silent, the cost of the coded hash cannot be balanced by the improvement in the PSNR of the decoded sequence, thereby diminishing the overall compression performance. As explained in Section 4.4, this issue is resolved by our novel HDVC system, which vastly reduces the required hash rate overhead, while maintaining high quality side information. The experimental evaluation of the HDVC system is given in Section 4.6.3.

Lastly, the presented TDFDVC codec is compared against two conventional low-complexity predictive codecs, namely, the H.264/AVC Intra and the H.264/AVC Inter codec without performing motion estimation (No Motion). H.264/AVC Intra is considered one of the most efficient intra-frame coding schemes and usually acts as a benchmark to evaluate transform-domain DVC systems. H.264/AVC No Motion enables partial temporal redundancy removal via simple differential coding principles. Note that both H.264/AVC configurations exhibit higher encoding complexity than the proposed TDFDVC solution, since they feature loop filtering, intra prediction and mode decision. When irregular motion content is coded, e.g., Foreman, Soccer and Carphone, it is well known that both H.264/AVC Intra and No Motion are very efficient compared to DVC solutions, as corroborated in Figure 4.16, Figure 4.17, and Figure 4.18. Nevertheless, the proposed hash-based system notably reduces the performance mismatch between the DISCOVER codec and both H.264/AVC configurations. Yet, when low-motion video content is encountered, the proposed codec outperforms H.264/AVC Intra, introducing BD rate savings of, for example, 22.36% in Silent GOP8.

4.6.3 Compression Results of the HDVC System

In the following, we compare the proposed HDVC system against several relevant state-of-the-art codecs, namely, our previous SDUDVC and TDFDVC codecs, DISCOVER [24], H.264/AVC Intra and No Motion [8], and the hash-based codec in [119].

For consistency, we maintain the test conditions introduced in Section 4.6.2.2. That is, tests were carried out for all frames of Foreman, Soccer, Carphone, and Silent.
sequences, at QCIF resolution, a frame rate of 15Hz, and GOP sizes of 2, 4 and 8. As mentioned in Section 4.6.2.2, these sequences exhibit a variety of object and camera motion characteristics. Furthermore, similar to Section 4.6.2.2, to assess the coding performance in terms of the Bjøntegaard Delta (BD) [138] metric, four RD points have been drawn corresponding to QMs 1, 5, 7 and 8 of [24] (see Figure 3.7). The QPs of the H.264/AVC Intra encoder are carefully selected in order to maintain a constant decoding quality.

In the OBMEC/SSM module, an overlap step size of $\varepsilon = 1$, a search range of $\rho = 8$ pixels and a block size $B = 8$ are employed. We observe that these values define the same number of predictors per pixel and the same search space for OBMEC/SSM, with the corresponding values to configure OBMEC in Section 4.6.1. Like so, we ensure a fair and meaningful comparison between our bit-plane OBMEC in Section 4.2 and the proposed OBMEC/SSM method in Section 4.4.2.

Figure 4.16 through Figure 4.19 depict the RD performance of the proposed HDVC codec, equipped with online SID estimation, against a relevant set of state-of-the-art low-cost video encoding schemes. The compression results, expressed in Bjøntegaard [138] rate (%) and relative PSNR (dB) deltas with respect to the benchmark codecs, are also summarized in Table 4-II.

Initially, the proposed codec is compared against our previous SDUDVC [103] scheme presented in Section 4.2.1. The results, tabulated in Table 4-II, reveal that the proposed codec substantially advances over SDUDVC [103], yielding gains of up to 43.33% and 45.53% in BD rate reduction for Foreman and Soccer, GOP8, respectively. This is because HDVC operates in the transform-domain and employs a feedback channel for optimal rate control, components which are not included in SDUDVC.

Subsequently, the proposed HDVC is evaluated against our hash-based TDFDVC codec detailed in Section 4.3. The experimental results, illustrated in Figure 4.16 through Figure 4.19 and tabulated in Table 4-II, show that HDVC systematically outperforms our previous TDFDVC system. The introduced compression improvements are more significant in sequences with low and medium motion activity because of the considerable amount of hash information coded by TDFDVC. Furthermore, the gains increase with the size of the considered GOP as more WZ frames are coded. Specifically, according to Table 4-II, HDVC advances over TDFDVC by bringing BD rate savings of to 7.02%, 12.05%, 14.10% and 18.16% in Soccer, Silent, Carphone and Foreman, GOP8, respectively.
Table 4-II: Bjøntegaard deltas [138] obtained using the proposed HDVC system against various state-of-the-art DVC systems. Negative rate and positive PSNR delta values represent gains. The compression results of their codec on Carphone and Silent have not been provided by Ascenso et al. in [119].

<table>
<thead>
<tr>
<th></th>
<th>GOP2</th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>ΔR(%)</td>
<td>ΔPSNR(dB)</td>
<td>ΔR(%)</td>
<td>ΔPSNR(dB)</td>
<td>ΔR(%)</td>
<td>ΔPSNR(dB)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Foreman</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>vs. SDUDVC</td>
<td>-19.23</td>
<td>1.388</td>
<td>-25.74</td>
<td>1.630</td>
<td>-43.33</td>
<td>2.926</td>
<td></td>
<td></td>
</tr>
<tr>
<td>vs. TDFDVC</td>
<td>-7.01</td>
<td>0.412</td>
<td>-11.63</td>
<td>0.680</td>
<td>-18.16</td>
<td>1.087</td>
<td></td>
<td></td>
</tr>
<tr>
<td>vs. DISCOVER</td>
<td>-3.66</td>
<td>0.240</td>
<td>-18.79</td>
<td>1.167</td>
<td>-31.41</td>
<td>2.078</td>
<td></td>
<td></td>
</tr>
<tr>
<td>vs. TDWZ</td>
<td>4.13</td>
<td>-0.210</td>
<td>-12.58</td>
<td>0.753</td>
<td>-24.86</td>
<td>1.577</td>
<td></td>
<td></td>
</tr>
<tr>
<td>vs. [119]</td>
<td>-2.54</td>
<td>0.168</td>
<td>-14.46</td>
<td>0.896</td>
<td>-24.93</td>
<td>1.581</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Soccer</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>vs. SDUDVC</td>
<td>-28.88</td>
<td>1.653</td>
<td>-34.40</td>
<td>1.991</td>
<td>-45.53</td>
<td>2.655</td>
<td></td>
<td></td>
</tr>
<tr>
<td>vs. TDFDVC</td>
<td>-2.16</td>
<td>0.085</td>
<td>-6.59</td>
<td>0.326</td>
<td>-7.02</td>
<td>0.347</td>
<td></td>
<td></td>
</tr>
<tr>
<td>vs. DISCOVER</td>
<td>-12.49</td>
<td>0.700</td>
<td>-24.77</td>
<td>1.500</td>
<td>-30.95</td>
<td>2.014</td>
<td></td>
<td></td>
</tr>
<tr>
<td>vs. TDWZ</td>
<td>-11.75</td>
<td>0.631</td>
<td>-22.77</td>
<td>1.334</td>
<td>-25.39</td>
<td>1.578</td>
<td></td>
<td></td>
</tr>
<tr>
<td>vs. [119]</td>
<td>-2.96</td>
<td>0.170</td>
<td>-9.15</td>
<td>0.503</td>
<td>-10.65</td>
<td>0.626</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Carphone</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>vs. SDUDVC</td>
<td>-9.26</td>
<td>0.661</td>
<td>-21.93</td>
<td>1.384</td>
<td>-42.58</td>
<td>2.399</td>
<td></td>
<td></td>
</tr>
<tr>
<td>vs. TDFDVC</td>
<td>-8.68</td>
<td>0.554</td>
<td>-9.77</td>
<td>0.552</td>
<td>-14.10</td>
<td>0.796</td>
<td></td>
<td></td>
</tr>
<tr>
<td>vs. DISCOVER</td>
<td>-3.70</td>
<td>0.251</td>
<td>-8.11</td>
<td>0.487</td>
<td>-16.59</td>
<td>0.948</td>
<td></td>
<td></td>
</tr>
<tr>
<td>vs. TDWZ</td>
<td>-0.44</td>
<td>0.052</td>
<td>-6.11</td>
<td>0.357</td>
<td>-13.89</td>
<td>0.754</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Silent</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>vs. SDUDVC</td>
<td>-9.63</td>
<td>0.618</td>
<td>-7.02</td>
<td>0.437</td>
<td>-12.28</td>
<td>0.694</td>
<td></td>
<td></td>
</tr>
<tr>
<td>vs. TDFDVC</td>
<td>-7.38</td>
<td>0.478</td>
<td>-10.24</td>
<td>0.631</td>
<td>-12.05</td>
<td>0.697</td>
<td></td>
<td></td>
</tr>
<tr>
<td>vs. DISCOVER</td>
<td>1.90</td>
<td>-0.067</td>
<td>2.21</td>
<td>0.003</td>
<td>-2.48</td>
<td>0.269</td>
<td></td>
<td></td>
</tr>
<tr>
<td>vs. TDWZ</td>
<td>5.34</td>
<td>-0.305</td>
<td>1.98</td>
<td>-0.059</td>
<td>-0.62</td>
<td>0.116</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

The obtained gains are attributed to the novel components of the HDVC system. Namely, the OBMEC/SSM technique, presented in Section 4.4.2, significantly improves over our previous OBMEC method in Sections 4.2.2 and 4.2.3. Experimental results depicted in Figure 4.20, show that, although using only a quarter of the hash information needed by OBMEC (configured with $b=1$), OBMEC/SSM delivers on average 0.4dB and 0.8dB higher side information quality (calculated with respect to the original uncoded frames) in Carphone and Soccer, GOP2, respectively. What is more, the proposed hash codec in Section 4.4.1, significantly reduces the required hash coding rate over our previous approach in Section 4.2.1, while not using temporal prediction and being less complex. In particular, in Figure 4.20, the hash rate for OBMEC/SSM using the proposed codec is 12.3kbps and 10.6kbps for Carphone and Soccer, respectively. Conversely, the
hash rate for performing OBMEC (configured with $b = 1$) using the codec discussed in Section 4.2.1 is significantly higher, i.e., 48.4kbps and 47.6kbps, for Carphone and Soccer, respectively.

![Graph](image)

Figure 4.20: Side information quality for (a) Carphone, and (b) Soccer at GOP2. The dyadically sub-sampled MSB is used by OBMEC/SSM, while the entire MSB is needed to perform OBMEC. The intra frames’ quality is identical for both methods (QP=25).

To further investigate the improvements due to the proposed hash formation and coding approaches in Section 4.4.1, Table 4-III reports the required rate to code the hash in the proposed HDVC system and its relative reduction with respect to the hash rate required by the TDFDVC codec, for the all sequences and GOPs. One notices that the proposed approach efficiently compresses the hash, diminishing the required hash coding overhead. We highlight that, compared to the TDFDVC system, the proposed codec diminishes the required hash rate by up to 78% in
Carphone, while improving the overall RD performance by 0.80dB in BD PSNR sense (see Table 4-II). We also remark that, since it performs binary edge adaptive prediction, the higher the intra correlation, the more efficient our hash coding technique becomes. As a consequence, the highest compression ratio is achieved for Soccer, while the lowest is obtained in Silent, i.e., 22% and 43% compression with respect to the uncoded sub-sampled MSB, respectively. This is also the reason why in Silent we obtain the least reduction in hash rate compared to our previous approach in Section 4.2.1, which exploits temporal correlation to code the hash.

Table 4-III: Hash coding rate (in kbps) of the HDVC codec and the percentage reduction of the hash rate with respect to the one in the TDFDVC codec for various GOPs.

<table>
<thead>
<tr>
<th>Sequence</th>
<th>GOP 2</th>
<th>GOP 4</th>
<th>GOP 8</th>
</tr>
</thead>
<tbody>
<tr>
<td>Foreman</td>
<td>14.99 (76.9%)</td>
<td>22.48 (76.3%)</td>
<td>26.22 (77.6%)</td>
</tr>
<tr>
<td>Soccer</td>
<td>10.60 (78.1%)</td>
<td>15.84 (78.2%)</td>
<td>18.52 (78.1%)</td>
</tr>
<tr>
<td>Carphone</td>
<td>12.30 (77.4%)</td>
<td>18.42 (76.7%)</td>
<td>21.52 (78.0%)</td>
</tr>
<tr>
<td>Silent</td>
<td>20.28 (49.6%)</td>
<td>30.43 (48.4%)</td>
<td>35.51 (51%)</td>
</tr>
</tbody>
</table>

Furthermore, the proposed codec is assessed against the state-of-the-art DISCOVER codec [24]. The RD performance of DISCOVER [24] has been obtained using the codec’s executable, as found on the DISCOVER website [24]. The experimental results, illustrated in Figure 4.16 through Figure 4.19 and tabulated in Table 4-II, show that the proposed codec generally outperforms DISCOVER. The reported gains, which are notable, are increasing with the GOP size and the amount of motion in a sequence. Only in Silent at low rates DISCOVER slightly surpasses the proposed codec, since for low motion content, MCI can deliver a good prediction. OBMEC/SSM requires additional hash rate, which is considerable at low rates in Silent due to low spatial correlation, thereby explaining this performance difference. Overall, in sequences with low to medium amount of motion activity, the proposed codec yields compression improvements of up to 2.48% and 16.59% BD rate reduction in Silent and Carphone, GOP8, respectively. However, when irregular motion content is coded, the gains brought by the proposed codec against DISCOVER, increase up to 31.41% and 30.95% BD rate reduction in Foreman and Soccer, GOP8, respectively. A similar behavior can be observed when the proposed HDVC codec is compared against our previous TDWZ codec – without DC-OBMEC.
side information refinement\(^{17}\) – presented in Section 4.5.1. However, due to the fact that our TDWZ surpasses DISCOVER, the gains introduced by the proposed HDVC system over the former are less, as corroborated in Table 4-II. In general, the RD improvements brought by HDVC over DISCOVER and our MCI-based TDWZ system highlight the capacity of OBMEC/SSM in capturing difficult motion even in large GOPs, conditions under which MCI falls short in providing accurate prediction [25, 89].

To assess the performance of the proposed scheme against that of the latest hash-based DVC prior art, the coding results of the system in [119] are also tabulated in Table 4-II. The latter creates side information by combining MCI with hash-based motion estimation using low quality intra coded WZ frame blocks. Although, this system outperforms DISCOVER, it is significantly falling behind the proposed HDVC scheme. In particular, the proposed codec improves over the system in [119] by 24.93\% and 10.65\% BD rate reduction in Foreman and Soccer, GOP8, respectively\(^{18}\). These performance gains indicate the superior quality of side information generated by OBMEC/SSM in comparison to the state-of-the-art hash-driven approach of [119].

In Figure 4.16 through Figure 4.19, we also report the performance of H.264/AVC Intra and H.264/AVC No Motion. Recall that the assessed H.264/AVC codecs are significantly more complex than the proposed DVC solution. The results show that, similarly to DISCOVER, the proposed scheme outperforms H.264/AVC Intra for low motion sequences at all GOPs, e.g., gains of up to 31.89\% BD rate reduction in Silent GOP8 are achieved. Contrary to the other DVC codecs, however, the proposed scheme manages to partially outperform H.264/AVC Intra in Foreman and Carphone at all the evaluated GOP sizes, and H.264/AVC No Motion in Foreman GOP4 and GOP8. In Soccer, a sequence comprising very irregular motion content, the proposed codec significantly diminishes the performance gap of DVC with respect to H.264/AVC Intra and No Motion.

Table 4-IV reports the percentages of the H.264/AVC Intra, the hash and the LDPCA rate in the total rate of the proposed codec. Results are synopsized for the two extreme RD points of Foreman and Carphone, GOP of 2, 4, and 8. We observe that the intra and the WZ (i.e., hash and LDPCA) rate percentages agree with the portion of key and WZ frames in a GOP. Note that, for a given sequence and GOP size, the hash rate is unvarying and hence, its percentage decreases with increasing

\(^{17}\) Our TDWZ system is configured without DC-OBMEC refinement as this is not employed by HDVC either. This ensures a fair and meaningful comparison.

\(^{18}\) Ascenso et al. have not reported the compression performance of their codec in [119] on the Carphone and Silent sequences.
rate. The proposed codec efficiently compresses the hash, diminishing the required hash rate overhead. Recall that the hash is coded in order to enable the creation of accurate side information at the decoder, and thus to increase the Wyner-Ziv coding performance. Table 4-IV also reports the relative standard deviation (RSD) of the decoded frames’ PSNR (both key and WZ), given by \( \text{RSD}(\%) = 100 \times \frac{\sigma_{PSNR}}{\mu_{PSNR}} \), where \( \sigma_{PSNR} \) is the standard-deviation and \( \mu_{PSNR} \) is the mean of the PSNR values. The very low RSD values show that the proposed codec yields a quasi-constant decoding quality. Bearing in mind that the number of LDPCA coded bit-planes per RD point is the same with prior art [24], one concludes that the proposed novel techniques, i.e., hash-based OBMEC/SSM (see Sections 4.4.1 and 4.4.2) and online SID correlation channel estimation (see Section 5.4.2), significantly increase the efficiency of Wyner-Ziv coding.

Table 4-IV: Percentage contribution in the total bit-rate, and the RSD of the decoded sequence for the proposed system.

<table>
<thead>
<tr>
<th>GOP</th>
<th>RD point</th>
<th>Intra (%)</th>
<th>Hash (%)</th>
<th>LDPCA (%)</th>
<th>RSD (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Foreman</td>
<td>2</td>
<td>53.14</td>
<td>17.11</td>
<td>29.75</td>
<td>2.77</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>47.49</td>
<td>3.08</td>
<td>49.43</td>
<td>1.85</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>28.69</td>
<td>26.19</td>
<td>45.12</td>
<td>2.64</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>23.46</td>
<td>4.35</td>
<td>72.19</td>
<td>2.06</td>
</tr>
<tr>
<td></td>
<td>8</td>
<td>16.38</td>
<td>30.83</td>
<td>52.79</td>
<td>2.69</td>
</tr>
<tr>
<td></td>
<td>8</td>
<td>12.44</td>
<td>4.89</td>
<td>82.67</td>
<td>2.33</td>
</tr>
<tr>
<td>Carphone</td>
<td>2</td>
<td>55.19</td>
<td>14.90</td>
<td>29.90</td>
<td>2.71</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>48.41</td>
<td>2.93</td>
<td>48.65</td>
<td>1.83</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>30.97</td>
<td>23.71</td>
<td>45.32</td>
<td>2.56</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>24.31</td>
<td>4.22</td>
<td>71.47</td>
<td>2.04</td>
</tr>
<tr>
<td></td>
<td>8</td>
<td>17.99</td>
<td>28.58</td>
<td>53.43</td>
<td>2.61</td>
</tr>
<tr>
<td></td>
<td>8</td>
<td>12.85</td>
<td>4.79</td>
<td>82.37</td>
<td>2.31</td>
</tr>
</tbody>
</table>

4.6.4 Assessment of OBMEC-Based Side Information Refinement

This section evaluates the performance of the proposed TDWZ codec equipped with our novel MCI (see Appendix B) and DC-OBMEC (see Section 4.5.2) techniques to generate side information at the decoder. The experimental results are presented in two stages. At first, we evaluate the performance of the proposed TDWZ system using our MCI method, detailed in Appendix B. Next, we assess the compression gains brought by the proposed OBMEC-based side information refinement method, described in Section 4.5.2.

To appraise the coding efficiency of the proposed DVC codecs, various relevant state-of-the-art codecs have been selected as benchmarks, namely, DISCOVER [24], H.264/AVC Intra [8] and the successively refined TDWZ systems found in [121, 133]. As in the previous sections, the test conditions include all frames of the
Figure 4.21: Compression performance of the proposed MCI-based TDWZ codec with and without DC-OBME refinement in the Foreman QCIF sequence at 15Hz: (a) GOP2, (b) GOP4, and (c) GOP8.
Figure 4.22: Compression performance of the proposed MCI-based TDWZ codec with and without DC-OBME refinement in the Soccer QCIF sequence at 15Hz; (a) GOP2, (b) GOP4, and (c) GOP8.
Figure 4.23: Compression performance of the proposed MCI-based TDWZ codec with and without DC-OBME refinement in the Carphone QCIF sequence at 15Hz; (a) GOP2, (b) GOP4, and (c) GOP8.
Figure 4.24: Compression performance of the proposed MCI-based TDWZ codec with and without DC-OBME refinement in the Silent QCIF sequence at 15Hz; (a) GOP2, (b) GOP4, and (c) GOP8.
Foreman, Soccer, Carphone and Silent sequences, at QCIF resolution, at a frame rate of 15Hz, and at a GOP of size 2, 4 and 8.

Regarding the presented MCI-based system’s setup, the MCI algorithm was configured exactly as in Section 4.6.1. In the side information refinement method, OBMEC was assigned an overlap step size of $\varepsilon = 4$, the size of the overlapping blocks was set to $B = 16$ and the motion search was performed in an exhaustive manner at integer-pel accuracy within a search range of $\pm 20$ pixels. The QPs of the H.264/AVC Intra coded frames were exactly the same with the ones given in [24].

In the beginning, we evaluate the performance of the proposed TDWZ codec without the refinement method. The experimental results, depicted in Figure 4.21 to Figure 4.24, show that the in-house MCI-based TDWZ codec delivers higher compression performance than the DISCOVER codec. The corresponding Bjøntegaard Deltas [138] indicate that the developed TDWZ codec yields average BD rate savings of 8.45%, 6.66%, 3.89% and 2.74% in Foreman, Soccer, Carphone and Silent, GOP8, respectively. Since both codecs employ the same H.264/AVC Intra frame codec and the same QPs, one concludes that the presented MCI-based TDWZ codec performs more efficient Wyner-Ziv coding compared to DISCOVER. This performance gain is attributed to the employed MCI method, which in contrast to [80, 81], comprises bidirectional OBMC. For further details we refer to Appendix B.

In the following, we evaluate the increase in the side information quality due to the employment of the proposed DC-OBME side information refinement method, presented in Section 4.5.2. Figure 4.25 plots the PSNR of the initial side information, produced by means of MCI (see Appendix B), with the one obtained after DC-OBME refinement, for every WZ frame in Carphone GOP2 and Soccer GOP8. The results show that the higher the GOP size and the amount of motion in the sequence, the higher the improvement in the quality of side information brought by the proposed DC-OBME refinement method. This is anticipated, due to the fact that when the sequence contains irregular motion and when the distance between the reference frames increases, blind motion estimation at the decoder by means of MCI is deficient. In essence, the results in Figure 4.25 show that the side information quality obtained with the DC-OBME refinement technique advances over the one produced by MCI by 2.37dB and 6.26dB on average in Carphone, GOP2 and Soccer, GOP8, respectively.

Subsequently, we quantify the compression gains obtained by this improvement in the side information quality offered by the DC-OBME method. The experimental results in Figure 4.21 to Figure 4.24, show that our TDWZ codec equipped with DC-OBME refinement consistently yields the best RD performance among the evaluated DVC codecs for all sequences and GOPs. Compared to our MCI-based TDWZ codec,
the proposed codec with DC-OBME brings BD savings of up to 22.88%, 24.60%, 16.40% and 17.83% in Foreman, Soccer, Carphone and Silent, GOP8, respectively. When compared to the state-of-the-art DISCOVER codec, the presented TDWZ with DC-OBME yields respective BD rate reductions of up to 30.07%, 30.51%, 19.87% and 18.95%. We observe that the obtained (significant) gains increase with the amount of irregular motion in the coded sequence and with the length of the considered GOP. We also notice that, due to the application of pixel-based multi-hypothesis prediction in the DC-OBME method, the obtained compression gains are systematic across all rates. This contrasts with other works in the literature, e.g., the codec of Martins et al. [121], in which compression gains are mainly obtained in medium and high rates.

![Graph A](image1.png)

**(a)**

![Graph B](image2.png)

**(b)**

*Figure 4.25: Side information quality before and after DC-OBME refinement for (a) Carphone GOP2, and (b) Soccer GOP8. The key frames quality is the same for both methods (QP=25).*
When compared against H.264/AVC [8] Intra the proposed codecs with and without refinement obtain higher RD performance in low-motion sequences, bringing respective BD rate savings of up to 30.28% and 41.98% in Silent. In Foreman and Carphone, the proposed codec (with refinement) outperforms H.264/AVC Intra only at low to medium rates. In Soccer, a sequence with very irregular motion characteristics, the proposed codec diminishes the performance gap of DVC with respect to H.264/AVC Intra. Recall, however, that H.264/AVC Intra inflicts much higher computational complexity at the encoder than the proposed codec, which is a tailback for low-power applications.

Finally, we endeavor a comparison against alternative state-of-the-art side information refinement techniques available in the literature. In particular, the proposed DC-OBME technique outperforms the method of Ye et al. [133] by up to 20.73% and 21.12% in DB rate reduction in Foreman and Soccer, GOP8, respectively. Furthermore, compared to the method of Martins et al. [121], the proposed technique brings average BD rate savings of up to 14.94% and 15.03% in Foreman and Soccer, GOP8, respectively. We highlight that, contrary to the proposed DC-OBME technique, which refines the side information only once, the work in [121] proposes side information refinement upon decoding the coefficients of each DCT band in the WZ frame. Namely, in the highest RD point, corresponding to QM8 in Figure 3.7, the codec in [121] performs motion estimation for 15(!) times per WZ frame. These results highlight that, by using OBMEC, the proposed side information refinement delivers superior compression performance with respect to alternative methods, e.g., [121], while vastly confining the inflicted structural latency.

4.7 CONCLUSIONS

Aiming at the development of efficient distributed video coding schemes, this chapter has concentrated on the problem of effective side information generation. Intrinsically, we have proposed OBMEC, an innovative concept that allows for multi-hypothesis pixel-based motion-compensated prediction at the decoder. Conversely to MCI techniques, the proposed OBMEC-based methods extend the motion information during motion compensation to more than one motion vector per block, thereby producing side information of higher quality, especially under difficult motion conditions, namely, irregular motion content and large GOPs. An additional benefit of our OBMEC-based techniques versus alternative approaches in the literature, e.g., [80, 81, 119], is that it reduces blocking artifacts, thus having a positive impact on the subjective quality of the decoded sequence.

Different OBMEC versions have been coupled with several novel DVC
architectures featuring various advantages and yielding state-of-the-art compression performance. The characteristics and the performance of the proposed OBMEC-driven systems are summarized in Table 4-V.

Firstly, by encoding a hash code consisting of the $b$ MSBs of the luma component of a WZ frame, our new bit-plane-based OBMEC stimulates the design of a novel codec that exhibits very low-encoding complexity and does not employ Slepian-Wolf codes. The proposed SDUDVC system operates in the pixel-domain and, most importantly, it does not require a feedback channel. Experimental results have shown that the proposed SDUDVC compares favorably against the state-of-the-art feedback-channel-based pixel-domain system in [85], introducing an average gain of approximately 1dB.

Secondly, including a transform-domain Wyner-Ziv coding layer with a feedback channel has brought about a novel hash-based Wyner-Ziv video coding system. Although the proposed TDFDVC system maintains the basic hash codec of SDUDVC, it improves upon the latter by operating in the transform-domain and by employing LDPCA codes with a feedback channel. Specifically, meticulous experimentation has confirmed that the proposed TDFDVC system drastically advances over SDUDVC, typically yielding superior RD performance with respect to the state-of-the-art DISCOVER codec. The experimental results have shown that, for motion activity characterized as medium to high, our hash-based TDFDVC system outperforms DISCOVER by up to 24.55% in BD rate savings. This significant improvement in the compression efficiency is credited to the capacity of the proposed bit-plane-based OBMEC to generate better side information than MCI. However, when easy-to-predict motion content is coded, the codec falls behind DISCOVER by 8.20%. This performance shortage in low-motion sequences is caused by the hash rate required by the TDFDVC scheme.

Thirdly, we have developed a new hash-based DVC architecture, called HDVC, \[19\] Essentially, the Wyner-Ziv encoding complexity is negligible when compared to the complexity for coding the key frames (approximately 5-10\% [88]).

