THE UNIVERSITY OF WARWICK

Library Declaration and Deposit Agreement

1. STUDENT DETAILS

Please complete the following:

Full name: Xin Lu

University ID number: 1053500

2. THESIS DEPOSIT

2.1 I understand that under my registration at the University, I am required to deposit my thesis with the University in BOTH hard copy and in digital format. The digital version should normally be saved as a single pdf file.

2.2 The hard copy will be housed in the University Library. The digital version will be deposited in the University's Institutional Repository (WRAP). Unless otherwise indicated (see 2.3 below) this will be made openly accessible on the Internet and will be supplied to the British Library to be made available online via its Electronic Theses Online Service (EThOS) service.
[At present, theses submitted for a Master's degree by Research (MA, MSc, LLM, MS or MMedSci) are not being deposited in WRAP and not being made available via EthOS. This may change in future.]

2.3 In exceptional circumstances, the Chair of the Board of Graduate Studies may grant permission for an embargo to be placed on public access to the hard copy thesis for a limited period. It is also possible to apply separately for an embargo on the digital version. (Further information is available in the *Guide to Examinations for Higher Degrees by Research*.)

2.4 *If you are depositing a thesis for a Master's degree by Research, please complete section (a) below. For all other research degrees, please complete both sections (a) and (b) below:*

(a) Hard Copy

I hereby deposit a hard copy of my thesis in the University Library to be made publicly available to readers (please delete as appropriate) EITHER immediately OR after an embargo period of ……..... months/years as agreed by the Chair of the Board of Graduate Studies.

I agree that my thesis may be photocopied. YES / ~~NO~~ (*Please delete as appropriate*)

(b) Digital Copy

I hereby deposit a digital copy of my thesis to be held in WRAP and made available via EThOS.

Please choose one of the following options:

EITHER My thesis can be made publicly available online. YES / ~~NO~~ (*Please delete as appropriate*)

OR My thesis can be made publicly available only after…..[date] (Please give date)
YES / NO (*Please delete as appropriate*)

OR My full thesis cannot be made publicly available online but I am submitting a separately identified additional, abridged version that can be made available online.
YES / NO (*Please delete as appropriate*)

OR My thesis cannot be made publicly available online. YES / NO (*Please delete as appropriate*)

3. GRANTING OF NON-EXCLUSIVE RIGHTS

Whether I deposit my Work personally or through an assistant or other agent, I agree to the following:

Rights granted to the University of Warwick and the British Library and the user of the thesis through this agreement are non-exclusive. I retain all rights in the thesis in its present version or future versions. I agree that the institutional repository administrators and the British Library or their agents may, without changing content, digitise and migrate the thesis to any medium or format for the purpose of future preservation and accessibility.

4. DECLARATIONS

(a) I DECLARE THAT:

- I am the author and owner of the copyright in the thesis and/or I have the authority of the authors and owners of the copyright in the thesis to make this agreement. Reproduction of any part of this thesis for teaching or in academic or other forms of publication is subject to the normal limitations on the use of copyrighted materials and to the proper and full acknowledgement of its source.

- The digital version of the thesis I am supplying is the same version as the final, hard-bound copy submitted in completion of my degree, once any minor corrections have been completed.

- I have exercised reasonable care to ensure that the thesis is original, and does not to the best of my knowledge break any UK law or other Intellectual Property Right, or contain any confidential material.

- I understand that, through the medium of the Internet, files will be available to automated agents, and may be searched and copied by, for example, text mining and plagiarism detection software.

(b) IF I HAVE AGREED (in Section 2 above) TO MAKE MY THESIS PUBLICLY AVAILABLE DIGITALLY, I ALSO DECLARE THAT:

- I grant the University of Warwick and the British Library a licence to make available on the Internet the thesis in digitised format through the Institutional Repository and through the British Library via the EThOS service.

- If my thesis does include any substantial subsidiary material owned by third-party copyright holders, I have sought and obtained permission to include it in any version of my thesis available in digital format and that this permission encompasses the rights that I have granted to the University of Warwick and to the British Library.

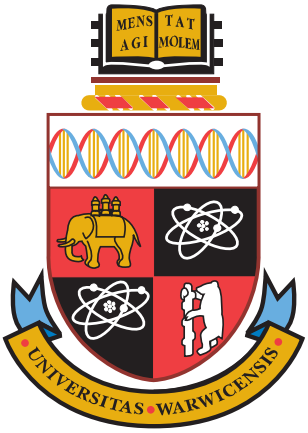
5. LEGAL INFRINGEMENTS

I understand that neither the University of Warwick nor the British Library have any obligation to take legal action on behalf of myself, or other rights holders, in the event of infringement of intellectual property rights, breach of contract or of any other right, in the thesis.

Please sign this agreement and return it to the Graduate School Office when you submit your thesis.

Student's signature: .....(Xin Lu)..... Date: .....30/11/2013.....

Efficient Algorithms for Scalable Video Coding



Xin Lu, BEng, MSc.

A thesis submitted in partial fulfilment of the requirements for the degree of Doctor of Philosophy in Computer Science

University of Warwick, Department of Computer Science
September 2013

Contents

List of Figures

List of Tables

Acknowledgements

First and foremost, I would like to take this opportunity to express my deepest gratitude and respect to my supervisor Dr. Graham Martin, who constantly offered fruitful guidance and strong support during all the time that I was in the Department of Computer Science at the University of Warwick. I benefitted greatly from his insightful advice, continuous encouragement and generous help and cannot help feeling lucky to be able to work with him. I am also looking forward to maintaining our collaboration in the future.

My parents, Lu Wenting and Wei Guangxian, and sister, Lu Na, also deserve cordial gratitude. Their love, support and encouragement have always been the source of my strength and the reason I have progressed this far. In particular, my wife, Gu Qian has unconditionally supported me throughout my studies at Warwick. Her selfless care, dedication, and love are my most valuable assets.

I also want to thank my friends and colleagues at Warwick, particularly Chen Chao, Gao Bo, Li Ruizhe, Xia Weixi, and Zhu Huanzhou for their support and friendship. They made life on the campus at Warwick enjoyable and joyful.

Last, but not least I wish to express my sincere thanks to my friend Jin Xuesong for his generous help and beneficial advice. I have benefitted a lot from his suggestions and elegant manner.

To my wife and parents

Declaration

I hereby declare that, except where acknowledged, the work presented in this thesis is my own work. No part of the work contained in this thesis has previously been accepted in substance for any degree nor submitted elsewhere for the purpose of obtaining an academic degree.

Xin Lu

Signature: _____

Date: 23 September 2013

Publications

1. **X. Lu**, G. Martin, "Performance comparison of the SVC, WSVC, and Motion JPEG2000 advanced scalable video coding schemes," accepted for publication in *Proc. IET Intelligent Signal Process.*, 6pp., Dec. 2013.
2. **X. Lu**, G. Martin, and X. Jin, "An improved rate control algorithm for SVC with optimised MAD prediction," in *Proc. York Doctoral Symposium (YDS 2013)*, 1pp., Oct. 2013.
3. **X. Lu**, G. Martin, "Rate control for scalable video coding with rate-distortion analysis of prediction modes," in *Proc. IEEE Multimedia Signal Process.*, pp. 289-294, Sep. 2013.
4. **X. Lu**, G. Martin, "Fast implementation of the scalable video coding extension of the H.264/AVC standard," in *Proc. Imperial College Computing Student Workshop (ICCSW'13)* , pp. 65-72, Sep. 2013.
5. **X. Lu**, G. Martin, "Improved rate control algorithm for scalable video coding," in *Proc. Imperial College Computing Student Workshop (ICCSW'13)*, pp. 73–81, Sep. 2013.
6. **X. Lu**, G. Martin, "Fast mode decision algorithm for the H.264/AVC scalable video coding extension," *IEEE Trans. Circuits Syst. Video Technol.* vol.23, no.5, pp. 846-855, May 2013.
7. **X. Lu**, G. Martin, "A hierarchical mode decision scheme for fast implementation of spatially scalable video coding," in *Proc. IEEE Visual Commun. Image Process.*, pp. 1-6, Nov. 2012.
8. **X. Lu**, "A improved scheme for fast implementation of scalable video coding and the performance comparison of advanced image coding schemes" Technical Report, University of Warwick, Jul. 2012.
9. **X. Lu**, G. Martin, "Fast H.264/SVC inter-frame and inter-layer mode decisions based on motion activity," *IET Electron. Lett.*, vol.48, no.2, pp. 84-86, Jan. 2012.
10. **X. Lu**, "Novel schemes for fast implementation of scalable video coding," Technical Report, University of Warwick, Jun. 2011.

Efficient Algorithms for Scalable Video Coding

Xin Lu, BEng, MSc.
A thesis submitted to
The University of Warwick
For the degree of Doctor of Philosophy
September 2013

Summary

A scalable video bitstream specifically designed for the needs of various client terminals, network conditions, and user demands is much desired in current and future video transmission and storage systems. The scalable extension of the H.264/AVC standard (SVC) has been developed to satisfy the new challenges posed by heterogeneous environments, as it permits a single video stream to be decoded fully or partially with variable quality, resolution, and frame rate in order to adapt to a specific application. This thesis presents novel improved algorithms for SVC, including: 1) a fast inter-frame and inter-layer coding mode selection algorithm based on motion activity; 2) a hierarchical fast mode selection algorithm; 3) a two-part Rate Distortion (RD) model targeting the properties of different prediction modes for the SVC rate control scheme; and 4) an optimised Mean Absolute Difference (MAD) prediction model.

