

— IAIN E. RICHARDSON —

# CODING VIDEO

A PRACTICAL GUIDE TO HEVC AND BEYOND

with website



WILEY



**Coding Video**



# **Coding Video**

A Practical Guide to HEVC and Beyond

*Iain E. Richardson*  
*Director, Vcodex Ltd*  
*Delft*  
*The Netherlands*

**WILEY**

This edition first published 2024

© 2024 John Wiley & Sons Ltd

All rights reserved, including rights for text and data mining and training of artificial technologies or similar technologies. No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic, mechanical, photocopying, recording or otherwise, except as permitted by law. Advice on how to obtain permission to reuse material from this title is available at <http://www.wiley.com/go/permissions>.

The right of Iain E. Richardson to be identified as the author of this work has been asserted in accordance with law.

*Registered Offices*

John Wiley & Sons, Inc., 111 River Street, Hoboken, NJ 07030, USA

John Wiley & Sons Ltd, The Atrium, Southern Gate, Chichester, West Sussex, PO19 8SQ, UK

For details of our global editorial offices, customer services, and more information about Wiley products visit us at [www.wiley.com](http://www.wiley.com).

Wiley also publishes its books in a variety of electronic formats and by print-on-demand. Some content that appears in standard print versions of this book may not be available in other formats.

Trademarks: Wiley and the Wiley logo are trademarks or registered trademarks of John Wiley & Sons, Inc. and/or its affiliates in the United States and other countries and may not be used without written permission. All other trademarks are the property of their respective owners. John Wiley & Sons, Inc. is not associated with any product or vendor mentioned in this book.

*Limit of Liability/Disclaimer of Warranty*

While the publisher and authors have used their best efforts in preparing this work, they make no representations or warranties with respect to the accuracy or completeness of the contents of this work and specifically disclaim all warranties, including without limitation any implied warranties of merchantability or fitness for a particular purpose. No warranty may be created or extended by sales representatives, written sales materials or promotional statements for this work. This work is sold with the understanding that the publisher is not engaged in rendering professional services. The advice and strategies contained herein may not be suitable for your situation. You should consult with a specialist where appropriate. The fact that an organization, website, or product is referred to in this work as a citation and/or potential source of further information does not mean that the publisher and authors endorse the information or services the organization, website, or product may provide or recommendations it may make. Further, readers should be aware that websites listed in this work may have changed or disappeared between when this work was written and when it is read. Neither the publisher nor authors shall be liable for any loss of profit or any other commercial damages, including but not limited to special, incidental, consequential, or other damages.

*Library of Congress Cataloging-in-Publication Data applied for*

Hardback ISBN 9781118711781

Cover Design: Wiley

Cover Image: © Aleksandra Konoplia/Getty Images

Set in 9.5/12.5pt STIXTwoText by Straive, Pondicherry, India

*To Naomi*





## Contents

<b>About the Author</b>	<i>xvii</i>
<b>Acknowledgements</b>	<i>xix</i>
<b>About the Companion Website</b>	<i>xxi</i>

<b>1</b>	<b>Introduction</b>	<i>1</i>
1.1	Why Write This Book?	<i>1</i>
1.2	What Is in the Book?	<i>2</i>
1.3	How Should You Use This Book?	<i>3</i>
	References	<i>4</i>
<b>2</b>	<b>Video Coding and Video Quality</b>	<i>5</i>
2.1	Introduction	<i>5</i>
2.2	An Overview of Video Coding	<i>6</i>
2.2.1	Just How Much Data Are We Talking About?	<i>6</i>
2.2.2	Bandwidth Is Increasing Every Year, But...	<i>7</i>
2.3	Inputs and Outputs	<i>9</i>
2.3.1	Digital Video	<i>10</i>
2.3.2	Uncompressed Video Formats	<i>10</i>
2.3.2.1	Frames, Fields and Pixels	<i>10</i>
2.3.2.2	Colour and Colour Spaces	<i>11</i>
2.3.3	Side Information	<i>15</i>
2.3.4	Coded Video	<i>15</i>
2.4	Structural Elements	<i>15</i>
2.5	Prediction	<i>17</i>
2.5.1	Prediction in a Video Encoder	<i>17</i>
2.5.2	Prediction in a Video Decoder	<i>18</i>
2.5.3	Prediction Example	<i>18</i>
2.5.4	Choosing a Prediction	<i>21</i>
2.5.5	Inter and Intra Prediction	<i>23</i>
2.6	Transform and Quantisation	<i>25</i>
2.6.1	Transform	<i>25</i>
2.6.2	Quantisation	<i>26</i>
2.7	Bitstream Coding	<i>27</i>
2.7.1	Pre-processing	<i>28</i>