Tests were carried out using an x86 personal computer with a Pentium D CPU at 3.2GHz, 2048MB of RAM and Windows XP operating system. The proposed codec is executed in software written in Visual Studio C++ v9.0, and compiled in release mode.
featuring efficient coding of low-complex hash information. This information is exploited by OBMEC/SSM, a competitive technique enabling accurate motion estimation at the decoder. The designed HDVC architecture mitigates the required compression rate and the complexity of the hash with respect to the systems presented in Sections 4.2 and 4.3. Furthermore, OBMEC/SSM advances over the bit-plane-based OBMEC technique in Sections 4.2.2 and 4.2.3, by using a different matching scheme and predictors that improve the prediction quality and reduce the computational complexity. The engineered HDVC architecture is assessed against several relevant state-of-the-art distributed and standard video codecs. Notable compression gains are reported, e.g., up to 31.41% over DISCOVER [24]. What is more, the proposed HDVC system systematically outperforms both our previous hash-based architectures, namely, our SDUDVC [103] and our TDFDVC schemes introduced in Section 4.2.1 and 4.3, respectively.

Similar to the TDFDVC system, the HDVC codec features higher encoding complexity than DISCOVER due to hash coding. However, the proposed hash formation and compression process requires limited computational and memory resources (Section 4.4.1). These demands, which are lower than those of the TDFDVC’s hash codec, are causing a slight overhead of up to 1.19% of the DISCOVER’s encoding execution time. Further specifics on the overall complexity of the HDVC system will be provided in Section 5.5.3.

Fourthly, the work in this chapter has described a new MCI-based TDWZ codec that employs an advanced MCI technique featuring bidirectional OBMC. The presented MCI-based TDWZ systematically improves over the best MCI-based DVC system in the literature, namely, the DISCOVER codec.

Fifthly, further building on the conception of OBMEC, this chapter has introduced a new side information refinement approach. The proposed technique refines the side information by performing bidirectional OBMEC using the partially decoded WZ frame. Conversely to alternative approaches in the literature, e.g., [123, 132], the proposed DC-OBME method does not need an overcomplete DCT transform, hence relieving the corresponding computational load at the decoder. Moreover, unlike for instance [121, 134], by employing multi-hypothesis prediction based on OBMEC, the proposed technique vastly constrains the imposed structural delay without jeopardizing the RD performance. Concrete experimentation has reported significant and consistent compression gains over DISCOVER, and the successively refined TDWZ systems proposed in [133] and [121] by up to 30.51%, 21.12% and 15.03%, respectively.

Comparing the designed architectures one can remark that the SDUDVC system induces the minimum encoding complexity and operates without a feedback-
channel, which makes it ideal for lightweight unidirectional applications, e.g., visual sensors for data storage. For lightweight applications that support bidirectional communications, the hash-based HDVC codec should be preferable, as it delivers higher compression performance over DISCOVER and our MCI-based TDWZ codec without refinement, especially in case of irregular motion content and large GOPs. Furthermore, to the best of the author’s knowledge, HDVC is the best performing hash-based DVC system in the literature, systematically outperforming our SDUDVC and TDFDVC schemes as well as the state-of-the-art hash-based codec of Ascenso et al. [119].

Comparing our TDWZ codec with DC-OBMEC refinement against HDVC, we observe that the latter delivers higher RD performance when high motion content is coded but lower RD performance when low motion sequences are compressed. In addition, we notice that the codec with DC-OBMEC refinement offers lower encoding complexity, as it does not code a hash like HDVC. However, the codec with DC-OBMEC refinement is more complex at the decoder, as it performs motion estimation and compensation twice. What is more, DC-OBMEC induces a structural latency on the codec, since the decoding of high frequency coefficients needs to wait until the decoding of the DC band is finished.

As a concluding remark, each of the proposed coding systems features distinctive characteristics that render it appropriate for a given lightweight multimedia application scenario, depending on the prerequisites and demands of the latter. We highlight that these codecs have been developed so as to target traditional lightweight multimedia applications, like visual sensor networks and wireless low-power surveillance. Chapter 6, however, focuses on the design of a novel hash-based codec for the promising niche application of wireless capsule endoscopy. To this end, the codec in Chapter 6 is adapted to the requisites of wireless capsule endoscopes and to the characteristics of the video content they capture.

Chapter 5 concentrates on the second important challenge in practical DVC, meaning the modeling and the effective estimation of the correlation statistics at the decoder. In this regard, Chapter 5 introduces a new correlation channel modeling scheme and proposes a novel successively refinement algorithm to estimate the parameters of the model at the decoder. An additional strength of the presented method lies on the fact that it is codec-independent; thereby it can be deployed by any DVC architecture (including of course the ones presented in this chapter and in Chapter 6).
Table 4-V: The characteristics of the proposed OBMEC-driven DVC systems.

<table>
<thead>
<tr>
<th>Proposed Hash-Based Codecs</th>
<th>Proposed Codec with Side Information Refinement</th>
</tr>
</thead>
<tbody>
<tr>
<td>Domain</td>
<td></td>
</tr>
<tr>
<td>SDUDVC</td>
<td>TDFDVC</td>
</tr>
<tr>
<td>Pixel domain</td>
<td>Transform domain</td>
</tr>
<tr>
<td>Transform domain</td>
<td>Transform domain</td>
</tr>
<tr>
<td>Hash information</td>
<td></td>
</tr>
<tr>
<td>The $b \leq 2$ MSBs of the WZ frame (luma).</td>
<td>The $b \leq 2$ MSBs of the WZ frame (luma).</td>
</tr>
<tr>
<td>Temporal prediction at the encoder</td>
<td>Temporal prediction of the hash.</td>
</tr>
<tr>
<td>Bit-plane-based OBMEC.</td>
<td>Bit-plane-based OBMEC.</td>
</tr>
<tr>
<td>Wyner-Ziv coding</td>
<td></td>
</tr>
<tr>
<td>No Slepian-Wolf codes.</td>
<td>On the $M - b$ remaining bit-planes of the WZ frames.</td>
</tr>
<tr>
<td>No</td>
<td>On the entire WZ frames' waveform.</td>
</tr>
<tr>
<td>Feedback channel</td>
<td></td>
</tr>
<tr>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>Encoding complexity</td>
<td></td>
</tr>
<tr>
<td>Very much lower than DISCOVER's (no DCT, no Slepian-Wolf codes).</td>
<td>Slightly higher than DISCOVER's (~2.81% over DISCOVER's encoding execution time).</td>
</tr>
<tr>
<td>Compression performance</td>
<td></td>
</tr>
<tr>
<td>Outperforms the PDWZ codec of [85] (forefather of DISCOVER)</td>
<td>Outperforms DISCOVER (medium and high motion content–long GOPs). Behind DISCOVER (low motion sequences).</td>
</tr>
<tr>
<td>Outperforms DISCOVER, SDUDVC, TDFDVC and the codec in [119].</td>
<td>Outperforms DISCOVER and the codecs in [121, 133].</td>
</tr>
</tbody>
</table>
Chapter 5
SIDE-INFORMATION-DEPENDENT CORRELATION CHANNEL ESTIMATION

5.1 INTRODUCTION

In Wyner-Ziv coding, the correlation statistics between the source and the side information influence both the efficiency of Slepian-Wolf coding, namely, the compression rate, as well as the quality of the source reconstruction at the decoder.

In distributed source coding theory and in the code designs presented in Chapter 2, the encoder and the decoder are ideally assumed to have perfect knowledge of the correlation statistics. However, in a practical DVC system this is not the case, since the current frame is available only at the encoder and side information is produced only at the decoder. That is, the correlation channel noise can never be directly measured in a hands-on DVC system. To solve this problem, accurate correlation channel modeling is needed.

5.1.1 Related Work on Correlation Channel Estimation

Existing approaches construct an additive correlation channel model in which the noise is assumed to be independent of the channel input signal. In early works, a zero-mean Laplacian noise model is employed, of which the scaling parameter is assumed temporally and spatially stationary [12]. Nonetheless, later, Westerlaken et al. [139] argued that by differentiating the noise scaling parameter for occluded and nonoccluded areas the overall coding performance is improved. In [140], however, they showed that segmentation inaccuracies notably reduce the convenience of the non-stationary model.

In the state-of-the-art MCI-based TDWZ architecture [24], Brites and Pereira [86] proposed a spatially stationary Laplacian model and performed online estimation of its scaling parameter per WZ frame/DCT band at the decoder side. Since the quality
of the side information fluctuates spatially, they also proposed a block and pixel/DCT coefficient based estimation in order to offer improved adaptation to the varying spatial statistics. However, along the lines of [140], adaptation in smaller spatial regions does not necessarily lead to improved performance since online estimation becomes imprecise due to limited statistical support. Furthermore, such kind of correlation estimators [83-86] are exploiting a motion-compensated residual frame between the reference frames, assuming heuristic relations between the correlation noise energy and the energy of this residual frame.

To cope with the aforementioned challenges, Škorupa et al. [113, 114] proposed a correlation estimation method which exploits the quantization distortion and the spatial correlation in the video signal. Specifically, the scaling parameters of the correlation noise per DCT coefficient were derived by utilizing the corresponding pixel-domain correlation channel estimate and the spatial correlation of the noise signal. To further improve the accuracy of the estimation, progressive refinement of the noise variance upon decoding of each DCT band is proposed in [141, 142]. Using an alternative approach, Esmaili and Cosman [143] performed block-based classification of the Laplacian scaling parameter using profiles obtained by offline training.

Recently, a correlation channel estimation technique that exploits the inter-bit-plane correlation using particle filtering has been proposed by Stankovic et al. [144] in a pixel-domain architecture without a feedback channel. This concept has lately been extended to the state-of-the-art feedback-channel-based transform-domain Wyner-Ziv architecture in [145]. However, the method of [144, 145] considers joint decoding of the WZ bit-planes, which hampers progressive (scalable) bit-plane-per-bit-plane coding. This strategy also vastly increases the decoding complexity due to employing a much larger SW codeword. What is more, the use of particle filtering in a part of the decoding factor graph requires exchanging messages containing probabilities (and not LLRs), thereby greatly intensifying the decoding complexity.

5.1.2 Contributions

In contrast to alternative models in the literature, the research work resulted in this dissertation has introduced an additive correlation channel model in which the noise depends on the channel input signal. Since the side information signal is the input of the considered channel, the term side-information-dependent (SID) correlation noise model is introduced. This pioneering concept, which constitutes a radical shift in correlation channel modeling and estimation in distributed video coding, has resulted in several important publications in the related international literature [104, 111, 115-117] and to the filing of one international patent [106]
application.

Concentrating on the SID modeling concept, this chapter details the following contributions in the domain of correlation channel modeling and estimation in distributed video coding.

- At first, this chapter introduces the concept of *SID modeling* and validates it experimentally using a fitting error metric (e.g., conditional relative entropy, optimal Slepian-Wolf rate, etc.), minimized offline.

- Secondly, this chapter studies the *theoretical grounds* of our SID model and theoretically proves that the proposed SID model brings rate-distortion (RD) gains in Wyner-Ziv video coding when compared to a classical side-information-independent (SII) model.

- Thirdly, this chapter presents a novel algorithm, which performs *online successively refined estimation* of the SID correlation channel based on already decoded bit-planes. Effectively, this algorithm has been the first approach in the literature in which online correlation estimation follows the SID paradigm. Also, unlike other methods, i.e., [86, 113, 114, 141-143], the proposed algorithm enables *bit-plane-by-bit-plane* refinement of correlation channel estimation providing improved accuracy.

- Fourthly, an additional important asset of the proposed online SID correlation channel estimation algorithm is that its applicability *does not impose limitations* on the employed DVC architecture. This is in contrast to alternative techniques in the literature, which are either specifically developed for an MCI-based system, e.g., [86, 113, 114, 141, 143], or constrained by a fixed decoding order of the DCT frequency bands, e.g., [142]. To prove this characteristic, the proposed technique has been incorporated in the proposed HDVC codec, i.e., our best performing hash-based codec from Chapter 4, as well as in the proposed TDWZ codec with DC-OBMEC-based side information refinement (see Section 4.5).

- Fifthly, this chapter evaluates the complexity of the proposed SID correlation channel estimation in relation to the complexity of the different encoding and decoding components of our best-performing HDVC codec, presented in Section 4.4.

Experimental results, provided in this chapter, validate the accuracy of SID modeling and corroborate its theoretical rate-distortion gains over SII modeling. The experiments also show that, when implemented in the proposed HDVC and TDWZ with DC-OBMEC side information refinement codecs, the proposed algorithm for online SID correlation channel estimation yields *improved compression performance* compared to state-of-the-art techniques [86, 142]. Additional,
execution time measurements, similar to the tests that the techniques in [142, 145] were evaluated, show that the proposed algorithm imposes negligible complexity at the decoder.

The remainder of this chapter is structured as follows. The rationale of the SID model and its theoretical gain against SII modeling are given in Section 5.2. The role and the application of SID modeling in state-of-the-art DVC architectures are discussed in Section 5.3. In Section 5.4, our novel algorithm to realize online successively refined SID correlation estimation is proposed. The experimental validation of the proposed correlation estimation technique is given in Section 5.5. Finally, Section 5.6 draws the conclusions of this chapter.

5.2 SID CORRELATION CHANNEL MODELING

This section explicates the concept of side-information-dependent modeling and derives theoretically the rate-distortion performance improvements that it brings over the conventional side-information-independent model.

5.2.1 Correlation Channel Modeling in DSC

In DSC, the correlation is often expressed by an additive noise channel model, \( X = Y + N \), where the side information \( Y \) is the input, and the source \( X \) is the output of the channel – see [45] and Sections 2.5 and 2.6 for explicit details. Hence, by definition, the relation between the conditional pdf of the channel output and the conditional pdf of the noise, given the input, is expressed as [146]

\[
X | Y f (x | y) = X - Y | Y f(x - y | y) = X | Y f(n | y).
\] (5.1)

Assuming that the correlation noise is independent of the channel input signal \([12, 17, 57]\), i.e., \( N \) is independent of \( Y \) [45], the correlation channel model is simplified to

\[
X | Y f (x | y) = X | Y f(n | y) = f_N (n).
\] (5.2)

We observe that the assumption of independent noise is common in DVC. Indeed, most DVC modeling approaches, e.g., [12, 86, 139-143], assume the correlation noise as independent zero-mean Laplacian with standard-deviation \( \sigma \), i.e., \( N \sim \mathcal{L}(0, \sigma) \),

\[
f_N (n) = \frac{1}{\sigma \sqrt{2}} e^{-\frac{\sqrt{2} |n|}{\sigma}}.
\] (5.3)

Therefore, the pdf of the source, \( X \), given the side information \( Y \), is expressed by a Laplacian distribution centered on \( y \) having standard-deviation \( \sigma \), i.e.,
Side-Information-Dependent Correlation Channel Estimation

\[ f_{X|Y}(x|y) = \frac{1}{\sigma \sqrt{2\pi}} e^{-\frac{\sqrt{3}(x-y)}{\sigma^2}} . \] (5.4)

Since in the referred cases, the noise \( N \) is independent of the channel input, i.e., the side information \( Y \) – see Figure 5.1 for the schemata of the channel, we call such modeling approaches side-information-independent (SII) noise modeling [104, 105, 111, 115-117].

Recall that an input-independent noise signal \( N(t) \) (where \( t \) denotes the time stamp of a sample) can be stationary or non-stationary. In the wide sense, a noise signal \( N(t) \) is called stationary [see Figure 5.1(a)] if its first and second moments, i.e., the mean \( \mu \) and the variance \( \sigma^2 \), respectively, do not vary in time \( t \). Conversely, a noise signal \( N(t) \) is called non-stationary [see Figure 5.1(b)] if its first and second moments vary in time \( t \), that is, the mean \( \mu(t) \) and the variance \( \sigma^2(t) \) of the noise are functions of the time variable \( t \).

![Figure 5.1: Schema of (a) the input-independent (SII) stationary noise model; (b) the input-independent (SII) non-stationary noise model.](image)

In the DVC related literature, the independent noise component of (5.3) has been considered stationary at different levels. In pixel-domain systems, the noise \( \sigma \) parameter is estimated at sequence-level [12, 86], frame-level [86], block-level [86], or pixel-level [86]. Analogously, in transform-domain systems, the noise \( \sigma \) parameter is estimated per band of the sequence [86, 118], per band of each WZ frame (band-level) [74, 86], or per DCT coefficient (coefficient-level) [86, 141, 142]. We stress that, although using smaller noise stationarity levels (finer granularity) enable better adaptation to the varying statistics in a video sequence, the coding performance can be compromised by inaccurate online estimation of the noise variance because of limited statistical support information.
Contrary to the SII approach, we have proposed a different correlation channel modeling concept [115-117], in which the distribution of the noise depends on the channel input signal, that is, the side information. Specifically our side-information-dependent (SID) [115-117] channel model (see Figure 5.2) assumes the noise as being zero-mean Laplacian with standard-deviation \( \sigma(y) \), which varies depending on the realization \( y \) of the side information (input of the channel), that is, \( N \sim \mathcal{L}(0, \sigma(y)) \),

\[
f_{X|Y}(x|y) = \frac{1}{\sigma(y) \sqrt{2}} e^{-\frac{|x-y|}{\sigma(y)}}.
\] (5.5)

Since the correlation noise is additive, Eq. (5.5) implies that for every realization \( y \) of the side information alphabet \( \mathcal{A}_y \), the pdf of the channel output, \( X \), is given by a Laplacian distribution centered on \( y \), having a standard-deviation \( \sigma(y) \) which varies with \( y \), i.e.,

\[
f_{X|Y}(x|y) = \frac{1}{\sigma(y) \sqrt{2}} e^{-\frac{|x-y|}{\sigma(y)}}.
\] (5.6)

Analogously to input-independent models, an input-dependent noise signal can be stationary or non-stationary. For a given input realization \( y \), the mean \( \mu(y) \) and the variance \( \sigma^2(y) \) of an input-dependent stationary noise signal [see Figure 5.2(a)] do not vary in time \( t \). Conversely, for a given input realization, the mean \( \mu(y,t) \) and the variance \( \sigma^2(y,t) \) of an input-dependent non-stationary noise signal \( N(t) \) [see Figure 5.2(b)] vary in time \( t \). Namely, the mean \( \mu(t,y) \) and the variance \( \sigma^2(t,y) \) of the noise signal are functions of both the realization of the input of the channel and the time variable \( t \).

The motivation of the SID channel model stems from the observation of the empirical conditional pmf of video data. Figure 5.3(a) depicts an example of the empirically obtained conditional pmf, i.e., \( p_{X|Y}(x|y) \forall y \in \mathcal{A}_y \), between an original frame and its side information in the pixel domain. In this figure, the random variable \( X \) denotes the sample values of the 71st (WZ) frame of the Foreman sequence. Respectively, the random variable \( Y \) signifies the sample values of the side information of the 71st frame, as produced by OBMEC (see Chapter 4). The experimental conditional pmf has been acquired by empirically measuring the joint pmf, i.e., \( p_{X,Y}(x,y) \), and applying:

\[ p_{X|Y}(x|y) = p_{X,Y}(x,y) \sum_y p_{X,Y}(x,y), \forall y \in \mathcal{A}_y \].

The models’ parameters had been fitted by minimizing the conditional relative entropy \( D_{KL} \) [26]. One observes that, in contrast to the SII model [see Figure 5.3(b)], which considers the noise standard-deviation to be independent of \( y \) for the considered granularity level (i.e., frame-
Figure 5.3: Depiction of the correlation channel $f_{X|Y}(x|y)$, given $y \in A_y$, for the 71st frame of Foreman 30Hz, CIF, GOP2 (pixel-domain): (a) Experimental pmf; (b) SII spatially stationary model; (c) SID spatially stationary model.
level stationarity), the SID model [see Figure 5.3(c)] much more accurately approximates the empirical conditional pmf, for the same assumed granularity level.

### 5.2.2 Symmetric versus Asymmetric Channels

A graphical representation of the SII and SID models, for an assumed noise stationarity level, is given in Figure 5.4. Let the side information values stem from a discrete alphabet $\mathcal{A}_y$ with $K$ elements. Then, the projections of the SII and SID correlation channel pdfs onto the $(X,Y)$-plane are given in Figure 5.4(a) and Figure 5.4(b), respectively. For an assumed noise stationarity level, in the SII (or input-independent) model the noise variance is constant and independent of the side information [see Figure 5.4(a)]. In contrast, in the SID (or input-dependent) model, the noise variance depends on the realization of the side information [see Figure 5.4(b)]. Following the terminology of channel symmetry by Cover and Thomas [26], the SII model can be interpreted as a $K$-ary input, continuous output symmetric Laplacian channel. Conversely, the SID model is equivalent to a $K$-ary input, continuous output asymmetric Laplacian channel.

Notice that in case both $X$ and $Y$ have a binary alphabet, the SII and SID channel models come down to the binary symmetric channel (BSC) and the binary asymmetric channel (BAC) models, correspondingly.

To further highlight the difference between prior symmetric approaches and the proposed asymmetric modeling concept, we employ the following example from the recent DVC literature. In detail, Toto-Zarasoa et al., in [147], considered binary sources $X$, $Y$ and a correlation channel $X = Y \oplus Z$, where the noise $Z \sim B(p_c)$ is Bernoulli distributed and independent of the side information $Y$. That is, the crossover probability $p_c$ is independent of the realization $y$ of the side information. This corresponds to a BSC:

![Figure 5.4: Projection of the (a) SII and (b) SID correlation channel distribution models for an assumed noise stationarity level.](image-url)
where $\delta(x)$ is the Dirac delta function. Toto-Zarasoa et al. introduced the terminology predictive BSC to refer to this channel. They also considered a correlation channel $Y = X \oplus Z$, where the noise $Z \sim \mathcal{B}(p_c)$ is Bernoulli distributed and independent of the source $X$. That is, the crossover probability $p_c$ is independent of the realization $x$ of the source $X$. This corresponds to the following BSC:

$$p(x|y) = \begin{cases} (1 - p_c)\delta(x) + p_c\delta(x-1), & y = 0 \\ p_c\delta(x) + (1 - p_c)\delta(x-1), & y = 1 \end{cases}$$ (5.7)

Toto-Zarasoa et al. named this channel model additive BSC channel. According to Bayes’ rule, if the input of the channel is uniform then the so-called “predictive” and “additive” BSC channels are equivalent. However, if one considers the “additive” BSC channel and the input, i.e., $X$, is non-uniform then, according to Bayes’ rule, the log-likelihood ratios (LLRs) in the channel decoder have to change based on the distribution of the input probabilities.

The proposed SID channel model is fundamentally different from the models proposed by Toto-Zarasoa et al. in [147]. Specifically, in case both $X$ and $Y$ have a binary alphabet, the proposed SID channel model boils down to a BAC [26, 146] model:

$$p(x|y) = \begin{cases} (1 - p_c^0)\delta(x) + p_c^0\delta(x-1), & y = 0 \\ p_c^1\delta(x) + (1 - p_c^1)\delta(x-1), & y = 1 \end{cases}$$ (5.9)

where $p_c^0$, $p_c^1$, $p_c^0 \neq p_c^1$, are the crossover probabilities for $y = 0$ and $y = 1$, respectively. We highlight that, conversely to the “additive” and “predictive” channels of Toto-Zarasoa et al., in the proposed SID model the crossover probabilities of the channel vary with the realization of the channel’s input.

One may observe that if we consider the “additive” BSC channel of Toto-Zarasoa et al. and $X$ is non-uniform then if we apply Bayes’ rule to $p(y|x)$, yields an asymmetric transition matrix for $p(x|y) = p(y|x)p(x)/p(y)$. This is straightforward but also different to the proposed asymmetric channel model. Note, for example, that the asymmetric nature of the proposed channel is maintained even if the distribution of the input (in our model is $Y$) is uniform.

In the following, we analytically prove that, at the same noise stationarity level, SID channel modeling yields compression gains over classical SII modeling. Driven by this finding, we propose a novel transform-domain online SID estimation method.
which considers band-level SID noise stationarity, that is, it is applied per coded DCT band of each WZ frame. The proposed algorithm enables bit-plane-by-bit-plane successively refined SID correlation estimation, yielding significant gains over the offline SII band-level method of [86], and the online SII coefficient-level TRACE technique [142].

5.2.3 Theoretical Analysis of SID Correlation Channel Modeling

We consider a WZ scheme consisting of uniform scalar quantization followed by ideal Slepian-Wolf (SW) coding [61]. This scheme has been shown to deliver Wyner-Ziv coding performance equivalent to entropy-coded scalar quantization in nondistributed coding [61]. To simplify the calculations, known asymptotic results [61, 148] are employed when necessary. The following holds.

Lemma 5.1: The L-2 distortion for a Laplacian source quantized using a uniform scalar quantizer centered on its mean is given by

$$D = \frac{2}{\lambda^2} - \frac{2\Delta e^{-\Delta^2/2}}{\lambda(1-e^{-\Delta^2})},$$

(5.10)

where $\Delta$ is the cell size of the quantizer, $\lambda = \sqrt{2}/\sigma$, and $\sigma$ is the standard-deviation.

Proof: The proof is sketched in Appendix C.

Remark 5.1: According to high rate results for distributed quantization [61], the optimal SW-coded scalar quantizer for smooth pdfs is the uniform quantizer. Based on Lemma 5.1, the following is derived.

Lemma 5.2: Let $D_{SID}$, $D_{SII}$ be the L-2 distortion of the SID and SII Laplacian model, respectively. A necessary condition so that the SID distortion be equal to the SII distortion for any $\Delta$, that is,

$$\int_{-\infty}^{+\infty} \sigma^2_{SID}(y) f_{Y}(y) dy = D_{SID},$$

$$\forall \Delta,$$

is given by

$$\int_{-\infty}^{+\infty} \sigma^2_{SII}(y) f_{Y}(y) dy = D_{SII},$$

(5.11)

where $\sigma_{SID}(y)$, $\sigma_{SII}$ are the standard-deviations of the SID and SII models respectively, and $f_{Y}(y)$ is the pdf of the side information $Y$.

Proof: The proof is given in Appendix C.

Remark 5.2: We note that the employed uniform scalar quantizer is centered on the mean of each Laplacian distribution in order to achieve the upper bound in the WZ source coding gain [61]. An analysis of the WZ source coding loss depending on the position of the quantizer in relation to the mean of the distribution can be found in [91]. Based on these two lemmas the following holds.
Theorem 5.1: Under high rate assumptions and considering the SID and SII models’ distortions equal \( \forall \Delta \),

\[
R_{\text{SID}}(D) - R_{\text{SII}}(D) = E\left[\log_2 \sigma_{\text{SID}}(y)\right] - \log_2 \sigma_{\text{SII}} \leq 0,
\]

(5.12)

where \( R_{\text{SID}}(D) \) and \( R_{\text{SII}}(D) \) are rates for a distortion level as given by an SID and an SII Laplacian model, respectively, and \( E[\cdot] \) is the expectation operator.

Proof: The proof is given in Appendix C.

Remark 5.3: We deduce from Theorem 5.1 that the conditional entropy of the quantized source given the side information is reduced when the correlation noise depends on the realization of the side information, that is, \( H_{\text{SID}}(Q(X)|Y) \leq H_{\text{SII}}(Q(X)|Y) \). Therefore, in terms of mutual information, we have

\[
I_{\text{SID}}(Q(X);Y) \geq I_{\text{SII}}(Q(X);Y).
\]

(5.13)

Hence, for the capacity of the correlation channel, we have

\[
\max_{f_{\text{SID}}(y)} I_{\text{SID}}(Q(X);Y) \geq \max_{f_{\text{SII}}(y)} I_{\text{SII}}(Q(X);Y) \iff C_{\text{SID}}(D) \geq C_{\text{SII}}(D).
\]

(5.14)

In words, Theorem 5.1 specifies that, for a given L-2 distortion \( D \), an SID (asymmetric) channel exhibits higher or equal correlation channel capacity compared to an SII (symmetric) channel. As a consequence, for a given L-2 distortion \( D \), SW random binning for an SID channel is more efficient compared to that for an SII channel. This intrinsic coding gain is experimentally verified in Section 5.5.1.