The proposed fast inter-frame and inter-layer mode selection algorithm is based on the empirical observation that a macroblock (MB) with slow movement is more likely to be best matched by one in the same resolution layer. However, for a macroblock with fast movement, motion estimation between layers is required. Simulation results show that the algorithm can reduce the encoding time by up to 40%, with negligible degradation in RD performance.

The proposed hierarchical fast mode selection scheme comprises four levels and makes full use of inter-layer, temporal and spatial correlation as well as the texture information of each macroblock. Overall, the new technique demonstrates the same coding performance in terms of picture quality and compression ratio as that of the SVC standard, yet produces a saving in encoding time of up to 84%. Compared with state-of-the-art SVC fast mode selection algorithms, the proposed algorithm achieves a superior computational time reduction under very similar RD performance conditions.

The existing SVC rate distortion model cannot accurately represent the RD properties of the prediction modes, because it is influenced by the use of inter-layer prediction. A separate RD model for inter-layer prediction coding in the enhancement layer(s) is therefore introduced. Overall, the proposed algorithms improve the average PSNR by up to 0.34dB or produce an average saving in bit rate of up to 7.78%. Furthermore, the control accuracy is maintained to within 0.07% on average.

As a MAD prediction error always exists and cannot be avoided, an optimised MAD prediction model for the spatial enhancement layers is proposed that considers the MAD from previous temporal frames and previous spatial frames together, to achieve a more accurate MAD prediction. Simulation results indicate that the proposed MAD prediction model reduces the MAD prediction error by up to 79% compared with the JVT-W043 implementation.

Keywords: Fast mode decision, Inter-layer prediction, Rate control, Scalable Video Coding (SVC), SVC extension of H.264/AVC.

Abbreviations

2D	Two-Dimensional
3D	Three-Dimensional
3DTV	Three-Dimensional Television
AC	Alternating Current (high frequency)
AVC	Advanced Video Coding
B-frame	Bi-directional predictive-coded frame
BDBR	Bjϕntegaard Bit Rate
BDPSNR	Bjϕntegaard PSNR
BL	Base Layer
BLMVP	Base Layer Motion Vector Predictor
BPC	Bit Plane Coder
BR	Bit Rate
CABAC	Context-Adaptive Binary Arithmetic Coding
CAVLC	Context-Adaptive Variable Length Coding
CBR	Constant Bit Rate
CCF	Cross-Correlation Function
CCTV	Closed-Circuit Television
CD-ROM	Compact Disc Read-Only Memory
CDF	Cohen-Daubechies-Feauveau
CGS	Coarse Grain Scalability
CIF	Common Intermediate Format
CUP	Cleanup Pass
DC	Direct Current (lowest frequency)
DCT	Discrete Cosine Transform
DFT	Discrete Fourier Transform
DHT	Discrete Hadamard Transform
DPCM	Differential Pulse-Code Modulation
DST	Discrete Sine Transform
DVD	Digital Versatile Disc
DWT	Discrete Wavelet Transform

EBCOT	Embedded Block Coding with Optimised Truncation
EL	Enhancement Layer
EOB	End-of-Block
ESCOT	Embedded Subband Coding with Optimal Truncation
FGS	Fine Grain Scalability
fps	frames per second
GOP	Group of Pictures
HD	High Definition
HDTV	High Definition Television
HEVC	High Efficiency Video Coding
HHI	Heinrich Hertz Institute
HM	HEVC Test Model
HPF	High-Pass FIR Filter
HVS	Human Visual System
I-frame	Intra-coded frame
IDCT	Inverse Discrete Cosine Transformation
IDR	Instantaneous Decoding Refresh
IEC	International Electrotechnical Commission
IPTV	Internet Protocol Television
ISDN	Integrated Services Digital Network
ISO	International Standards Organisation
ITU-T	International Telecommunications Union Telecommunication Standardisation Sector
JCT-VC	Joint Collaborative Team on Video Coding
JPEG	Joint Photographic Experts Group
JSVM	Joint Scalable Video Model
JTC1	Joint Technical Committee 1
JVT	Joint Video Team
KLT	Karhunen Loève Transform
LPF	Low-Pass FIR Filter
LSB	Least Significant Bit
MAD	Mean Absolute Difference
MAE	Mean Absolute Error
MB	Macroblock
MCTF	Motion-Compensated Temporal Filtering
MGS	Medium Grain Scalability
MPEG	Moving Picture Experts Group
MRC	Magnitude Refinement Coding
MRP	Magnitude Refinement Pass
MSB	Most Significant Bit
MSE	Mean Squared Error
MSRA	Microsoft Research Asia
MV	Motion Vector

MVD	Motion Vector Difference
MVP	Motion Vector Predictor
P-frame	Predictive-coded frame
PA	Prediction Accuracy
PCC	Pearson Correlation Coefficient
PSNR	Peak Signal-to-Noise Ratio
PSTN	Public Switched Telephone Network
QCIF	Quarter Common Intermediate Format
Qp	Quantisation Parameter
R-Q	Rate-Quantisation
RD	Rate Distortion
RDO	Rate Distortion Optimisation
RLC	Run Length Coding
RM	Reference Model
RS	Reference Software
SAD	Sum of Absolute Differences
SC	Sign Coding
SC29	SubCommittee 29
SDTV	Standard Definition Television
SG16	Study Group 16
SHVC	Scalable High-efficiency Video Coding
SIF	Source Input Format
SNR	Signal-to-Noise Ratio
SPP	Significance Propagation Pass
SVC	Scalable Video Coding
TM5	Test Model 5
TMN8	Test Model Near-term 8
TR	Time Reduction
UHDTV	Ultra High Definition Television
VBR	Variable Bit Rate
VCEG	Video Coding Experts Group
VidWav	Video Wavelet
VLC	Variable Length Coding
VM8	Verification Model 8
VOD	Video on Demand
VOP	Video Object Plane
WG11	Working Group 11
WSVC	Wavelet Scalable Video Coding
ZC	Zero Coding

Chapter 1

Introduction

Video communication has become an indispensable part of the modern world. Take the video sharing website Youtube for example. In 2011, over one trillion online videos were viewed and, on average, every person on earth watched around 140 playbacks through Youtube [1]. A major factor in the huge number of video applications is the rapid development of video compression techniques. During the last few decades, video coding techniques have not only involved many new research developments, but have also achieved much commercial success. Video compression is concerned with reducing the number of bits required to represent a video sequence without significantly reducing the perceptual quality. With continual advances in network infrastructure, storage capability, and processing power, a variety of video applications set ever greater requirements for compression technology. Alternative solutions are required and this has resulted in new developments in digital video coding.

1.1 Fundamental Techniques in Video Compression

Digital video source data contains a large amount of redundancy which can be reduced or eliminated, resulting in compression. Uncompressed video sequences can require enormous amounts of storage and very large bandwidths for transmission. Assuming a video sequence of European broadcasting Standard Definition Television (SDTV) [2,3] resolution of 704×576 pixels, a frame rate of 25 frames per second (fps), 4:2:0 YCbCr colour representation (see Fig. 1-1), and 8 bits per component, the bandwidth required for transmission is:

$$(704 \times 576 \times 25 \times 8) + 2 \times (352 \times 288 \times 25 \times 8) = 116.02 \text{ Mbits/s} \tag{1.1}$$

where the first part on the left side of the equation refers to the luminance component and the second part is the chrominance component.

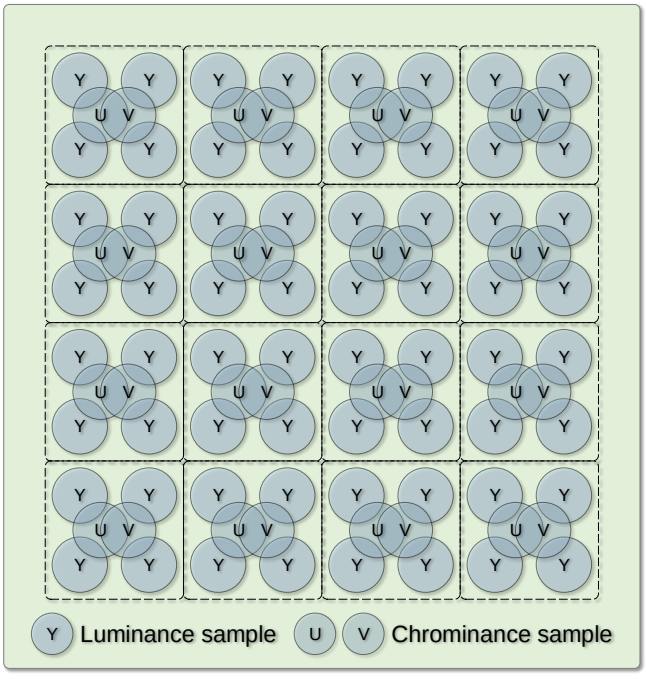


Fig. 1-1 4:2:0 chrominance component subsampling.