2.7.2	Entropy Encoding	29
2.8	The Coded Bitstream	30
2.9	Storing and Transmitting the Coded Bitstream	30
2.10	The Decoder	32
2.10.1	Entropy Decoding	32
2.10.2	Rescaling and Inverse Transform	32
2.10.3	Reconstruction	33
2.10.4	Decoder Output	33
2.11	The Video Codec Model	33
2.12	Video Codec Performance	35
2.12.1	Video Quality	36
2.12.2	Subjective Quality Measurement	37
2.12.2.1	Factors Influencing Subjective Quality	37
2.12.2.2	Subjective Quality Assessment Methods	37
2.12.3	Objective Quality Measurement	38
2.12.3.1	PSNR	38
2.12.3.2	Other Objective Quality Metrics	39
2.13	Conclusion	40
	References	41
<b>3</b>	<b>A History of Video Coding and Video Coding Standards</b>	<b>43</b>
3.1	Introduction	43
3.2	The Foundations of Video Coding, 1950–1990	44
3.2.1	Entropy Coding: 1952–1987	44
3.2.2	Frame Differencing: 1956–1972	46
3.2.3	The Discrete Cosine Transform: 1974–1984	47
3.2.4	The Development of the Video Codec Model: 1977–1985	48
3.2.5	Early Advances in Motion-Compensated Prediction: 1989–1995	51
3.2.5.1	Bidirectional Prediction	51
3.2.5.2	Variable Block Size Motion Compensation	52
3.2.5.3	Multiple Reference Frames	52
3.2.5.4	Motion Compensation with Fractional Pixel Accuracy	52
3.3	Video Coding Standards and Formats: 1990–2021	53
3.3.1	How Are the Standards Developed?	54
3.3.1.1	MPEG and VCEG	54
3.3.1.2	The Development Process	55
3.3.1.3	Standards and Intellectual Property Rights	55
3.3.2	H.120, H.261 and MPEG-1: Early Video Coding Standards	56
3.3.3	MPEG-2: Digital TV and DVD-Video	58
3.3.4	H.263 and MPEG-4 Part 2: Internet and Mobile Video	59
3.3.5	H.264: Multi-Purpose Video Coding	60
3.3.6	H.265/HEVC	61
3.3.7	H.266/VVC	62
3.3.8	Other Standards and Formats	62
3.3.8.1	VC-1	62

3.3.8.2	VP8, VP9 and AV1	63
3.3.8.3	Essential Video Coding and Low-Complexity Enhancement Video Coding	64
3.4	Comparing Video Coding Standards	65
3.5	Conclusions	66
	References	66
<b>4</b>	<b>Structures</b>	<b>69</b>
4.1	Introduction	69
4.1.1	How Does a Video Codec Use Structures?	69
4.1.2	Separating a Sequence into Processing Units	70
4.1.3	Regular or Irregular Structures	70
4.1.4	Transmission and Access	72
4.2	Coded Video: Sequence to Picture	72
4.2.1	Coded Sequence	73
4.2.2	Coded Frame or Picture	74
4.2.3	Multi-picture Structures	74
4.3	Coded Video: Picture to Basic Unit	75
4.3.1	Basic Units	76
4.3.2	Slices	76
4.3.3	Tiles	78
4.4	Coded Video: Basic Unit to Block	80
4.4.1	Coding Structures	80
4.4.2	Prediction Structures	80
4.4.3	Transform Structures	81
4.5	HEVC Coding Structures	82
4.5.1	Coded HEVC Sequence	82
4.5.2	Parameter Sets	82
4.5.3	Profiles and Levels	85
4.5.4	Reference Picture Sets	86
4.5.5	HEVC Picture Types and Random Access	88
4.5.5.1	Random Access Picture	88
4.5.5.2	Examples: RAPs, Trailing and Leading Pictures	90
4.5.5.3	Random Access and Error Recovery	92
4.5.5.4	Bitstream Splicing Operations	92
4.5.6	HEVC-Coded Pictures	94
4.5.7	Slices and Slice Segments	95
4.5.8	Tiles	97
4.5.9	Combining Slices and Tiles in HEVC	99
4.5.10	HEVC Structures and Parallel Processing	101
4.5.10.1	Parallel Decoding Using Tiles	101
4.5.10.2	Parallel Decoding Using Slices	102
4.5.10.3	Wavefront Parallel Processing/Parallelising CTU Rows	102
4.5.11	Coding Tree Units and Coding Units	103
4.5.11.1	Quantisation Groups	104

