

Complexity-Aware High Efficiency Video Coding

Guilherme Corrêa • Pedro Assunção
Luciano Agostini • Luis A. da Silva Cruz

Complexity-Aware High Efficiency Video Coding

 Springer

Guilherme Corrêa
Computação
Centro de Desenvolvimento Tecnológico,
Universidade Federal de Pelotas
Pelotas, Brazil

Luciano Agostini
Computação
Centro de Desenvolvimento Tecnológico,
Universidade Federal de Pelotas
Pelotas, Brazil

Pedro Assunção
Instituto de Telecomunicações (IT)
Instituto Politécnico de Leiria
Leiria, Portugal

Luis A. da Silva Cruz
Instituto de Telecomunicações (IT)
Dep. Eng. Electrotécnica e de
Computadores
Faculdade de Ciências e Tecnologia
Universidade de Coimbra
Coimbra, Portugal

ISBN 978-3-319-25776-1

ISBN 978-3-319-25778-5 (eBook)

DOI 10.1007/978-3-319-25778-5

Library of Congress Control Number: 2015954576

Springer Cham Heidelberg New York Dordrecht London

© Springer International Publishing Switzerland 2016

This work is subject to copyright. All rights are reserved by the Publisher, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, reuse of illustrations, recitation, broadcasting, reproduction on microfilms or in any other physical way, and transmission or information storage and retrieval, electronic adaptation, computer software, or by similar or dissimilar methodology now known or hereafter developed.

The use of general descriptive names, registered names, trademarks, service marks, etc. in this publication does not imply, even in the absence of a specific statement, that such names are exempt from the relevant protective laws and regulations and therefore free for general use.

The publisher, the authors and the editors are safe to assume that the advice and information in this book are believed to be true and accurate at the date of publication. Neither the publisher nor the authors or the editors give a warranty, express or implied, with respect to the material contained herein or for any errors or omissions that may have been made.

Printed on acid-free paper

Springer International Publishing AG Switzerland is part of Springer Science+Business Media
(www.springer.com)

Preface

In the last decades, the rapid advances of semiconductor technologies fostered a large development in the field of multimedia systems, mainly due to the continuous increase of computational resources and the availability of reliable communication infrastructures. Several video compression standards have been developed in this period, aiming at reducing transmission bit rates without decreasing the video quality. The High-Efficiency Video Coding (HEVC) standard, recently launched by the Joint Collaborative Team on Video Coding (JCT-VC), is the state of the art in video compression and is expected to gradually substitute its predecessor, the H.264/AVC standard. HEVC provides improved compression ratios in comparison to previous standards, but such gains are associated with large increases in the encoding computational complexity and consequently longer processing times, which may compromise the encoder operation in portable devices and in real-time systems, especially for high-resolution videos. This book addresses the subject of computational complexity of HEVC encoders with contributions extending from the analysis of HEVC compression efficiency and computational complexity to the reduction and scaling of its encoding complexity. Besides the introductory chapters, which present an overview of the HEVC standard and the state-of-the-art works on the field, this book also introduces four main contributions from the authors, which target on analysing and solving complexity-related issues in HEVC encoders. The first contribution is an investigation and detailed analysis of the HEVC encoding tools which allowed identifying the most computationally demanding operations of the encoding process. The second contribution of this book comprises a set of five new algorithms for reducing and dynamically scaling the encoding complexity of HEVC encoders. All of them take advantage from the flexibility of the frame partitioning structures allowed by the standard, namely, the coding units and the prediction units, which were identified as responsible for a large share of the encoding computational complexity. The best complexity scaling algorithm presented in this book allows downscaling the encoding complexity to 50 % of its original value with negligible loss of compression efficiency and down to 20 % with medium to small loss. The third book contribution consists of a set of early termination methods based on data mining techniques, which are able to reduce the computational

complexity required to find the best frame partitioning structures, namely, the coding trees, the prediction units and the residual quadrees, in up to 65 % with very small compression efficiency loss. Finally, the fourth contribution of this book is an encoding time control system that employs the three previous contributions to adjust the encoding time whenever necessary and maintain it under a specified target. The system uses predefined encoding configurations created by combining the early termination schemes and by changing the parameterisation of the most computationally demanding tools of HEVC. Overall, the methods proposed in this book are especially useful in power-constrained portable multimedia devices to reduce energy consumption and to extend the battery life. Besides, they can also be applied to portable and non-portable multimedia devices operating in real time with limited computational resources.