5.3 The Role of SID Modeling in DVC

This section discusses the role of SID modeling and the usage of online SID correlation channel estimation (CCE) in state-of-the-art transform-domain Wyner-Ziv (TDWZ) video coding architectures. The details of the proposed SID CCE algorithm will be given in Section 5.4.

Recall that the correlation channel statistics are exploited in a twofold way at the decoder of DVC systems. First, the correlation channel statistics are needed to derive the soft-input information in the Slepian-Wolf decoder. Second, the correlation channel statistics are used to perform inverse quantization, i.e., reconstruction, of the quantized WZ source information.
We observe that state-of-the-art TDWZ video coding systems, including our novel codecs proposed in Sections 4.3-4.5, are built on the principles of layered, i.e., bit-plane-per-bit-plane, Wyner-Ziv coding \[17\] (see Section 2.7.2). Specifically, at the encoder of state-of-the-art TDWZ systems (see Figure 5.5), the WZ frame’s pixel values are transformed, and then each DCT band, \( \beta \), is independently quantized with \( 2^{L_\beta} \) levels. After quantization, the quantization indices are converted into bit-planes (i.e., binary codewords), which are progressively Slepian-Wolf encoded. Hence, it is of paramount significance to engineer a novel SID CCE method that is compatible with such kind of TDWZ coding and also takes advantage of its progressive (layered) nature.

At this point, we notice that the employed Slepian-Wolf coding needs to cope with the asymmetric nature of the SID correlation channel (see Section 5.2). In \[149\], McEliece argued that good communication codes with Turbo-like decoders could be effectively used on asymmetric channels. Furthermore, in \[150\] Wang et al. extended Density Evolution – a strong analytical tool of low-density parity-check (LDPC) codes – to asymmetric channels and proposed good code constructions. Thus, in the designed DVC systems, Slepian-Wolf coding is realized using the rate-adaptive LDPC accumulate (LDPCA) codes of \[48\], of which the performance is not degraded even when the correlation channel features strong asymmetries, e.g., Z-channel statistics \[48\].

It is also important to mention that, as shown in Chapter 4, there can be several methods to generate side information in DVC, each of which having distinctive advantages depending on the target application and video content. It is therefore necessitated to design a novel SID CCE algorithm that is not confined to a specific side information generation method, as it is the case with the techniques in \[86, 141, ... \]
neither does it impose a fixed decoding order of the DCT frequency bands, unlike the TRACE [142] algorithm which impedes the decoding order of the DCT bands in our state-of-the-art TDWZ codec with DC-OBMEC side information refinement (see Section 4.5).

The proposed SID CCE algorithm, which is described in Section 5.4.2, complies with the aforementioned requirements in DVC; namely, it is compatible with layered Wyner-Ziv coding and it is independent of the side information generation method.

After the side information frame is obtained and transformed, the decoder performs the proposed novel online SID CCE algorithm (see Figure 5.5). Per band \( \beta \) of the WZ frame, the decoder produces soft-input estimates (i.e., LLRs) to decode the WZ bit-planes based on the SID channel estimates, \( \sigma_\beta(y_k) \), and the value of each side information coefficient. Per bit-plane of each band, the calculation of the LLRs is performed as explained in Section 2.7.2. After decoding each bit-plane of a band \( \beta \) of the WZ frame, the bit-plane is stored in a buffer and the algorithm is executed again enabling bit-plane-by-bit-plane progressive refinement of the \( \sigma_\beta(y_k) \) estimates. After decoding the final WZ bit-plane of the band \( \beta \) of the WZ frame, the \( \sigma_\beta(y_k) \) estimates are again updated yielding improved estimation for the reconstruction process. After decoding all the bit-planes of all WZ coded bands of the WZ frame, MMSE reconstruction [87] and inverse DCT are carried out, yielding the reconstructed WZ frame.

5.4 SID CORRELATION CHANNEL ESTIMATION

In this section, we describe our novel algorithm for online bit-plane-by-bit-plane progressively refined SID correlation channel estimation. First, we propose an offline SID channel estimator that serves as a reference enabling the accuracy assessment of the proposed online algorithm.

5.4.1 Offline SID Correlation Channel Estimation

Offline SID correlation estimation is an ideal but unrealistic approach, since it assumes that the original frame pixel values are present at the decoder. Under this assumption, the decoder can form the transformed noise frame, i.e.,

\[
N(t) = X(t) - Y(t), \quad t = (\beta_i, \zeta_i),
\]

where \( X(t) \) and \( Y(t) \) denote the DCT transformed WZ and side information frame’s coefficient of the band \( \beta_i \) and the block \( \zeta_i \). Per coded band of a WZ frame, offline SID correlation channel estimates are independently determined.

We note that DCT coefficients are real numbered, so practically, in order to have
a discrete number of SID sigma values, one needs to group the side information frame coefficients of each band \( \beta \). Then, for every coded band \( \beta \) and transformed side information quantization index \( y_k \), the corresponding offline SID estimate is

\[
\sigma_\beta(y_k) = \sqrt{E[N^2(t)] - E^2[N(t)]},
\]

\[
\forall t \in \{ t = (\beta_t, \zeta_t) | \beta_t = \beta, Q_{L_\beta}(Y(t)) = y_k, y_k \in [0, 2^{L_\beta} - 1] \},
\]

(5.15)

where \( Q_{L_\beta}(\cdot) \) denotes the quantization of the side information frame coefficients of band \( \beta \), and \( 2^{L_\beta} \) is the number of quantization levels (QLs) for band \( \beta \). Eq. (5.15) implies that, per band \( \beta \) of a WZ frame, \( \sigma_\beta(y_k) \) is estimated as the standard-deviation of the transformed noise frame coefficients of which the corresponding side information coefficient value falls into the quantization bin indexed by \( y_k \).

For both the proposed offline and online SID algorithms, the best RD performance is obtained based on a balance between the SID noise stationarity level, the number of QLs of the side information frame coefficients per band, and the statistical support to accurately estimate the SID \( \sigma_\beta(y_k) \) parameters. Concerning stationarity, SID estimation can be applied to any small number of spatially neighboring DCT coefficients \( t = (\beta_t, \zeta_t) \) that belong to the same DCT band, i.e., \( \beta_t = \beta \). However, note that, though using small spatial stationarity levels enables better adjustment to fluctuating spatial statistics, precision can be challenged by inaccurate estimation of the \( \sigma_\beta(y_k) \) parameters due to narrow statistical support. Additionally, the number of QLs used in the quantization of the side information coefficients affects the accuracy of channel estimation. A low number of QLs drives the system to the SII paradigm. On the other hand, a high number of QLs reduces the statistical support, hence deteriorating the RD performance. We have empirically observed that the highest RD performance for the proposed offline and online SID algorithms is obtained when the SID channel is considered stationary at band-level and the number of QLs per band is identical to the number of QLs employed to quantize the values of the transformed original frame at the encoder.

### 5.4.2 Online Successively Refined SID Correlation Channel Estimation

The proposed online SID algorithm notably advances over contemporary online

---

\( ^{21} \) Notice that grouping (quantization) of the side information coefficients is only performed for discretizing their alphabet in order to enable SID correlation channel estimation. When deriving the soft decoding log-likelihood ratios and during MMSE reconstruction, the actual values of the side information coefficients are used.
correlation estimation methods.

First, apart from few exceptions, e.g., [142], most online methods [86, 113, 114, 141, 143], are solely created for MCI-based systems. On the contrary, the proposed technique is intended to not limit its applicability to a specific DVC architecture.

Second, in contrast to the TRACE [142] technique, the proposed algorithm does not impose a fixed decoding order of the DCT frequency bands.

Third, the proposed technique is the first to capture the dependency of the correlation noise on the side information, thereby exploiting the inherent coding gain of SID (asymmetric) channel modeling.

Fourth, conversely to alternative transform-domain correlation estimation methods [141, 142], which refine the SII model parameters across DCT bands of a WZ frame, the proposed technique improves the estimation accuracy by enabling a finer refinement of the SID model parameters, namely, across the bit-planes of a band of a WZ frame.

Fifth, unlike the technique in [144, 145], the proposed algorithm, is compatible (actually exploits) progressive bit-plane-per-bit-plane coding, thereby supporting bit-plane-based quality scalable Wyner-Ziv video coding.

Sixth, conversely to the method from [144, 145], which vastly increases the decoder’s computation demands, the proposed algorithm imposes a negligible burden on the decoding complexity.

As explained above, the proposed online algorithm guarantees adequate statistical support for accurate SID estimation by balancing the SID stationarity level and the number of QLs of the side information coefficients per band. For any coded band $\beta$ of a WZ frame, the successive refinement algorithm is initiated after SW decoding of the first bit-plane of the band. To facilitate the applicability of the algorithm to any DVC scheme, similar to [74], the $\sigma_{\beta}(y_k) \forall y_k \in \left[0, 2^{L_{\beta}} - 1\right]$ estimates, required to decode the first bit-plane of a band $\beta$, are obtained using (5.15) in which $N(t)$ is approximated by the transformed noise of the reconstructed previous WZ frame and its corresponding side information $^{22}$. Since the correlation between the first WZ bit-plane and the side information is high, or else, since WZ quantization is very coarse, the estimation error caused by this approach does not notably influence the RD performance, as shown in Section 5.5.1.

In a nutshell, the principle of our online bit-plane-by-bit-plane successive refinement SID algorithm is as follows. Per coded band $\beta$ of a WZ frame, the algorithm combines the already SW decoded bit-planes of the band to derive

---

$^{22}$ Only for the very first WZ frame, one (SII) sigma per coded WZ band is estimated at the encoder similarly to [96] and communicated to the decoder. This is done in order to keep the required rate to code the sigmas as low as possible.
quantization indices of the WZ frame coefficients of the band. In this way, apart from the side information coefficients, the algorithm is aware of a coarse description of the WZ frame’s coefficients of the band. Then, based on the available WZ quantization indices and the discretized side information coefficients of the band, i.e., \( y_k \in \left[ 0, 2^{L_{Y}} - 1 \right] \), the algorithm empirically estimates the correlation channel conditional pmf for any given \( y_k \). Based on the obtained pmf for any \( y_k \), the algorithm derives the corresponding conditional pdf. Notice that, according to the SID model, the conditional pdf for a given \( y_k \) is Laplacian with a parameter \( \sigma_{\beta}(y_k) \) which depends on \( y_k \). The derived SID estimates \( \sigma_{\beta}(y_k) \) are used to decode the next bit-plane. Thereafter, the decoder has access to a finer description of the WZ coefficients of the band and the algorithm is repeated. After SW decoding all the bit-planes of the band \( \beta \), the algorithm is executed again so as to refine the \( \sigma_{\beta}(y_k) \) estimates for reconstruction.

For any coded DCT band \( \beta \) of a WZ frame, the progressive refinement process of the proposed algorithm is thoroughly described in the next steps:

Step 1) Let \( m, 1 \leq m \leq L_{\beta} \), denote the number of decoded bit-planes of the WZ coefficients of the band \( \beta \). These bit-planes are denoted by the binary \( M \)-tuples \( b_{1}^{M}, b_{2}^{M}, \ldots, b_{m}^{M} \), where, \( M \) is the size of the WZ frame band. The decoder combines the available \( b_{1}^{M}, b_{2}^{M}, \ldots, b_{m}^{M} \) binary \( M \)-tuples to produce a coarse description of the WZ coefficients of the band \( \beta \), denoted by \( q_{m}^{M} \). The latter contains quantization indices of the WZ coefficients of the band \( \beta \) in the range \( q_{m} \in \left[ 0, 2^{m} - 1 \right] \). Also, for the purpose of channel estimation, the decoder discretizes the side information coefficients of the band \( \beta \) (see Section 5.4.1) thereby producing the \( y_{k}^{M} \)-tuple, containing the indices \( y_{k} \in \left[ 0, 2^{L_{Y}} - 1 \right] \).

Step 2) Using the available \( q_{m}^{M} \) and \( y_{k}^{M} \) \( M \)-tuples, the correlation estimator approximates the joint pmf, \( P_{Q_{m},Y}(q_{m},y_{k}) \), where, \( Q_{m} \) denotes the random variable of the quantization indices of the WZ coefficients, \( q_{m,\vartheta} \) denotes the \( \vartheta^{th} \) quantization index, or bin, \( q_{m,\vartheta} \in \left[ 0, 2^{m} - 1 \right] \), and \( y_{k} \in \left[ 0, 2^{L_{Y}} - 1 \right] \). This approximation is performed using the histogram.

Step 3) From the estimated joint pmf, the algorithm calculates the empirical conditional pmf, i.e.,

\[
P_{Q_{m},Y}(q_{m,\vartheta} | y_{k}) = \frac{P_{Q_{m},Y}(q_{m,\vartheta}, y_{k})}{\sum_{\vartheta=0}^{2^{m}-1} P_{Q_{m},Y}(q_{m,\vartheta}, y_{k})}. \tag{5.16}
\]

\(23\) In particular, given the available bit-planes \( b_{1}, b_{2}, \ldots, b_{m} \) for a given WZ coefficient, the quantizer cell indexed by \( q_{m} \) is formed by merging the original, encoder-side quantizer cells for which the binary representation is prefixed by \( b_{1}, b_{2}, \ldots, b_{m} \).
For every index $y_k \in [0, 2^{L_{\beta}} - 1]$ and $q_{m, \vartheta} \in [0, 2^m - 1]$, Eq. (5.16) gives the transition matrix $\left[ p_{Q_{m, \vartheta}}(q_{m, \vartheta} | y_k) \right]$ of a discrete correlation channel having the discretized side information coefficients as input and the quantized (with $2^m$ levels) WZ coefficients as output, namely,

$$
\left[ p_{Q_{m, \vartheta}}(q_{m, \vartheta} | y_k) \right] = 
\begin{bmatrix}
  p_{Q_{m, \vartheta}}(q_{m, 0} | y_0) & \cdots & p_{Q_{m, \vartheta}}(q_{m, \vartheta} | y_0) \\
  \vdots & \ddots & \vdots \\
  p_{Q_{m, \vartheta}}(q_{m, 0} | y_{K-1}) & \cdots & p_{Q_{m, \vartheta}}(q_{m, \vartheta} | y_{K-1})
\end{bmatrix}
$$

(5.17)

Step 4) Each row of the transition matrix in (5.17), i.e., $p_{Q_{m, \vartheta}}(q_{m, \vartheta} | y_k) \forall q_{m, \vartheta} \in [0, 2^m - 1]$, is a conditional pmf for a given $y_k$. The algorithm then, attempts to derive the conditional pdf for a given $y_k$, which would yield this empirical pmf. According to the SID paradigm, $p_{Q_{m, \vartheta}}(q_{m, \vartheta} | y_k) \forall q_{m, \vartheta} \in [0, 2^m - 1]$, is a pmf which would be derived by scalar quantization of a Laplacian distribution which has a scaling parameter $\lambda_{\beta}(y_k) = \sqrt{2/\sigma_{\beta}(y_k)}$, and it is centered on $y'_k$, namely, the inversely quantized side information value derived from the quantization index $y_k$. Therefore, to get the scaling parameter $\lambda_{\beta}(y_k)$ of each Laplacian distribution $\forall y_k \in [0, 2^{L_{\beta}} - 1]$, one needs to find the root of the following function:

$$
g(\lambda_k) = p_{Q_{m, \vartheta}}(q_{m, \vartheta} | y_k) - \int_{q_L}^{q_H} \frac{\lambda_k}{2} e^{-\lambda_k |y_k - \gamma_k|} dy_k, \forall y_k \in [0, 2^{L_{\beta}} - 1], (5.18)
$$

where $q_L$, $q_H$ are the lower and upper bounds of the quantization bin with index $q_{m, \vartheta}$, and $\lambda_k$ has replaced $\lambda_{\beta}(y_k)$ for simplicity. The derivation and the proof of the uniqueness of the root of Eq. (5.18) for every $y_k \in [0, 2^{L_{\beta}} - 1]$ are detailed in Appendix C.

After SW decoding of a bit-plane of the band $\beta$ of the WZ frame Steps 1-4 are executed again. The process is executed recursively, hence delivering a more accurate estimate with every additional decoded bit-plane of a band $\beta$ of the frame. This is of paramount importance since for every additional bit-plane WZ quantization becomes finer, thus the impact of inaccurate channel estimation on the coding efficiency increases. This also explains why for the first bit-plane of each band, one can afford a less accurate estimate without experiencing a significant penalty in coding performance.
5.5 Experimental Results

This section evaluates the proposed SID correlation channel estimation method as part of the best performing coding systems proposed in Chapter 4. Namely, in order to prove the codec-independent character of the proposed SID CCE technique, the latter has been incorporated in both our best performing hash-based system, i.e., the HDVC scheme in Section 4.4, as well as in our TDWZ codec with DC-OBMEC side information refinement, presented in Section 4.5.

First, we compare the accuracy of the proposed offline and online SID CCE method versus conventional offline SII estimation [86]. Next, we assess the RD improvements in the compression performance of the HDVC system brought by the proposed online SID CCE algorithm compared to the state-of-the-art TRACE method [142]. Subsequently, we quantify the impact of the proposed SID CCE algorithm on the RD performance of the proposed TDWZ codec with side information refinement. At the end, we give a complexity assessment of the proposed SID CCE technique in relation to other coding modules of our state-of-the-art HDVC system.

To be systematic, we follow the test conditions introduced in Chapter 4. Namely, experiments were performed for all frames of Foreman, Soccer, Carphone, and Silent sequences, at QCIF resolution, a frame rate of 15Hz, and GOP sizes of 2, 4 and 8. Recall from Section 4.6.2.2 that the selected sequences exhibit different motion content characteristics. Moreover, the QMs of the WZ frames and the QPs of the key frames were kept unaltered with respect to the parameters used for drawing the results presented in Sections 4.6.2.2, 4.6.3 and 4.6.4. Furthermore, the configuration of the OBMEC/SSM and the DC-OBMEC module was exactly the same with the one mentioned in Section 4.6.3 and 4.6.4, respectively.

5.5.1 Validation of the Proposed SID Model’s Accuracy

To demonstrate the precision of SID versus SII modeling, we compare the ideal SW rate, namely, the conditional entropy $H(Q(X)|Y)$, calculated using the offline band-level SID (see Section 5.4.1), the offline band-level SII [86], and the proposed online band-level SID correlation estimation algorithm (see Section 5.4.2). The numerical results, which are obtained with the presented HDVC codec, are computed for the DC coefficient band and averaged over all WZ frames. Seven rate points are depicted, corresponding to QM8 of DISCOVER [24]. The results, illustrated in Figure 5.6, show that, at the same stationarity level (i.e., band-level), input-dependent (SID) channel modeling offers a significant reduction of the ideal SW rate compared to input-independent (SII) [86] estimation. Notice that at high
rates the theoretical gain of SID over SII modeling, which is given by (5.12), agrees with the experimental measurements, thereby confirming our theory.

![Graph showing SW rate per number of coded bit-planes for DC band](image)

Figure 5.6: Ideal SW rate per number of coded bit-planes of the DC band for (a) Foreman GOP2, and (b) Soccer GOP8. The theoretical gain at high rates as derived by (5.12) is (a) 0.38 bits/sample, and (b) 0.61 bits/sample.

Regarding the accuracy of the proposed online SID algorithm, one observes that, due to its successively refined nature, it performs closely to offline SID estimation irrespective of the rate, the amount of motion in the sequence or the GOP size. Figure 5.6 also shows that the SW rate for decoding the MSB using online SID estimation is approximately the same with that of using offline SII [86]. This validates our choice to initialize the proposed algorithm with the statistics taken from the previously decoded WZ frame and the corresponding side information, as explained in Section 5.4.2. Nonetheless, the higher the SW rate the higher the improvement brought by online SID versus offline SII [86] estimation, thus justifying the potential of the proposed method.
5.5.2 Performance Evaluation of SID Correlation Channel Estimation

In the following, we quantify the influence of the proposed online correlation channel estimation algorithm on the compression performance of our best performing DVC systems presented in Chapter 4.

5.5.2.1 SID Correlation Channel Estimation in the HDVC System

First, we evaluate the compression improvements brought by the proposed online SID CCE algorithm in our HDVC scheme (see Section 4.4). Figure 5.7 and Figure 5.8 depict the RD comparison of the proposed online band-level SID algorithm against the offline band-level SII channel estimation in [86], and the state-of-the-art coefficient-level SII TRACE method in [142]. Figure 5.7 and Figure 5.8 also include the coding results obtained with offline band-level SID estimation, which serves as an ideal SID channel estimate. The results reveal that offline SID estimation yields the best coding performance among the assessed methods, thus verifying the capacity of SID versus SII modeling, as theoretically anticipated. We also notice that the RD performance achieved with online SID is close to that obtained with offline SID estimation, hence confirming the ability of the proposed online SID algorithm to accurately estimate the SID channel statistics.

It is also noteworthy that online band-level SID estimation consistently outperforms the offline band-level SII estimation method [86], yet the latter is impractical. One observes that the coding gain brought by the proposed online SID method compared to the offline band-level SII method [86] increases with the rate, since progressive refinement allows the proposed online SID method to improve its estimation accuracy as more information is decoded. This trend is also shown in Figure 5.6. In terms of the average Bjøntegaard rate metric, the proposed online band-level SID algorithm outperforms offline band-level SII estimation by 2.88%, 4.4%, 4.05%, and 3.74% in Foreman GOP2, Soccer GOP2, Carphone GOP4 and Silent GOP8, respectively.

We point out that the proposed online SID algorithm delivers higher compression efficiency compared to the state-of-the-art TRACE technique [142], which models the correlation noise as input-independent (SII) and non-stationary (coefficient-level). Unlike the online coefficient-level method in [86], which is solely designed for MCI-based schemes, TRACE can be employed in the proposed system. Also, TRACE enables progressively refined estimation across decoded DCT bands, outperforming the method from [86]. Nevertheless, compared to the proposed SID estimation method, TRACE builds on the SII paradigm and supports a coarser refinement, namely, band-based versus bit-plane-based in the proposed SID method.

The results summarized in Table 5-I show that compared to TRACE, the
proposed online SID algorithm yields average Bjøntegaard rate reductions of 1.63%, 2.61%, and 3.35% for a GOP size of 2, 4, and 8, respectively. We observe that the RD gains increase with the length of the GOP, since the number of coded WZ frames grows. Moreover, the gains increase, that is, up to 5.17% Bjøntegaard rate reduction for Soccer GOP8, when irregular video content is coded. This is anticipated since as the quality of the side information deteriorates, the impact of accurate correlation estimation on the RD performance of the system increases.

Table 5-I: Bjøntegaard [138] gains obtained using the proposed online SID algorithm compared to TRACE [142].

<table>
<thead>
<tr>
<th>Sequence</th>
<th>GOP2</th>
<th>GOP4</th>
<th>GOP8</th>
</tr>
</thead>
<tbody>
<tr>
<td>Foreman</td>
<td>ΔR (%)</td>
<td>ΔPSNR (dB)</td>
<td>ΔR (%)</td>
</tr>
<tr>
<td>Soccer</td>
<td>-2.46</td>
<td>0.12</td>
<td>-3.99</td>
</tr>
<tr>
<td>Carphone</td>
<td>-1.39</td>
<td>0.09</td>
<td>-2.50</td>
</tr>
<tr>
<td>Silent</td>
<td>-1.48</td>
<td>0.10</td>
<td>-1.75</td>
</tr>
</tbody>
</table>

5.5.2.2 SID Correlation Channel Estimation in the TDWZ System with Side Information Refinement

In the next experimental set, we assess the benefit of incorporating our novel SID CCE algorithm in the proposed TDWZ codec with DC-OBMEC side information refinement, which was presented in Section 4.5. To show the effect of our algorithm, our TDWZ codec with side information refinement was equipped with the proposed online progressive SID CCE as well as with the offline band-level SII CCE algorithm in [86]. The online SII coefficient-level channel estimator TRACE [142] could not be integrated in the proposed TDWZ system, since the fixed reverse order in which the DCT frequency bands need to be decoded according to the TRACE algorithm interferes with the pursued side information refinement strategy. This highlights the generic applicability of the proposed SID correlation channel estimation algorithm, as mentioned in Sections 5.3 and 5.4.2.

The compression performance of the proposed TDWZ system with side information refinement for the different CCE methods is shown in Figure 5.9 and Figure 5.10. The results reveal that adopting the proposed SID CCE algorithm in our TDWZ system with side information refinement increases the compression performance compared to the offline SII band-level method in [86]. Respective Bjøntegaard [138] rate savings of 3.42%, 4.68%, 4.82% and 3.16% for Foreman GOP8, Soccer GOP8, Carphone GOP4 and Silent GOP4 are obtained. These gains underline the potential of the proposed algorithm in accurately estimating the correlation channel statistics. Notice that the offline SII band-level method from [86] is an unrealistic approach, which cannot be adopted in a practical DVC system.
Figure 5.7: RD performance comparison of correlation channel estimation methods in our HDVC system for (a) Foreman, GOP2, and (b) Soccer, GOP2.
Figure 5.8: RD performance comparison of correlation channel estimation methods in our HDVC system for (a) Carphone, GOP4, and (b) Silent, GOP8.
Figure 5.9: RD performance comparison of correlation channel estimation methods in our TDWZ system with side information refinement for (a) Foreman, GOP8, and (b) Soccer, GOP8.
Figure 5.10: RD performance comparison of correlation channel estimation methods in our TDWZ system with side information refinement for (a) Carphone, GOP4, and (b) Silent, GOP4.
5.5.3 Complexity Evaluation

This section evaluates the computational complexity of the proposed SID correlation channel estimation method with respect to the complexity of different components of our HDVC system (see Section 4.4). To this end, we conducted execution time tests under controlled conditions (Pentium D CPU at 3.2GHz, 2048MB of RAM and Windows XP operating system), as in [24, 142].

At the encoder, the proposed HDVC system features the same Intra and Wyner-Ziv codecs as DISCOVER [24], thus the encoding complexity is dominated by the coding of the key frames. Recall from Chapter 4 that, conversely to DISCOVER, the proposed encoder allocates additional resources to code the hash, yet this imposes very low computational and memory demands as mentioned in Section 4.4.1. In particular, experimentation has shown that the hash encoder causes a negligible overhead of 1.15% and 1.19% on the encoding execution time for Foreman and Soccer, GOP8, QM8, respectively.

Table 5-II assesses the decoding execution time of the proposed codec with respect to DISCOVER’s executable [24]. The results, which are averaged over the four RD points considered in our experimental settings, show that most of the total execution time (90.79% on average) is consumed by Slepian-Wolf decoding (SWD), which performs repeated LDPCA decoding using the feedback channel. OBMEC/SSM, is the second most demanding operation taking on average 9.17% of the total decoding time. Recall from Section 4.4.2 that for the same number of predictors per pixel and considering the same search space for each predictor, OBMEC/SSM has a significantly reduced complexity compared to the OBMEC [103] technique presented in Sections 4.2.2 and 4.2.3. This is because, by construction, OBMEC/SSM requires only one fourth of operations for each block comparison with respect to OBMEC, as explained in Section 4.4.2.1.

Concerning the complexity of correlation channel estimation (CCE), we notice that the computational demands of offline SID estimation are similar to those of offline band-level SII estimation. When comparing the offline with online SID estimation, one observes that the latter is more complex due to its bit-plane-by-bit-plane successively refined behavior. However, the proposed online SID estimator comprises simple histogram measuring and solving non-linear equations. Both of these steps are efficiently performed using elegant algorithms, as detailed in Section 5.4.2. Moreover, contrary to the demanding operations of side information creation and LDPCA decoding, the online SID algorithm has a minor contribution to the total decoding computation demands, namely, 0.04% on average. In this regard, it is important to highlight that, according to the results in Table 5-II, although online
Table 5-II: Comparison of Wyner-Ziv decoding execution times (in sec/WZ frame).