For High Definition Television (HDTV) [4] with a resolution of 1920×1280 pixels, the

bandwidth required increases to

$$(1920 \times 1080 \times 25 \times 8) + 2 \times (960 \times 540 \times 25 \times 8) = 593.26 \text{ Mbits/s} \tag{1.2}$$

These huge amounts of data incur significant requirements for transmission bandwidth and make storage prohibitively expensive. Elaborate video coding techniques have been developed to remove the redundant information without noticeable degradation in visual quality. Video compression algorithms typically exploit four types of redundancy to reduce the bits used to represent the original data [5].

1. **Perceptual redundancy**: The Human Visual System (HVS) is less sensitive to chrominance than to luminance, and it is more difficult to see high frequency distortion [6, 7]. Thus, information to which the HVS is insensitive can be reduced without significantly affecting the subjective quality of the picture [8].
2. **Temporal redundancy**: Successive frames in a video sequence tend to be highly correlated. Temporal redundancy is also named inter-frame redundancy. Removing the redundancy between adjacent frames by coding their difference leads to more efficient video compression.
3. **Spatial redundancy**: Each pixel is likely to have the same or a very similar value to those of its neighbouring pixels. This spatial redundancy is usually removed through spatial prediction and transform coding.
4. **Statistical redundancy**: There exists a high correlation between the quantised coefficients, Motion Vectors (MVs), and other coding coefficients. Entropy coding is employed to further reduce this redundancy and thus improve the coding efficiency.

1.1.1 Predictive Coding

The idea behind predictive coding is to exploit the correlation between adjacent pixels within a frame or between frames [9]. In predictive coding, the adjacent pixels in the same frame or in a previous frame, or their combination, are used to predict the value of the

current pixel. In this way, the current pixel is not coded directly, but the prediction error is coded instead. After predictive coding, compared to the original video sequence with high temporal and spatial redundancy, the prediction error exhibits a concentrated distribution and weak correlation. The derived prediction error usually requires fewer bits to code it, thus leading to more efficient compression. Predictive coding is classified as either temporal predictive coding or spatial predictive coding. The former is also known as inter-frame predictive coding and the latter as intra-frame predictive coding.

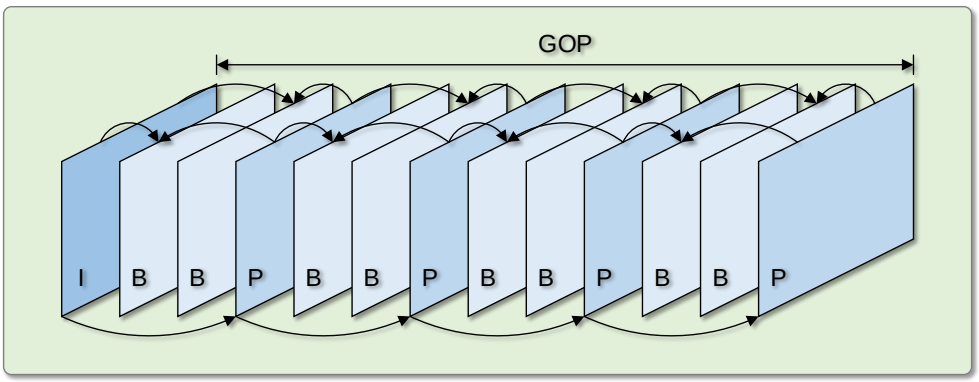


Fig. 1-2 I-, P- and B-frames in a Group of Pictures (GOP).

In most existing video coding standards, a video sequence comprises three types of frame, as shown in Fig. 1-2.

1. **I-frame (Intra-coded frame)**: I-frames are encoded without reference to any other frames, and contain only intra-coded macroblocks (these are blocks within an intra-coded frame as described in subsection 2.2.1). From the perspective of compression, an I-frame is the least efficient of the three types of frame.
2. **P-frame (Predictive-coded frame)**: P-frames are encoded using the coding results of previous I- and P-frames, and also used as a reference for the inter-frame coding of subsequent frames. In the latest standards, macroblocks in P-frames are either intra-coded or predictive-coded.
3. **B-frame (Bi-directional predictive-coded frame)**: B-frames reference both the previ-

ous and subsequent frames and contain intra-coded, predictive-coded, and bi-directional-predictive-coded macroblocks. Generally, a B-frame requires the least number of bits to encode, which makes it the most efficient of the three types of frame.

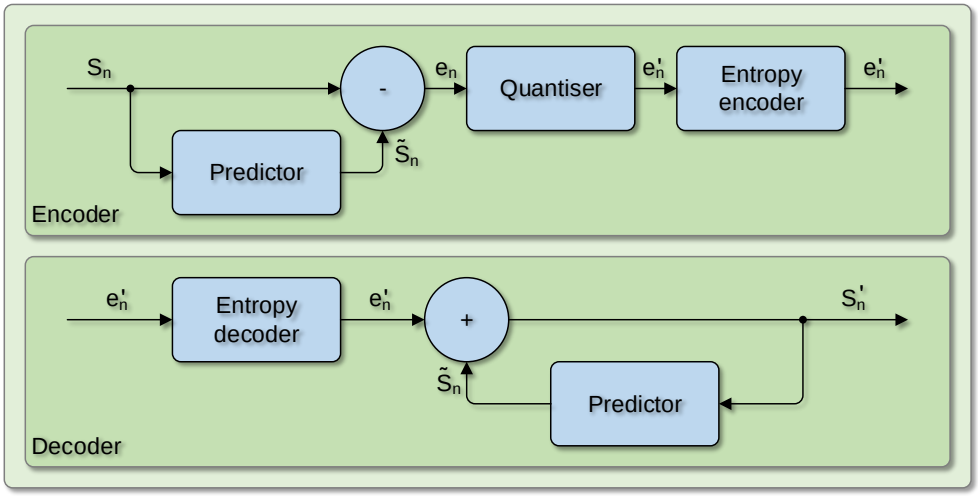


Fig. 1-3 Block diagram of a typical predictive codec.

The basic principle of predictive coding is to use the previous samples to estimate the value of the current sample. The residual or prediction error between the actual value and the estimate then forms the signal to be further processed. It is obvious that, the more accurate the prediction the lower is the resulting redundancy, and the more efficient the coding process. The block diagram of a typical predictive codec is illustrated in Fig. 1-3; notations are also introduced at appropriate points. In the diagram, the prediction $\tilde{S}_n$ of the current sample S_n is a linear combination of weighted previous samples S_i, i=1, ..., $n-1$.

$$\tilde{S}_n = \sum_{i=1}^{n-1} \alpha_i S_i \tag{1.3}$$

Instead of directly coding the current sample S_n, the prediction error e_n, which normally has less variance and energy than the original sample S_n, is coded.

$$e_n = S_n - \tilde{S}_n \tag{1.4}$$

Some quantisation noise q_n is added by the quantiser to the prediction error.

$$e'_n = e_n + q_n \tag{1.5}$$

At the decoder, the inverse procedure is performed to restore the original sample. The reconstructed prediction $\tilde{S}_n$ is added to e'_n to form the reconstructed output sample S'_n.

$$S'_n = e'_n + \tilde{S}_n = e_n + q_n + \tilde{S}_n = S_n + q_n \tag{1.6}$$

Note that the difference between the original sample and the reconstructed sample at the decoder is the quantisation noise q_n.

1.1.2 Transforms and Quantisation

Transform coding has proved to be an efficient tool to eliminate spatial redundancy in images or videos, therefore it forms an important component of almost all video coding systems [10]. Although different transformations are used in different video coding systems, they all share the same function of mapping a group of pixel samples into a different domain. As the transformation does not generate any compression, it is generally followed by quantisation and entropy coding. In the quantisation operation, the transform coefficient is assigned one of a finite number of discrete values by rounding and truncation. Consequently, the number of possible values of transform coefficients is reduced, thus resulting in compression.

Transforms

The motivation for transforming a signal from the time domain to the frequency domain is to acquire a more compact representation of the signal. Due to the prevalence of homogenous content in natural images, the transform concentrates the most energy in the lower frequency components, while the high frequency components have little energy. Since human vision is less sensitive to detailed information, the less important information is

discarded or reduced by applying a coarse quantisation to the higher frequency components. When applying the above operations, compression is obtained, but a quantisation error is necessarily introduced [11].

In a block-based coding scheme, each frame is divided into a number of blocks and the blocks are transformed to another domain. Without loss of generality, the transform process can be written as $\mathbf{R} = \mathbf{TST^t}$, where **S** denotes a block in the pixel domain, **R** refers to the representation of the block in a specific transform domain, **T** is a transform matrix and $\mathbf{T^t}$ is the transpose matrix of **T**. From the perspectives of feasible implementation and compaction efficiency, some of the transforms considered for image and video compression are the Discrete Fourier Transform (DFT), Discrete Hadamard Transform (DHT) [12], Discrete Cosine Transform (DCT) [13], and Discrete Sine Transform (DST). Among these, the DCT transform has become the common choice for most image and video coding standards.