4.5.12	Prediction Units and Transform Units	104
4.5.12.1	Prediction Unit Structures	105
4.5.12.2	Transform Unit, Transform Block and Transform Sub-Block	106
4.5.13	HEVC Structures Example	107
4.6	Structures in Versatile Video Coding/H.266	111
4.6.1	Sequence Down to Picture	111
4.6.2	Picture Down to Coding Unit	111
4.7	Conclusion	114
	Reference	114
<b>5</b>	<b>Intra Prediction</b>	<b>115</b>
5.1	Introduction	115
5.2	The Intra Prediction Process	118
5.3	Intra Prediction Modes	118
5.3.1	DC Prediction	120
5.3.2	Planar Prediction	122
5.3.3	Directional Prediction	123
5.3.4	Sample Filtering	126
5.4	Prediction Block Sizes	126
5.5	Signalling Intra Prediction Choices	128
5.6	Choosing a Prediction	129
5.7	HEVC Intra Prediction	130
5.7.1	HEVC Intra Prediction Modes	131
5.7.2	HEVC Intra Prediction Filtering	132
5.7.3	Signalling HEVC Intra Prediction Modes	133
5.7.3.1	HEVC Luma Prediction Mode Signalling	133
5.7.3.2	HEVC Chroma Prediction Mode Signalling	133
5.7.4	HEVC Intra Prediction Examples	134
5.8	VVC Intra Prediction	139
5.8.1	Intra Prediction Modes	139
5.8.2	Intra Prediction of Rectangular Blocks	141
5.8.3	Intra Interpolation Filters	141
5.8.4	Cross-Component Prediction	142
5.8.5	Extra Reference Lines	142
5.8.6	Intra Sub-Partitions	142
5.8.7	VVC Intra-coding Performance	143
5.9	Conclusions	144
	References	144
<b>6</b>	<b>Inter Prediction</b>	<b>145</b>
6.1	Introduction	145
6.2	Inter Prediction – the Basics	146
6.2.1	Motion Vectors for Inter Prediction: The Terminology Problem	152
6.3	Forward, Backward and Biprediction	155
6.3.1	Picture or Slice Types	155

6.3.2	Forward Prediction from Past Frames	155
6.3.3	Backward Prediction from Future Frames	156
6.3.4	Biprediction from Two Frames	158
6.4	Inter Prediction Block Sizes	159
6.5	Motion Vectors	162
6.5.1	How Does the Encoder Choose Each Motion Vector?	167
6.6	Sub-Pixel Interpolation	167
6.6.1	What Does a Sub-Pixel Interpolated Reference Frame Look Like?	170
6.6.2	Interpolation Filters	171
6.6.3	Trade-Offs in Sub-Pixel Interpolation	173
6.7	Reference Pictures	176
6.7.1	Reference Pictures for Inter Prediction	176
6.7.2	Reference Picture Structures	181
6.7.2.1	Low-Delay Picture Structures	181
6.7.2.2	Random Access Picture Structures	182
6.7.3	Reference Picture Buffers	183
6.7.3.1	Adding Pictures to the Reference Picture Buffer	184
6.7.3.2	Removing Pictures from the Reference Picture Buffer	184
6.7.3.3	Ordering Reference Pictures	184
6.7.3.4	Communicating Reference Picture Structures	186
6.8	Signalling Inter Prediction Choices	186
6.8.1	Inter Prediction Parameters	186
6.8.2	Signalling the Base Motion Vector Parameters	189
6.8.3	Signalling the Delta Motion Vector Parameters	191
6.8.4	Signalling Reference Picture Choices	191
6.9	Skip Mode	192
6.10	Loop Filter	193
6.11	When Inter Prediction Does Not Find a Good Match	194
6.12	HEVC Inter Prediction	197
6.12.1	Overview	197
6.12.1.1	HEVC Inter Prediction Example	198
6.12.2	HEVC Inter prediction Unit Partitioning	200
6.12.2.1	Slice Types and PU Types	200
6.12.2.2	Symmetric PU Partitions	200
6.12.2.3	Asymmetric PU Partitions	201
6.12.2.4	CU Partitionings into Inter-PUs: Examples	202
6.12.3	HEVC Inter-PU Coding	204
6.12.3.1	Skip or Merge Coding	205
6.12.3.2	Inter-PU Coding: Advanced Motion Vector Prediction	208
6.12.3.3	HEVC Inter-PU Coding: Examples	210
6.12.4	HEVC Inter Prediction Sources	212
6.12.5	HEVC Biprediction and Weighted Prediction	213
6.12.6	HEVC Sub-Pixel Interpolation	213
6.12.6.1	Luma Sample Interpolation	213
6.12.6.2	Chroma Sample Interpolation	215