Pelotas, Brazil
Leiria, Portugal
Pelotas, Brazil
Coimbra, Portugal

Guilherme Corrêa
Pedro Assunção
Luciano Agostini
Luis A. da Silva Cruz

Contents

1	Introduction	1
1.1	Terminology for Computational Complexity.....	2
1.2	Book Organisation and Contribution	3
1.2.1	Encoding Performance and Complexity Assessment	3
1.2.2	Algorithms for Computational Complexity Scaling	4
1.2.3	Data Mining for Computational Complexity Reduction	4
1.2.4	Complexity Reduction and Scaling Applied to Encoding Time Control.....	5
	References.....	5
2	Video Coding Background	7
2.1	Basic Concepts.....	7
2.2	Lossless and Lossy Compression.....	8
2.3	Hybrid Video Compression.....	9
2.4	Distortion Metrics	12
2.4.1	Bjøntegaard Model.....	12
2.5	Rate-Distortion Optimisation.....	14
2.6	High-Efficiency Video Coding.....	15
2.6.1	Encoding Structures	17
2.6.2	Encoding Process	22
2.6.3	Test Conditions, Temporal Configurations and Profiles	31
2.7	Computational Complexity of HEVC.....	34
	References.....	35
3	State of the Art on Computational Complexity of Video Encoders	37
3.1	Modelling Computational Complexity	37
3.1.1	Computational Complexity Modelling for H.264/AVC.....	38
3.1.2	Computational Complexity Modelling for HEVC.....	41
3.2	Computational Complexity Reduction	42

3.2.1	Computational Complexity Reduction for H.264/AVC	42
3.2.2	Computational Complexity Reduction for HEVC	46
3.3	Computational Complexity Scaling	51
3.3.1	Computational Complexity Scaling for H.264/AVC	52
3.3.2	Computational Complexity Scaling in HEVC	56
3.4	Challenges and Conclusions	57
	References	58
4	Performance and Computational Complexity Assessment of HEVC	63
4.1	Analysis of HEVC Encoding Tools	63
4.1.1	Experimental Setup and Methodology	64
4.1.2	Identification of Relevant Parameters	64
4.1.3	Relevant Encoding Configurations	65
4.1.4	Encoding Performance and Complexity Trade-Off Analysis	70
4.1.5	Performance and Complexity as a Function of QP	75
4.1.6	Experiment Conclusions	75
4.2	Analysis of HEVC Frame Partitioning Structures	78
4.2.1	Experimental Setup and Methodology	79
4.2.2	Frame Partitioning Configurations Tested	79
4.2.3	Encoding Performance and Complexity Trade-Off Analysis	80
4.2.4	Experiment Conclusions	84
	References	85
5	Complexity Scaling for HEVC Using Adaptive Partitioning Structures	87
5.1	Introduction and Motivation	87
5.2	Fixed Depth Complexity Scaling	89
5.2.1	Algorithm Overview	89
5.2.2	Results for FDCS and Performance Evaluation	92
5.3	Variable Depth Complexity Scaling	93
5.3.1	Algorithm Overview	94
5.3.2	Results for VDCS and Performance Evaluation	96
5.4	Motion-Compensated Tree Depth Limitation	99
5.4.1	Algorithm Overview	99
5.4.2	Results for MCTDL and Performance Evaluation	103
5.5	Coding Tree Depth Estimation	104
5.5.1	Algorithm Overview	105
5.5.2	Results of CTDE and Performance Evaluation	107
5.6	Constrained Coding Units and Prediction Units	108
5.6.1	Rate-Distortion-Complexity Relationship in CUs and PUs	109
5.6.2	Encoding Constrained CTUs	110