For an 8×8 block of pixels, the Two-Dimensional (2D) DCT is expressed as

$$F(u,v) = \frac{1}{4}c(u)c(v)\sum_{x=0}^{7}\sum_{y=0}^{7} f(x,y)\cos\left[\frac{(2x+1)u\pi}{16}\right]\cos\left[\frac{(2y+1)v\pi}{16}\right] \tag{1.7}$$

where $0 \leq u \leq 7$, $0 \leq v \leq 7$ and

$$c(u), c(v) = \begin{cases} \frac{1}{\sqrt{2}} & \textbf{if } u, v = 0 \\ 1 & \textbf{otherwise} \end{cases} \tag{1.8}$$

In equation (1.7), $f(x,y)$ denotes the intensity of a pixel sample at position (x,y) and $F(u,v)$ refers to the DCT transform coefficients. The transform coefficient $F(0,0)$ represents the Direct Current (DC) value of the block and the remaining 63 transform coefficients are the Alternating Current (AC) coefficients. The inverse 2D DCT is thus defined as

$$f(x,y) = \frac{1}{4}\sum_{u=0}^{7}\sum_{v=0}^{7} c(u)c(v)F(u,v)\cos\left[\frac{(2x+1)u\pi}{16}\right]\cos\left[\frac{(2y+1)v\pi}{16}\right] \tag{1.9}$$

where $0 \leq x \leq 7$, $0 \leq y \leq 7$.

The wide usage of the DCT is credited to the following benefits that it offers [14]: 1) The DCT provides approximate compaction efficiency to the optimum Karhunen Loève Transform (KLT) for images containing homogeneous content; 2) Due to the separability of the 2D DCT, it can be implemented through a series of 1D DCTs in the horizontal direction, followed by the vertical direction; 3) The DCT is independent of the image content; 4) Fast DCT and Inverse DCT (IDCT) algorithms are widely available for efficient hardware and software implementation.

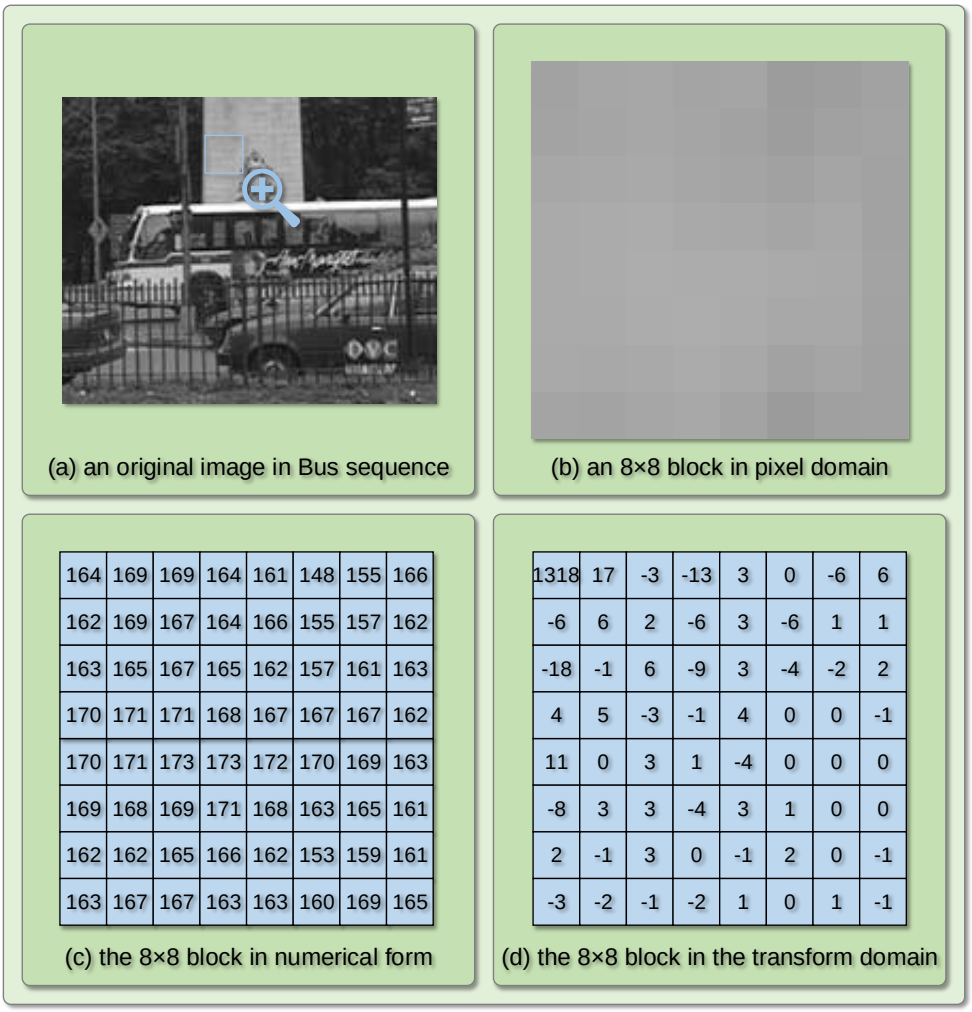


Fig. 1-4 An example of the forward DCT transform.

In order to better illustrate the property of the DCT, the procedure for transforming an

8×8 spatial block to a block of 8×8 frequency coefficients is presented in Fig. 1-4.

Fig. 1-4(a) is the 50th frame of the 'Bus' video test sequence; Fig. 1-4(b) shows the highlighted 8×8 block of Fig. 1-4(a) in the pixel domain; the 8×8 block is represented in numerical form in Fig. 1-4(c); Fig. 1-4(d) shows the transform coefficients of the 8×8 block after application of the DCT. From Fig. 1-4(d), it can be seen that the values of the DCT coefficients decrease as the horizontal and vertical frequencies increase. Most of the energy of the 8×8 block is concentrated towards the top-left corner corresponding to the low horizontal and low vertical frequency regions.

Quantisation

The quantisation operation is usually performed after the transform, and is a necessary procedure in lossy video compression. The transform coefficients are mapped to a finite set of discrete amplitudes represented by a finite number of bits. The less important transform coefficients, which do not have a significant influence on the picture quality, are removed or eliminated. The more important transform coefficients are retained. Specifically, a coarse quantisation is performed on the high frequency components and a fine quantisation on the low frequency components, since the HVS is more sensitive to the uniform regions. Quantisation therefore leads to a significant reduction in the bit rate and hence provides compression.

Quantisation rounds or truncates a transform coefficient to the nearest integer. Generally, this process is irreversible, as it is a many-to-one mapping.

A general quantisation process is described as

$$Q(u,v) = \mathbf{round}\left[\frac{F(u,v)}{Q_s}\right] \tag{1.10}$$

where $Q(u,v)$ refers to the quantisation index, Q_s is a quantisation step size, and **round** indicates the rounding function. The transform coefficient is reconstructed by rescaling the quantisation index $Q(u,v)$ after inverse quantisation, but is slightly different to the

original value.

$$\tilde{F}(u,v) = Q_s \cdot Q(u,v) \tag{1.11}$$

where the $\tilde{F}(u,v)$ is the reconstructed transform coefficient. A uniform quantiser with a quantisation step size Q_s and an uniform quantiser with a 'dead zone' are illustrated in Fig. 1-5. In Fig. 1-5(b), the dead zone is an enlarged interval around zero, which is used to reduce to zero more small transform coefficients. The quantisation results of the DCT coefficients in Fig. 1-4(d) using the uniform quantiser without dead zone and the quantiser with dead zone are illustrated in Fig. 1-6.

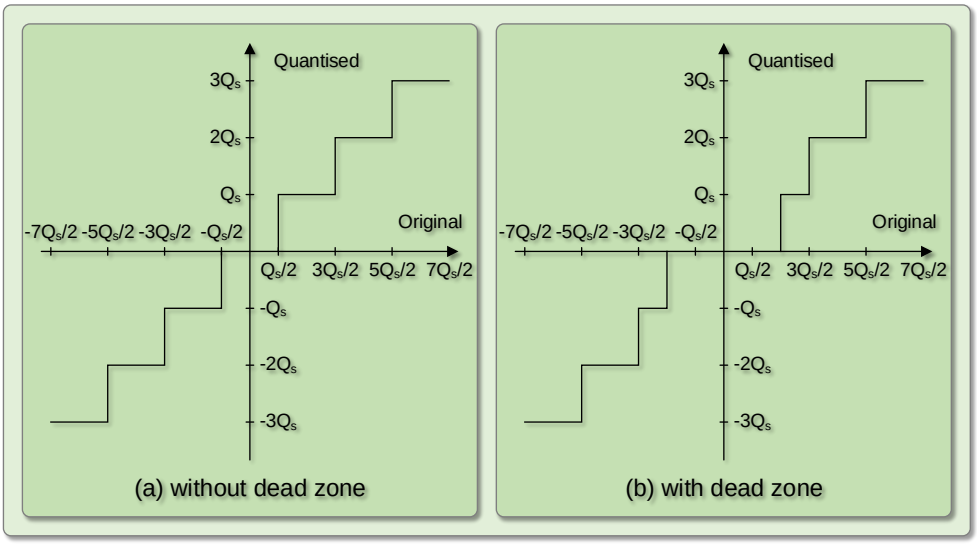


Fig. 1-5 Uniform quantisers.

1.1.3 Entropy Coding

Entropy coding is used to achieve further compression by reducing the statistical redundancy within the quantised coefficients or symbols. Relatively shorter codewords are assigned to the symbols that occur more frequently, and vice versa. The entropy coder attempts to minimise the average number of bits per symbol that are required to represent a sequence of symbols. Huffman and arithmetic coding are two types of entropy coding

commonly used in image and video compression standards.