<table>
<thead>
<tr>
<th>Sequence</th>
<th>OBMEC/SSM</th>
<th>CCE</th>
<th>SWD</th>
<th>CCE</th>
<th>SWD</th>
<th>CCE</th>
<th>SWD</th>
<th>Total</th>
<th>DISCOVER</th>
<th>Gain (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Foreman, GOP2</td>
<td>6.160</td>
<td>0.008</td>
<td>59.902</td>
<td>0.005</td>
<td>69.354</td>
<td>0.027</td>
<td>64.044</td>
<td>70.231</td>
<td>86.272</td>
<td>18.59</td>
</tr>
<tr>
<td>Foreman, GOP8</td>
<td>10.121</td>
<td>0.011</td>
<td>69.831</td>
<td>0.008</td>
<td>89.980</td>
<td>0.039</td>
<td>80.038</td>
<td>90.197</td>
<td>125.027</td>
<td>27.86</td>
</tr>
<tr>
<td>Soccer, GOP2</td>
<td>6.729</td>
<td>0.009</td>
<td>77.837</td>
<td>0.006</td>
<td>92.882</td>
<td>0.032</td>
<td>85.756</td>
<td>92.517</td>
<td>105.698</td>
<td>12.47</td>
</tr>
<tr>
<td>Soccer, GOP8</td>
<td>9.721</td>
<td>0.011</td>
<td>89.066</td>
<td>0.008</td>
<td>104.303</td>
<td>0.040</td>
<td>94.400</td>
<td>104.162</td>
<td>137.828</td>
<td>24.43</td>
</tr>
</tbody>
</table>
SID estimation is more complex than offline band-level SII estimation, the overall decoding complexity is significantly reduced. This is due to the fact that correlation estimation becomes more accurate, and as a result SW decoding requires fewer feedback channel requests and decoding operations.

We point out that, although not optimized for speed, the total decoding time of the proposed codec – with online SID estimation – is vastly lower than DISCOVER’s, yielding a decrease of up to 27.86% in decoding execution time. This notable benefit is credited to the novel techniques encompassed by the proposed system, i.e., online SID correlation channel estimation and hash-based OBMEC/SSM, which significantly increase the decoding performance, thus reducing soft channel decoding operations. Furthermore, it is worth mentioning that these gains in decoding complexity versus DISCOVER’s highlight the advantage of the proposed online SID correlation estimation method compared to the techniques presented in [144, 145]. In particular, observe that the complexity analysis provided in [144, 145] shows that the associated decoding complexity is above ten times higher than DISCOVER’s.

## 5.6 CONCLUSIONS

In contrast to alternative approaches in the literature, which construct an additive SII correlation channel model, the research work presented in this chapter has introduced the novel concept of SID modeling of the correlation channel in DVC.

One of the main contributions of the work in this chapter has been the development of a theoretical formulation to compare the implications of the common SII modeling assumption with respect to our SID correlation channel model. Specifically, it has been theoretically proven that SID modeling leads to systematic Wyner-Ziv compression improvements compared to the conventional SII assumption.

Motivated by this theoretical finding, another contribution of this work has been the introduction of a novel algorithm enabling online SID correlation channel estimation.

The applicability of the designed algorithm is not limited to the employed architecture as being the case for other state-of-the-art methods, e.g., [86, 113, 114, 141, 143], which are specifically designed for MCI-based architectures. Furthermore, unlike the state-of-the-art TRACE [142] method, the proposed SID algorithm does not force a specific decoding order of the DCT frequency bands, thereby extending its applicability to architectures equipped with side information refinement.

Additionally, unlike alternative channel estimation algorithms [86, 113, 114, 141-
the proposed technique enables bit-plane-by-bit-plane progressive refinement of the model parameters. Therefore, by supporting a finer refinement of the correlation estimation, the presented method provides improved accuracy with respect to competitive techniques, e.g., [86, 142].

The proposed online algorithm also benefits from adequate statistical support to perform accurate SID estimation by balancing the SID stationarity level and the number of discretized side information values per band.

It is also important to emphasize that, as confirmed by experimentation, the computational demands of the presented online SID estimation algorithm are minimal. This is in contrast to other techniques in the literature, e.g., [144, 145], which impose a significant burden on the decoding complexity.

The proposed SID correlation channel estimation method has been integrated in our novel HDVC codec, detailed in Section 4.4, as well as in our innovative TDWZ codec with DC-OBMEC side information refinement, presented in Section 4.5.

The experimental results have verified the theoretically derived compression improvements of SID compared to SII modeling. In addition, methodical experimentation over several test video sequences and GOP sizes validate the RD improvements of the proposed online SID algorithm versus competing state-of-the-art correlation channel estimation methods [86, 142]. These gains are systematic in both our state-of-the-art HDVC and TDWZ with side information refinement codecs.

One concludes that with careful optimizations, our novel distributed video coding systems, equipped with the proposed online SID correlation channel estimation method, can become competitive candidates for a wide variety of uplink-oriented power-constrained lightweight multimedia applications, e.g., wireless visual sensors and wireless surveillance sensors.

In fact, a niche application from the medical domain, which has been investigated for the first time in the literature in this dissertation, is wireless capsule endoscopy [112, 151]. Unlike standard video content characteristics, as encountered in the aforementioned non-medical lightweight applications, the video material acquired with a wireless capsule endoscope features distinctive traits as a result of low frame rates and erratic movements of the capsule [112, 151]. In the following chapter, we will extend the coding tools introduced so far, namely, OBMEC-based side information generation and online SID correlation channel estimation, so as to engineer a Wyner-Ziv video codec that can effectively address the strict requirements imposed by the important application of wireless capsule endoscopy.
Chapter 6
WYNER-ZIV CODING OF CAPSULE ENDOSCOPIC VIDEO

6.1 INTRODUCTION

Wyner-Ziv video coding, a.k.a. DVC, has been recognized as a potential strategic component for a wide range of lightweight video encoding applications, including visual sensor networks and wireless low-power surveillance [22, 152]. Motivated by this kind of lightweight applications, the previous chapters presented Wyner-Ziv video coding solutions that advanced over the state-of-the-art. In this chapter, a novel Wyner-Ziv video codec which addresses a unique medical application of particular interest in this dissertation, i.e., wireless capsule endoscopy, is presented. Rooted in the work presented in this dissertation, the research group that I belong to has been the first to propose an efficient and low complex Wyner-Ziv video codec for wireless capsule endoscopy [112, 151].

6.1.1 Prologue on Wireless Capsule Endoscopy

Conventional endoscopy, like colonoscopy or gastroscopy, has proven to be an indispensable tool in the diagnosis and remedy of various diseases of the gastrointestinal track. Significant advances in miniaturization have led to the emergence of endoscopic video capturing functionality in a pill [153]. Although the history of ingestible capsules for sensing purposes goes surprisingly back to 1957, it was the semiconductor revolution of the 1990’s that created a rush in the development of miniaturized devices performing detailed sensing and signal processing inside the body [154]. Among the latest achievements in this regard is wireless capsule endoscopy, which aims at providing visual recordings of the human digestive track.

A wireless capsule endoscope is a device at the size of a large pill, composed of a limited lifespan battery, a strong light source, an integrated chip video camera, and a
radio telemetry transmitter. These components are encapsulated in a biocompatible robust ingestible housing resistant to the gastrointestinal’s hostile environment. An illustration of an up-to-date wireless capsule endoscope is depicted in Figure 6.1. Once swallowed, the capsule transmits video of the human digestive track, namely, the esophagus, the stomach, the small bowel and the colon to a sensor array placed around the patient's abdomen [154]. Figure 6.2 shows visual examples of frames extracted from a video sequence obtained by a wireless capsule examination.

Figure 6.1: The Pill-Cam ESO2, a wireless capsule endoscope manufactured by Given Imaging, relative to a one euro coin. (©Nikolaos Deligiannis).

Nowadays, the primary clinical use of capsule endoscopy is to examine areas of the small intestine that cannot be visualized by conventional types of endoscopy such as colonoscopy or esophagogastroduodenoscopy [153]. Beyond small intestine pathology, capsule endoscopy is often preferable in other digestive tract pathologies compared to a more invasive gastroscopy or colonoscopy. For instance, it has been employed for the detection of oesophageal pathology in adults, and for the avoidance of gastroscopy and a general anesthetic in children. Additionally, capsule endoscopy offers a less unpleasant alternative to traditional endoscopy, lowering the threshold for preventive periodic screening procedures, where the large majority of patients are actually healthy. Though not yet a reality, capsule endoscopy is envisioned to constitute an alternative to conventional endoscopy as a first line investigation for many diseases.

The principal drawback of contemporary capsule endoscopes – when compared to conventional endoscopy means – is that they only detect and record but are unable to take biopsies or perform therapy. In case a pathology is diagnosed, a more uncomfortable or even surgical therapeutic procedure is necessary. Nevertheless, because of its valuable diagnostic potential the clinical use of capsule endoscopy has a bright future. Capsule endoscopy has been shown to have a superior positive
Figure 6.2: Snapshots of the (a) esophagus, (b) stomach and (c) small intestine taken by a wireless capsule endoscopic examination. The material was provided by Prof. Dr. Daniel Urbain, Gastroenterology Clinic, Universitair Ziekenhuis Brussel.
diagnosis rate compared to other methods, including push enteroscopy, barium contrast studies, computed tomographic enteroclysis, and magnetic resonance imaging [153].

6.1.2 Technical Challenges

From a technological perspective, wireless capsule endoscopic video transmission poses an interesting engineering challenge. Focusing on the video coding technology part, it is apparent that wireless capsule endoscopes are subjected to severe constraints in terms of available computational capacity and power consumption. Moreover, since the recorded video is used for medical diagnosis, providing high quality decoded video at an efficient compression ratio is of paramount importance.

Contemporary capsule video chips employ conventional coding schemes operating in a low-complexity, intra-frame mode, i.e., Motion JPEG [155], or even no compression at all. Current capsule endoscopic video systems operate at modest frame resolutions, e.g., 256×256 pixels, and frame rates, e.g. 2-5Hz, on a battery life time of approximately 7 hours. Future generations of capsule endoscopes are intended to transmit at increased resolution, frame rate and battery life time and will therefore require efficient video compression at a computational cost as low as possible. Additionally, a video coding solution supporting scalability has an attractive edge, enabling increased focus (in terms of the temporal resolution or the quality level) during the relevant stages of the capsules bodily journey. DVC is a strong candidate to fulfill the technical demands imposed by wireless capsule endoscopy, offering low-cost encoding, scalability and high compression efficiency [22, 112, 151, 152].

Generating high quality side information plays a vital role in the compression performance of a DVC system, as shown in Chapter 4. Intrinsically, producing accurate motion-compensated predictions at the decoder for a wide range of video content, while at the same time constraining the encoding complexity and guaranteeing high compression performance, poses a major challenge. This problem becomes even more intricate in the largely unexplored application of DVC in wireless capsule endoscopy, in which the recorded video material contains extremely irregular motion, due to low frame acquisition rates and the erratic movement of the capsule along the gastrointestinal track. In case of such motion content, blind motion estimation at the decoder, by means of MCI for example, fails to deliver adequate prediction quality. As explained in Chapter 4, two directions have been identified in the DVC related literature to cope with accurate capturing of irregular motion content:

A. One direction refers to the method of joint decoding and motion estimation
which principally enables successive refinement of the side information as more and more Wyner-Ziv information is decoded. We refer to Section 4.5 for our contribution in this context. In the specific application of wireless capsule endoscopy, such an approach has particular consequences. Firstly, as explained in Chapter 4, it introduces structural latency to the codec, due to the fact that the computationally expensive operation of side information generation is performed more than one time at the decoder. Secondly, as it only exploits the temporal correlation in the sequence, highly erratic motion content, where temporal prediction cannot provide a good estimate of the original frame, poses a challenge.

B. The other direction to overcome this problem is to perform hash-driven motion estimation at the decoder. In this way, good quality side information can be produced at the decoder with the aid of the transmitted hash information, without imposing a significant decoding delay. Moreover, the hash can be formed in such a way so as to substitute temporal prediction when the motion characteristics demand so.

For the abovementioned reasons, this chapter focuses on the design of a new efficient hash-driven Wyner-Ziv video codec that successfully addresses the austere requirements of wireless capsule endoscopic video content. The contributions of the presented work are summarized next.

6.1.3 Contributions

The key contribution of this work is that it leads the way in the application of DVC systems in lightweight medical imaging, where the Wyner-Ziv video coding system presented in this chapter achieves high compression efficiency with the additional benefit of low computational encoding complexity.

Concentrating to the particular niche application of wireless capsule endoscopy and referring to the technical aspects, the presented hash-based DVC architecture introducing the following novelties:

- Firstly, in contrast to our hash-based DVC architectures [103, 110, 111] presented in Chapter 4, which employed a bit-plane hash, the presented system generates the hash as a downscaled and subsequently conventionally intra coded version of the original frames. In this fashion, the designed hash conveys more (texture) information about the WZ frame waveform with respect to a bit-plane hash.

- Secondly, unlike our OBMEC [103], and OBMEC/SSM [111] techniques discussed in Chapter 4, the hash is exploited in the design of a novel motion-compensated multi-hypothesis prediction scheme, which is able to adapt to
the regional variations in temporal correlation in a frame by extracting information from the hash when temporal prediction is untrustworthy. Compared to alternative techniques, i.e. [24, 81, 102, 119], the proposed methodology delivers superior performance under strenuous conditions, namely, when irregular motion content is encountered as in for example endoscopic video material, where gastrointestinal contractions can generate severe morphological distortions in conjunction with extreme camera panning.

- Thirdly, this chapter includes a thorough experimental evaluation of the proposed hash-based distributed video coding scheme on (i) conventional test sequences, numerous (ii) traditional endoscopic as well as (iii) wireless capsule endoscopic video content. The experimental results show that, in endoscopic sequences, the proposed DVC outperforms alternative state-of-the-art DVC schemes, including DISCOVER [24], the hash-based DVC from [119], our previous best-performing HDVC [111] codec in Section 4.4 and our TDWZ codec [102] in Section 4.5.1, as well as conventional codecs, namely, Motion JPEG and H.264/AVC [8] Intra.

- Fourthly, this chapter incorporates a detailed study of the encoding complexity and buffer size requirements of the proposed system. The analysis offered pays attention to the applicability of the proposed Wyner-Ziv video coding system, appraising its hands-on potential in wireless capsule endoscopy.

This chapter continues as follows. Section 6.2 covers a comprehensive discussion on the designed DVC architecture for wireless capsule endoscopy. The proposed method for hash formation and compression is detailed in Section 6.3. Section 6.4 expands on the new approach to perform OBME, which except for temporal predictors uses predictors from the hash as well. The presented codec is experimentally evaluated in Section 6.5, not only using standard test sequences but using endoscopic test video as well. Section 6.6 draws the conclusions of the work given in this chapter.

6.2 DVC ARCHITECTURE FOR WIRELESS CAPSULE ENDOSCOPY

A graphical overview of our DVC architecture, targeting the application of wireless capsule endoscopy, is depicted in Figure 6.3. We briefly refer to this codec as Endoscopic Distributed Video Coding, a.k.a., EDVC.
6.2.1 The Encoder

Every incoming frame – captured by the sensor’s camera – is categorized as a key or a WZ frame, denoted by $I$ and $X$ respectively, as to construct groups of pictures (GOP) of the form $IX...X$. Similar to the architectures proposed in Chapter 4, the key frames are separately coded using a conventional intra codec, e.g., H.264/AVC Intra [8] or Motion JPEG. The WZ frames on the other hand are encoded in two stages.

In the first stage, for every WZ frame, the encoder first generates and codes a hash, which will assist the decoder during the motion estimation process. The hash formation and compression procedures are explicitly detailed in Section 6.3.1.

In the second stage, a layered transform-domain Wyner-Ziv bit-stream is formed for each WZ frame, providing efficient compression [12] and scalable coding [17]. In line with the DVC architectures introduced in Chapter 4, the WZ frames are first transformed with a $4\times4$ integer approximation of the DCT [8] and the obtained coefficients are subsequently assembled in frequency bands. Each DCT band is independently quantized using a collection of predefined QMs [24], where the DC and the AC bands are quantized with a uniform and double-deadzone scalar quantizer, respectively. The quantized symbols are translated into binary codewords and passed to a LDPC Accumulate (LDPCA) encoder [48], assuming the role of SW encoder.

Recall that the LDPCA [48] encoder realizes Slepian and Wolf’s random binning argument [45] through linear channel code syndrome binning. An overview of the LDPCA code can be found in Section A.3. The encoder stores the accumulated syndrome bits, derived by the LDPCA encoder, in a buffer and transmits them incrementally upon the decoder’s request using a feedback channel, as explained in Section A.3. Considering that the human body is no place for operational failure, contemporary wireless (implantable) sensors – including capsule endoscopes – support bidirectional communication between the implantable biosensor and the external control terminal [154, 156, 157]. Supporting bidirectional communication enables important functionality, including robust transmission of data, capsule navigation and control. For instance, the wireless biomedical implant system studied in [156] supported bidirectional communication between the sensor and the control terminal, where the transmission in each direction took place in a half-duplex mode.
Figure 6.3: Schematic overview of the proposed Wyner-Ziv video codec for wireless capsule endoscopy.
Hence, a feedback channel from the encoder to the decoder is a viable solution for the pursued application of wireless capsule endoscopy. The effect of the employed feedback channel on the decoding delay, and in turn on the buffer requirements at the encoder of a wireless capsule endoscope, is studied in Section 6.5.4.

The focus of this work is to successfully target various lightweight applications by improving the compression efficiency of Wyner-Ziv video coding while maintaining low computational cost at the encoder. Hence, in order to accurately evaluate the impact of the proposed techniques on the RD performance, the proposed system employs LDPCA codes, which are also used in the state-of-the-art reference codecs of [24, 102]. These reference codecs are used to benchmark the RD performance of the proposed system. Recall from 2.5.3 that for distributed compression under a noiseless transmission scenario the syndrome-based Slepian-Wolf scheme [45] is optimal [49]. Nevertheless, in order to address distributed joint source-channel coding (DJSCC) in a noisy transmission scenario the parity-based [49] Slepian-Wolf scheme needs to be deployed. In the latter, parity-check bits are employed to indicate the Slepian-Wolf bins, thereby achieving equivalent Slepian-Wolf compression performance at the cost of an increased codeword length [49].

It is important to mention that the proposed Wyner-Ziv encoder encodes the entire original WZ frame, instead of coding the difference between the original frame and the reconstructed hash. We note that this feature is also appearing in the bit-plane-based codec given in Section 4.4; but it is in contrast to other hash-driven Wyner-Ziv schemes, e.g., [119], and to our TDFDVC system presented in Section 4.3. In the codec presented in this chapter the motivation for this decision is twofold. The first reason stems from the nature of the hash. Namely, coding the difference between the WZ frame and the reconstructed hash would require decoding and interpolating the hash at the encoder, an operation which is computationally demanding (see Section 6.3.2) and would pose an additional strain on the encoder’s memory demands. Second, compressing the entire WZ frame with linear channel codes enables the extension of the scheme to the DJSCC case [49], thereby providing error-resilience for the entire WZ frame if a parity based Slepian-Wolf approach is followed.

### 6.2.2 The Decoder

The decoder first conventionally intra decodes the key-frame bit stream and stores the reconstructed frames in the reference frame buffer. The hash bit-stream is decoded with the appropriate conventional intra codec and the reconstructed hash is then upscaled to the original WZ frame’s resolution. Further comments on the interpolation of the hash are given in Section 6.3.2. Previously decoded frames and
the interpolated hash frame are then used to produce a prediction of the WZ frame. As explained in Section 6.4, the OBME method is properly adjusted so as to support online tuning to the spatial variations in temporal correlation in a frame by obtaining information from the coded hash in case temporal prediction is unreliable.

The derived prediction is subsequently DCT transformed and converted to soft-input information using the online successively refined correlation channel estimation algorithm presented in Section 5.4.2. Recall that the latter algorithm is specifically engineered so as to be independent of the considered DVC architecture. Once all the bit-planes of a DCT band of a WZ frame are LDPCA decoded, the obtained binary tuples are combined to form the decoded quantization indices of the coefficients of the band.

The decoded quantization indices are thereafter fed to the MMSE reconstruction module (see Section 2.7.2), which performs inverse quantization using the side information and the lastly updated SID correlation channel statistics. Finally, the inverse DCT transform provides the reconstructed frame $\hat{X}$ in the spatial domain. The reconstructed frame is now ready for display and is stored in the reference frame buffer, serving as a reference for future temporal prediction.

### 6.3 Hash Formation and Compression

Apart from assisting motion estimation at the decoder – as in contemporary hash based systems – the proposed hash code is designed to also act as a candidate predictor for pixels for which the temporal correlation is low. To this end, the proposed hash formation and coding scheme conveys more texture information with respect to existing methods in which the hash consists of a number of most significant WZ frame bit-planes (see Chapter 4), of coarsely sub-sampled and quantized versions of blocks [118], or of quantized low frequency DCT bands [126] in the WZ frames. This feature is of particular significance especially when difficult-to-capture endoscopic video content is coded.
6.3.1 Hash Encoding

Our Wyner-Ziv video encoder creates an efficient hash that consists of a low quality version of the downsized original WZ frames. We refer to Figure 6.4 for a schema of the proposed hash formation and encoding procedures. In contrast to our previous SDUDVC and TDFDVC architectures (see Section 4.2.1 and 4.3, respectively), where the dimensions of the hash were equal to the dimensions of the original input frames, coding a hash based on the downsampled WZ frames reduces the computational complexity. In particular, every WZ frame undergoes a downscaling operation by a factor \( d \in \mathbb{Z}_+ \). To limit the involved operations, straightforward downsampling is applied. Foregoing a low-pass filter to bandlimit the signal prior to downsampling runs the risk of introducing undesirable aliasing artifacts. However, experimental evidence has shown that the impact on the overall RD performance of the entire system does not outweigh the computational complexity incurred by the use of state-of-the-art downsampling filters, e.g. Lanczos filters [158].

After the dimensions of the original WZ frames have been reduced, the result is coded using a conventional intra video codec, exploiting spatial correlation within the hash frame only. The quality at which the hash is coded has been experimentally selected and constitutes a trade-off between (i) obtaining a constant quality of the decoded frames, which is of particular interest in medical applications, (ii) achieving high RD performance for the proposed system and (iii) maintaining a low hash rate overhead. We notice that constraining the hash overhead comes with the additional benefit of minimizing the hash encoding complexity. On the other hand, ensuring sufficient hash quality, so that the accuracy of the hash-based motion estimation at the decoder is not compromised or so that even pixels in the hash itself could serve as predictors, is important.

We wish to note that, in contrast to other hash-based DVC solutions, [118, 119], the proposed architecture avoids block-based decisions on the transmission of the hash at the encoder side. Although this can increase the hash rate overhead when easy-to-predict motion content is coded, it comes at the benefit of constraining the encoding complexity, in the sense that the encoder is not burdened by expensive block-based comparisons or memory requirements necessary for such mode decision. An additional key advantage of the presented hash code is that it facilitates accurate side information creation using pixel-based multi-hypothesis compensation at the decoder, as explained in Section 6.4. In this way, the presented hash code enhances the RD performance of the proposed system especially for irregular motion content, e.g., (capsule) endoscopic video material.
6.3.2 Hash Decoding

At the decoder-side, the conventionally intra decoded hash bit-stream is upscaled to the original WZ frame’s resolution. The ideal upscaling process consists of upsampling followed by ideal interpolation filtering. The ideal interpolation filter is a perfect low-pass filter with gain $d$ and cut-off frequency $\pi/d$ without transition band [159]. However, such a filter corresponds to an infinite length impulse response $h_{ideal}$, to be precise, a sinc function $h_{ideal}(\nu) = \text{sinc}(\nu/d)$, $\nu \in \mathbb{Z}$, which cannot be implemented in practice.

Therefore, the presented system employs a windowing method [159] to create a filter with finite impulse response $h(\nu)$, namely,

$$h(\nu) = h_{ideal}(\nu) \cdot z(\nu), \quad |\nu| < 3 \cdot d,$$

where the window function $z(\nu)$ corresponds to samples taken from the central lobe of a sinc function, that is,

$$z(\nu) = \text{sinc}\left(\frac{\nu}{3 \cdot d}\right), \quad |\nu| < 3 \cdot d.$$

Such interpolation filter is known in the literature as a Lanczos3 filter [158]. In order to reduce the computational complexity, the process of upsampling followed by interpolation filtering to generate the upscaled frame has been replaced by an equivalent polyphase filterbank with interpolation filters $h_{\gamma}(\nu)$, $0 \leq \gamma < d$ as shown in Figure 6.5. The filters $h_{\gamma}(\nu)$, $0 \leq \gamma < d$ are straightforwardly derived from $h(\nu)$ as

$$h_{\gamma}(\nu) = h(\nu \cdot d + \gamma).$$

As in [160] and similar to the Lanczos downscaling filters, the resulting filter taps are normalized in our implementation to obtain unit DC gain. Note that $h_0(n) = 1$ so that the input samples are preserved by the upscaling process.
6.4 Side Information Generation

After the hash has been restored to the same frame size as the original WZ frames, it is used to perform decoder-side motion estimation. We recall that the quality of the side information is an important factor on the overall compression performance of any Wyner-Ziv codec, since the higher the quality the less channel code rate is required for Wyner-Ziv decoding.

In Chapter 4, we have introduced OBMEC, a novel method which enables multi-hypothesis pixel-based prediction. We have shown that, in contrast to state-of-the-art techniques, e.g., [12, 24, 119, 121], OBMEC mitigates the motion uncertainty at a pixel level, by deriving motion information from more than one motion vector per pixel. Moreover, blocking artifacts are drastically reduced, increasing the subjective quality, which is a strong prerequisite in medical imaging applications. Therefore, building upon our work presented in Chapter 4, the proposed side information generation algorithm for Wyner-Ziv coding of capsule endoscopic video performs a modified bidirectional OBME method.

6.4.1 Modified OBME

Similar to the prediction structures used in state-of-the-art systems, e.g., [12, 24, 119] and in our codecs presented in Sections 4.2, 4.3, 4.4 and 4.5, temporal prediction is carried out in a hierarchical bidirectional organization, using the available hash information and a past and a future reconstructed WZ and/or key frame as references. It is important to note that conversely to our prior OBME methods presented in Chapter 4, in which motion estimation was based on bit-planes (see Sections 4.2 and 4.4) or on partially decoded frames (see Section 4.5.2), this work follows a different approach.

Due to the particular nature of the hash, before motion estimation is initiated, the reference frames are appropriately preprocessed. Specifically, the reference frames are first low-pass filtered undergoing the same downsampling and upsampling operation with the formation of the hash frame (see Section 6.3). In this fashion, the consistency of the resulting motion vectors, and in turn the prediction quality, is improved. Notice that this preprocessing step is the equivalent of (a) performing motion search on the \( b \) MSBs of the references frames, as carried out in Section 4.2, and (b) re-organizing the MSB of each reference frame into four sub-sampled reference frames, as performed in Section 4.4.2.

To offer a clear yet concise presentation of the proposed modified OBMEC algorithm for wireless capsule endoscopy, we briefly introduce the employed notation. Let \( \hat{X} \) be the decoded and upsampled hash of a WZ frame, let \( Y \) be the...
derived side information frame and let \( \tilde{R}_k \) be the preprocessed versions of the reference frames \( R_k \), \( k \in \{0,1\} \), respectively. Also, in line with the notation used in Section 4.2.2, denote by \( Y_m \), \( R_{k,m} \), \( \tilde{X}_m \), \( \tilde{R}_{k,m} \) the blocks of size \( B \times B \) pixels with top-left coordinates \( m = (m_1, m_2) \) in \( Y \), \( R_k \), \( \tilde{X} \) and \( \tilde{R}_k \), respectively.