6.12.7	HEVC Reference Pictures	219
6.13	Inter Prediction in VVC	219
6.13.1	Inter Prediction Partitions	219
6.13.2	Extended Merge Mode	220
6.13.3	Affine Motion-Compensated Prediction	220
6.13.4	Biprediction Enhancements	221
6.13.5	Sub-Pixel Interpolated Motion Compensation	221
6.14	Conclusions	222
	References	222
<b>7</b>	<b>Transform and Quantisation</b>	<b>223</b>
7.1	Introduction	223
7.1.1	Transform: An Overview	224
7.1.2	Quantisation: An Overview	225
7.2	Residual Blocks	226
7.3	Block Transforms	228
7.3.1	What Is the Purpose of a Block Transform?	228
7.3.2	Spatial Frequencies and Basis Patterns	230
7.3.3	What Does the Transform Do?	231
7.3.4	Types of Block Transform	233
7.3.4.1	The Karhunen–Loeve Transform	233
7.3.5	The Discrete Cosine Transform	233
7.3.6	The Hadamard Transform	234
7.3.7	Discrete Sine Transform	234
7.3.8	The Discrete Cosine Transform in Detail	234
7.3.9	The Discrete Cosine Transform: What It Means	240
7.3.10	The Discrete Cosine Transform: Significant Coefficients	244
7.4	Quantisation	246
7.4.1	Combining Transform and Quantisation	249
7.4.2	Designing the Quantiser	251
7.5	Transform and Quantisation in Practice	253
7.6	HEVC Transform and Quantisation	254
7.6.1	Overview of HEVC Transform and Quantisation	254
7.6.2	The HEVC Transforms	256
7.6.2.1	DCT-Based Transforms	256
7.6.2.2	DST-Based Transform	258
7.6.2.3	The HEVC Forward and Inverse Transform and Quantisation Processes	259
7.6.3	HEVC Quantisation and Rescaling	260
7.6.3.1	HEVC QP and Qstep	261
7.6.3.2	HEVC Forward Quantiser	264
7.6.3.3	HEVC Rescaling	264
7.6.3.4	HEVC Quantisation Matrices	265
7.6.4	Example: The Complete HEVC Transform and Quantise Processes	267
7.6.4.1	Forward Transform and Quantise	268
7.6.4.2	Rescale and Inverse Transform	269

7.6.5	Bypassing Transform and/or Quantisation in HEVC	271
7.6.5.1	Transform Bypass	271
7.6.5.2	Transform and Quantisation Bypass	271
7.6.5.3	PCM Mode	272
7.7	Transform and Quantise in H.266 Versatile Video Coding	272
7.7.1	Transforms in H.266	272
7.7.2	Quantisation in H.266	272
7.8	Conclusions	273
	References	273
<b>8</b>	<b>Entropy Coding</b>	<b>275</b>
8.1	Introduction	275
8.2	Entropy Coding for Video Compression	278
8.2.1	Entropy Coding to Maximise Compression Efficiency	278
8.2.2	Types of Entropy Coding	280
8.2.2.1	Fixed-Length Coding	280
8.2.2.2	Variable-Length Coding	281
8.2.2.3	Arithmetic Coding	281
8.2.2.4	Binary Arithmetic Coding	281
8.2.3	Designing an Entropy Coder	282
8.2.3.1	Compression Efficiency	282
8.2.3.2	Computational Complexity	282
8.2.3.3	Synchronisation	282
8.3	Pre-processing	283
8.3.1	Pre-processing Examples	283
8.4	Probability Models and Context Adaptation	285
8.5	Variable-Length Coding	286
8.5.1	Unary Coding	287
8.5.2	Exponential Golomb Coding	287
8.5.3	Huffman Coding	289
8.5.4	Precalculated VLC Tables	290
8.5.5	Context-Adaptive VLC	290
8.6	Arithmetic Coding	291
8.7	Binary Arithmetic Coding	294
8.7.1	Binarisation	295
8.7.2	Encoding a Bin	296
8.7.3	Decoding a Bin	297
8.7.4	BAC Encoding and Decoding: Example	298
8.7.4.1	All the Possible Coding Outcomes	299
8.7.5	Binary Fractions	302
8.7.6	Sending Bits: BAC Encoder	303
8.7.6.1	Renormalisation and Outputting Bits at the Encoder	304
8.7.7	Receiving Bits at the BAC Decoder	307
8.7.8	Binary Arithmetic Coding: Putting It All Together	308
8.8	Context-Adaptive Binary Arithmetic Coding (CABAC)	309