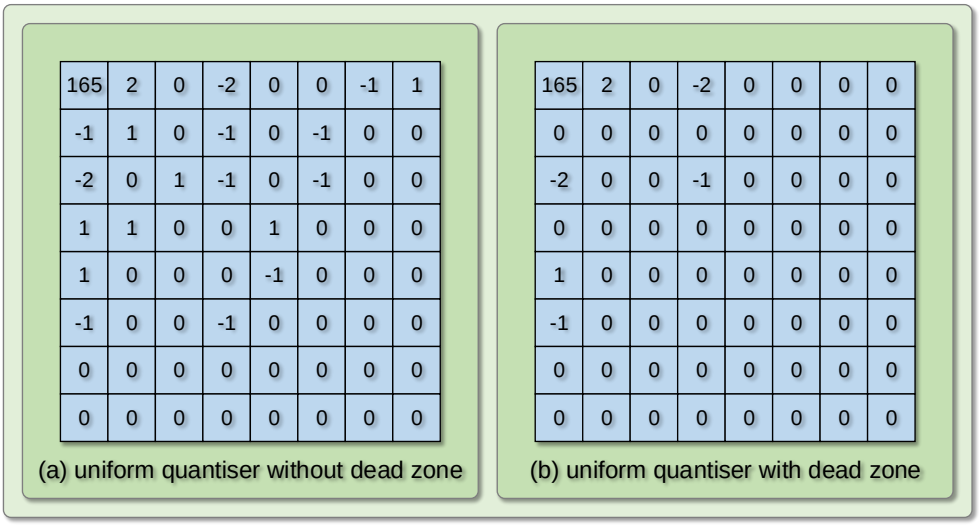


Fig. 1-6 Quantisation results with quantisation step size Q_s=8.

Many blocks contain a few significant non-zero coefficients and a large number of zero coefficients after transformation and quantisation, as shown in Fig. 1-6. These sparse blocks are normally coded by the following steps.

1. **Reorder the quantised coefficients**

   For a natural image, the non-zero transform coefficients are concentrated close to the top-left DC coefficient, and the magnitude of the transform coefficients decreases rapidly along the horizontal and vertical directions towards the bottom-right. The transform coefficients are therefore required to be reordered in a more compact representation. The optimum scan manner would be to reorder the coefficients in descending order of their magnitude, thus resulting in a series of zeros at the end of the reordered sequence. In practice, scanning in a zig-zag manner performs well, efficiently grouping the zero and non-zero coefficients. As shown in Fig. 1-7, the zig-zag scan of the transform coefficients commences at the top-left corner of the 8×8 transform coefficient matrix, where the lower frequency components reside, to the higher frequency components at the bottom-right.

2. Run Length Coding (RLC)

As the result of the zig-zag scan order operation, a 1D array is produced in which most non-zero coefficients are encountered before the zero coefficients. Instead of coding each coefficient individually, so called Run Length Coding (RLC) is employed to reduce the redundancy within the series of coefficients. The codewords of RLC comprise a series of (run, level) pairs, where 'run' represents the number of zero coefficients that precede a non-zero coefficient and 'level' refers to the value of the non-zero coefficient. An End-of-Block (EOB) symbol is used to indicate that all remaining coefficients are zero.

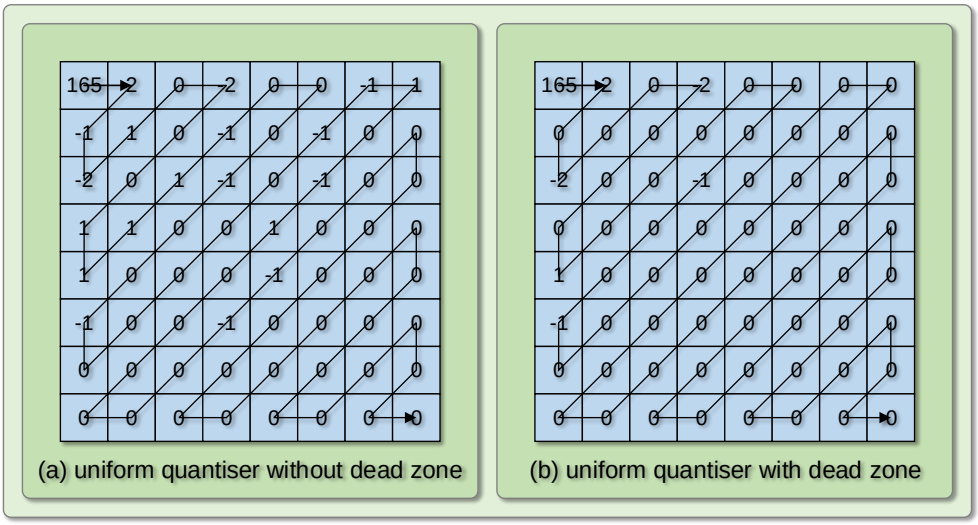


Fig. 1-7 Zig-zag scan of quantisation coefficients.

Consider the zig-zag ordered coefficients derived in Fig. 1-7(b),

165, 2, 0, -2, 0, 0, -2, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, -1, 0, 0, -1, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0.

RLC converts the coefficients into the following (run, level) pairs,

(0,165), (0,2), (1,-2), (2,-2), (3,1), (6,-1), (2,-1), EOB.

3. Variable Length Coding (VLC)

A variable length entropy coding algorithm, such as Huffman coding [15] or arithmetic coding [16], is employed to encode the (run, level) data into a set of compact binary bitstreams. The entropy coder assigns a shorter codeword to a (run, level) pair with high probability of occurrence and a longer codeword to an infrequently occurring pair, so that the average bit rate is minimised.

1.2 International Video Coding Standards

The standardisation of video coding algorithms has greatly stimulated the development of video compression technology. This has enabled a variety of new visual applications in the fields of communication, multimedia, and broadcasting [17]. Video coding standardisation work has been conducted since the early 1980s, and a series of video coding standards have been developed, each targeted at different application scenarios. The Video Coding Experts Group (VCEG) in the International Telecommunications Union Telecommunication Standardisation Sector (ITU-T), the Moving Picture Experts Group (MPEG) in the International Standards Organisation (ISO) and International Electrotechnical Commission (IEC), and a combination of the two known as the Joint Video Team (JVT) are the major organisations in the development of the standards. The evolution of video compression standards over the last three decades is summarised in Fig. 1-8.

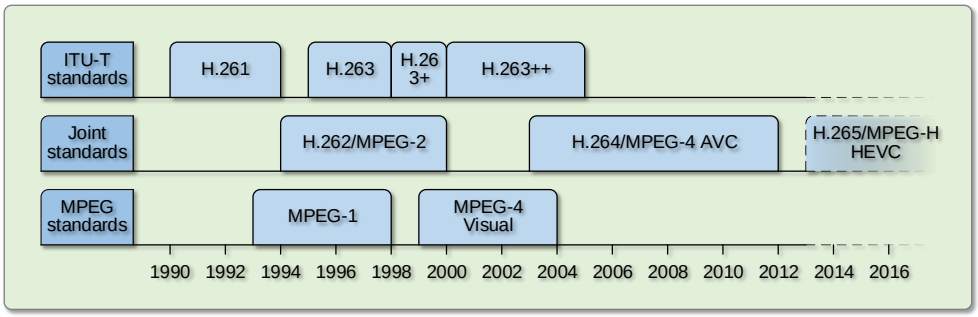


Fig. 1-8 Progression of the international video coding standards.

1.2.1 ITU-T H.261 and H.263

Generally, the ITU-T contributes work for real-time telecommunication applications, such as the H.261 standard [18] for transmission over Integrated Services Digital Network (ISDN) lines and the H.263 standard [19] for very low bit rate communications over Public Switched Telephone Network (PSTN) channels. The H.261 and H.263 standards are designed to offer high compression ratios for full colour video transmission with very low delay.

H.261

The ITU-T H.261 standard was developed for video telephony, video conferencing and other audiovisual services over ISDN channels at bit rates of $n\times64$ kbits/s, where n is an integer with values between 1 and 30. H.261 was the first video coding framework that was widely used in practical terms. It adopted a hybrid DCT/Differential Pulse-Code Modulation (DPCM) coding scheme with integer pixel motion compensation. Two frame formats are supported in H.261: Common Intermediate Format (CIF) and Quarter Common Intermediate Format (QCIF) using a 4:2:0 chroma sampling scheme, i.e. the resolution of the luminance component is 352×288 for CIF and 176×144 for QCIF and the horizontal and vertical chrominance resolutions are half those of the luminance component. The concept of a macroblock was originally suggested in H.261, where a macroblock is a basic processing unit comprising 16×16 pixels. H.261 elaborated the video coding techniques of prediction with motion compensation, DCT transform coding, quantisation, VLC and rate control.

A block diagram of an H.261 encoder is illustrated in Fig. 1-9. Inter-frame prediction is used to remove the temporal redundancy and transform coding is used to remove the spatial redundancy. As H.261 is designed to operate in real-time video telephony and video conferencing applications, motion compensation is optional and only forward motion estimation is allowed. H.261 was a successful video coding standard and was regarded as a starting point for the development of more sophisticated standards. Many coding meth-

ods in H.261 were adopted for future video coding standards, and H.261 was the first example of a transform-based video coder.