- 5.6.3 Algorithm Overview 112
- 5.6.4 Results for CCUPU and Performance Evaluation 114
- 5.7 Conclusions..... 120
- References..... 124
- 6 Complexity Reduction for HEVC Using Data Mining Techniques..... 125**
 - 6.1 Introduction to Data Mining and Decision Trees..... 126
 - 6.2 Methodology 127
 - 6.3 Early Termination for Determining Coding Trees 129
 - 6.4 Early Termination for Determining Prediction Units 135
 - 6.5 Early Termination for Determining the RQT..... 140
 - 6.6 Experimental Results 145
 - 6.6.1 Experimental Setup and Methodology..... 145
 - 6.6.2 Rate-Distortion-Complexity Results for Single Schemes..... 146
 - 6.6.3 Rate-Distortion-Complexity Results for Joint Schemes 152
 - 6.7 Results and Discussion 155
 - 6.8 Conclusions..... 157
 - References..... 157
- 7 Complexity Reduction and Scaling Applied to HEVC Encoding Time Control 159**
 - 7.1 Finding the Encoder Operating Points..... 160
 - 7.1.1 Parameters Selection for R-D-C Analysis 162
 - 7.1.2 Operating Points at the Pareto Frontier..... 163
 - 7.2 Encoding Time Control System..... 166
 - 7.2.1 Medium-Granularity Time Control..... 167
 - 7.2.2 Fine-Granularity Time Control 169
 - 7.2.3 Encoding Time Limitation Algorithm 170
 - 7.3 Experimental Results 170
 - 7.4 Comparison with Previous Works..... 176
 - 7.5 Conclusions..... 177
 - References..... 177
- 8 Conclusions..... 179**
 - 8.1 Final Remarks 179
 - 8.2 Research Contributions 180
 - 8.3 Future Work 181
- Appendix A Common Test Conditions and Video Sequences..... 183**
- Appendix B Obtained Decision Trees..... 203**
- Appendix C Encoder Configurations Tested in the R-D-C Analysis..... 213**
- Index..... 221**

Abbreviations

AI	All intra
ALF	Adaptive loop filter
AMP	Asymmetric motion partition
ARFF	Attribute-relation file format
AVC	Advanced video coding
BD	Bjøntegaard delta
BD-rate	Bjøntegaard delta bit rate
BD-PSNR	Bjøntegaard delta PSNR
CABAC	Context-adaptive binary arithmetic coding
CB	Coding block
CBF	Coded block flag
CCUPU	Constrained coding units and prediction units
CIF	Common intermediate format
CTB	Coding tree block
CTC	Common test conditions
CTDE	Coding tree depth estimation
CTU	Coding tree unit
CU	Coding unit
DBF	Deblocking filter
DCT	Discrete cosine transform
DMV	Differential motion vector
DPB	Decoded picture buffer
DST	Discrete sine transform
EPZS	Enhanced predictive zonal search
Fc	Constrained frame
FDCR	Fixed depth complexity reduction
FME	Fast motion estimation
fps	Frames per second
FRME	Fractional motion estimation
FS	Full search
Fu	Unconstrained frame

GBFOS	Generalised Breiman, Friedman, Olshen and Stone Algorithm
GOP	Group of pictures
GPB	Generalised P and B picture
HD	High definition
HE	High efficiency
HETR	High-efficiency encoding time reduction
HEVC	High-efficiency video coding
HM	HEVC model
HP	High profile
I/O	Input/output
IBD	Internal bit depth
IBDD	Internal bit depth decrease
IBDI	Internal bit depth increase
IDR	Instantaneous decoding refresh
IGAE	Information gain attribute evaluation
IEC	International electrotechnical commission
IME	Integer motion estimation
IP	Intra-prediction
IQ	Inverse quantisation
ISO	International organisation for standardisation
IT	Inverse transform
JCT-VC	Joint collaborative team on video coding
KDD	Knowledge discovery from data
KLD	Kullback-Leibler divergence
LC	Low complexity
LCB	Largest coding block
LCTC	Low-complexity encoding time control
LD	Low delay
LDP	Low delay P
LM	Linear mode
MB	Macroblock
MC	Motion compensation
MCTDL	Motion-compensated tree depth limitation
MD	Mode decision
ME	Motion estimation
MPEG	Moving Picture Experts Group
MPM	Most probable modes
MSE	Mean-squared error
MSM	Merge/SKIP mode
MTDM	Maximum tree depth map
MV	Motion vector
MVP	Motion vector prediction
NSQT	Non-square transforms
PB	Prediction block
PRD	Power-rate-distortion