Similar to the OBME method in Section 4.2.2, the available hash frame is divided into overlapping spatial blocks, \( \tilde{X}_u \), with top-left coordinates \( u = (u_1, u_2) \), using an overlapping step size \( \varepsilon \in \mathbb{Z}_+ \), \( 1 \leq \varepsilon \leq B \). For each overlapping block \( \tilde{X}_u \), the best matching block within a specified search range \( \rho \), is found in the reference frames \( \tilde{R}_k \). In contrast to our earlier bit-plane based OBME method in Sections 4.2 and 4.4.2, the proposed algorithm retains the motion vector \( v = (v_1, v_2) \), \( -\rho < v_1, v_2 \leq \rho \), which minimizes the SAD metric between \( \tilde{X}_u \) and a block \( \tilde{R}_{k,u-v} \), \( v = (v_1, v_2) \), in one of the preprocessed reference frames. The motion search is executed at integer-pel accuracy and the obtained motion field is extrapolated to the original reference frames \( R_k \), \( k \in \{0,1\} \).

As explained in Section 4.2.2, after the execution of the OBME, a temporal predictor block \( R_{k,u-v} \) for every overlapping block \( Y_u \), \( u = (u_{c,1}, u_{c,2}) \) has been identified in one reference frame. Recall that each pixel \( Y(s) \) at position \( s = (i,j) \) in the side information frame belongs to several overlapping blocks \( Y_u \), and therefore, it has a number of associated temporal predictors \( r_{k,u-v} \) being the co-located pixels in the blocks \( R_{k,u-v} \).

### 6.4.2 Reliability Screening

However, some temporal predictors per pixel may stem from rather unreliable motion vectors. Especially when the input sequence was recorded at low frame rates or when the motion content is highly irregular – as might be the case in wireless capsule endoscopic sequences – temporal prediction is not the preferred method for all blocks at all times.

Therefore, to avoid quality degradation of the side information due to untrustworthy predictors, all obtained motion vectors are subjected to a reliability screening. Namely, when the SAD, based on which the motion vector associated with a temporal predictor \( r_{k,u-v} \) was determined, is not smaller than a certain threshold \( T \), the associated temporal predictor is labeled as unreliable. In this case, this particular temporal predictor \( r_{k,u-v} \) for the side information pixel \( Y(s) \) is replaced by the co-located pixel of \( Y(s) \) in the upsampled hash frame, that is, \( \tilde{X}(s) \). In other words, when motion compensation is considered not to be trusted, the hash itself is assumed to convey more dependable information.

This feature of OBME is referred to as hash-predictor-selection (HPS). We remark that the HPS module is tailored to the characteristics of (capsule) endoscopic
video content and is adapted to the hash designed in this chapter. This module is not appearing in our hash-based architectures discussed in Sections 4.2, 4.3 and 4.4, since in these systems the hash is only used for motion estimation and does not contain sufficient information to act as a pixel predictor.

6.4.3 MSE-Optimal Motion Compensation

To continue with the proposed multi-hypothesis pixel-based motion compensation method, we introduce the following notations. Let $\psi_{(s)} = (\psi_0, \psi_1, ..., \psi_\zeta, ..., \psi_{Z-1})$ denote the $Z$ distinct values of the $C$ candidate temporal predictors $r_{k, u-v}$, $c = \{0, 1, ..., C-1\}$, of the side information pixel $Y(s)$. That is, each predictor value $\psi_\zeta$ appears $C_\zeta$ times in the set $r_{k, u-v}$, $c = \{0, 1, ..., C-1\}$ counting $C$ candidate temporal predictors. Also, denote by $X, Y, \Psi$ the random variables corresponding to the WZ frame, the side information frame, and the candidate predictor pixel values, respectively. Moreover, let $f_{X|\Psi}(x \mid \Psi = \psi_\zeta)$ represent the conditional pdf of the WZ source random variable given the candidate predictor value $\psi_\zeta$. This conditional pdf is typically modeled by a peaked distribution (e.g., Laplacian) centered on the candidate predictor value $\psi_\zeta$. For further specifics on this argument the reader is referred to Section 5.2 and to references [107-109, 111, 115-117] in the literature. The proposed motion compensation performs multi-hypothesis pixel-based prediction by combining the values in $\psi_{(s)}$ into a single value $y_{opt} = Y(s)$, which is optimal in MSE sense. The following holds.

**Lemma 6.1:** The expected value of the elements of $\psi_{(s)}$ corresponds to the MSE optimal side information value $y_{opt}$ for the original WZ frame sample $x = X(s)$.

**Proof:** The expected mean squared error when predicting an original sample value $x$ by $y$, given the collection of conditional pdfs $f_{X|\Psi}(x \mid \Psi = \psi_\zeta)$, as defined by the elements of $\psi_{(s)}$, is expressed by

$$\text{MSE}(y) = \sum_{\zeta=0}^{Z-1} p_{\Psi}(\Psi = \psi_\zeta) \int_{-\infty}^{\infty} (x - y)^2 f_{X|\Psi}(x \mid \Psi = \psi_\zeta) dx, \quad (6.4)$$

where $p_{\Psi}(\Psi)$ represents the marginal distribution of the candidate predictor random variable. By putting the partial derivative $\frac{\partial \text{MSE}(y)}{\partial y}$ to zero and solving for $y$ we derive

$$\sum_{\zeta=0}^{Z-1} p_{\Psi}(\Psi = \psi_\zeta) \int_{-\infty}^{\infty} 2(y - x) f_{X|\Psi}(x \mid \Psi = \psi_\zeta) dx = 0 \quad (6.5)$$
\[ \sum_{\zeta=0}^{Z-1} p_{\Psi} (\Psi = \psi_{\zeta}) \int_{-\infty}^{+\infty} f_{X|\Psi}(x \mid \Psi = \psi_{\zeta}) \, dx \]

\[ = \sum_{\zeta=0}^{Z-1} p_{\Psi} (\Psi = \psi_{\zeta}) \int_{-\infty}^{+\infty} x f_{X|\Psi}(x \mid \Psi = \psi_{\zeta}) \, dx. \quad (6.6) \]

Bearing in mind that \( f_{X|\Psi}(x \mid \Psi = \psi_{\zeta}) \) is modeled by a peaked distribution centered on the candidate predictor value \( \psi_{\zeta} \), (6.6) leads to

\[ \Rightarrow y_{\text{opt}} = \sum_{\zeta=0}^{Z-1} \psi_{\zeta} p_{\Psi} (\Psi = \psi_{\zeta}), \quad (6.7) \]

which ends the proof.

By assuming that \( p_{\Psi}(\psi) \) follows a uniform distribution, then from \textit{Lemma 6.1} we have:

\[ y_{\text{opt}} = \frac{1}{C} \sum_{\zeta=0}^{Z-1} C_{\zeta} \psi_{\zeta}. \quad (6.8) \]

Therefore, during compensation, every side information pixel \( Y(s) \) is calculated as the mean value of the predictor values \( \psi_{k, u_c} \), namely,

\[ Y(s) = \frac{1}{C_{k, u_c}} \sum_{u_c} \psi_{k, u_c}, \quad (6.9) \]

where \( C_{k, u_c} \) denotes the number of predictors for pixel \( Y(s) \) and \( \psi_{k, u_c} = r_{k, u_c} \) when \( r_{k, u_c} \) is \textit{reliable} or \( \psi_{k, u_c} = \tilde{X}(s) \) when \( r_{k, u_c} \) is \textit{unreliable}.

We mention that, similar to the approach presented in Section 4.2, the multi-hypothesis motion field generated by OBME is employed in an analogous manner to estimate the chroma components of the side information frame from the chroma components of the reference frames \( R_k \) or the upsampled hash (taking the chroma subsampling into account).

For a concise presentation, the main features of the proposed OBMEC technique with HPS functionality are summarized in Table 6-I, in regard to the characteristics of our OBMEC/SSM technique (see Section 4.4.2).
Table 6-I: The characteristics of the proposed OBMEC with HPS in relation to the features of our OBMEC/SSM technique.

<table>
<thead>
<tr>
<th></th>
<th>OBMEC/SSM</th>
<th>OBMEC with HPS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Method</td>
<td>Hash-based side information generation technique.</td>
<td>Hash-based side information generation technique.</td>
</tr>
<tr>
<td>Motion search (based on)</td>
<td>The MSB of sub-sampled WZ frame (luma component).</td>
<td>Decoded, upsampled and interpolated hash frame.</td>
</tr>
<tr>
<td>Matching criterion</td>
<td>Minimizes the hamming distance.</td>
<td>Minimizes the SAD metric.</td>
</tr>
<tr>
<td>Reference information for</td>
<td>Binary sub-sampled reference frames.</td>
<td>Low-pass filtered reference frames.</td>
</tr>
<tr>
<td>motion search</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Compensation method</td>
<td>Even pixel positions: Weighted average of predictors.</td>
<td>HPS followed by averaging of the predictors.</td>
</tr>
<tr>
<td></td>
<td>Other positions: Average of predictors.</td>
<td></td>
</tr>
<tr>
<td>Output</td>
<td>The side information frame.</td>
<td>The side information frame.</td>
</tr>
<tr>
<td>Additional concealment</td>
<td>No</td>
<td>No</td>
</tr>
</tbody>
</table>

6.5 Experimental Evaluation

The proposed hash-based Wyner-Ziv video codec was evaluated on a broad collection of video material. The experimental results have been divided into three distinct parts. Namely, first, the proposed system is compared against a set of relevant alternative video coding solutions using traditional test sequences. Like so, the performance of the system is compared against the state-of-the-art under typical test video conditions. The second part comprises the experimental validation of our system in the application of wireless capsule endoscopy, comparing its performance mainly against coding solutions currently used for the compression of endoscopic video. The third and last part elaborates on the encoding complexity and the encoder buffer size requirements of the proposed architecture.

We begin by defining the configuration elements of the proposed system, which are common to both types of input video. Namely, the motion estimation algorithm was configured with an overlap step size $e = 4$, the size of the overlapping blocks was set to $B = 16$ and the threshold was chosen $T = 400$. The motion search was executed in an exhaustive manner at integer-pel accuracy within a search range of $\rho = \pm 16$ pixels. The downscaling factor to create the hash was fixed at $d = 2$. 
6.5.1 Evaluation on Conventional Test Sequences

Regarding the performance evaluation of the proposed hash-based DVC on conventional video sequences, comparisons were conducted against a collection of state-of-the-art reference codecs, namely, DISCOVER \(^{24}\) [24], the hash-based scheme in [119], H.264/AVC [8] Intra, and our previous SDUDVC, TDFDVC, HDVC and TDWZ \(^{25}\) systems presented in Sections 4.2.1, 4.3, 4.4 and 4.5.1, respectively.

Complying with the test conditions followed throughout this dissertation (see also Sections 4.6 and 5.5), comparative tests were carried out on the complete Foreman, Soccer, Carphone and Silent sequences, at QCIF resolution and at a frame rate of 15Hz. To assess the RD performance of the codec in different GOPs, results are depicted for GOP sizes of 2, 4 and 8 frames. Also, analogous to Sections 4.6 and 5.5, in order to express the difference in the coding performance in terms of the Bjøntegaard Delta (BD) \(^{138}\) metric, four RD points have been drawn corresponding to QMs 1, 5, 7 and 8 of [24].

In this experimental setting, the hash and the key frames of our proposed system were coded with the H.264/AVC Intra codec (Main profile), since the assessed codecs employ H.264/AVC Intra as well. For a fair comparison, the employed quality parameters (QPs) (per RD point and per sequence) of the key frames are exactly the same with the ones employed in the reference Wyner-Ziv codecs (see [24] and Section 4.6). Similar to the method used to find the key frames’ QPs in [24], an offline iterative scheme has been employed to determine the hash QPs in our codec. The process \(^{26}\) was carried out on the first 15 frames of a sequence and on a GOP of 2 (this GOP size was also used in [24] to determine the QPs for the key frames). The relative standard deviation (RSD) of the PSNR values was used as a metric of the quality fluctuation of the decoded sequence (see also Section 4.6.3). The parameters used and the resulting RSD per sequence and RD point are reported in Table 6-II. Although the proposed codec supports chroma (YUV) encoding (see Section 6.4), the experimental results presented in this section are only obtained for the

\(^{24}\) The experimental results of DISCOVER [24] have been obtained using the executable of the DISCOVER codec which is available on the project’s website [24].

\(^{25}\) For a meaningful and fair comparison, side information refinement by means of DC-OBME is switched off in the TDWZ system. This is because side information refinement is not included in the presented EDVC system either.

\(^{26}\) Given an RD point and a number of iterations, the process starts from a specific hash QP value (QP_hash=QP_key+1), and calculates the total and the hash rate, and the resulting RSD of the decoded frames (both key and WZ frames). If the RSD is lower that a strict threshold, the QP and the rate values are stored; otherwise they are discarded. Next, the hash QP is increased and the algorithm continuous till it reaches a given number of iterations. Out of the retained QPs, the one which minimizes the total rate is chosen as the best for the specific rate point. In case of equal total rates the highest QP value is selected.
luma (Y) component to allow a meaningful comparison with prior art [24, 119].

Table 6-II: Employed quantization parameters for the key, the hash and the WZ frames as well as the resulting RSD for the entire sequence. The RSD is given by

\[ RSD(\%) = 100 \times \frac{\sigma_{PSNR}}{\mu_{PSNR}}, \]

where \( \sigma_{PSNR} \) and \( \mu_{PSNR} \) are the standard deviation and the mean of the PSNR values.

<table>
<thead>
<tr>
<th>Carphone</th>
<th>RD point 1 (QPM1)</th>
<th>RD point 2 (QPM4)</th>
<th>RD point 3 (QPM7)</th>
<th>RD point 4 (QPM8)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Key frame QP</td>
<td>40</td>
<td>34</td>
<td>29</td>
<td>25</td>
</tr>
<tr>
<td>Hash QP</td>
<td>41</td>
<td>40</td>
<td>39</td>
<td>38</td>
</tr>
<tr>
<td>RSD(%)</td>
<td>2.56</td>
<td>2.89</td>
<td>2.78</td>
<td>2.43</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Foreman</th>
<th>RD point 1 (QPM1)</th>
<th>RD point 2 (QPM4)</th>
<th>RD point 3 (QPM7)</th>
<th>RD point 4 (QPM8)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Key frame QP</td>
<td>40</td>
<td>34</td>
<td>29</td>
<td>25</td>
</tr>
<tr>
<td>Hash QP</td>
<td>41</td>
<td>40</td>
<td>39</td>
<td>38</td>
</tr>
<tr>
<td>RSD(%)</td>
<td>2.92</td>
<td>2.97</td>
<td>2.58</td>
<td>1.96</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Silent</th>
<th>RD point 1 (QPM1)</th>
<th>RD point 2 (QPM4)</th>
<th>RD point 3 (QPM7)</th>
<th>RD point 4 (QPM8)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Key frame QP</td>
<td>37</td>
<td>33</td>
<td>29</td>
<td>24</td>
</tr>
<tr>
<td>Hash QP</td>
<td>40</td>
<td>39</td>
<td>38</td>
<td>37</td>
</tr>
<tr>
<td>RSD(%)</td>
<td>2.29</td>
<td>1.02</td>
<td>0.54</td>
<td>2.38</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Soccer</th>
<th>RD point 1 (QPM1)</th>
<th>RD point 2 (QPM4)</th>
<th>RD point 3 (QPM7)</th>
<th>RD point 4 (QPM8)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Key frame QP</td>
<td>44</td>
<td>36</td>
<td>31</td>
<td>25</td>
</tr>
<tr>
<td>Hash QP</td>
<td>45</td>
<td>42</td>
<td>41</td>
<td>38</td>
</tr>
<tr>
<td>RSD(%)</td>
<td>4.41</td>
<td>3.29</td>
<td>2.96</td>
<td>2.73</td>
</tr>
</tbody>
</table>

The experimental evaluation of the proposed EDVC scheme against DISCOVER, the hash-based codec in [119] and H.264/AVC [8] Intra is illustrated in Figure 6.6 to Figure 6.9, whereas the Bjøntegaard deltas [138] against our previously presented DVC systems are tabulated in Table 6-III.

The experimental results in Figure 6.6 to Figure 6.9 show that the proposed hash-based DVC regularly outperforms the DISCOVER [24] codec. Notice that when the size of the GOP and the amount of motion in the sequence increases, the overall compression performance of the DISCOVER codec notably decreases with respect to the proposed DVC, which is mainly due to the quality degradation of DISCOVER’s MCI-based side information generation. Hence, the proposed system outperforms DISCOVER in Foreman, Carhone and Soccer, all of which contain medium to complex motion patterns (above all Soccer), and this for all GOP sizes. Especially for a GOP size of 8, the recorded gains are significant with BD rate savings [138] of 24.77%, 7.53% and 32.13%, in Foreman, Carhone and Soccer, respectively. Comparing the compression performance of both DVC systems on the Silent sequence, which contains a low amount of motion activity, the MCI-based DISCOVER slightly surpasses the proposed DVC. This is due to the fact that in low-motion sequences a hash is not required to accurately capture the motion pattern at the decoder, as this can be simply achieved via interpolation. The incurred loss in RD performance is albeit reasonable, at the level of 7.9% for GOP2, and decreasing
with growing GOP size to 5.4% for GOP8.

Analogous observations can be made when comparing the proposed EDVC system with our state-of-the-art MCI-based TDWZ codec (without DC-OBME refinement) given in Section 4.5.1. Namely, for the reasons explained above, in low-motion sequences and short GOPs the proposed EDVC scheme falls behind our MCI-based codec. However, the numbers change drastically when the motion in the sequence becomes irregular – as it is the case in (capsule) endoscopic video content – and when the GOP size grows larger. Specifically, observe that in Carphone, Foreman and Soccer, in GOP8, corresponding BD rate savings of 5.20%, 18.17% and 26.35% have been obtained versus TDWZ. We remark that our MCI-based TDWZ systematically outperforms DISCOVER, and therefore the gains brought by EDVC compared with the former are less than the ones against DISCOVER. Essentially, this experimentation demonstrates the superiority of the EDVC system in competition with MCI-based codecs in difficult-to-capture motion content. In this regard, in the next experimental setting, i.e., using endoscopic sequences, remarkable compression gains over our MCI-based TDWZ system will be reported.

To further evaluate the performance of our proposed scheme against DVC architectures that deploy hash-based motion estimation, the coding results of [119] are included in Figure 6.6 and Figure 6.7 (Ascenso et al. [119] do not report compression results on Carphone and Silent). Recall that the hash-based Wyner-Ziv video codec in [119] combines MCI with hash-driven motion estimation using low quality H.264/AVC Intra coded Wyner-Ziv blocks to generate side information. Even though the codec of Ascenso et al. [119] typically advances over MCI-based systems, e.g., DISCOVER, our proposed hash-based solution generally exhibits higher performance bringing BD rate savings of 17.68% and 12.18% in Foreman and Soccer, in GOP8, respectively.

Comparing the proposed EDVC system with our previously presented hash-based DVC schemes, the results in Table 6-III, presented in terms of Bjontegaard deltas [138], clearly show that the former brings notable compression improvements over the basic SDUDVC architecture as well as its extension to the transform domain, namely, the TDFDVC scheme. Except for better compression performance, the EDVC system comes with the additional benefit that it applies Wyner-Ziv coding on the entire WZ frames’ pixel values. Therefore, in contrast to the SDUDVC and TDFDVC architectures, the proposed system can directly be extended to the DJSCC case, thereby providing error resilience to the WZ frames’ waveform.
Figure 6.6: Experimental results obtained on traditional test sequences. The proposed EDVC is compared against DISCOVER, the system in [119] and H.264/AVC Intra. The figure shows the RD performance corresponding to Foreman at QCIF resolution, a frame rate of 15Hz, and a GOP of (a) 2, (b) 4 and (c) 8. Only the Y component is coded.
Figure 6.7: Experimental results obtained on traditional test sequences. The proposed EDVC is compared against DISCOVER, the system in [119] and H.264/AVC Intra. The figure shows the RD performance corresponding to Soccer at QCIF resolution, a frame rate of 15Hz, and a GOP of (a) 2, (b) 4 and (c) 8. Only the Y component is coded.
Figure 6.8: Experimental results obtained on traditional test sequences. The proposed EDVC is compared against DISCOVER and H.264/AVC Intra. The figure shows the RD performance corresponding to Carphone at QCIF resolution, a frame rate of 15Hz, and a GOP of (a) 2, (b) 4 and (c) 8. Only the Y component is coded.
Figure 6.9: Experimental results obtained on traditional test sequences. The proposed EDVC is compared against DISCOVER and H.264/AVC Intra. The figure shows the RD performance corresponding to Silent at QCIF resolution, a frame rate of 15Hz, and a GOP of (a) 2, (b) 4 and (c) 8. Only the Y component is coded.
Table 6-III: Bjøntegaard deltas [138] obtained using the proposed EDVC system against our previously presented DVC systems. Negative rate and positive PSNR delta values represent gains.

<table>
<thead>
<tr>
<th></th>
<th>GOP2</th>
<th></th>
<th>GOP4</th>
<th></th>
<th>GOP8</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>ΔR(%)</td>
<td>ΔPSNR(dB)</td>
<td>ΔR(%)</td>
<td>ΔPSNR(dB)</td>
<td>ΔR(%)</td>
<td>ΔPSNR(dB)</td>
</tr>
<tr>
<td>Foreman</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>vs. SDUDVC</td>
<td>-14.07</td>
<td>1.009</td>
<td>-18.61</td>
<td>1.096</td>
<td>-37.00</td>
<td>2.385</td>
</tr>
<tr>
<td>vs. TDFDVC</td>
<td>-1.20</td>
<td>0.021</td>
<td>-3.59</td>
<td>0.174</td>
<td>-10.26</td>
<td>0.604</td>
</tr>
<tr>
<td>vs. TDWZ</td>
<td>10.31</td>
<td>-0.550</td>
<td>-5.27</td>
<td>0.311</td>
<td>-18.17</td>
<td>1.126</td>
</tr>
<tr>
<td>vs. HDVC</td>
<td>5.95</td>
<td>-0.362</td>
<td>8.37</td>
<td>-0.475</td>
<td>8.81</td>
<td>-0.480</td>
</tr>
<tr>
<td>Soccer</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>vs. SDUDVC</td>
<td>-28.44</td>
<td>1.598</td>
<td>-33.77</td>
<td>1.904</td>
<td>-46.69</td>
<td>2.769</td>
</tr>
<tr>
<td>vs. TDFDVC</td>
<td>-0.96</td>
<td>0.021</td>
<td>-4.20</td>
<td>0.199</td>
<td>-9.10</td>
<td>0.455</td>
</tr>
<tr>
<td>vs. TDWZ</td>
<td>-10.70</td>
<td>0.575</td>
<td>-20.68</td>
<td>1.233</td>
<td>-26.35</td>
<td>1.668</td>
</tr>
<tr>
<td>vs. HDVC</td>
<td>1.22</td>
<td>-0.055</td>
<td>0.67</td>
<td>-0.025</td>
<td>-2.53</td>
<td>0.134</td>
</tr>
<tr>
<td>Carphone</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>vs. SDUDVC</td>
<td>-3.01</td>
<td>0.254</td>
<td>-11.89</td>
<td>0.764</td>
<td>-32.77</td>
<td>1.727</td>
</tr>
<tr>
<td>vs. TDFDVC</td>
<td>-2.89</td>
<td>0.170</td>
<td>-1.45</td>
<td>0.067</td>
<td>-5.80</td>
<td>0.341</td>
</tr>
<tr>
<td>vs. TDWZ</td>
<td>5.89</td>
<td>-0.299</td>
<td>2.74</td>
<td>-0.093</td>
<td>-5.20</td>
<td>0.306</td>
</tr>
<tr>
<td>vs. HDVC</td>
<td>6.20</td>
<td>-0.395</td>
<td>8.96</td>
<td>-0.512</td>
<td>9.42</td>
<td>-0.506</td>
</tr>
<tr>
<td>Silent</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>vs. SDUDVC</td>
<td>-4.11</td>
<td>0.313</td>
<td>0.72</td>
<td>-0.004</td>
<td>-5.80</td>
<td>0.351</td>
</tr>
<tr>
<td>vs. TDFDVC</td>
<td>-1.89</td>
<td>0.139</td>
<td>-2.80</td>
<td>0.171</td>
<td>-5.18</td>
<td>0.283</td>
</tr>
<tr>
<td>vs. TDWZ</td>
<td>11.49</td>
<td>-0.614</td>
<td>10.26</td>
<td>-0.469</td>
<td>6.98</td>
<td>-0.260</td>
</tr>
<tr>
<td>vs. HDVC</td>
<td>5.95</td>
<td>-0.359</td>
<td>7.87</td>
<td>-0.451</td>
<td>7.45</td>
<td>-0.402</td>
</tr>
</tbody>
</table>

When compared with our previous best-performing HDVC system, given in Section 4.4, EDVC typically lacks in performance, as shown in Table 6-III. Intrinsically, only in sequences with very complex motion attributes, e.g., Soccer, EDVC can meet or surpass the performance of our previous HDVC architecture. This behavior is mostly accredited to the relatively high amount of hash information communicated by EDVC with respect to the corresponding hash rate required by the HDVC system. As discussed in Section 6.3, the hash in EDVC provides more reliable prediction information when the motion becomes very complex. Unlike HDVC, the EDVC codec presented in this chapter is specifically designed for the video technology requirements of the wireless capsule endoscopy application. In fact, as shown later, when capsule endoscopic video content is coded, EDVC systematically outperforms HDVC.

Lastly, the proposed DVC is compared with H.264/AVC Intra, which represents the low-complexity configuration of the state-of-the-art traditional coding paradigm. One can observe from Figure 6.9 that in low-motion sequences the proposed codec
is superior to H.264/AVC Intra, bringing BD rate savings of up to 26.7% in Silent, GOP8. However, under difficult motion conditions – like in Soccer – H.264/AVC Intra is very efficient compared to DVC systems, which is in agreement with the results shown in Figure 6.7. We emphasize that the encoding complexity of H.264/AVC Intra is much higher than any of the presented DVC solutions, as discussed in Section 6.5.3.

In Figure 6.10 we schematically depict the contribution of the LPDCA, the hash, and the key-frame rate to the total rate of the proposed coding system. The results show that as the GOP size increases, namely as more WZ frames are coded, the hash and the LDPCA rates increase, whereas the key-frame rate decreases. Notice also that, for a given sequence and GOP size, as the total rate increases from RD point 1 to 4, the contribution of the hash rate diminishes in favor of the LDPCA rate.

Figure 6.10: Contribution of the LDPCA (WZ), the hash and the key-frame rate to the total rate of the proposed EDVC system for (a) Foreman and (b) Soccer. The results are provided for GOP sizes of 2, 4, and 8 frames and the four rate points.
Furthermore, the relative contribution of the hash rate to the total rate becomes smaller when high-motion sequences are coded [see Figure 6.10(b)] since relatively more LDPCA rate is spent.

Examples of the subjective performance of the proposed system with respect to DISCOVER for Foreman and Soccer are illustrated in Figure 6.11 and Figure 6.12, respectively. From the depicted sequence snapshots it is evident that the proposed codec delivers significantly better visual quality than DISCOVER. Particularly, due to the employment of the proposed pixel-based multi-hypothesis technique, which generates high quality side information, WZ frames decoded with the presented codec do not suffer from blocking artifacts and ghosting effects.

*Figure 6.11: Visual snapshots from the decoded Foreman sequence. (left) The DISCOVER codec (GOP8, 186.48kbps, 30.6dB), (right) the proposed hash-based EDVC (GOP8, 192.84kbps, 32.56dB).*

*Figure 6.12: Visual snapshots from the decoded Soccer sequence. (left) The DISCOVER codec (GOP8, 368.9kbps, 33.99dB), (right) the proposed EDVC system (GOP8, 362.17kbps, 35.70dB).*
Figure 6.13: Experimental results on data acquired from a wireless capsule endoscope: (a) “Capsule Test Video 1”, (b) “Capsule Test Video 2”, (c) “Capsule Test Video 3”, and (d) “Capsule Test Video 4”. The RD performance of EDVC (with and without HPS) is compared against that of Motion JPEG and HDVC. All three Y, U, and V components are coded. The average YUV PSNR is given by $\text{PSNR}_{YUV} = \frac{4 \times \text{PSNR}_Y + \text{PSNR}_U + \text{PSNR}_V}{6}$. 