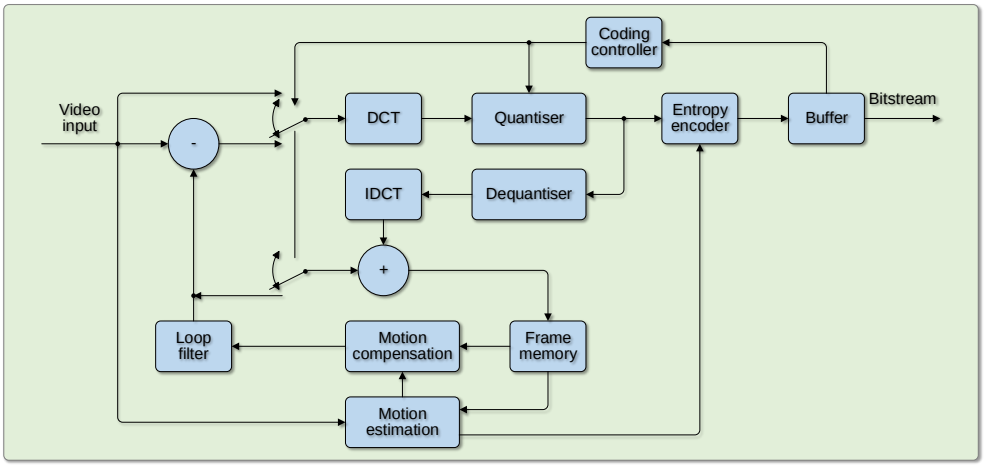


Fig. 1-9 Block diagram of a typical H.261 video encoder.

H.263

The ITU-T H.263 video coding standard was developed to support low delay video telephony applications over PSTN networks at bit rates of less than 64 kbits/s. The original version of H.263 was approved as a standard in early 1996; afterwards some new features and improvements were introduced (also known as H.263+ and H.263++) in 1998 and 1999 respectively.

The coding structure of H.263 is inherited from H.261, but more optional modes and even more frame formats, from SQCIF (128×96 pixels) to 16CIF (1408×1152 pixels), are supported. These new features allow H.263 to be applied in various application scenarios and transmission circumstances.

The following improved features enable H.263 to offer obvious superiority over H.261.

1. **Half pixel precision motion compensation**: In the H.261 codec, only integer pixel precision motion estimation and compensation were defined, whereas in H.263 half pixel precision was used to achieve better motion compensation. The prediction accuracy is improved and the high frequency components in the spatial domain are reduced.

2. **Three-Dimensional (3D) VLC**: The VLC codeword in H.263 was extended to a 3D format of (last, run, level). Similar to the codeword in H.261, 'run' indicates a run length of zero coefficients that precede a non-zero coefficient and 'level' refers to the value of the non-zero coefficient. The EOB element of H.261 is replaced by a new element 'last', which is a binary variable. '0' means that there are more non-zero coefficients in the block, and 1 means that this is the last non-zero coefficient in the block.
3. **Improved MV coding**: The MVs of the three neighbouring macroblocks are used to predict the MV of the current macroblock. Instead of directly encoding the MV, the prediction error is encoded using VLC.

By applying the above techniques, compared to H.261, H.263 achieves 50% or more savings in the bit rate needed to represent video at a given perceptual quality at very low bit rates. In terms of Signal-to-Noise Ratio (SNR), H.263 can provide about 3dB gain over H.261 at these very low bit rates.

1.2.2 ISO/IEC MPEG-1, MPEG-2, and MPEG-4 Visual

MPEG, formally, Working Group 11 (WG11) of ISO/IEC Joint Technical Committee 1 (JTC1)/ SubCommittee 29 (SC29) was formed to set standards for audio and video compression [20, 21]. The most notable MPEG standards to date are MPEG-1 [22, 23] for audiovisual data storage on CD-ROM, MPEG-2 [24–26] for high quality moving picture applications and MPEG-4 Visual [27] for the coding of audiovisual objects.

MPEG-1

MPEG-1 was the first MPEG standard and targeted at the storage of moving pictures and audio on hard disks at the bit rate of 1.5 Mbits/s to 2 Mbits/s. It accommodates progressive scan video and operates on Source Input Format (SIF) video (352×240, 352×288, or 320×240 pixels) at 25 fps and 29.97 fps. In fact, MPEG-1 is built on the work of the Joint Photographic Experts Group (JPEG) [28] and the ITU-T H.261 standard. Therefore, it adopts

many of coding techniques used in the JPEG and H.261 standards. Both MPEG-1 and H.261 adopted the hybrid DCT/DPCM codec scheme. However, the following features defined in MPEG-1 distinguish it from H.261.

1. **Types of frame**: Only forward prediction is allowed in H.261, while in MPEG-1 a so called B-frame is defined which is predicted from frames in both the forward and backward directions. Furthermore, a unique frame type is used in MPEG-1 to facilitate fast forward and backward preview, namely D-frame, which is encoded using only DC transform coefficients.
2. **Accuracy of motion estimation**: Half pixel precision is used for motion estimation in MPEG-1, while H.261 restricts the motion estimation to integer pixel accuracy. Although the half pixel precision increases the computational complexity of the codec, a better coding efficiency is gained.
3. **Motion search range**: H.261 is used mainly for video telephony and video conferencing, where the motion activity is simple and slow. Unlike H.261, MPEG-1 is normally used for the coding of movies, which contain larger movement and more complex activity. Consequently, a larger MV search range is supported in MPEG-1.

MPEG-2

The MPEG-2 standardisation activity was started in 1991 and targeted at high quality video coding at bit rates of 4-15 Mbits/s for Video on Demand (VOD), digital broadcast television, and digital storage media such as DVD. Three years later in 1994, MPEG-2 was approved by ISO/IEC as an international video coding standard. This standard was also recommended by the ITU-T as H.262. MPEG-2 was developed from MPEG-1 but included some advanced features to accommodate video applications of higher picture quality, interlaced coding, and a more flexible syntax. MPEG-2 achieved tremendous commercial success and was employed in digital terrestrial TV broadcasting, digital cable TV, and many other areas.

The basic coding structure of MPEG-2 is the same as that of MPEG-1, but the following enhancements enable MPEG-2 to show substantial superiority over MPEG-1 and H.261.

1. **Supported source format**: MPEG-2 supports a set of larger frame sizes ranging from CIF (352×288 pixels) to HDTV (1920×1080 pixels). Both progressive and interlaced scan coding are supported in MPEG-2, while MPEG-1 was designed only for progressive video coding.
2. **Scalability function**: The scalability modes of MPEG-2 enable interoperability among different services or accommodation of different receivers and networks upon which a single service may operate. MPEG-2 allows the decoder to decode a subset of the full bitstream in order to display a video sequence at a reduced quality, spatial or temporal resolution.
3. **Alternate scan order**: As well as the zig-zag scan order of DCT coefficients used in MPEG-1 and H.261, MPEG-2 has an alternate scan order to accommodate interlaced video.
4. **Quantisation of DCT coefficients**: Both linear and non-linear quantisation of DCT coefficients are supported in MPEG-2 [29, 30]. The non-linear quantisation increases the precision of quantisation at high bit rates by employing lower quantiser scale values. This improves picture quality in low contrast areas.

MPEG-4

There are two video coding standards defined in MPEG-4: MPEG-4 part 2 (also known as MPEG-4 Visual) and MPEG-4 part 10 (also known as MPEG-4 AVC) [27, 31, 32]. In contrast to conventional block-based video coding standards, MPEG-4 Visual was the first object-based visual compression scheme which enables not only efficient compression, but also enhanced flexibility, extensibility and accessibility for a wide range of applications. In the object-based MPEG-4 video coding system, each video scene is made up of a number of Video Object Planes (VOPs), and each VOP is characterised by intrinsic properties such as

shape, motion and texture. Each VOP is encoded independently to others using a coding algorithm similar to H.263 and context-based arithmetic coding is employed to code the shape of the VOP. In this way, MPEG-4 achieves the following primary features.

1. **Higher compression efficiency**: A number of advanced coding tools are used to increase the coding efficiency of MPEG-4 including: DC prediction, AC prediction, alternate scan order, and global motion compensation.
2. **Interactive functionality**: A video scene can be coded as a set of VOPs rather than a series of frames. This is a novel feature in MPEG-4 and allows both foreground and background to be coded independently. This feature enables the user to access and manipulate individual objects in a video scene.
3. **Universal access functionality**: Several mechanisms were incorporated into MPEG-4 to handle transmission errors and maintain successful video transmission in error-prone network environments. MPEG-4 also supports spatial and temporal scalability, which provide flexible solutions over a wide range of transmission bit rates.

1.2.3 ITU-T H.264/MPEG-4 AVC

The video coding experts from ITU-T VCEG and ISO/IEC MPEG formed the JVT in 2001. The collaborative result, H.264 or MPEG-4 part 10 Advanced Video Coding (AVC) was finalised in March 2003 [32–37]. This standard aimed to achieve a coding efficiency at least twice better than that of the earlier video codecs, such as H.263 or MPEG-4 Visual.