- 8.8.1 The CABAC Encoder 310
- 8.8.2 The CABAC Decoder 311
- 8.8.3 CABAC Context Models 311
- 8.8.4 CABAC Context Adaptation 312
- 8.9 Entropy Coding in HEVC 314
  - 8.9.1 Pre-processing in HEVC 315
    - 8.9.1.1 HEVC Transform Blocks and Sub-blocks 315
    - 8.9.1.2 Coding Transform Sub-blocks 318
    - 8.9.1.3 Pre-processing HEVC Prediction Parameters 321
    - 8.9.1.4 Pre-processing HEVC Quantisation Parameters 321
  - 8.9.2 HEVC Binarisation 322
  - 8.9.3 The HEVC Binary Arithmetic Coder 322
    - 8.9.3.1 HEVC BAC Encoding 322
    - 8.9.3.2 HEVC BAC Decoding 325
    - 8.9.3.3 HEVC BAC Coding Example 327
  - 8.9.4 HEVC Context Modelling 329
  - 8.9.5 HEVC Context Models and Initialisation 331
- 8.10 Entropy Coding in H.266/VVC 334
- 8.11 Conclusion 334
  - References 335
  
- 9 Coded Video Filtering 337**
  - 9.1 Introduction 337
  - 9.2 Filtering and Video Coding 339
  - 9.3 Detecting and Correcting Video Coding Artefacts 341
    - 9.3.1 Blocking Artefacts 341
      - 9.3.1.1 Characteristics of Blocking Artefacts 341
      - 9.3.1.2 Causes of Block Artefacts 342
      - 9.3.1.3 Detecting and Correcting Block Artefacts 343
    - 9.3.2 Ringing Artefacts 344
      - 9.3.2.1 Characteristics and Causes of Ringing Artefacts 344
      - 9.3.2.2 Detecting and Correcting Ringing Artefacts 345
    - 9.3.3 Magnitude Offset Artefacts 345
      - 9.3.3.1 Characteristics and Causes of Magnitude Offset Artefacts 345
      - 9.3.3.2 Detecting and Correcting Magnitude Offset Artefacts 347
  - 9.4 HEVC In-Loop Filtering 347
    - 9.4.1 Deblocking Filter 348
      - 9.4.1.1 HEVC Deblocking Filter Example 350
    - 9.4.2 Sample Adaptive Offset (SAO) Filter 351
      - 9.4.2.1 SAO Edge Offset 352
      - 9.4.2.2 SAO Band Offset 352
  - 9.5 VVC Filtering 354



9.6	Conclusions	355
	References	355
<b>10</b>	<b>Storing and Transporting Coded Video</b>	<b>357</b>
10.1	Introduction	357
10.2	Storing and Delivering Coded Video	359
10.3	Coded Video File Formats	361
10.3.1	The MPEG-4 File Format	363
10.4	Transport of Coded Video	365
10.4.1	Broadcast	365
10.4.2	Video Calling	366
10.4.3	Video Streaming	367
10.4.3.1	Progressive Download	367
10.4.3.2	Streaming Servers	369
10.4.3.3	DASH and HLS: HTTP Adaptive Bitrate Streaming	370
10.4.3.4	Content Delivery Networks	373
10.5	Video Rate Control	374
10.6	Error Handling	379
10.7	Conclusions	381
	References	382
<b>11</b>	<b>Implementation and Performance</b>	<b>383</b>
11.1	Introduction	383
11.2	Implementing Video Codecs	384
11.3	Software Implementation	384
11.3.1	Comparing HM and x265 Encoding	387
11.4	Hardware Implementation	388
11.5	Video Codec Performance	390
11.5.1	Measuring Codec Performance	390
11.5.2	Measuring Rate-Distortion Performance	392
11.5.3	Optimising Video Codec Performance	394
11.5.3.1	Computational Optimisation	394
11.5.3.2	Rate-Distortion Optimisation	394
11.5.3.3	Software Optimisation	397
11.6	Getting Started with Experiments	397
11.6.1	Installing Software	398
11.6.2	Source Video Material	398
11.6.3	Adjusting Source Video Clips	399
11.6.4	Encode, Decode and Play Video Clips	399
11.6.5	Measure Rate-Distortion Performance	400
11.6.6	Analysing Coded Bitstreams	402
11.7	Conclusion	405
	References	405

<b>12</b>	<b>Conclusions</b>	<i>407</i>
12.1	What This Book Has and Has Not Covered	<i>407</i>
12.2	Where Is Video Coding Going Next?	<i>408</i>
12.3	Where Should You Go Next?	<i>409</i>
	References	<i>409</i>
	<b>Glossary</b>	<i>411</i>
	<b>Index</b>	<i>415</i>

## About the Author

Iain E. Richardson is an internationally recognised expert on video compression and digital video communications. He is the author of four other books about video coding, which include two widely cited books on the H.264 Advanced Video Coding standard. For over thirty years, he has carried out research in the field of video compression and video communications, as a Professor at the Robert Gordon University in Aberdeen, Scotland, and as an independent consultant with his own company, Vcodex. He advises companies and delivers lectures on video compression technology and is sought after as an expert witness in litigation cases involving video coding.