PRDO	Power-rate-distortion optimisation
PSNR	Peak signal-to-noise ratio
PU	Prediction unit
Q	Quantisation
QP	Quantisation parameter
QZB	Quasi-zero-blocks
RA	Random access
RAP	Random access point
R-D	Rate-distortion
R-D-C	Rate-distortion-complexity
RDCO	Rate-distortion-complexity optimisation
RDO	Rate-distortion optimisation
RDOQ	Rate-distortion optimised quantisation
RGB	Red, green and blue
RMD	Rough mode decision
RQT	Residual quadtree
SA	Search area
SAD	Sum of absolute differences
SAO	Sample adaptive offset
SATD	Sum of absolute transformed differences
SCB	Smallest coding block
SMP	Symmetric motion partition
SSE	Sum of squared error
SUMHexS	Simple unsymmetrical-cross multi-hexagon-grid search
T	Transform
TB	Transform block
TSS	Three-step search
TU	Transform unit
TZS	Test zone search
UMHexS	Unsymmetrical-cross multi-hexagon-grid search
VCEG	Video coding experts group
VDCR	Variable depth complexity reduction
VGA	Video graphics array
YCbCr	Luminance, blue chrominance and red chrominance
Y-PSNR	Luminance PSNR

List of Figures

Fig. 2.1	Generic hybrid video compression system (encoder).....	10
Fig. 2.2	Generic hybrid video compression system (decoder).....	11
Fig. 2.3	Rate-distortion (R-D) points and optimal curve	15
Fig. 2.4	Block diagram of a typical HEVC encoder	16
Fig. 2.5	Video sequence divided into a number of GOPs	17
Fig. 2.6	Frame divided into two slices and several CTUs.....	18
Fig. 2.7	Coding tree structure of a CTB divided into CBs.....	19
Fig. 2.8	Inter-frame and intra-frame PB splitting modes available in HEVC	20
Fig. 2.9	RQT structure of a CB divided into TBs	21
Fig. 2.10	A frame divided into nine tiles.....	21
Fig. 2.11	Wavefront parallel processing encoding order.....	22
Fig. 2.12	Neighbouring samples used for intra-prediction in an $N \times N$ PB of size 16×16	23
Fig. 2.13	Intra-prediction modes tested for each PB.....	24
Fig. 2.14	Examples of subpixel interpolations from full-pixel samples	26
Fig. 2.15	Multiple reference frame prediction	27
Fig. 2.16	Graphical presentation of the AI configuration	32
Fig. 2.17	Graphical presentation of the LD configuration	33
Fig. 2.18	Graphical presentation of the RA configuration	33
Fig. 4.1	Normalised computational complexity for encoding each video sequence under all 17 configurations (QP 32).....	67
Fig. 4.2	Normalised bit rate for each video sequence encoded under all 17 configurations (QP 32).....	67
Fig. 4.3	Delta Y-PSNR for each video sequence encoded under all 17 configurations (QP 32).....	68
Fig. 4.4	Normalised computational complexity for encoding each video sequence under all 16 configurations (QP 32).....	70
Fig. 4.5	Normalised bit rate for each video sequence encoded under all 16 configurations (QP 32).....	71

Fig. 4.6	Delta Y-PSNR for each video sequence encoded under all 16 configurations (QP 32).....	71
Fig. 4.7	BD-rate values for each configuration using CFG 1 as reference..	72
Fig. 4.8	BD-PSNR values for each encoding configuration using CFG 1 as reference	72
Fig. 4.9	R-D efficiency of HEVC with CFG 1, CFG 12, CFG 15 and CFG 16 and H.264 HP for the (a) BQMall, (b) vidyo4, (c) ParkScene and (d) Traffic videos (QPs 22, 27, 32, 37)	76
Fig. 4.10	Encoding time of HEVC with CFG 1, CFG 12, CFG 15 and CFG 16 and H.264 HP for the (a) BQMall, (b) vidyo4, (c) ParkScene and (d) Traffic videos (QPs 22, 27, 32, 37)	77
Fig. 4.11	Normalised computational complexity for encoding each video sequence under the seven frame partitioning configurations (QP 32).....	81
Fig. 4.12	Normalised bit rate for each video sequence encoded under the seven frame partitioning configurations (QP 32).....	81
Fig. 4.13	Delta Y-PSNR for each video sequence encoded under the seven frame partitioning configurations (QP 32).....	82
Fig. 4.14	BD-rate values for each configuration using PAR 1 as reference	82
Fig. 4.15	BD-PSNR values for each configuration using PAR 1 as reference	83
Fig. 5.1	Computational complexity of encoding CUs in each coding tree depth.....	88
Fig. 5.2	Maximum coding tree depth of random co-localised 64×64 areas for sequences (a) BQTerrace (QP 27), (b) BasketballDrive (QP 32) and (c) Cactus (QP 37)	90
Fig. 5.3	Diagram for the FDCS algorithm	91
Fig. 5.4	Average R-D performance of the FDCS method	93
Fig. 5.5	Example of operation of the VDCS strategy	94
Fig. 5.6	Diagram for the VDCS algorithm.....	95
Fig. 5.7	Adjustment of N_c value according to β	95
Fig. 5.8	N_c and CP^N variations when encoding sequence BasketballDrive, QP32.....	97
Fig. 5.9	Accuracy of the VDCS complexity scaling method	97
Fig. 5.10	Average R-D performance of the VDCS method	98
Fig. 5.11	Example of MTDM motion compensation.....	100
Fig. 5.12	Diagram for the MCTDL algorithm	101
Fig. 5.13	MTDM fragment and average MVs for the (a) 58th and (b) 59th frames of the BasketballDrive sequence (QP 32)	102
Fig. 5.14	Accuracy of the MCTDL complexity scaling method.....	103
Fig. 5.15	Diagram for the CTDE algorithm.....	106
Fig. 5.16	Pseudocode for the maximum coding tree depth decision in CTDE	107