H.264/MPEG-4 AVC has been used for many applications: broadcasting HDTV over cable, terrestrial and satellite channel video transmission, storage of high quality video on optical and magnetic storage devices, and some other emerging video applications such as VOD, Internet Protocol Television (IPTV) and mobile video communications.

H.264/MPEG-4 AVC introduced a number of new features that provide better coding efficiency than its predecessors. Some of the notable features are as follows:

1. **Multiple frame motion-compensated prediction**: The number of reference frames is

increased up to 16 in a more flexible fashion than that of earlier standards.

2. **Variable block size motion compensation**: The block sizes used in H.264/MPEG-4 AVC range from 4×4 to 16×16 pixels.
3. **Quarter pixel precision for motion compensation**
4. **Weighted prediction**
5. **Directional spatial prediction for intra-coding**
6. **Exact match transform**
7. **Hierarchical block transform**
8. **In-loop deblocking filtering**
9. **Decoupling of referencing order from display order.**

The above and some other techniques enable H.264/MPEG-4 AVC to achieve a significant improvement over earlier standards under a wide range of circumstances.

1.2.4 ITU-T H.265/MPEG-H HEVC

H.265/MPEG-H High Efficiency Video Coding (HEVC) is the most recent joint standardisation work of the Joint Collaborative Team on Video Coding (JCT-VC), which is formed from the ITU-T VCEG and the ISO/IEC MPEG standardisation bodies [38].

The first version of the H.265/MPEG-H HEVC standard [39] was published in January 2013, and further improvements are still under development. A scalable extension and multi-view extension of H.265/MPEG-H HEVC are being developed and will be finalised in the near future. In order to satisfy new challenges posed by emerging video applications such as Ultra High Definition Television (UHDTV) and Three-Dimensional Television (3DTV), HEVC aims to further reduce the bit rate by a further 50% compared to the current state-of-the-art H.264/MPEG-4 AVC standard [40]. H.265/MPEG-H HEVC features a comprehensive suite of coding tools enabling a significant improvement over the prior standards. The new features of H.265/MPEG-H HEVC along with those of H.264/MPEG-4 AVC are summarised in Table 1-1.

Table 1-1
Comparison of tools in H.264/MPEG-4 AVC and H.265/MPEG-H HEVC*

Features		H.264/AVC	HEVC
Partitioning	Coding Partitioning	16×16 macroblock	Variable, large size
	Prediction Partitioning	Quadtree-based structure	Irregular, large size
	Transform Partitioning	4×4 and 8×8	Rectangular, large size
Motion	MVP Derivation	Median	MV competition
	Motion Sharing	No	Yes
	Motion Inference	B_DIRECT, SKIP	P_DIRECT; Enhanced B_DIRECT; SKIP; Template matching
Inter Prediction	Subpel Interpolation Filter	6-tap FIR; bilinear filter	Fixed filter; Wiener-based adaptive filter
	Parametric OBMC	No	Yes
	MV Precision	1/4-pel	1/2-, 1/4-, 1/8-, 1/12-pel adaptive
	Spatial-Temporal Prediction	No	Yes. Intra-prediction for inter residual
	Weighted Prediction	Signalled at slice level	Signalled at partition level; Modified offset; Illuminance prediction
Intra Prediction	Texture Synthesis	No	Yes. Template matching average
	Pre-filtering	High-profile intra 8×8 only	On/Off for all block partitions
	Post-filtering	No	Yes. Filters applied on predictor
	Plane Prediction	Yes	Yes. Bilinear; Plane-fitting
	Directional Prediction	At most 8 directions	More than 8 directions

(Continued on next page)

Table 1-1
Comparison of tools in H.264/MPEG-4 AVC and H.265/MPEG-H HEVC*

(Continued from last page)

Features		H.264/AVC	HEVC
Intra Prediction	Short-term Prediction	No. Only long-term prediction	Yes. Minimise distance between reference and predicted pixels
	Chroma Prediction	Independent prediction	Refer to segment information of reconstructed luma samples
Transform Coding and Quantisation	Directional Transform	No. Only integer DCT	Yes. MDDT; Switchable transform; Rotational transform
	Quantisation Matrix Adaptation	No. Fixed weighting matrix	Yes. Context-adaptive selection of weighting matrices
In-Loop Filter	De-blocking Filter	Horizontal and vertical edges	Simplified design; De-banding algorithm
	Adaptive Loop Filter	No	Yes
Entropy Coding	Entropy Coder	VLC; CAVLC; CABAC	Modified CAVLC; CABAC; V2V
	Parallelization	No. Only serial processing	Yes. Entropy slice-, syntax-, bin-level parallelization
	Adaptive Coefficient Scanning	No. Only zig-zag scanning	Yes. Switchable scanning order
Calculation Accuracy	Bit Depth	8 bits	Internal bit-depth increasing; Dynamic data range extension

*(taken from [41])

1.3 Research Contributions

The main contributions detailed in this thesis relate to improved algorithms for techniques employed in the scalable extension of H.264/AVC standard.

The mode selection process in SVC requires a much larger amount of computation than the scalable profiles of previous video coding standards, as SVC intends to support temporal, spatial and quality scalability. Furthermore, SVC demonstrates significantly improved coding efficiency compared with existing video coding standards. However, this is achieved at the cost of additional computation. This thesis describes methods to reduce the computational complexity of the SVC encoder without significantly degrading the Rate Distortion (RD) performance. Chapter 4 presents a simple and efficient mode selection algorithm and this process is extended into a hierarchical scheme in chapter 5. The proposed fast mode selection algorithm makes full use of inter-layer, temporal and spatial correlation, as well as the texture information of each macroblock. It produces state-of-the-art performance in terms of encoding time reduction.

An inter-layer prediction mechanism to reuse the coded lower layer data for encoding the corresponding enhancement layer is employed in SVC. However, the effects of inter-layer prediction are not taken into consideration in the rate control scheme of SVC. The existing RD models cannot accurately represent the RD properties of the prediction modes. Chapter 6 analyses the RD statistical properties of the different prediction modes and develops a more accurate RD model for the spatial enhancement layers. Furthermore, some encoding results of the base layer can be used to inform the encoding of the enhancement layers, thus benefiting from the bottom-up coding structure of SVC. An optimised Mean Absolute Difference (MAD) prediction model for the spatial enhancement layers is detailed. Simulation results show that the proposed methods achieve better rate control accuracy than the default rate control scheme of SVC, Furthermore, the proposed algorithms attain higher coding efficiency.

1.4 Thesis Outline

This chapter provided a brief review of fundamental techniques in video coding, including predictive coding, transform coding, quantisation and entropy coding. Building on the video compression principles mentioned above, currently used and evolving international video coding standards: H.261, MPEG-1, H.262/MPEG-2, H.263, MPEG-4 Visual, H.264/MPEG-4 AVC, and H.265/MPEG-H HEVC, are briefly described. This background knowledge is important to further discussions in this thesis.

The next chapter discusses the importance and advantages of scalable coding in video communication and also describes the basic principles of the scalable extension of the H.264/AVC standard (SVC). Chapters 3 through to 6 describe my main contributions to the field. These include performance analysis of advanced scalable video codecs (chapter 3) and improvements made to techniques employed in the SVC standard (chapters 4, 5, and 6). Finally, concluding remarks and suggestions for future work are included in chapter 7. The following provides a detailed outline of each chapter.

† Chapter 2 reviews the functional structure of SVC. The basic concept of scalability types as well as the advanced techniques incorporated in SVC are then explained to provide the prerequisite knowledge required for the remaining chapters. Two problems to be addressed in this thesis are also presented.

† Chapter 3 provides an analytic comparison of the three advanced scalable video coding schemes, SVC, Motion JPEG2000, and Wavelet Scalable Video Coding (WSVC). Coding efficiency in terms of RD performance is examined in detail.

† Chapter 4 initially reveals the relationship between best coding mode and motion activity, then presents a fast inter-frame and inter-layer mode selection algorithm utilising the motion activity in the video sequence.

† Chapter 5 proposes a hierarchical fast mode decision scheme that exploits the temporal, spatial and inter-layer correlation. Extensive simulation results are provided and a

performance comparison with state-of-the-art algorithms is presented.

† Chapter 6 presents a novel rate control algorithm for the enhancement layers of SVC, in which a new RD model and an optimised MAD prediction model are described. With the proposed rate control algorithm, good bit rate control and a higher coding efficiency is achieved.

† Chapter 7 summarises the thesis, conclusions are drawn, and some directions for future work discussed.

Chapter 2

Scalable Video Coding

This chapter reviews the scalable extension of the H.264/AVC standard by describing its functional structure, basic modes of scalability and some associated techniques. In order to generate a scalable video bitstream, a layer-based coding scheme is employed. A H.264/AVC compatible encoder is used for the Base Layer (BL) and a scalable video encoder for the Enhancement Layers (ELs). In each layer, motion-compensated prediction and intra-prediction are employed as for single layer coding. Furthermore, when encoding the enhancement layers, the inter-layer prediction mechanisms are introduced to remove the redundancy between layers. SVC mainly supports three kinds of scalability, in temporal frame rate, spatial resolution, and reconstruction quality. In the next section, the layer-based coding scheme and the three basic types of scalability are described in detail. In section 2.2, SVC is discussed in terms of its prediction, transform, quantisation, and entropy coding tools. Sections 2.3 and 2.4 highlight two problems which the work presented in this thesis seeks to address. Finally, section 2.5 summarises this chapter.