6.5.2 Evaluation on Endoscopic Video Sequences

A major contribution in this chapter is the assessment of Wyner-Ziv coding for endoscopic video data, characterized by its unique content. Similar to the process described in Section 6.5.1, the quantization parameters of the WZ frames, the key frames, and the hash are meticulously selected so as to retain high and quasi-constant decoded frames’ quality, as demanded by medical applications. Furthermore, in order to deliver high quality decoding under the strenuous conditions of highly irregular motion content and low frame acquisition rates, the proposed codec employs a GOP size of 2.

Initially, in order to prove the potential of its application in contemporary wireless capsule endoscopic technology, the proposed codec has been appraised using four capsule endoscopic test video sequences visualizing diverse areas of the gastrointestinal track. These sequences were extracted from extensive capsule endoscopic video material of 2 capsule examinations from 2 arbitrary volunteers performed at the Gastroenterology Clinic of the Universitair Ziekenhuis Brussels, Belgium. In the aforementioned clinical examinations, the capsule acquisition rate was two frames per second with a frame resolution of 256×256 pixels. The obtained test video sequences are termed “Capsule Test Video 1” to “Capsule Test Video 4” in the remainder of this dissertation.

In the set of experiments comprising capsule endoscopic video content, Motion JPEG has been set as benchmark, since this technology is commonly employed in up-to-date capsule endoscopes [155]. Furthermore, the results obtained with our previous best performing hash-based codec, i.e., the HDVC scheme from Section 4.4, have been included in the comparison. To enable a fair comparison, Motion JPEG has also been employed to code the key and the hash frames in the proposed EDVC as well as the key frames in the HDVC system. We note that in this experimental setting the luma (Y) and the chroma (U and V) components were coded. The results, which are illustrated in Figure 6.13, show that the proposed codec generally outperforms Motion JPEG for the capsule endoscopic sequences. In particular, in “Capsule Test Video 1” and “Capsule Test Video 2” the proposed codec brings average Bjøntegaard [138] rate savings of respectively 6.16% and 9.33% against Motion JPEG.

What is more, the experimental results in Figure 6.13 corroborate that the proposed EDVC system systematically outperforms our previous HDVC, bringing

---

27 These volunteers presented no evidence of gastrointestinal pathologies.
28 These sequences were transformed to the YUV 4:2:0 format supported by the proposed codec.
significant BD rate savings of up to 14.03%, 17.79%, 12.07% and 13.53% in “Capsule Test Video 1” to “Capsule Test Video 4”, respectively. The corresponding BD PSNR gains are up to 1.30dB, 1.31dB, 0.81dB and 0.86dB. These quality improvements are very important for a correct diagnosis in medical imaging applications. These gains are due to the new coding concepts introduced in this chapter, namely, the novel hash and the modified OBMEC with HPS functionality. Conversely to the HDVC system, the aforementioned features enable the codec to select side information predictors not only from temporally adjacent frames but also from the hash itself.

Figure 6.13 also evaluates the impact of the flexible scheme that enables the proposed OBME method to identify erroneous motion vectors and to replace the temporal predictor pixel with the decoded and interpolated hash. The results show that the proposed system with the HPS module remarkably advances over its equivalent that solely retains predictors from the reference frames. Specifically, in “Capsule Test Video 1” to “Capsule Test Video 4”, adding the HPS functionality results in BD [138] rate improvements of 21.1%, 16.02%, 12.93% and 12.06%, respectively.

The visual assessment of the proposed codec (with HPS) compared to Motion JPEG for a WZ frame of “Capsule Test Video 1” and “Capsule Test Video 2” is depicted in Figure 6.14 and Figure 6.15, respectively.

Future generations of capsule endoscopic technology aim at diminishing the quality difference with respect to conventional endoscopy by increasing the frame rate and resolution. Therefore, to confirm its capability under these conditions, the proposed Wyner-Ziv video codec is evaluated using conventional endoscopic video sequences monitoring diverse parts of the digestive track of several patients. The endoscopic test video sequences considered in this experimental setting have a frame rate of 30Hz with a frame resolution of 480×320 pixels. These endoscopic test video sequences are further referred to as “Endoscopic Test Video 1” to “Endoscopic Test Video 6”. In this experiment, the proposed codec employs H.264/AVC Intra (Main profile) to code the key and the hash frames. Notice that the H.264/AVC Intra codec constitutes a recognized reference for medical video compression, e.g. [161].
Figure 6.14: Visual snapshots from the decoded “Capsule Test Video 1” sequence: (a) Motion JPEG (72.94 kbps, 39.07dB), (b) the proposed EDJC scheme (77.1 kbps, 39.50dB).
Figure 6.15: Visual snapshots from the decoded “Capsule Test Video 2” sequence: (a) Motion JPEG (49.74kbps, 38.56dB), (b) the proposed EDVC scheme (50.88kbps, 39.25dB).
In Figure 6.16 the proposed DVC system (with and without HPS) is evaluated against the H.264/AVC Intra and our previous TDWZ [102] codec in Section 4.5.1. Recall that the latter features an MCI framework comprising overlapped block motion compensation to generate side information at the decoder. The TDWZ [102] codec provides state-of-the-art MCI-based DVC performance, outperforming the DISCOVER [24] codec. Notice that the DISCOVER codec could not be included in the comparison since it does not support the frame resolution of the video data. The results are provided only for the luma component of the sequences “Endoscopic Test Video 1” to “Endoscopic Test Video 4”, since the TDWZ [102] codec does not support chroma encoding. The experimental results – depicted in Figure 6.16 show that the proposed codec (with HPS) delivers significant compression gains over the state-of-the-art TDWZ [102] codec. Specifically, in “Endoscopic Test Video 1” and “Endoscopic Test Video 2” the proposed codec introduces average BD [138] rate savings of 43.14% and 43.37%, respectively. These remarkable compression gains clearly motivate the proposed hash-based Wyner-Ziv architecture comprising our novel motion-compensated multi-hypothesis prediction scheme over MCI-based solutions.

Compared to H.264/AVC [8] Intra, the experimental results in Figure 6.16 show that the proposed codec delivers BD rate savings of 4.1% in “Endoscopic Test Video 2”. In “Endoscopic Test Video 1” and “Endoscopic Test Video 3” the proposed codec falls behind H.264/AVC Intra, incurring a BD rate loss of 3.84% and 0.20%, respectively. Only in “Endoscopic Test Video 4”, which comprises highly irregular motion, the experienced Bjøntegaard [138] rate overhead is notable, mounting to 15.68%. Notice that the benefit of the HPS functionality of the proposed codec is reduced in case of conventional endoscopic video with respect to the capsule endoscopic sequences. This is due to the fact that the former sequences were recorded at a much higher frame rate and contain more temporal correlation. Nevertheless, in “Endoscopic Test Video 4” the HPS module brings BD rate savings of 4.63%.

To benchmark the performance of the presented codec (with HPS) against H.264/AVC when all three Y, U and V components are coded, both systems are also tested on “Endoscopic Test Video 5” and “Endoscopic Test Video 6”. The results, given in Figure 6.17, show that the proposed codec outperforms the competition. Specifically, the proposed codec delivers a significant BD rate reduction of 12.41% and 18.61% in “Endoscopic Test Video 5” and “Endoscopic Test Video 6”, respectively.
Figure 6.16: Experimental results obtained on data acquired from conventional endoscopy: (a) “Endoscopy Test Video 1”, (b) “Endoscopy Test Video 2”, (c) “Endoscopy Test Video 3”, and (d) “Endoscopy Test Video 4”. The RD performance of the proposed EDVC system (with and without HPS) is compared against that of H.264/AVC Intra and that of our TDWZ codec. Only the Y component is coded.
Figure 6.17: Experimental results obtained on data acquired from conventional endoscopy; (a) “Endoscopy Test Video 5” and (b) “Endoscopy Test Video 6”. The RD performance of the proposed EDVC system (with HPS) is compared against that of H.264/AVC Intra. All three Y, U, and V components are coded. The average YUV PSNR is given by $\text{PSNR}_{YUV} = \frac{4 \times \text{PSNR}_Y + \text{PSNR}_U + \text{PSNR}_V}{6}$.

Visual comparisons between our TDWZ [102] system and the proposed codec are given in Figure 6.18 and Figure 6.19. The proposed codec yields significantly better visual quality and does not suffer from blocking artifacts, typically affecting TDWZ [102] at this rate. The superior visual quality delivered by the proposed system compared to the state-of-the-art MCI-based TDWZ codec of [102] confirms the potential of the former in medical imaging applications, where high visual quality is a fundamental demand.
Figure 6.18: Visual snapshots from the decoded “Endoscopy Test Video 1” sequence. (a) Our TDWZ codec [102] (1206kbps, 40.3dB), (b) the proposed EDVC scheme (1204.7kbps, 44.2dB).
Figure 6.19: Visual snapshots from the decoded “Endoscopy Test Video 2” sequence. (a) Our TDWZ codec [102] (1281kbps, 35.43dB), (b) the proposed EDVC scheme (1242kbps, 39.15dB).
6.5.3 Encoding Complexity

Motivated by the low power processing capacity of inbody sensors, low-cost encoding is a key aspect when designing a video compression system for wireless capsule endoscopy. The proposed encoder consists of intra coding of the key frames, as well as Wyner-Ziv coding together with hash formation and compression for the WZ frames. Notice that Wyner-Ziv encoding engenders very low computational demands, simply consisting of the integer approximation of the DCT, quantization, bit-planes extraction and LDPCA encoding. The latter is performed by the multiplication of a binary array with a parity-check matrix. This matrix is sparse, meaning that a low amount of memory (for storage) is needed at the encoder. Conversely, apart from the integer approximation of the DCT and quantization, components that are similar with the ones employed by Wyner-Ziv encoding, the H.264/AVC Intra encoder in the Main Profile deploys several 4×4 and 16×16 intra prediction modes, rate-distortion optimal mode selection, and entropy coding. These features impose considerable computational demands at the encoder. Intrinsically, it is well-proven [24, 88] that the complexity associated to Wyner-Ziv encoding is very low in relation to that of the H.264/AVC Intra encoder.

In contrast to pure TDWZ codecs, e.g. DISCOVER [24], the proposed encoder exhibits higher encoding complexity due to hash coding. However, this complexity overhead is kept low, given that the hash data is at 1/d² of the original frame resolution, and is coarsely coded with H.264/AVC Intra frame coding, which employs fast RDO. In addition, recall that contrary to other hash-based Wyner-Ziv video codecs in the literature, the proposed system does not perform block-based mode decision [118, 119], or temporal correlation exploitation [103, 120], to code the hash data.

Let us formulate a simple yet illustrative model to assess the encoding complexity of the proposed codec, in relation to those of H.264/AVC Intra and the conventional TDWZ [12, 24, 102] architecture. Denote by \( C_{KEY} \) the computational cost spent by H.264/AVC Intra to encode a key frame at a specific quality. Also, let the computational cost paid by the Wyner-Ziv encoder to code a WZ frame at the same quality be signified by \( C_{WZ} = \xi \cdot C_{KEY} \), where \( 0 < \xi \ll 1 \) is a complexity scaling factor. Furthermore, let the computational overhead devoted to hash frame encoding at a lower resolution be given by \( C_{HASH} = C_{KEY} / d^2 \). Notice that the cost associated to the hash formation is trivial, as straightforward downsampling is used. Observe also that for the sake of simplicity the aforementioned hash model overestimates the actual hash computational cost. In reality, the complexity of the hash encoder is further constrained by coding the hash at a low quality and by using fast RDO.
Let \( U \cdot G + 1 \) be the total number of encoded frames of a video sequence, where \( G \) is the GOP size and \( U \) denotes the number of coded GOPs. Remember that due to hierarchical coding the last encoded frame must always be a key frame. In case H.264/AVC Intra codec is used, then the computational complexity to encode the video sequence is

\[
C_{\text{INTRA}} = C_{\text{KEY}} \cdot (U \cdot G + 1).
\]  

(6.10)

Analogously, if a conventional TDWZ architecture is used, the overall encoding computational complexity is

\[
C_{\text{TDWZ}} = (U + 1) \cdot C_{\text{KEY}} + U \cdot (G - 1) \cdot C_{\text{WZ}} = \left( (U + 1) + U \cdot \xi \cdot (G - 1) \right) \cdot C_{\text{KEY}},
\]  

(6.11)

where \((U + 1)\) and \(U \cdot (G - 1)\) represent the number of key and WZ frames in the sequence, respectively. Similarly, the overall encoding computational cost, when the proposed codec is used to encode the video sequence, is given by

\[
C_{\text{EDVC}} = (U + 1) \cdot C_{\text{KEY}} + U \cdot (G - 1) \cdot \left( C_{\text{WZ}} + C_{\text{HASH}} \right) = (U + 1) \cdot C_{\text{KEY}} + U \cdot (G - 1) \cdot \left( \xi \cdot C_{\text{KEY}} + \frac{C_{\text{KEY}}}{d^2} \right) = C_{\text{KEY}} \cdot \frac{(U + 1) \cdot d^2 + U \cdot (G - 1) \cdot (d^2 \cdot \xi + 1)}{d^2}.
\]  

(6.12)

Based on Eqs. (6.10), (6.11), and (6.12), for a given number of coded frames, one can relate the encoding complexity of the proposed hash-based Wyner-Ziv codec with that of H.264/AVC Intra and TDWZ, as parameterized by the GOP size \( G \) and the Wyner-Ziv complexity scaling factor \( \xi \). As explained above, the complexity of Wyner-Ziv encoding is minor with respect to that of H.264/AVC Intra encoding, thus, typical values of \( \xi \) are in the range \( 0 < \xi \ll 1 \). Extensive execution time measurements under controlled conditions\(^{29}\), as performed by the partners of the VISNET I project\(^{24, 88}\) and by the author of this dissertation, demonstrate that this scaling factor is approximately given by \( \xi = 0.15 \). Yet, in order to enable a broad and instructive examination, Table 6-IV compares the encoding complexity of the proposed system with that of H.264/AVC Intra and TDWZ, for several values of

---

\(^{29}\) Encoding execution time tests using the executables of the JM implementation of H.264/AVC and the proposed system were conducted under the same hardware and software conditions. The employed hardware was a personal computer with Pentium D CPU at 3.2GHz, and with 2048MB of RAM. As regards the software conditions, a Windows XP operating system was used, while the executables were obtained using the Visual Studio C++ v8.0 compiler in release mode. We note that both the JM implementation of H.264/AVC and the implementation of our codec have not been optimized for speed.
The results confirm that the encoding complexity of the proposed codec is dominated by H.264/AVC Intra frame coding of the key frames. As a consequence, the lower the number of key frames in the encoded video sequence, i.e., the longer the GOP, the higher the gain in complexity offered by the proposed EDVC with respect to H.264/AVC intra frame coding. In particular, for \( \xi = 0.15 \), the results show that the proposed codec brings an encoding time reduction of approximately 29.79\%, 48.58\%, 56.78\% for a GOP of size 2, 4 and 8, respectively, compared to H.264/AVC Intra.

Table 6-IV: Encoding complexity comparison of the proposed EDVC codec against H.264/AVC Intra and TDWZ for 145 coded frames. Several GOP sizes and Wyner-Ziv complexity scaling factors are considered.

<table>
<thead>
<tr>
<th>( d = 2 )</th>
<th>( \xi = 0.10 )</th>
<th>( \xi = 0.15 )</th>
<th>( \xi = 0.20 )</th>
<th>( \xi = 0.30 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \xi = 0.10 )</td>
<td>32.28</td>
<td>48.58</td>
<td>56.78</td>
<td></td>
</tr>
<tr>
<td>( \xi = 0.15 )</td>
<td>29.79</td>
<td>44.84</td>
<td>52.41</td>
<td>18.33</td>
</tr>
<tr>
<td>( \xi = 0.20 )</td>
<td>27.31</td>
<td>41.11</td>
<td>48.04</td>
<td>17.08</td>
</tr>
<tr>
<td>( \xi = 0.30 )</td>
<td>22.34</td>
<td>33.63</td>
<td>39.31</td>
<td>15.99</td>
</tr>
</tbody>
</table>

As mentioned above, compared to traditional TDWZ codecs, e.g., [12, 24, 102], the proposed codec increases the encoding computational cost because of the additional resources allocated to code the hash. The results in Table 6-IV indicate that this complexity overhead rises with the GOP size, since more hash frames are encoded. For short GOPs, the results show that this overhead is somewhat small, i.e., 17.68\% for \( \xi = 0.15 \).

When compared to Motion JPEG, the proposed codec (although currently not optimized for speed) exhibits similar encoding time but offers superior compression performance. It is worth mentioning that compared to DISCOVER [24], the TDWZ from [102] or Motion JPEG, the proposed codec offers a significant reduction of the encoding rate for a given distortion level. For instance, recall that the experimental results in Figure 6.13 and Figure 6.16 have shown that EDVC brings BD rate savings of 9.33\% and 43.37\% over Motion JPEG and TDWZ, respectively. Such a notable rate reduction causes an important diminution of the processing power.

As shown in [24, 88] and confirmed by our experimentation, the Wyner-Ziv encoding complexity does not notably vary with the WZ frame quantization parameter. However, this is not the case for the encoding complexity of H.264/AVC Intra. To maintain the simplicity of our model, the considered \( \xi \) values correspond to average values over different quality levels.
consumed by the transmission part of a wireless capsule endoscope. Given the
scaling between the processing requirements of multimedia coding and
transmission, the proposed codec can reduce the overall power consumption of a
wireless capsule endoscope compared to the aforementioned codecs.

6.5.4 Encoder Buffer Requirements

The proposed system links the encoder to the decoder via a feedback channel.
Such a reverse channel implies that the encoder is forced to store Wyner-Ziv data in
a buffer pending the decoder’s directives, as mentioned in Section 6.2.1. We analyze
the buffer size requirements imposed on the presented system’s encoder due to the
decoding delay for the wireless capsule endoscopy application scenario. Recall that
the GOP size in this scenario is restricted to 2 frames (see Section 6.5.2). The prime
factors determining the delay are the frame acquisition period $t_F$, the time $t_{SI}$ to
generate a side information frame, the transmission time (time-of-flight) $t_{TOF}$
between encoder and decoder, and the LDPC soft-input soft-output decoding time,
denoted by $t_{SISO}$.

For simplicity, the combined intra encoding and decoding time is at most $t_F$ [94].
Given the fact that the presented system applies bidirectional motion estimation, the
decoding of a WZ frame can commence only after the next key frame (in a GOP of
2) has been decoded. This induces a structural latency, starting from receiving the
WZ frame, of $3 \times t_F$, which corresponds to the acquisition time of two frames, that
is, the WZ frame proper and the next key frame, as well as the encoding and
decoding of the latter. Adding the time to generate the side information and to
perform Wyner-Ziv decoding yields the total time delay\(^{31}\). Hence, the total time that
the WZ frame bits need to be stored at the encoder, measured from the time the
frame is received, is given by

$$3t_F + t_{SI} + 2F \times t_{TOF} + F \times t_{SISO}$$

(6.13)

where $F$ represents the number of feedback requests, soliciting additional
syndrome bits and inducing another LDPC decoding attempt. Since in a GOP of 2,
the encoder receives a WZ frame every $2 \times t_F$, the size of the encoder’s buffer,
expressed in number of frames $L$, is given by

$$L = \text{ceiling}\left[\frac{3t_F + t_{SI} + 2F \times t_{TOF} + F \times t_{SISO}}{2t_F}\right]$$

(6.14)

\(^{31}\) We can fairly assume that Wyner-Ziv encoding and hash compression can be performed
within the period of one (WZ) frame, i.e., $t_F$ ms. The decoder also performs online SID
correlation channel estimation, as detailed in Section 5.4.2. However, as shown in Section.
5.5.3, the presented SID estimation algorithm is very fast, and thus the associated delay can
safely be considered negligible.
Continuing our analysis, the reported capsule and the conventional endoscopic sequences were recorded using a camera with an acquisition rate of 2Hz and 30Hz respectively, corresponding to an acquisition period $t_F$ of 500ms and 33.33ms. Note that the assumption to limit the total intra encoding and decoding time to $t_F$ is not extreme, since this time is typically much lower, in particular for capsule endoscopic sequences.

An estimation of the transmission time $t_{TOF}$ through the body can be made by calculating the velocity of a uniform plane in a lossy medium [162], characterized by its dielectric properties, i.e., the conductivity and permittivity. These values can be calculated based on [163] and [164] for a wide range of body tissues and frequencies. It can be verified that at a frequency of 433MHz [157] the velocity is always greater than 10% of the speed of light through all body tissue cases included in [164], leading to a time-of-flight $t_{TOF}$ in the order of 15ns through 0.5m$^3$ of tissue.

It is clear that the time $t_{SI}$ to generate a side information frame is dominated by OBME. Fortunately, several VLSI designs for hardware implementation of block-based motion estimation have been proposed. Considering the state-of-the-art architecture of [165], full integer-pel motion search can be executed at $4\rho^2 + B - 1$ cycles per macroblock (MB), where $\rho$ and $B$ are the search range and MB size respectively. However, our presented scheme employs bidirectional OBME. Specifically, the total number of overlapping blocks per frame is $(H \cdot V)/\varepsilon^2$, where $H$ and $V$ are the horizontal and vertical frame dimensions and $\varepsilon$ is the overlap size. Hence, based on the VLSI architecture in [165], the total number of cycles per frame is given by $2 \cdot \left(4 \cdot \rho^2 + B - 1\right) \cdot H \cdot V / \varepsilon^2$, where the factor 2 stems from the bidirectionality. Considering a simplified decoding device with a single core CPU running at 800MHz with a 1DIMPS/MHz/core and instantiating the OBME parameters for $\rho = 16$, $\varepsilon = 4$ and $B = 16$, yields a delay of 10.63ms and 24.9ms per frame for the capsule endoscopic ($H = V = 256$) and the endoscopic ($H = 480$, $V = 320$) sequences, respectively.

LDPC decoding of a WZ frame is performed in $t_{SISO}$. If decoding is unsuccessful,
a request through the feedback channel is issued. Supposing that the encoder responds immediately, the decoder receives the next chunk of syndrome bits in $2 \times t_{TOF}$ after which a new LDPC decoding attempt is made. Table 6-V tabulates the average number of feedback channel requests per WZ frame for the four considered RD points for the capsule endoscopic sequences. Taking recent advances in LDPC decoder implementations into account, decoding latencies in the order of 6 $\mu$s have been reported for a codeword length of 2304 bits (802.16e standard) and 25 iterations or of 270 $\mu$s for a codeword length of 64800 bits (DVB-S2 standard) and 50 iterations [166]. For parallel architectures a latency of 280 ns per iteration can be reached [167] for a rate $\frac{1}{2}$ code of 2304 bit codewords. Our system employs a LDPC codeword of 6336 bits and 50 decoding iterations. Considering a worst-case scenario, namely decoding codewords as big as in the DVB-S2 standard and with $F = 100$ feedback requests (see Table 6-V), the total LDPC decoding latency $F \times t_{SISO}$ would be roughly 27 ms.

Table 6-V: Average feedback channel requests per WZ frame for the capsule endoscopic video sequences.

<table>
<thead>
<tr>
<th></th>
<th>RD point 1</th>
<th>RD point 2</th>
<th>RD point 3</th>
<th>RD point 4</th>
</tr>
</thead>
<tbody>
<tr>
<td>Capsule Test Video 1</td>
<td>35.4</td>
<td>41.2</td>
<td>59.3</td>
<td>86.6</td>
</tr>
<tr>
<td>Capsule Test Video 2</td>
<td>33.7</td>
<td>39.7</td>
<td>59.6</td>
<td>85.4</td>
</tr>
<tr>
<td>Capsule Test Video 3</td>
<td>34.1</td>
<td>39.4</td>
<td>55.8</td>
<td>80.9</td>
</tr>
<tr>
<td>Capsule Test Video 4</td>
<td>32.7</td>
<td>40.5</td>
<td>59.0</td>
<td>88.3</td>
</tr>
</tbody>
</table>

Based on the above approximations, (6.14) yields an estimated buffer size of $L = 2$ and $L = 3$ frames for the capsule ($t_F = 500$ ms) and the conventional endoscopic ($t_F = 33.33$ ms) sequences, respectively. Therefore, the analysis provided above shows that the buffer requirements at the encoder are restricted to the WZ syndrome/parity bits of a limited number of WZ frames. This confirms the applicability of the proposed scheme and its hands-on capacity in meeting the stringent demands imposed by the application of wireless capsule endoscopy.

All these figures correspond to Soft-Input Soft-Output (SISO) decoders.
Table 6-VI: The characteristics of the proposed EDVC system relative to our HDVC scheme.

<table>
<thead>
<tr>
<th></th>
<th>HDVC</th>
<th>EDVC</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Domain</strong></td>
<td>Transform domain</td>
<td>Transform domain</td>
</tr>
<tr>
<td><strong>Hash information</strong></td>
<td>The MSB of downsampled WZ frame (luma component).</td>
<td>Downsampled and coarsely intra coded WZ frame.</td>
</tr>
<tr>
<td><strong>Temporal prediction at the encoder</strong></td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td><strong>Side information generation</strong></td>
<td>OBMEC/SSM</td>
<td>OBMEC with HPS</td>
</tr>
<tr>
<td><strong>Feedback channel</strong></td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td><strong>Encoding complexity</strong></td>
<td>Slightly higher than DISCOVER’s due to the hash codec (~1.19% over DISCOVER’s encoding execution time)</td>
<td>Lower than H.264/AVC Intra’s. Similar to Motion JPEG’s. Higher than DISCOVER’s.</td>
</tr>
<tr>
<td><strong>Compression performance</strong></td>
<td>Outperforms DISCOVER, SDUDVC, TDFDVC and the codec in [119].</td>
<td>Standard sequences: Outperforms DISCOVER (medium and high motion sequences), the codec in [119], SDUDVC, TDFDVC. Endoscopic sequences: Outperforms Motion JPEG, H.264/AVC Intra, HDVC, TDWZ.</td>
</tr>
</tbody>
</table>

6.6 CONCLUSIONS

Though Wyner-Ziv video coding schemes have been meticulously researched, little has been achieved in the applicability of such systems. The work presented in this chapter is the first in the literature to demonstrate the effective practical capacity of Wyner-Ziv video coding in the unique and important application of wireless capsule endoscopy. In this context, this chapter has described a novel hash-based Wyner-Ziv video coding architecture, called EDVC, which has been explicitly tailored to the characteristics of wireless capsule endoscopic video content. The characteristics of the proposed EDVC system, relative to our HDVC scheme in Section 4.4, are summarized in Table 6-VI.

Driven by the highly irregular motion patterns typically encountered in
endoscopic video content, a new way to form and compress a hash has been proposed. The engineered hash communicates more WZ frame texture information to the decoder in comparison to our previous bit-plane-based hash formations in Sections 4.2.1 and 4.4. This has been achieved by generating the hash as a downscaled and subsequently coarsely intra coded version of the WZ frame.

The designed hash is utilized by a novel hash-driven motion estimation technique, which generates accurate side information at the decoder. Conversely to other schemes in the literature as well as our previously presented approaches in Chapter 4, the proposed technique performs motion-compensated multi-hypothesis prediction by embracing a new HPS functionality that enables replacing temporal predictors with hash pixel values, when temporal correlation is poor. This feature, which is in tandem with the developed hash, has been experimentally confirmed to significantly boost the compression performance of the system, particularly when highly irregular motion content is coded. Namely, a type of content typically encountered in endoscopic video material, which exhibits highly erratic motion characteristics, e.g., low frame acquisition rates and extreme camera panning, caused by gastrointestinal contractions. Concerning correlation channel estimation, the proposed codec deploys our state-of-the-art online SID estimation method detailed in Chapter 5.