Fig. 5.17 Accuracy of the CTDE complexity scaling method 108

Fig. 5.18 Pseudocode for the CCUPU method..... 113

Fig. 5.19 Complexity scalability accuracy for ten video sequences using QP 27 (a), QP 32 (b), QP 37 (c) and QP 42 (d) 117

Fig. 5.20 Effect of computational complexity scalability over the R-D performance..... 118

Fig. 5.21 R-D performance for each target complexity for videos (a) Traffic and (b) RaceHorses1 encoded with QPs 27, 32, 37 and 42..... 119

Fig. 5.22 Average encoding time per frame in each GOP for different target complexities: (a) PartyScene, QP 27; (b) vidyo1, QP 32; (c) BasketballDrive, QP 37; and (d) Traffic, QP 42 121

Fig. 5.23 Average encoding time per frame in each GOP for different target complexities, considering video RaceHorses1 and QP 32..... 122

Fig. 5.24 Target complexity versus running complexity for the complexity scaling methods 122

Fig. 5.25 BD-rate increase for the five complexity scaling methods 123

Fig. 5.26 BD-PSNR for the five complexity scaling methods 123

Fig. 6.1 Example of a binary classification tree 126

Fig. 6.2 Example of an ARFF file 128

Fig. 6.3 Occurrence of (a) 64×64 CUs, (b) 32×32 CUs and (c) 16×16 CUs split and not split into smaller CUs according to the PU mode chosen (i.e. Partition) 131

Fig. 6.4 Neighbouring CTUs used in the calculation of ΔNeighDepth 132

Fig. 6.5 Occurrence of (a) 64×64 CUs, (b) 32×32 CUs and (c) 16×16 CUs split and not split into smaller CUs according to the average of CU depths in neighbouring CTUs 132

Fig. 6.6 Occurrence of (a) 64×64 CUs, (b) 32×32 CUs and (c) 16×16 CUs split and not split into smaller CUs according to the ratio between the 2N×2N and MSM R-D costs..... 134

Fig. 6.7 CTUs divided into CUs in the 100th frame of the BasketballDrill video sequence encoded with QP 32 by (a) the original HM encoder and the (b) HM encoder with the coding tree early termination 135

Fig. 6.8 Frequency of occurrence of each inter-PU splitting mode in the BasketballDrive sequence encoded with (a) QP 27, (b) QP 32, (c) QP 37 and (d) QP 42..... 136

Fig. 6.9 Inter-frame PU splitting mode decision with the proposed early termination 137

Fig. 6.10 Occurrence of (a) 64×64 CUs, (b) 32×32 CUs, (c) 16×16 CUs and (d) 8×8 CUs split and not split into PUs smaller than $2N \times 2N$ according to the ratio between the R-D costs of $2N \times 2N$ and MSM modes..... 138

Fig. 6.11 Occurrence of (a) 32×32 CUs, (b) 16×16 CUs and (c) 8×8 CUs split and not split into PUs smaller than $2N \times 2N$ according to the splitting decision at the upper coding tree depth..... 139

Fig. 6.12 PU boundaries in the 100th frame of the BasketballDrill video sequence encoded with QP 32 by (a) the original HM encoder and the (b) HM encoder with the PU early termination 141