## Acknowledgements

This book would never have left my desk without the dedication, support and encouragement of my colleague Naomi Reid, who read many drafts, acted as a sounding board and kept the manuscript moving forwards until it was complete. Thank you, Naomi! I would also like to thank the ever-amazing Pat Ballantyne for reading and commenting on the final draft and for her constant support. Thanks to the editorial team at John Wiley & Sons, to Elecard, Parabola Research and Solveig Multimedia for providing access to their visualisation tools and to everyone who has asked me to explain video coding to them over the years.



## About the Companion Website

This book is accompanied by a companion website

[www.wiley.com/go/richardson/codingvideo1](http://www.wiley.com/go/richardson/codingvideo1)







# 1

## Introduction

The scene: my video coding research lab at the Robert Gordon University, Aberdeen, Scotland, 1994. ‘You’re working on video compression?’ asked my visitor. ‘Isn’t that going to be unnecessary in a few years? After all, network bandwidth is increasing every year. We’ve already got 10 Mbps Ethernet and we’ll soon have 100 Mbps. Video compression will be redundant by the year 2000’.

My visitor predicted that research into video compression would fade away as ever-increasing bandwidth and storage capacity made it possible to send uncompressed, full-quality video with ease. He offered me a wager, that my research topic, video coding, would be redundant by the year 2000. If only I had taken that bet! In fact, at the present time, the need for efficient and effective video coding is perhaps greater than ever. Generation and consumption of video content have grown more quickly than available bandwidth has in recent years. This is because of the emergence of user-generated content that comes most notably from mobile devices. Consumption has also increased through changing viewing patterns – often with multiple simultaneous streams being viewed in a single household – and the move to higher-resolution content, such as High Definition and beyond.

This book is about video coding, also known as video compression. A video encoder converts digital video into a compressed form, that is, a form that takes up less storage or transmission capacity, known as coded video. A video decoder does the opposite, converting coded video into uncompressed digital video. The combination of a video encoder and a video decoder is known as a video codec, and the processes of compression and decompression are often described as video coding.

### 1.1 Why Write This Book?

I have written four books about video coding. Why write another one?

It is now 2024, and video coding is embedded in modern life as never before. Compressed video made a major impact in the 1990s with the advent of digital television and DVD video. Other applications followed throughout the early 2000s, and now compressed video

is a fundamental part of an ever-expanding list of consumer and business applications. These include, but are not limited to:

- Recording, playing and sharing video on mobile devices.
- Video calling and videoconferencing, both of which received a massive boost following the COVID lockdowns of 2020.
- Video streaming, which is now overtaking digital TV broadcasting as the dominant way in which we watch video.
- Video in social media, with platforms such as Facebook, Instagram and TikTok increasingly relying on video as well as still images.
- Security and surveillance, from commercial security cameras to police bodycams and video-enabled doorbells.
- Automotive video, from dashcams to in-car entertainment.

Since I wrote my last book in 2010, we have seen a significant increase in the use of video compression and a shift in the way coded video is used, as well as the release of a series of new industry standards and formats for video compression. These formats include H.265/High Efficiency Video Coding (HEVC), H.266/Versatile Video Coding, VP9, AV1 and more.

These new standards and formats are publicly available documents. However, the published standards are not intended for the casual reader and can be challenging to understand. This is due in part to their focus on highly detailed technical descriptions of how standard-based video codecs work. Even a reader with a solid background in computer science or electronic engineering may need a more approachable way into the video coding standards.

I have spent much of my career explaining video coding to students, researchers, engineers and professionals. I have always tried to find approachable and intuitive explanations of how video coding works. I wrote this book primarily to gather this material in one place, and I hope that it can provide a more user-friendly guide to video coding that will enable readers to engage with the video coding standards and understand how to get the most out of video compression.