The proposed EDVC system has undergone meticulous experimentation based on a rich experimental setting consisting of (i) standard test video sequences, (ii) wireless capsule endoscopic video content, and (iii) conventional endoscopic video material. The experimental results in standard test video sequences have proven that the proposed EDVC scheme delivers state-of-the-art RD performance, regularly outperforming both MCI-based codecs – e.g., DISCOVER and our previous TDWZ codec in Section 4.5.1 – as well as hash-based systems, including the codec of Ascenso et al. and our SDUDVC and TDFDVC solutions from Sections 4.2.1 and 4.3. The experimental results in standard test video conditions have also demonstrated that, apart from high-motion sequences, the proposed EDVC system typically falls behind our HDVC architecture discussed in Section 4.4. However, in sequences obtained with a wireless capsule endoscope, the proposed EDVC notably improves over the state-of-the-art, that is, Motion JPEG (see [155]), as well as our previous HDVC system. In essence, in contrast to the latter, EDVC delivers Bjøntegaard [138] PSNR gains of up to 1.31dB, which correspond to significant quality improvements of the decoded video. In addition, in sequences acquired with conventional endoscopes (both from gastroscopy and colonoscopy) the experimental evaluation of the EDVC system shows compression improvements against H.264/AVC Intra and our state-of-the-art MCI-based TDWZ codec, which can be up
to 32.13% and 43.37% in terms of Bjøntegaard rate savings.

A detailed study on the encoding complexity has been provided, confirming that, in comparison to H.264/AVC Intra, the proposed EDVC system offers a notable reduction of the encoding computational cost. The achieved reduction in encoding complexity increases with the GOP size as more WZ frames are coded in the sequence. Additionally, considering the rate savings obtained over competing schemes, including, our TDWZ, and HDVC systems as well as Motion JPEG, the proposed codec can prolong the battery life of a wireless capsule endoscope.

Last but not least, motivated by the existing link between the decoder and the encoder via a feedback channel, an analysis on the encoder buffer size requirements of the proposed system has been given. Based on primary elements determining the decoding delay, which have been properly instantiated according to up-to-date hardware implementations, our analysis shows that a relatively small buffer of approximately 2 frames is required for a GOP of size 2. If the quality in capsule endoscopy will raise to that of conventional endoscopy, the buffer size is approximately 3 frames.

Based on the aforementioned observations, we conclude that the proposed Wyner-Ziv video coding system is a promising video compression candidate in wireless capsule endoscopy.
Chapter 7
EPILOGUE AND PROSPECTIVE WORK

7.1 CONCLUSIONS

This dissertation has aimed at the development of novel effective distributed video coding architectures that meet the austere requirements of low encoding complexity, scalability and high compression performance, as imposed by various emerging wireless lightweight multimedia applications. Obligating the decoder to exploit the redundancy in the video signal, efficient distributed video compression poses severe challenges. Novel solutions to these challenges have been proposed that advance over the state-of-the-art both conceptually as well as in terms of the delivered compression performance. Furthermore, an attractive application scenario from the medical domain, namely, wireless capsule endoscopy, has been investigated.

Delivering high-quality motion-compensated prediction at the decoder—without having access to the original frame to be encoded—is a major challenge in DVC, vastly affecting the overall compression performance. The first part of the work included in this dissertation has dealt with the aforementioned design aspect in several DVC architectures. Specifically, the work presented in Chapter 4 has proposed the deployment of overlapped block motion estimation and compensation (OBMEC) as a side information generation tool to engineer efficient DVC systems. Unlike contemporary approaches, the proposed OBMEC supports multi-hypothesis pixel-based motion-compensated prediction, boosting the codec’s rate-distortion performance and the visual quality of the decoded video, especially under irregular motion patterns and large GOPs.

A novel bit-plane-based version of OBMEC has first been introduced as part of SDUDVC, a new hash-based DVC scheme operating in the pixel domain and featuring very low encoding complexity. Although being liberated from the use of a feedback channel, SDUDVC was experimentally proven to outperform the state-of-the-art in feedback-channel-based pixel-domain DVC [85]. Due to the
aforementioned characteristics, one concludes that the proposed SDUDVC system can serve as a competent solution for applications involving low-power visual sensors for data collection and storage. In such an application scenario, the SDUDVC encoder runs at the sensor node encoding the key frames and the MSBs of the luma component of the intermediate frames in the GOP. In case the compressed data is collected from the sensor node, OBMEC is deployed at a powerful decoding device to predict the missing information.

The basic SDUDVC architecture was subsequently expanded to include a new efficient Wyner-Ziv layer, coding the difference between the hash and the original WZ frames in the transform domain. The proposed hash-based TDFDVC system employed a feedback channel for optimal rate control and was designed based on a layered Wyner-Ziv approach offering quality scalability. As a result of these complementary coding tools, TDFDVC has been showed to systematically deliver higher compression performance and to achieve a broader range of rates compared to the SDUDVC scheme. Furthermore, due to the ability of the proposed hash-driven OBMEC to accurately capture the motion at the decoder, our TDFDVC system has been shown to outperform the state-of-the-art DISCOVER [24] codec for sequences with medium to high motion content. A performance diminution versus DISCOVER was observed in low-motion sequences, which was credited to the amount of the hash rate in TDFDVC.

Focusing on hash-based DVC architectures, a novel codec was proposed, which advanced over our prior SDUDVC and TDFDVC schemes. First, the proposed HDVC architecture required a much less amount of hash rate in comparison with our prior schemes. Second, despite diminishing the communicated hash rate, the proposed HDVC system included a new OBMEC/SSM method that improved upon our previous bit-plane-based OBMEC. Third, the entire WZ frame – rather that its difference with the hash – was coded thereby promoting the extension of the codec to the DJSCC case. On account of these developments, significant compression gains were obtained with respect to our previous SDUDVC and TDFDVC systems. Furthermore, the proposed HDVC system has been experimentally confirmed to notably outperform state-of-the-art codecs in the literature, including, DISCOVER and the hash-based system in [119]. It is worth noticing that, compared to DISCOVER, the HDVC scheme imposes a slight increase in the encoding computational resources (due to the hash) but offers a vast reduction of the decoding complexity.

It can therefore be concluded that due to its abovementioned attractive characteristics, the developed HDVC codec can constitute a viable video coding solution for lightweight multimedia applications, such as, wireless visual sensors
networks and wireless surveillance sensors. Except for operating in tandem with a hash, the OBMEC concept has also inspired the design of a novel technique that refined the obtained side information upon decoding the DC coefficient band of a WZ frame. The proposed side information refinement technique has been integrated in a novel TDWZ codec which deploys a new improved MCI method including bidirectional OBMC. On account of our new MCI technique, the developed TDWZ codec improved over the state-of-the-art DISCOVER [24] codec. When our side information refinement technique was switched on, the compression performance was further improved at the expense of additional computational resources and structural latency. Experimentation has reported significant and consistent compression gains over DISCOVER as well as alternative state-of-the-art TDWZ systems featuring side information refinement [121, 133]. It is worth emphasizing that, unlike other methods in the literature, e.g., [121], which perform refinement in a recursive manner, these significant RD improvements were obtained with a single refinement pass. This notable achievement highlights the capacity of the proposed OBMEC as a side information generation tool.

After generating high-quality side information, the Wyner-Ziv video decoder needs to accurately predict the statistical dependency between the source and the side information. As it critically influences the performance of DVC systems, the second part of this dissertation has elaborated on the problem of accurate correlation channel modeling and estimation. Intrinsically, the work presented in Chapter 5 has introduced the innovative concept of SID correlation channel modeling. Unlike prior SII correlation models, the proposed SID paradigm constructed an additive correlation channel model in which the noise depends on the channel input signal, that is, the side information. The performance of Wyner-Ziv coding under the assumption of SID modeling has been theoretically evaluated against that obtained by considering the SII modeling approach. It has been analytically proven and experimentally corroborated that, at the same noise stationarity level, the proposed SID modeling brings Wyner-Ziv compression improvements over SII modeling.

To accurately estimate the SID channel at the decoder, a novel online SID correlation channel estimation technique has been introduced in Chapter 5. Apart from being the first method in the literature to online estimate the SID correlation channel, the proposed technique features further advantages over the state-of-the-art as well. Namely, unlike alternative methods, e.g., [86, 141-143], the proposed algorithm has been designed to support bit-plane-by-bit-plane progressive refinement. By enabling such a fine correlation channel estimation refinement, the algorithm has been proven to bring performance improvements against state-of-the-art techniques [86, 142]. An additional important property of the proposed technique
is that it is not confined to a specific architecture; therefore, it can be deployed by any DVC system. This is not the case for alternative state-of-the-art methods in the literature, e.g., [86, 141-143].

The third part of this dissertation has concentrated on extending the applicability of DVC to a promising field in the domain of medical imaging, namely, wireless capsule endoscopy. In consequence of low acquisition frame rates and erratic movements of the smart pill, capsule endoscopic video generally contains highly irregular motion characteristics. Since such a type of video content notably differs from typical video material, a novel hash-based EDVC architecture has been engineered in Chapter 6. The proposed codec incorporated a new hash, which conveyed more texture information to the decoder compared to our prior hash-based architectures in Chapter 4. In this way, a novel modified OBMEC technique was developed in which the hash was not only exploited to drive motion estimation but also to act as a reliable side information predictor in areas where the temporal correlation was estimated to be low. The proposed EDVC system has been experimentally evaluated on standard test video sequences where it was shown to achieve state-of-the-art Wyner-Ziv video compression performance. When conventional endoscopic and capsule endoscopic video material was compressed, the EDVC system was proven to bring significant and consistent compression improvements over a germane set of state-of-the-art codecs, comprising H.264/AVC Intra, Motion JPEG, our HDVC system and our MCI-based TDWZ codec.

The visual performance of the proposed EDVC codec on broad endoscopic video content was examined by Prof. Dr. Daniel Urbain, head of the Gastroenterology clinic of the Universitair Ziekenhuis Brussel. According to Prof. Dr. Daniel Urbain, the proposed codec clearly increases the visual quality of the decoded endoscopic video, allowing to better distinguish tissue structures and in turn leading to better diagnoses.

It has also been shown that the EDVC system’s encoder is simple, leading to a radical decrease in computational complexity compared to other state-of-the-art codecs such as H.264/AVC Intra. Due to this important advantage, the proposed codec can lead to increased diagnostic yield, since endoscopic capsules – equipped with the proposed codec – can further approach the end of the colon thanks to an increased battery lifetime. The decrease in power requirements offered by our novel codec can also be exploited to improve the capabilities of wireless capsule endoscopes by increasing the frame acquisition rate, by using sensors providing higher spatial resolution, or by adding further functionality, such as monitoring the human body’s parameters (temperature, pressure, pH), tissue sampling (e.g. optical biopsy) and therapeutic tasks (drug delivery).
Finally, another key benefit of the proposed EDVC is that it supports *quality and temporal scalability*, thereby providing the possibility to online modify the acquisition frame rate and the quality of the decoded video. Like so, the medical practitioner can focus on areas of particular significance during the capsule’s passage through the digestive track.

To sum up, the work in the context of this dissertation has developed *novel DVC systems* and solutions that effectively target a wide spectrum of wireless lightweight multimedia applications. In this regard, our work has attracted the interest of both academia and industry. On the one hand, concerning the academic record, the work described in this dissertation has led to *eleven* ISI indexed journal articles, *two* book chapters and *seventeen* papers published in conference proceedings. On the other hand, the potential of the designed DVC schemes in wireless visual sensor networks and wireless capsule endoscopy has been recognized by key industrial players in the corresponding fields. Therefore, aiming at valorization, parts of the work included in this thesis have been protected by IBBT under a filed *international patent application*.

### 7.2 Prospective Work

Aside from fulfilling its goal to enhance the compression performance of DVC and to explore new application scenarios, the work presented in this dissertation stimulates further interesting research directions that deserve to be investigated in the future.

Considering the vast evolution that traditional predictive systems have undergone over the last decades, one can deduce that there is still room to enhance the compression efficiency and the obtained visual quality in DVC. The work in this dissertation has highlighted the capacity of hash-based DVC systems to cope with various motion motives. In such hash-based architectures, suitable *rate allocation mechanisms*, which employ refined rate-distortion modeling to determine on-the-fly the amount of the transmitted hash, are expected to bring additional RD improvements.

Another research track in this field would concentrate on the design of efficient *successively refined hash-based DVC architectures*. A first effort towards this direction has been presented in our recent work in [168], where the system presented in Chapter 6 was equipped with the DC-OBMEC side information refinement method described in Section 4.5. Extending over this approach, we envision a new successively refined hash-based Wyner-Ziv video coding scheme that assigns the DCT coefficients to several distinct refinement levels depending on the distribution
of their frequencies. The number of refinement levels would constitute a trade-off between the achieved RD performance improvements and the decoding complexity and structural latency. The foreseen system would improve the side information per refinement level by reinitiating OBMEC. Every time a refinement level would be decoded the DCT coefficients in all previous levels would be reconstructed again, thus improving the overall decoding quality.

Furthermore, the use of different coding modes has shown great gains in traditional prediction video coding as well as in MCI-based DVC [90-92] schemes. However, the extension of these techniques to (successively refined) hash-based DVC systems remains generally unexplored. Therefore, in order to add to the coding efficiency of the developed hash-based DVC systems, identification and employment of proper techniques enabling the switch between different coding modes, i.e., intra-coding, Wyner-Ziv coding or skip, is of critical substance.

Low-delay temporal prediction structures are additional functional components that could be included in the designed hash-based architectures. Complying with the state-of-the-art, e.g., [24, 119], the codec architectures presented in this work have been evaluated using a hierarchical bidirectional motion prediction structure. Although side information creation techniques based on bidirectional prediction provide the finest performance, they introduce an inherent coding delay of one GOP. This also increases the memory footprint of the encoder, as the intermediate WZ frames in a GOP need to be buffered pending the encoding of the key frames (see Section 6.5.4). Because of the hash-based nature of the designed systems, the temporal prediction structure can be modified in a flexible manner by altering the appropriate reference frame(s). Furthermore, inspired by recent developments in traditional predictive systems [169], the foreseen system can preserve low encoding delay but still support temporal scalability.

In the developed systems, a typical decoder-driven rate control using a feedback channel was implemented. In prior collaborative work, the feedback channel was constrained [94, 95]. Towards completely removing the feedback channel in our hash-based architectures, a major focus of forthcoming work would be to model the side information at the encoder in an accurate yet low complex manner. This encoder-side approximation of the side information would serve as a basis to calculate an encoder-side SID estimate of the required rate for successful channel decoding. In addition, sophisticated reconstruction techniques would be applied to lower the distortion should the assigned rate be insufficient.

Seeing the improved modeling accuracy and the promising gains compared to SII models (see Chapter 5), prospective research work should also build on the implications of the proposed SID modeling on distributed source coding. For
instance, novel Slepian-Wolf codes, tailored to the statistical properties of asymmetric channel models are anticipated to augment the contribution of the inherent coding gain of the SID paradigm. Moreover, in Section 5.4, it has been explained that for a practical realization of the SID channel the side information signal needs to be quantized into a number of quantization bins. In this context, the design of optimal quantizers for the side information that yield the best RD performance is foreseen to be an interesting research subject.

Lastly, a major research topic revolves around the extension of the presented DVC schemes to the distributed joint source-channel coding (DJSCC) scenario. In this context, future research should focus on performing error resilience tests against special communication channels typically encountered in the wireless lightweight scenarios targeted by the designed systems.
Appendix A

CHANNEL CODES FOR DISTRIBUTED SOURCE COMPRESSION

A.1. INTRODUCTION

Since the groundbreaking publication of Shannon’s information theory [28], an enormous amount of research has been conducted into the construction of practical codes with good error-correcting capabilities, and the development of effective decoding algorithms. The research in channel coding has resulted in the design of several codes, such as Golay, Hamming, BCH, Reed-Solomon (RS), convolutional codes, etc. In 1993, the revolutionary concept of Turbo codes [170], landmarked the construction of advance designs, performing very close to the Shannon bound. Two years later, MacKay and Neal [171] re-invented LDPC codes, originally engineered by Gallager in 1963 [172], and demonstrated their excellent performance. In 2002, a new family of channel codes, called fountain codes has been pioneered by Luby [173] and expanded by Shokrohalli [174] to Raptor codes in 2004. Fountain and Raptor codes are Luby Transform (LT) codes of fixed dimension that present limitless block-length.

From Golay to Raptor codes, the primary employment of channel coding involves the addition of a controlled amount of redundancy to the transmitted source data so as to provide the means for detecting and correcting transmission errors. However, the revolutionary concept of random binning by Slepian and Wolf [9], has inaugurated a new epoch for the design and deployment of channel coding principles. As detailed in Chapter 2, it is since the landmark paper of Wyner [45], that channel coding is recognized to provide a pragmatic platform for the realization of algebraic binning [44]. Nowadays, efficient channel codes with soft decoders, e.g., Turbo and LDPC codes, are key ingredients of diverse applications of distributed source coding—see for example [152] for a comprehensive overview.

In essence, two structurally different channel coding types are distinguished in
Appendix A

coding theory, namely, block and convolutional codes. In convolutional coding, the encoder accepts information symbols as a continuous stream and generates a continuous stream of it at a higher rate. Encoding is realized by sending the input streams over linear filters. The name convolutional code/encoder stems from the fact that this filtering operation can be expressed as a convolution. In DSC, convolutional codes were first used in [46, 69]. Additionally, convolutional codes constitute the individual codecs of Turbo codes, which have been widely deployed in distributed source coding and in distributed video coding, in particular. However, Turbo codes are not used in this dissertation and therefore are not described further. More information on convolutional and Turbo codes can be found in [175]. Furthermore, we refer to [176] for a profound demonstration of the design and implementation of a Turbo-based Slepian-Wolf codec for efficient DVC.

Block codes are operating on codewords of a specific length. A particular class of high-capacity linear block codes is defined by LDPC codes. As LDPC codes have shown to outperform Turbo codes for Slepian-Wolf coding [48], in this research work, we employ the former to conceive highly efficient DVC systems. Furthermore, LDPC codes are near-capacity achieving on several channel types, both symmetric and asymmetric. The latter is of vital significance in this research work, in which we have shown the asymmetric behavior of the correlation channel in DVC (see Chapter 5).

The rest of this Appendix is structured as follows. Section A.2 sketches a short overview of linear block codes. Then, Section A.2.2 describes the construction and the iterative decoding process of LDPC codes. Finally, Section A.3 focuses on LDPCA codes, namely, the rate-adaptive LDPC coding scheme employed in this dissertation.

A.2. BLOCK CODES

A block code $C$ of length $n$ and cardinality $\Omega$ over a field $\mathbb{F}$ with $m$ symbols is defined as a subset of $\mathbb{F}^n$ with $\Omega$ $m$-ary elements of length $n$, called codewords: $C = \{(c_{11}, c_{12}, \ldots, c_{1n}), (c_{21}, c_{22}, \ldots, c_{2n}), \ldots, (c_{\Omega 1}, c_{\Omega 2}, \ldots, c_{\Omega n})\}$. The length $n$ of the codewords is called the block length. Codewords of block length $n$ are typically generated by encoding messages of $k$ information symbols using an invertible encoding function. Hence, the number of codewords $\Omega$ included by the block code $C$ equals $m^k$. When encoding with a block code, the encoder successively processes information in $k$-symbol blocks and for each $k$ symbols, generates a block of $n$ symbols where $n > k$. With this respect, the dimensionless ratio [175] $k/n$ is defined as the rate of the code, or simply code-rate. When the $k$ source symbols
appear per se in a codeword, the code is said to be *systematic*. That is, a codeword of
a systematic code separately contains the *k* source and \((n-k)\) parity or redundant
symbols.

Furthermore, most established codes are *linear*. A linear code is one in which
linear combinations of codewords are again codewords. This also means that the all-
zero codeword is always a codeword. The received version of the codeword is called
the *received word*. The block decoder operates on the received word to determine
whether the information and parity symbols satisfy the encoding rules. It is noted
that symbols are bits when a binary \((m=2)\) encoder is used and multiple bits when
a non-binary \((m>2)\) encoder is used.

In the following section, we concentrate on LDPC codes, which are the basis for
the Slepian-Wolf codes employed in this dissertation.

### A.2.1. General Principles of Linear Block Codes

In general, *linear block codes* are described in terms of matrices, namely, the
generator matrix \(G\) of dimension \(k \times n\) and its dual parity-check matrix \(H\) of
dimension \((n-k) \times n\). The generator matrix \(G\) represents a set of basis vectors
within a \(k\)-dimensional subspace of field \(\mathbb{F}^n\) such that any codeword is a linear
combination of the rows of \(G\). Therefore, the generator matrix \(G\) defines the
mapping of a source word \(w\) of dimension \(k \times 1\) to a codeword \(c\) of dimension
\(n \times 1\) as the matrix multiplication

\[
c = G^T w .
\]

It is common to consider \(G\) in its *systematic* form, i.e., \(G = [I_k | P]\) so that the first
transmitted \(k\) symbols are the source symbols.

Each row of the parity-check matrix \(H\) defines a linear constraint satisfied by all
codewords, i.e.,

\[
Hc = 0.
\]

Also, \(HG^T = 0\) such that \(H\) can be used to detect errors in the received word \(r\),
which is corrupted by a noise vector \(n: Hr = H(c + n) = HG^T w + Hn = Hn \triangleq s\),
where \(s\) is called the *syndrome*. In fact, the syndrome tells which parity-check
equations are not satisfied. If the syndrome is null, we assume that there are no
errors. Otherwise, the decoding problem is to find the most likely noise vector that
explains the observed syndrome given the assumed properties of the channel.
Observe that if \(G\) is written in systematic form as above, then \(H\) has the form

\[
\left[ P | I_{(n-k)} \right] .
\]

### A.2.2. Low-Density Parity-Check Codes

An LDPC code is a linear block code as described in Section A.2.1. The main
The characteristics of an LDPC code are that (i) the parity-check matrix $H$ is sparse, i.e., a matrix with a low number of ones and a large number of zeros, and (ii) the decoding is performed iteratively using a so-called message-passing algorithm.

The iterative decoding process is easily explained using a Tanner graph representation of the parity-check matrix. A Tanner graph consists of $n$ variable nodes and $n-k$ check nodes. Observe that the number of variable nodes corresponds to the number of bits in a codeword. Connections between the two sorts of nodes are realized according to the position of the ones in the matrix and are called edges. A regular LDPC code is characterized by a low and fixed number of ones in the columns (also called variable degree $d_v$) and a low and fixed number of ones in the rows (also called check degree $d_c$). LDPC codes with a variable amount of ones in the rows and columns are called irregular LDPC codes. An example of a regular parity-check matrix with $(d_v,d_c) = (2,4)$ and its associated Tanner graph is given in Figure A.1.

The idea of iterative decoding is to compute, for each bit $c_i$ of the transmitted codeword, the a posteriori probability (APP) $\Pr(c_i = 1|\mathbf{r})$ that bit $c_i$ is equal to 1 given the received word $\mathbf{r}$. The numerically stable log-domain APP ratio is typically used and referred to as log-likelihood-ratio (LLR) $L(c_i)$,

$$L(c_i) \triangleq \log \frac{\Pr(c_i = 0|\mathbf{r})}{\Pr(c_i = 1|\mathbf{r})}. \quad (A.3)$$

During one half of each iteration, each variable node $v_i$, processes its incoming messages and passes a resulting message $m_{ij}^\uparrow$ to each check-node $u_j$ to which it is connected. This is illustrated in Figure A.2(a). This message $m_{ij}^\uparrow$ contains the probability that the received bit $r_i$ at the level of variable $v_i$ is correct given the received word $\mathbf{r}$. Similarly, during the second half of each iteration, each check node $u_j$ processes its incoming messages and passes a resulting message $m_{ij}^\downarrow$ to each variable node $v_i$ to which it is connected. The message $m_{ij}^\downarrow$ contains the probability that the check-equation (A.2) is satisfied. It is observed that the
messages \( m_{r_{ij}} \), \( m_{s_{ji}} \) only contain extrinsic information. This means that the message \( m_{r_{ij}} \) (\( m_{s_{ji}} \)) that is passed from a variable node \( v_i \) (check node \( u_j \)) to a check node \( u_j \) (variable node \( v_i \)) contains information from all incoming messages of the variable node \( v_i \) (check node \( u_j \)) except from the message of the check node \( u_j \) (variable node \( v_i \)) itself.

\[
\begin{align*}
    m_{r_{ij}} \quad &\text{(b) Channel information and bit value} \\
    m_{s_{ji}} \quad &\text{(a) Channel information and bit value}
\end{align*}
\]

*Figure A.2: Iteration of message-passing algorithm. (a) Extrinsic passing of messages \( m_{r_{ij}} \) of variable nodes \( v_i \) to check nodes \( u_j \) and (b) of messages \( m_{s_{ji}} \) of check nodes \( u_j \) to variable nodes \( v_i \).*

The iterative decoding process, when transmitting a codeword over a BSC, is demonstrated hereafter. The sum-product algorithm in the log-domain is used as message-passing algorithm [175]. This decoding process is employed in our LDPC code implementation.

We introduce the following notation:

\[
L(t_{ji}) = \log \left( \frac{t_{ji}(0)}{t_{ji}(1)} \right), \quad L(q_{ji}) = \log \left( \frac{q_{ji}(0)}{q_{ji}(1)} \right), \quad L(Q_i) = \log \left( \frac{Q_i(0)}{Q_i(1)} \right),
\]

where \( q_{ji}(b) = \Pr(c_i = b \mid r) \), \( t_{ji}(b) = \Pr(H_j r_j = 0 \mid r) \) and \( Q_i(b) \) is the pseudo-LLR of variable node \( v_i \) that is computed at the end of each iteration with \( b = \{0,1\} \). Using these notations, the iterative sum-product algorithm is performed as:

1. \( \forall i = \{1, \ldots, n\} \), initialize \( L(q_{ji}) \) for a BSC\(^{36}\) with crossover probability \( p_c \),

\[
L(q_{ji}) = (-1)^i \log \left( \frac{1 - p_c}{p_c} \right).
\]

2. Update \( \{L(t_{ji})\} \) as follows:

\[
L(t_{ji}) = \prod_{i \in V_j \setminus i} L(t_{ji}) \cdot \Phi \left( \sum_{i' \in V_j \setminus i} L(q_{i'j}) \right),
\]

where \( \Phi(x) = -\log \left[ \tanh \left( x / 2 \right) \right] = \log \left( \left( e^x + 1 \right) / \left( e^x - 1 \right) \right) \), \( \alpha_{\bar{y}} = \text{sign} \left[ L(q_{ji}) \right] \).

\(^{36}\) The BSC is used as an example to explain the operation of the message passing algorithm in LDPC decoding. The latter algorithm has proven its generic applicability is a broad set of symmetric and/or asymmetric channels [62, 150, 175].
Appendix A

$\beta_{ij} = |L(q_{ij})|$ and $V_j$, $U_i$ correspond to the set of all variable nodes connected to $u_j$, and all check nodes connected to $v_i$.

3. Update $\{L(q_{ij})\}$ as

$$L(q_{ij}) = L(c_i) + \sum_{j' \in U_i \setminus j} L(t_{ij'}).$$

4. Update $\{L(Q_i)\}$ as

$$L(Q_i) = L(c_i) + \sum_{j \in U_i} L(t_{ji}).$$

5. $\forall i = \{1, \ldots, N\}$ make the following hypothesis:

$$\hat{c}_i = \begin{cases} 1, & \text{if } L(Q_i) < 0 \\ 0, & \text{otherwise} \end{cases}$$

6. If $\hat{H} = 0$ or if the iterations equal the maximum number of iterations, stop. Otherwise, iterate from step 2.

In order to design good LDPC codes with prescribed properties, the methods described in [62, 63], and [64] have been employed in this dissertation.