Fig. 6.13 Occurrence of (a) 32×32 and (b) 16×16 TUs split and not split into smaller TUs according to the absolute sum of luminance residues..... 142

Fig. 6.14 Occurrence of (a) 32×32 and (b) 16×16 TUs split and not split into smaller TUs according to the number of nonzero luminance coefficients after the transform 143

Fig. 6.15 Occurrence of (a) 32×32 and (b) 16×16 TUs split and not split into smaller TUs according to the R-D cost for the current TU depth 144

Fig. 6.16 TUs in the 100th frame of the BasketballDrill video sequence encoded with QP 32 by (a) the original HM encoder and the (b) HM encoder with the RQT early termination 145

Fig. 6.17 Number of bits per frame in the SlideEditing sequence (QP 32) encoded with the original HM and the modified HM with the RQT early termination 149

Fig. 6.18 Fragment of the 11th frame in the SlideEditing sequence (QP 32) encoded with (a, b) the original HM and (c, d) the modified HM with the RQT early termination..... 150

Fig. 6.19 Rate-distortion efficiency for the (a) SlideEditing, (b) BQTerrace, (c) Kimono and (d) PeopleOnStreet video sequences encoded with the original HM and the three early terminations implemented separately..... 151

Fig. 6.20 Rate-distortion efficiency for the (a) BQTerrace, (b) BQSquare, (c) Kimono and (d) PeopleOnStreet video sequences encoded with the original HM and the four joint early termination schemes implemented..... 154

Fig. 7.1 Encoding time control system integrated with the HEVC encoder 160

Fig. 7.2 Example of R-D-C space and 3-D Pareto frontier for a set of 240 encoding parameter configurations..... 161

Fig. 7.3 (a) R-C and (b) D-C average results for the 240 configurations tested. Squares represent Pareto-efficient configurations in the R-C space, circles represent Pareto-efficient configurations in the D-C space and × symbols represent the remaining configurations..... 165

Fig. 7.4 Encoding time control system integrated with the HEVC encoder..... 168

Fig. 7.5 Pseudocode for the encoding time limitation system 171

Fig. 7.6 (a) Encoding times, (b) configuration index k , (c) $N_{c(i)}^1$ parameter and (d) $N_{c(i)}^2$ parameter for each GOP in the Beauty sequence (QP 22)..... 173

Fig. 7.7 (a) Encoding times, (b) configuration index k , (c) $N_{c(i)}^1$ parameter and (d) $N_{c(i)}^2$ parameter for each GOP in the YachtRide sequence (QP 32) 174

Fig. 7.8 Encoding times for each GOP in the concatenation of the Beauty and YachtRide sequences (QP 32), with the scene change in GOP 150 175