## 1.2 What Is in the Book?

In this book, I give an overview of digital video and video compression and a short history of some of the key concepts and standards. I then discuss each of the main core concepts of video coding. I explain each concept in general terms and discuss how each is put into practice in certain video coding standards. I have chosen to focus on the widely used H.26x series of standards, with a particular focus on H.265/HEVC. Each chapter is liberally illustrated with diagrams and examples.

In Chapter 2, I provide an overview of a video codec and introduce the concepts of performance and visual quality.

Chapter 3 is a short history of video coding. This chapter begins with a description of the basic concepts that developed between the 1950s and 1990s and goes on to describe the major video coding standards from H.261 through to H.266 and AV1.

Chapter 4 explains the structural elements used in video coding. A video codec processes a video by breaking it down into structures such as Groups of Pictures, pictures, slices, tiles, basic coding units, prediction blocks and transform blocks.

Chapters 5 and 6 deal with prediction. The information in a block of video pixels can be significantly reduced if we can predict the pixels from previously sent information. Intra prediction (see Chapter 5) involves predicting blocks using information that is available in the same frame of video. This is known as intra-frame or spatial prediction. Inter prediction (see Chapter 6) creates predictions using information that is available in other, previously coded frames. This is known as inter-frame or temporal prediction.

Chapter 7 examines transforms and quantisation. A transform converts a block of pixels, or difference samples, into another domain in which the important visual information is concentrated into a smaller number of values. Quantisation removes some of the resulting information, deliberately reducing the precision of the data in order to compress it, hopefully without sacrificing too much visual quality.

Chapter 8 looks at the final stage in a video encoder, entropy coding, in which data are converted into a compressed bitstream. I discuss the main types of entropy coding, with a particular emphasis on arithmetic coding, which has become the method of choice for entropy coding in recent video coding standards.

Chapter 9 deals with filtering, in particular the types of video filtering that are carried out as part of a video encoding or decoding process. In this context, video pixels are filtered to try to improve the performance of all the other compression processes.

In Chapter 10, I discuss how coded video can be stored and transmitted. Video is often stored together with audio in container files such as an MP4 file. Transport or transmission of video is an area that continues to develop, with the emergence of adaptive bitrate streaming as one of the most important ways of sending video across the internet.

Chapter 11 considers how video codecs are implemented in software or hardware and how the performance of video codecs can be measured and compared. I provide some suggestions for trying this yourself, for example, by experimenting with the various public-domain software video codecs and by analysing coded video to see how it is actually put together.

### 1.3 How Should You Use This Book?

I wrote this book with several audiences in mind. Each chapter starts by setting out basic concepts and continues into more technical detail. I have deliberately not assumed specialised background knowledge on the part of the reader. I hope that the book will be useful to:

- Students who are studying multimedia processing and communications.
- Academics and researchers, as the book presents a way to understand and approach the extensive research literature on video coding and provides a platform to develop ideas and research topics.
- Engineers and implementers, as a bridge between the basic concepts of video coding and the often challenging density of the video coding standards.
- Other professionals, as a more accessible way into this important technical topic.

I have written the book so that it can be read from start to finish, and I hope that at least some readers will do just that. Of course, you may want to dip in and out and concentrate on particular topics of interest. I would recommend reading Chapter 2, which will give an overview of how each topic fits into the overall video codec system or model. From Chapter 4 onwards, each chapter begins with an explanation of the basic concepts of the video compression process such as prediction and transform. The chapter then explains how the process is put into practice according to specific standards with a particular focus on H.265/HEVC.

At some point, you may want to go deeper into the topics presented here. For example, you might want to find out exactly how motion vectors are communicated in H.265 or H.266. There are many excellent articles in journals that can explain topics further, such as the *IEEE Transactions on Circuits and Systems for Video Technology*, which has published several special issues and special sections on H.265/HEVC [1] and a special section on H.266/Versatile Video Coding [2]. If there is any doubt or ambiguity about a particular process that has been specified in one of the standards, the standard document itself should be consulted.

Experimenting is an excellent way to deepen your understanding of a topic. Chapter 11 gives practical guidance on how to start experimenting with standard-based codecs. Examples of experiments can include changing encoding parameters to understand how they can affect the compressed video or using syntax analysers to visualise the way the video is encoded. Please visit the companion website, <https://www.vcodex.com/coding-video-book>, for further examples and resources.

I hope you are now ready to walk through the processes of coding video, starting with an overview of the video codec and the visual information it is designed to compress.