A.3. RATE-ADAPTIVE LDPC CODES FOR SLEPIAN-WOLF COMPRESSION

Concerning the important issue of rate control for LDPC-based Slepian-Wolf compression, in early DSC designs [17, 47], the correlation channel statistics are assumed to be stationary and perfectly known at both the encoder and the decoder. Under this assumption, the encoder and the decoder agreed on an efficient code-rate driven by the SW rate, i.e., the conditional entropy of the source given the side information. Nevertheless, in real-world applications, DSC frameworks face intricate barriers impeding perfect, a priori information on the correlation statistics. Specifically, first, the correlation channel exhibits highly non-stationary properties. Secondly, the fact that in DSC schemes the source is available at the encoder whereas the side information is only formed at the decoder obstructs direct measurement of the correlation noise. Moreover, sophisticated correlation channel models, e.g., the proposed SID (asymmetric) model described in Chapter 5, encapsulates the side information dependency of the noise, thereby complicating the problem of online correlation channel estimation further. Due to the aforementioned realistic limitations, a rate-adaptive Wyner-Ziv scheme employing a feedback channel from the decoder to the encoder seems an appealing rate control solution for many applications.
In the considered rate-adaptive LDPC coding scheme, the encoder initially transmits a weak channel code and the decoder attempts decoding based on the estimated channel statistics. In case of successful decoding, the decoder informs the encoder to continue with the next block of source data. Conversely, if decoding fails, the encoder supplements the transmitted channel code by creating a longer syndrome based on a lower-rate code. This progression is carried on until the syndrome is eligible for successful decoding. Obviously, this method is reasonable under feedback channel availability and low-delay requirements.
Early rate-adaptive Slepian-Wolf codes were realized using punctured Turbo codes [12, 178]. However, extending this scheme to LDPC codes, while maintaining efficient performance is not straightforward. In fact, the performance of punctured LDPC codes, e.g., [179], significantly deteriorates when the codes are subject to medium and high puncturing. This is because punctured LDPC graphs contain unconnected or individually connected nodes, impeding effective transfer of information via the iterative message-passing algorithm (see Section A.2.2 above).

To set an illustrative example, consider the LDPC factor graph of Figure A.3. The source bits $x = (x_0, x_1, \ldots, x_7)$ and the syndrome bits $s = (s_0, s_1, \ldots, s_7)$ constitute the variable and the check nodes of the LPDC factor graph, respectively. The schema of Figure A.3(a) corresponds to the non-compression case. Notice that, based on the Gaussian approximation principles, decoding is always successful in this case, independent of the correlation channel statistics. Now, consider puncturing the above-described LDPC code in order to obtain a compression ratio of two, alias an LDPC code rate of $1/2$. Let for instance puncturing be performed by not transmitting the even indexed syndrome bits. As shown in Figure A.3(b), the decoding graph resulting by conventional puncturing of the syndrome bits would be severely degraded, incurring inefficient iterative decoding.

An important contribution to resolve this issue, that is, to achieve various LDPC puncturing rates without affecting the codes’ performance, has been made in [48]. Varodayan et al. [48] introduced a novel class of LDPC accumulate (LDPCA) codes, which consists of an LDPC syndrome-based code concatenated with an accumulator. As in conventional syndrome-based LDPC schemes, the LDPCA encoder produces syndrome bits $s$ by performing mod-2 addition of the source bits $x$ according to the LDPC factor graph. However, contrary to traditional approaches, the derived syndrome bits are subsequently mod-2 accumulated, thus producing the
accumulated syndrome tuple $\alpha$. For the sake of demonstration, this operation is sketched in Figure A.4. The encoder stores the accumulated syndrome bits in a buffer and transmits them incrementally to the decoder.

Notice that the LDPCA decoder can achieve various rates by on-the-fly altering the LDPC decoding factor graph upon reception of an additional increment of the accumulated syndrome [48]. In this way, the graph resulting from the different punctured syndrome patterns consistently maintains the degree of all variable nodes, as defined by the parity-check matrix of the non-punctured LDPC code [see Figure A.3(c)]. As a consequence, this method enables the exchange of a higher amount of soft information between the variable and the punctured check nodes, thereby yielding a more effective iterative decoding than conventional punctured LDPC codes.

As a concluding remark, notice that the encoding and decoding complexity of the LDPCA codes [48] is linear in the number of factor graph edges. As the number of edges is linear in the codeword length (for a fixed degree distribution), the complexity of LDPCA encoding and decoding is $O(n)$, where $n$ is the codeword length.
Appendix B
MOTION-COMPENSATED INTERPOLATION

B.1. INTRODUCTION

The objective of employing motion-compensated interpolation, or briefly MCI, in the decoder of a Wyner-Ziv video coding system is to create a motion-compensated prediction of an encoded WZ frame based on two already decoded reference frames, that is, one previous and one following frame. Seeing the fact that this prediction acts as side information to the original WZ frame, the higher its resemblance to the latter, the higher the compression performance of the developed Wyner-Ziv video coding system. This appendix summarizes the state-of-the-art MCI [80-82], method employed in the DISCOVER [24] reference DVC system, incorporating modifications and extensions proposed by the research conducted in the context of this dissertation [102].

The schematic representation of the developed MCI technique is depicted in Figure B.1. As explained in Section B.2, the frame interpolation module first performs block-based motion estimation between two already decoded reference frames (i.e., one previous and one following frame), and the resulting motion vectors are intercepted in the current frame. Next, as portrayed in Section B.3, for each block in the current frame, the closest intersecting vector is split into two parts and treated as a bidirectional motion vector. The vectors are then further refined with half-pel accuracy. The ensuing motion field is smoothed using a median filter described in Section B.4. In DISCOVER’s MCI method, the side information frame is generated block-by-block using simple bidirectional motion compensation. However, as mentioned in Section B.5, the MCI technique employed in this dissertation advances over the latter by deploying bidirectional overlapped block motion compensation.
B.2. **Forward Motion Estimation**

In the first stage, for each WZ frame $X$ forward block-based motion estimation with integer-pixel accuracy is performed between the previous and the next reference frame, denoted by $X_p$ and $X_n$, respectively. As a hierarchical motion predictive structure is considered (see Figure B.2) the reference frames are the already decoded previous and next key and/or WZ frames. Note that the reference frames are initially low-pass filtered (with a $3 \times 3$ mean filter) in order to improve the reliability of the motions vectors. In this setting, for each block in the next reference frame, forward block motion estimation involves finding the best matching block in the past reference frame, within a certain search range. This operation is sketched in Figure B.3. Similar to [81], the error metric ($EM$) employed for block matching is a modified version of the sum of absolute differences (SAD) metric, which favors smaller motion vectors [81], that is,

$$
EM(x, y, v = (v_x, v_y)) = (1 + k \|v\|) \cdot \sum_{j=0}^{N-1} \sum_{i=0}^{N-1} |X_n(Nx + i, Ny + j) - X_p(Nx + i + v_x, Ny + j + v_y)|,
$$

where, $x$ and $y$ respectively denote the top-left coordinates of the block for which motion estimation is performed, $N$ denotes the block size, $i$ and $j$ are respectively the column and row coordinates of the pixels in the block, $v = (v_x, v_y)$ represents the candidate motion vector and $k$ is a constant set to $k = 0.05$ [81]. In compliance with prior art [80, 81], a block size of $N = 16$ and a search range of $\rho = 32$ pixels is employed in the forward motion estimation algorithm.
B.3. **Bidirectional Motion Estimation**

The resulting unidirectional motion field, denoted by $MF_F$, is thereafter used to derive the bidirectional motion field, $MF_B$, between the interpolated frame and the reference frames, as depicted in Figure B.4. In particular, similar to [80], the points where the motion vectors of $MF_F$ intercept the interpolated frame are determined first. For each block in the interpolated frame, the motion vector for which the intercept point is closest to the top-left corner of the block is selected. This motion vector $v$ is subsequently scaled with the ratio between the distance of the interpolated frame to the previous reference frame and the distance between both reference frames, yielding the new forward motion vector for the block. Observe that this ratio is always $\frac{1}{2}$ since hierarchical prediction is used. Similarly, the backward motion vector of the interpolated block is determined by scaling the inverted motion vector $-v$ by $\frac{1}{2}$. This operation generates the initial bidirectional motion field between the interpolated frame and both reference frames.

Subsequently, the obtained bidirectional motion field is further improved. Similar to [80], the algorithm searches for symmetric motion vector pairs, corresponding to linear motion trajectories, around the initially determined motion vector pair. This operation is schematically described in Figure B.5. The procedure employs the SAD between the referred blocks in the previous and next reference frames as an error metric and supports half-pel motion estimation accuracy [82]. The required interpolation for half-pel motion estimation is performed using the 6-tap interpolation filter of H.264/AVC [8]. We refer to [82] for further details.
Figure B.4: Extrapolation of the bidirectional motion field from the unidirectional one. For each block in the current (i.e., interpolated) frame, the closest intercepting vector is found, shifted to the top-left corner of the block, and then divided into a bidirectional motion vector.

Figure B.5: Improvement of the initially obtained bidirectional motion field by using symmetric motion vector pairs.

B.4. **Spatial Motion Vector Smoothing**

The ensuing bidirectional motion field is spatially smoothed by applying a weighted vector media filter [80, 81]. This is done in order to improve the spatial coherence of the obtained bidirectional motion field, thereby aiming at removing outliers, i.e., motion vectors that are far from the true motion field. Specifically, for each block $B_j$ in the interpolated frame, the weighted median vector filter proposed in [80] looks for candidate motion vectors at neighboring blocks, which can represent better the motion in the block. According to this method, the spatially
smoothed motion vector for an interpolated block $B_l$ is given by

$$
v = \arg \min_{v_l} \left\{ \sum_{m=1}^{M} w_m \|v_l - v_m\| \right\}, \tag{B.2}$$

where $v_{m=1}, v_{m=2}, \ldots, v_{m=M}$ are the motion vectors derived by the refined bidirectional motion estimation of the block under consideration and its $M - 1$ neighbors, and $w_m$, $m = 1, \ldots, M$, are weights determining the strength of the median filter. These weighting factors are obtained as

$$w_m = \frac{1}{SAD(B_l, v_m)}, \tag{B.3}$$

where the SAD metric evaluates the matching error between the reference blocks for each block $B_l$ compensated with the bidirectional vector $v_m$.

It is important to mention that extensive experimentation has shown that spatial motion vector smoothing is mainly beneficial for CIF rather than for QCIF sequences\(^{37}\). Based on this evaluation, when a QCIF sequence is coded no subsequent motion field smoothing is performed in our MCI method, as explained in [102].

### B.5. MOTION COMPENSATION

Once the final bidirectional motion field is derived, the side information frame is obtained by performing motion compensation. In the MCI [80-82] method developed for the DISCOVER [24] codec a simple bidirectional motion compensation approach is followed. In particular, for the pixels belonging to block $B$, bidirectional motion compensation is defined by

$$Y_B(i, j) = \frac{1}{2} \left\{ X_B^p(i-v_x, j-v_y) + X_B^n(i+v_x, j+v_y) \right\}, \tag{B.4}$$

where $Y_B(i, j)$ corresponds to a pixel location in the block $B$ in the motion-compensated frame, and $X_B^p(i-v_x, j-v_y), X_B^n(i+v_x, j+v_y)$ represent the corresponding pixels in the best matching blocks, identified by the derived symmetric bidirectional motion vector $v = (v_x, v_y)$, in the previous and next reference frame, respectively.

In contrast to the latter method, in the MCI [102] algorithm engineered in the context of this dissertation, bidirectional overlapped block motion compensation (OBMC) is performed. Rather than predicting by using a single symmetric motion vector per block, OBMC predicts using motion vectors from the blocks in a

---

\(^{37}\) This experimental observation has also been confirmed by Dr. J. Ascenco (i.e., the first author of [80]) during private communication, 2011.
neighborhood around the interpolated block. Hence, by introducing OBMC, the presented MCI technique produces an interpolated frame which exhibits reduced prediction error energy at the pixel level, and in turn increases the performance of Wyner-Ziv coding. Moreover, blocking artifacts, typically appearing at block boundaries, are vastly diminished, thereby increasing the visual quality of the decoded frame.

In the proposed MCI technique, bidirectional OBMC is performed as follows. Initially, based on the previously obtained bidirectional motion field, the proposed approach derives a forward and a backward overlapped block motion-compensated frame, denoted by \( \tilde{Y}^f \) and \( \tilde{Y}^b \), respectively. Each of these frames is obtained by using the corresponding motion field (i.e., forward or backward) and by applying OBMC. Note that OBMC may employ an adaptive non-linear predictor. However, in the designed methodology a fixed linear predictor, in particular a raised cosine window is used. This means that OBMC is actually implemented as a windowed motion compensation method (see Figure B.6). In this case, translated blocks are first scaled by the window, and the overlapping fractions are summed. In this way, a pixel in position \((i, j)\) in the block \(B\) in the forward overlapped block motion-compensated frame \( \tilde{Y}^f \) is predicted as

\[
\tilde{Y}^f_B(i, j) = \sum_{m=1}^{M} w(i_m, j_m) X^f_B(m, i-v_{m,x}, j-v_{m,y}),
\]

where \( m = 1, \ldots, M \) indexes the block of which the window contains the compensated pixel, \( w(i_m, j_m) \) is the corresponding scaling factor of the \( m \)th overlapping window, and \( X^f_B(m, i-v_{m,x}, j-v_{m,y}) \) is the candidate predictor pixel in the previous reference frame belonging to the \( m \)th overlapping window. Similarly, the backward overlapped block motion-compensated frame \( \tilde{Y}^b \) can be found as

\[
\tilde{Y}^b_B(i, j) = \sum_{m=1}^{M} w(i_m, j_m) X^b_B(m, i+v_{m,x}, j+v_{m,y}).
\]

Finally, the derived forward and backward overlapped block motion-compensated frames are averaged yielding the final side information pixel values, that is,

\[
Y_B(i, j) = \frac{1}{2} \{ \tilde{Y}^f_B(i, j) + \tilde{Y}^b_B(i, j) \}.
\]
per prediction direction) and then bidirectional compensation is performed. Furthermore, in [125], the weight of each block in OBMC is given by the value of the matching error between its reference (forward and backward) blocks. Conversely, as mentioned above, in the proposed approach, a fixed linear predictor, namely, a raised cosine window is employed to determine the OBMC weights.

Figure B.6: Example of overlapped block motion compensation with a fixed linear predictor. The blocks in the block lattice are represented by continuous lines, the overlapping windows are indicated by dashed lines, and the motion vectors by dotted lines. For convenience, the blocks and their corresponding overlapping windows are sketched on the right. The pixel $A$ belonging to the black block is predicted by four motion vectors, namely, the motion vector of the black block and the motion vectors of its three neighboring blocks.
Appendix C
THEORETICAL FORMULATIONS

C.1. PROOFS AND DERIVATIONS

This Appendix summarizes the proofs of Lemma 5.1, Lemma 5.2 and Theorem 5.1 as stated in Chapter 5. Furthermore, this Appendix includes the theoretical formulations leading to the derivation and the uniqueness of the proof of uniqueness of Equation (5.18) in Chapter 5.

C.1.1. Proof of Lemma 5.1

Let \( f_X(x) = \lambda e^{-|x| - \mu}/2 \) be a generic Laplacian distribution and assume that the random variable \( X \) is quantized using a uniform scalar quantizer centered on the mean \( \mu \) of the distribution. The distortion resulting from quantizing \( X \) is given by

\[
D = \sum_{k=-\infty}^{\infty} \int_{\mu + (k+1/2)\Delta}^{\mu + (k-1/2)\Delta} (x - (\mu + k\Delta))^2 f_X(x) \, dx. \tag{C.1}
\]

Replacing \( x - \mu \) with \( x \) and expressing the above as a sum of three terms, leads to

\[
D = \frac{\lambda}{2} \sum_{k=-\infty}^{1} \int_{(k-1/2)\Delta}^{(k+1/2)\Delta} (x - k\Delta)^2 e^{\lambda x} \, dx + \frac{\lambda}{2} \int_{-\Delta/2}^{\Delta/2} x^2 e^{-\lambda|x|} \, dx + \frac{\lambda}{2} \sum_{k=1}^{\infty} \int_{(k-1/2)\Delta}^{(k+1/2)\Delta} (x - k\Delta)^2 e^{\lambda x} \, dx. \tag{C.2}
\]

By changing \( k \) in \(-k\) and subsequently \( x \) in \(-x\) in the first term above shows that the first and the last sums are equal. Hence,

\[
D = \frac{\lambda}{2} \int_{0}^{\Lambda/2} x^2 e^{-\lambda x} \, dx + \lambda \sum_{k=1}^{\infty} \int_{(k-1/2)\Delta}^{(k+1/2)\Delta} (x - k\Delta)^2 e^{\lambda x} \, dx. \tag{C.3}
\]

In the second term, replacing again \( x - k\Delta \) with \( x \) leads to

\[
D = \lambda \left( \int_{0}^{\Lambda/2} x^2 e^{-\lambda x} \, dx \right) + \lambda \left( \int_{-\Delta/2}^{\Delta/2} x^2 e^{\lambda x} \, dx \right) \sum_{k=1}^{\infty} (e^{-\lambda\Delta})^k. \tag{C.4}
\]
The last summation term is a geometric progression converging to 
\[ e^{-\lambda \Delta} / (1 - e^{-\lambda \Delta}) \]. The other two finite integrals can be easily computed by parts, leading to

\[ D = \frac{2}{\lambda^2} - \frac{2\Delta e^{-\lambda \Delta/2}}{\lambda (1 - e^{-\lambda \Delta})}, \]  
(C.5)

and thereby ending the proof. ■

C.1.2. Proof of Lemma 5.2

For a given \( y \), the corresponding \( f_{X|Y}(x|y) \) is Laplacian with mean \( y \). The distortions of the SID and SII models are of the form given by (C.5), that is,

\[ D_{SID}(y) = \frac{2}{\lambda(y)^2} - \frac{2\Delta e^{-\lambda(y) \Delta/2}}{\lambda(y) (1 - e^{-\lambda(y) \Delta})}, \]  
(C.6)

and

\[ D_{SII} = \frac{2}{\lambda_0^2} - \frac{2\Delta e^{-\lambda_0 \Delta/2}}{\lambda_0 (1 - e^{-\lambda_0 \Delta})}, \]  
(C.7)

where \( \lambda(y) = \sqrt{2/\sigma_{SID}}(y) \), while \( \lambda_0 = \sqrt{2/\sigma_{SII}} \) is a constant not depending on \( y \). The condition \( \int_{-\infty}^{+\infty} D_{SID}(y) f_Y(y) dy = D_{SII} \) for any \( \Delta \) is equivalent to

\[ \int_{-\infty}^{+\infty} \frac{2}{\lambda(y)^2} \left[ 1 - \frac{\lambda(y) \Delta e^{-\lambda(y) \Delta/2}}{1 - e^{-\lambda(y) \Delta}} \right] f_Y(y) dy = \frac{2}{\lambda_0^2} \left[ 1 - \frac{\lambda_0 \Delta e^{-\lambda_0 \Delta/2}}{1 - e^{-\lambda_0 \Delta}} \right], \forall \Delta. \]  
(C.8)

Let

\[ F(\lambda(y), \Delta) = \frac{\lambda(y) \Delta e^{\lambda(y) \Delta/2}}{e^{\lambda(y) \Delta} - 1}, \]  
(C.9)

then Eq. (C.8) can be equivalently written as

\[ \int_{-\infty}^{+\infty} \frac{2}{\lambda(y)^2} \left[ 1 - F(\lambda(y), \Delta) \right] f_Y(y) dy = \frac{2}{\lambda_0^2} \left[ 1 - F(\lambda_0, \Delta) \right], \forall \Delta. \]  
(C.10)

One can easily verify that \( \lim_{\Delta \to \infty} F(\lambda(y), \Delta) = 0 \). Therefore, a necessary condition to have the distortions of the SID and SII models equal for any \( \Delta \) is

\[ \int_{-\infty}^{+\infty} \frac{2}{\lambda(y)^2} f_Y(y) dy = \frac{2}{\lambda_0^2}. \]  
(C.11)

Replacing \( \lambda(y) = \sqrt{2/\sigma_{SID}}(y) \) and \( \lambda_0 = \sqrt{2/\sigma_{SII}} \) in (C.11) leads to the necessary condition of (5.11), that is,
\[ \int_{-\infty}^{\infty} \sigma_{\text{SID}}^2(y) f_Y(y) dy = \sigma_{\text{SII}}^2, \]  
\quad \text{(C.12)}

which ends the proof.

\section*{C.1.3. Proof of Theorem 5.1}

Supposing ideal Slepian-Wolf coding, the rate is given by the conditional entropy of the quantization index knowing the value of the side information \[61\], namely, \( H(Q(X)|y) \). Under high-rate assumptions, the output entropy of a uniform scalar quantizer is approximated as \[61, 148\],

\[ H(Q(X)|y) = h(X|y) - \log_2 \Delta. \]  
\quad \text{(C.13)}

For a Laplacian model, the differential entropy is given by

\[ h(X|y) = \log_2 \sqrt{2\pi e} \sigma(y). \]  
\quad \text{(C.14)}

Now, let us compare the SW rate given by an SII and an SID model. From Eqs. (C.13), and (C.14), we derive

\[ 2 \cdot \mathbb{E} \left[ H_{\text{SID}}(Q(X)|y) - H_{\text{SII}}(Q(X)|y) \right] = 2 \cdot \mathbb{E} \left[ \log_2 \sqrt{2\pi e} \sigma_{\text{SID}}(y) - \log_2 \sqrt{2\pi e} \sigma_{\text{SII}} \right] \]
\[ = 2 \cdot \mathbb{E} \left[ \log_2 \sigma_{\text{SID}}(y) - \log_2 \sigma_{\text{SII}} \right] \]
\[ = \mathbb{E} \left[ \log_2 \sigma_{\text{SID}}^2(y) \right] - \log_2 \sigma_{\text{SII}}^2. \]

Dividing by two and replacing the expectation operation, we have

\[ \mathbb{E} \left[ H_{\text{SID}}(Q(X)|y) - H_{\text{SII}}(Q(X)|y) \right] = \frac{1}{2} \int_{-\infty}^{\infty} \log_2 \sigma_{\text{SID}}^2(y) f_Y(y) dy - \log_2 \sigma_{\text{SII}}^2. \]

Applying Jensen’s inequality \[26\] in the integral of the last line and bearing in mind that \( g(u) = \log_2 u \) is a concave function, we have

\[ H_{\text{SID}}(Q(X)|y) - H_{\text{SII}}(Q(X)|y) \leq \frac{1}{2} \int_{-\infty}^{\infty} \log_2 \sigma_{\text{SID}}^2(y) f_Y(y) dy - \log_2 \sigma_{\text{SII}}^2, \]
\quad \text{(C.15)}

where \( H(Q(X)|y) = \mathbb{E} \left[ H(Q(X)|y) \right] \). Assuming the average SID distortion equal to the SII distortion for every RD point, Lemma 2 holds. Combining (C.15) with (5.11) leads to

\[ R_{\text{SID}}(D) \leq R_{\text{SII}}(D), \]  
\quad \text{(C.16)}

which ends the proof.

\section*{C.1.4. Derivation and Uniqueness of the Root of Equation (5.18)}

For every \( y_k \in \left[ 0, 2^{L_\beta} - 1 \right] \), we have \( \Theta = 2^m \) quantization indices \( q_{m,\vartheta} \), or else columns in the transition matrix of (5.17). That is, \( \forall y_k \in \left[ 0, 2^{L_\beta} - 1 \right] \), there are \( \Theta = 2^m \) equations of the form of (5.18), each of which is satisfied by the unknown
\( \hat{\lambda}_k = \lambda_\beta \left( y_k \right) \). Based on the position of \( y_k' \) in relation to a quantization bin with index \( q_{m,\beta} \), Eq. (5.18) is written as

\[
\begin{align*}
g_1 (\hat{\lambda}_k) & = 2 \cdot P_{Q_m} \left( q_{m,\beta} \left| y_k \right. \right) - e^{-2\lambda (q_L - y_k')} + e^{-2\lambda (q_H - y_k')} , \quad y_k' < q_L \\
g_2 (\hat{\lambda}_k) & = 2 \cdot P_{Q_m} \left( q_{m,\beta} \left| y_k \right. \right) - e^{2\lambda (q_H - y_k')} + e^{2\lambda (q_L - y_k')} , \quad y_k' > q_H \\
g_3 (\hat{\lambda}_k) & = 2 \cdot P_{Q_m} \left( q_{m,\beta} \left| y_k \right. \right) - 2 + e^{2\lambda (q_L - y_k')} + e^{-2\lambda (q_H - y_k')} , \quad q_L \leq y_k' \leq q_H
\end{align*}
\]

(C.17)

It can be derived that \( g_1 (\hat{\lambda}_k) \) is decreasing in \([0, \lambda_k^*] \) and increasing in \([\lambda_k^*, +\infty) \) with \( \lambda_k^* \) being equal to \( \lambda_k^* = \left( \frac{1}{2} \right) \ln \left( \frac{(q_H - y_k')/(q_L - y_k')}{1} \right) \). In addition, \( g_1 (0) \geq 0 \), \( \lim_{\lambda_k \to \infty} g_1 (\lambda_k) = 0 \). Therefore, \( g_1 (\hat{\lambda}_k) = 0 \) can have no, one or two solutions in \( \lambda_k \in [0, +\infty) \), depending on the sign of \( g_1 (\hat{\lambda}_k^*) \). Similarly, it can be found that \( g_2 (\hat{\lambda}_k) = 0 \) can have no, one or two solutions in \( \hat{\lambda}_k \in [0, +\infty) \).

Nevertheless, \( g_3 (\hat{\lambda}_k) \) is monotonically decreasing in \([0, +\infty) \), \( g_3 (0) \geq 0 \) and \( \lim_{\hat{\lambda}_k \to +\infty} g_3 (\hat{\lambda}_k) = 0 \), thus \( g_3 (\hat{\lambda}_k) = 0 \) has one and only one solution in \( \hat{\lambda}_k \in [0, +\infty) \). Hence, we choose the solution of \( g_3 (\hat{\lambda}_k) = 0 \) in order to get the online estimate \( \hat{\lambda}_k = \lambda_\beta \left( y_k \right) \) of the Laplacian scaling parameter per \( y_k \).

When \( y_k' = q_L \) or \( y_k' = q_H \), meaning when the realization of the inversely quantized side information value coincides with the upper or lower bound of the considered quantization bin, the solution of \( g_3 (\hat{\lambda}_k) = 0 \) is analytically found as

\[
\hat{\lambda}_\beta \left( y_k \right) = - \frac{\ln \left( 1 - 2 \cdot P_{Q_m} \left( q_{m,\beta} \left| y_k' \right. \right) \right)}{2^{m-m}}.
\]

(C.18)

Alternatively, when \( q_L < y_k' < q_H \) the solution of \( g_3 (\hat{\lambda}_k) = 0 \) is numerically found using the fast algorithm of [180] which combines bisection, secant, and inverse quadratic interpolation methods.
List of Publications

Publications in International Journals


**Patents**


**Publications in Book Chapters**


**Publications in Conference Proceedings**


List of Publications

Belgium, September 2011.


Processing, ICIP 2010, Hong Kong, September 2010.


REFERENCES


References


References


References


[88] F. Pereira, J. Ascenso, and C. Brites, "Studying the GOP size impact on the


References


[109] F. Verbist, N. Deligiannis, M. Jacobs, J. Barbarien, P. Schelkens, A. Munteanu, and


References


References


[141] X. Huang and S. Forchhammer, "Improved virtual channel noise model for transform domain Wyner-Ziv video coding," in *IEEE International Conference on Acoustics,
References

Speech and Signal Processing, ICASSP 2009, Taipei, Taiwan, April 2009.


[153] R. Sidhu, D. Sanders, and M. McAlindon, "Gastrointestinal capsule endoscopy: from


no. 8, pp. 1311-1321, August 2004.