Fig. A.1 BasketballDrillText 185

Fig. A.2 BasketballDrill 185

Fig. A.3 BasketballDrive..... 186

Fig. A.4 BasketballPass..... 186

Fig. A.5 Beauty 187

Fig. A.6 BlowingBubbles..... 187

Fig. A.7 Bosphorus 188

Fig. A.8 BQMall 188

Fig. A.9 BQSquare 189

Fig. A.10 BQTerrace 189

Fig. A.11 Cactus..... 190

Fig. A.12 ChinaSpeed 190

Fig. A.13 FourPeople 191

Fig. A.14 HoneyBee..... 191

Fig. A.15 Jockey..... 192

Fig. A.16 Johnny 192

Fig. A.17 Kimono 193

Fig. A.18 KristenAndSara..... 193

Fig. A.19 NebutaFestival..... 194

Fig. A.20 ParkScene..... 194

Fig. A.21 PartyScene..... 195

Fig. A.22 PeopleOnStreet 195

Fig. A.23 Poznan_CarPark..... 196

Fig. A.24 Poznan_Hall1 196

Fig. A.25 PoznanStreet 197

Fig. A.26 RaceHorses1 197

Fig. A.27 RaceHorses2 198

Fig. A.28	ShakeNDry.....	198
Fig. A.29	Shark	199
Fig. A.30	SlideEditing.....	199
Fig. A.31	SlideShow	200
Fig. A.32	SteamLocomotive	200
Fig. A.33	Tennis.....	201
Fig. A.34	Traffic.....	201
Fig. A.35	YachtRide.....	202
Fig. B.1	Coding tree early termination decision tree for 16×16 CUs.....	204
Fig. B.2	Coding tree early termination decision tree for 32×32 CUs.....	205
Fig. B.3	Coding tree early termination decision tree for 64×64 CUs.....	206
Fig. B.4	PU early termination decision tree for 8×8 CUs.....	206
Fig. B.5	PU early termination decision tree for 16×16 CUs.....	207
Fig. B.6	PU early termination decision tree for 32×32 CUs.....	208
Fig. B.7	PU early termination decision tree for 64×64 CUs.....	209
Fig. B.8	RQT early termination decision tree for 16×16 TUs.....	210
Fig. B.9	RQT early termination decision tree for 32×32 TUs.....	211

List of Tables

Table 2.1	PB splitting modes available for each CB size in HEVC.....	20
Table 2.2	Filter coefficients for half-pixel and quarter-pixel luminance sample interpolation.....	26
Table 2.3	$Q_{QP\%6}$ values used in the coefficient quantisation	29
Table 2.4	$IQ_{QP\%6}$ values used in the coefficient quantisation.....	30
Table 3.1	Computational complexity reduction strategies for H.264/AVC	45
Table 3.2	Computational complexity reduction strategies for HEVC.....	51
Table 3.3	Computational complexity scaling strategies for H.264/AVC	56
Table 3.4	Computational complexity scaling strategies for HEVC	57
Table 4.1	Encoder configurations for identifying relevant parameters	66
Table 4.2	Encoder configurations used for complexity and performance analysis	69
Table 4.3	Computational complexities (in seconds) for HEVC and H.264/AVC configurations under QPs 22, 27, 32, 37 and 42	73
Table 4.4	Frame partitioning structure configurations tested in the experiments.....	80
Table 4.5	BD-rate increase (%) per configuration.....	83
Table 4.6	Average computational complexity reduction (%) per configuration.....	84
Table 4.7	Ratio between BD-rate increase (%) and computational complexity reduction	84
Table 5.1	Average results obtained for the FDCS algorithm.....	93
Table 5.2	Average results obtained for the VDCC algorithm.....	98
Table 5.3	Weights for average MV calculation	102
Table 5.4	Average results obtained for the MCTDL algorithm.....	103
Table 5.5	Maximum depths used in neighbouring CTUs.....	104
Table 5.6	Average results obtained for the CTDE algorithm	108

Table 5.7	CU and PU configurations tested	110
Table 5.8	Comparison between encoding configurations.....	110
Table 5.9	Maximum depths used in neighbouring CTUs.....	111
Table 5.10	Average complexity and R-D results for CCUPU considering five different spatial resolutions.....	115
Table 5.11	Average complexity and R-D results for CCUPU.....	116
Table 6.1	Information gain attribute evaluation for the coding tree early termination	130
Table 6.2	Characteristics and performance of trees trained for the coding tree early termination	134
Table 6.3	Information gain attribute evaluation for the PU early termination.....	137
Table 6.4	Characteristics and performance of trees trained for the prediction unit early termination.....	140
Table 6.5	Information gain attribute evaluation for the RQT early termination.....	142
Table 6.6	Characteristics of trees trained for the RQT early termination.....	144
Table 6.7	R-D-C results for the coding tree early termination.....	147
Table 6.8	R-D-C results for the prediction unit early termination	147
Table 6.9	R-D-C results for the RQT early termination.....	147
Table 6.10	R-D-C results for joint coding tree and PU early terminations	152
Table 6.11	R-D-C results for joint PU and RQT early terminations.....	153
Table 6.12	R-D-C results for joint coding tree and RQT early terminations	153
Table 6.13	R-D-C results for joint coding tree, PU and RQT early terminations	153
Table 6.14	Comparisons with related works	156
Table 7.1	Encoding configurations tested in the R-D-C analysis.....	163
Table 7.2	Parameters, normalised computational complexity, BD-PSNR and BD-rate for the configurations belonging to the Pareto frontier	166
Table 7.3	Average encoding time error (TE), BD-rate and BD-PSNR results	172
Table 7.4	Comparison with related works.....	176
Table A.1	Video sequence spatial resolutions.....	184
Table C.1	Parameters, computational complexity, BD-PSNR and BD-rate for the 240 encoder configurations considered in the R-D-C analysis.....	214
Table C.2	Update table with ratios between normalised complexities of encoding configurations	220