## References

- 1 Gharavi, H. (2012). Combined issue on high efficiency video coding (HEVC) standard and research. *IEEE Transactions on Circuits and Systems for Video Technology* 22 (12): 1646–1646. doi: <https://doi.org/10.1109/TCSVT.2012.2226073>.
- 2 Boyce, J.M., Chen, J., Liu, S. et al. (2021). Guest editorial introduction to the special section on the VVC standard. *IEEE Transactions on Circuits and Systems for Video Technology* 31 (10): 3731–3735. doi: <https://doi.org/10.1109/TCSVT.2021.3111712>.

## 2

# Video Coding and Video Quality

## 2.1 Introduction

Video coding or video compression bridges the gap between displayable or uncompressed video and practical methods of storing or transmitting data. Digital video is made up of a series of frames or pictures, each of which contains hundreds of thousands or millions of picture elements or pixels. Storing or transmitting video in its original, uncompressed form can require an impractically large amount of storage capacity or bandwidth. A video encoder compresses video into a smaller and more compact form known as coded video, suitable for storage or transmission. A video decoder reverses this process, extracting displayable video from coded video (see Figure 2.1). The complementary pair of an encoder and a decoder is a video codec.

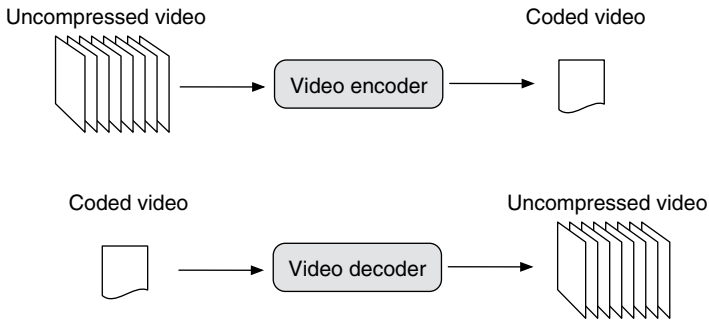
In this chapter, I will introduce each of the following aspects of a video codec, before covering them in further detail throughout the book:

**Inputs and outputs:** The data inputs and outputs for a video codec. The input to an encoder and the output of a decoder are uncompressed or raw video data, i.e. video that can be displayed. The output of an encoder and the input to a decoder are an encoded, compressed bitstream that is suitable for storage or transmission.

**Data structures:** A video codec processes data in a hierarchy from a complete video sequence, e.g. a video scene or programme, through groups of coded frames and individual coded frames, down to basic coding units (Coding Tree Units [CTUs], Macroblocks [MBs] or Superblocks) and sub-blocks.

**Prediction:** Some or all of the data to be coded are predicted from previously coded data. We can predict a block of video data from neighbouring data in the same video frame. This is known as intra prediction. Inter prediction uses video data from previously coded frames to predict a block of video data. Once the prediction is created, it is subtracted from the original block to form a residual block, which typically contains less information than the original block.

**Transform and quantisation:** Blocks of video data or residual data are transformed into another form, such as a frequency domain representation, and then quantised to reduce the data precision. This has the effect of removing or discarding much of the information in a block, at the expense of a loss of visual quality.



**Figure 2.1** Video encoder and decoder

**Bitstream coding:** Quantised blocks and information needed for decoding are converted into a compressed digital bitstream, using a form of entropy coding such as variable-length coding or arithmetic coding.

**Transport:** A coded video bitstream is arranged in a suitable form for storage and/or transmission. This might include splitting the data into packets of a convenient size for network transmission or streaming, or placing the video data in a container file, along with other information such as an audio track.

**Decoding:** A video decoder reverses the encoding processes to create a video sequence that can be displayed. Depending on the encoding process, the decoded version may not be identical to the original video sequence.

**Performance measurement:** We can measure the performance of a video codec by considering how much it compresses video, i.e. how small the compressed bitrate is; by considering how good the decoded video looks, i.e. how much it is distorted by, compared to the original video; and by considering how much computation is required at the encoder and the decoder. These three dimensions of compression, quality and computation can be used to compare the effectiveness of different video codecs.

## 2.2 An Overview of Video Coding

### 2.2.1 Just How Much Data Are We Talking About?

Figure 2.2 shows the same frame of a video clip at four different resolutions:

- 1) **Standard Definition (SD):** Contains  $720 \times 576$  pixels
- 2) **720p Definition<sup>1</sup>:** Contains  $1280 \times 720$  pixels
- 3) **High Definition (1080p):** Contains  $1920 \times 1080$  pixels
- 4) **Ultra-High Definition (UHD/4K):** Contains  $3840 \times 2160$  pixels.

One second of original, uncompressed SD video, captured at 25 frames per second, takes up around 15.5 Mbytes of storage space. This means that this video clip would require

<sup>1</sup> 720p is an intermediate resolution between Standard Definition and 1080p (Full HD).