2.1 Overview of the Scalable Extension of H.264/AVC

Early video coding systems encoded video at a fixed target bit rate for a specific application. An increasing number of applications imposed higher demands on the nature of the video service provided. High coding efficiency is not the only goal, but also the ability to meet various client terminal capabilities, network conditions, and user demands. In order to meet the requirements of these new video coding challenges, encoded video that supports a highly scalable, easily adaptable, and fully accessible bitstream has attracted much attention in both industry and academia [42, 43]. The simulcast technique provides a simple solution for scalable video. It independently encodes multiple versions of the video at different resolutions and transmits that version which is most appropriate for the bandwidth available. Due to the low efficiency of simulcast, a better solution that guarantees efficient data transmission to video clients with diverse needs over heterogeneous networks is desirable [44]. As shown in Fig. 2-1, scalable video coding aims to encode the original video once, but permits the compressed bitstream to be decoded with a lower frame rate, smaller spatial size or degraded quality in accordance with the device capabilities, network characteristics and user requirements. Temporal, spatial and quality scalability can be achieved by selectively transmitting and decoding the required substreams. Compared with simulcast, scalable video coding possesses a greater ability to satisfy various needs and to achieve higher coding efficiency. The challenges presented have meant that scalable video coding has become an active research topic, attracting extensive attention from experts in the field of video processing.

The JVT of ITU-T VCEG and ISO/IEC MPEG has defined a scalable extension to the H.264/AVC standard, namely SVC. SVC incorporates the many new coding tools which are employed in H.264/AVC to improve coding efficiency and robustness. These techniques include: 1) Variable size block matching for motion estimation and compensation, 2) Quarter pixel accuracy for motion estimation, 3) Multiple reference frames for motion-

compensated bi-prediction, 4) Directional spatial prediction for intra-frame coding, 5) Adaptive deblocking filtering within the motion-compensated prediction loop, 6) Hierarchical block transforms, 7) Exact inverse transforms, 8) Arithmetic entropy coding, and 9) Context-adaptive entropy coding. A further introduction of the new coding tools in SVC is presented in section 2.2.

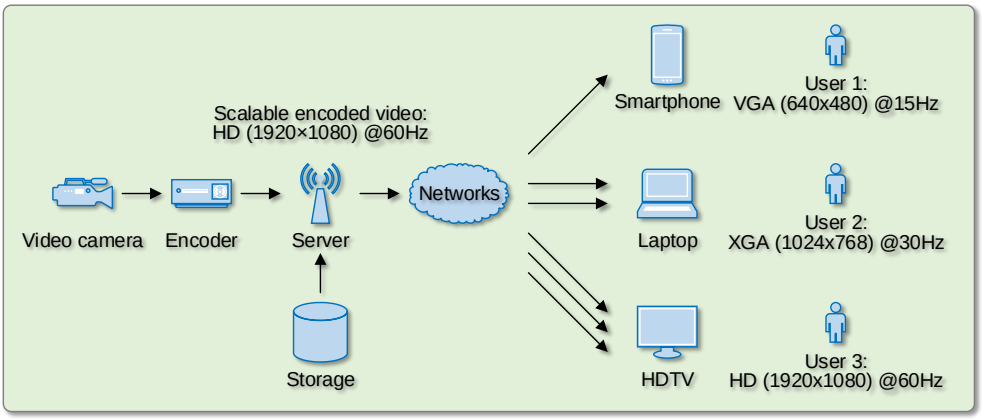


Fig. 2-1 Scalable video coding over heterogeneous networks with heterogeneous terminals.

In addition to the features mentioned above, some new tools [44] have been designed for SVC to implement flexible bitstream adaptation: 1) A hierarchical bi-directional prediction structure is utilised to achieve temporal scalability, 2) Inter-layer prediction techniques containing inter-layer motion prediction, inter-layer residual prediction and inter-layer intra-prediction are introduced to improve coding efficiency by exploiting information from the lower layer, called the reference layer, 3) Three types of quality scalability: Coarse Grain Scalability (CGS), Medium Grain Scalability (MGS) and Fine Grain Scalability (FGS) are supported.

2.1.1 Structure

Scalable video coding is often known as layer-based video coding. A bitstream generated by a layer-based coder consists of one base layer and a number of enhancement layers.

The base layer carries the most essential information and the enhancement layers contain the complementary information required to enhance the perceptual quality. In the case of bandwidth shortage, the less important enhancement layer data is intentionally discarded. When bandwidth resources permit, one or more enhancement layers are also transmitted. As expected, the more bits that are transmitted, the better the overall quality.

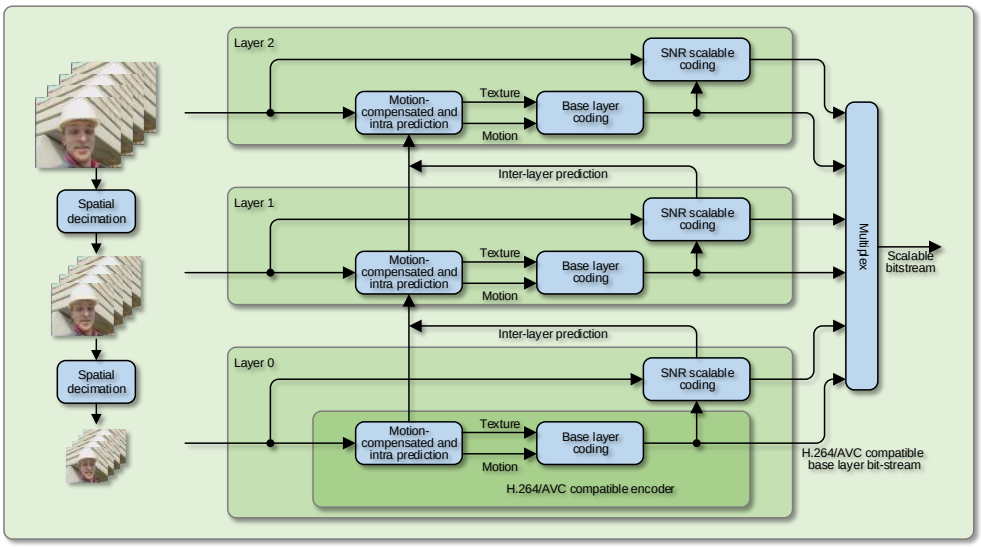


Fig. 2-2 The general coding structure of the scalable extension of H.264/AVC with three spatial layers.

The SVC development team aimed to develop a fully scalable video codec as an extension of the H.264/AVC standard, and all the well-designed coding tools of H.264/AVC were inherited. The primary design principle of SVC was that new tools should only be added if necessary to support the required types of scalability [44]. A typical encoder diagram supporting three spatial layers is illustrated in Fig. 2-2. Each layer is assigned a dependency identifier starting from 0 for the base layer and increasing by 1 from each layer to the next. For the base layer, the input video is coded by a H.264/AVC compatible encoder and can be independently decoded by a decoder conforming to the H.264/AVC standard. For the enhancement layers, the video is coded by a scalable encoder, layer by layer. In each layer, motion-compensated prediction and intra-prediction, which are in-

herited from H.264/AVC, are employed as for single layer coding. Furthermore, when encoding the enhancement layers, new inter-layer prediction mechanisms are introduced to remove the redundancy between layers. This results in significant improvements in the coding efficiency of the enhancement layers compared with simulcast.

2.1.2 Basic Modes of Scalability

A layer-based architecture is used in the SVC standard to produce a single bitstream that contains multiple versions of the same video content. This bitstream comprises a base layer and one or more enhancement layers. Enhancement layers are added to the base layer in an optional manner to improve the quality of the video sequence. The quality of the video is improved in three ways: frame rate, spatial resolution, and picture fidelity, corresponding to the temporal, spatial, and quality scalability functions [30].

In SVC, more flexible temporal scalability is provided by a hierarchical tree scheme [45]. This scheme allows both dyadic and non-dyadic temporal scalability, so that better bandwidth flexibility is provided [46]. The SVC standard not only supports spatial video coding with a dyadic resolution ratio, but even arbitrary resolution ratios. This is achieved by the multi-layer coding architecture. Furthermore, three types of quality scalability are defined in SVC, namely Coarse Grain Scalability (CGS), Medium Grain Scalability (MGS) and Fine Grain Scalability (FGS). The quality scalability is achieved by applying different quantisation step sizes to each quality layer. The above three types of scalability can be combined to form a single bitstream, which contains a number of representations with a variety of frame rates, spatial resolutions, and picture quality. The combination of the three types of scalability is often referred to as hybrid scalability or combined scalability.

Each scalability type and the hybrid scalability will be discussed individually in the following subsections, with particular focus on their features and the techniques involved.

Temporal Scalability

A video sequence is composed of a series of consecutive frames. Intuitively, video sequences with a higher frame rate, appear more smooth and natural, and result in better visual quality. Nevertheless, an increase in frame rate results in a significant increase in the quantity of data to be transmitted. Also, the end user's device needs to possess greater processing and display capability. Therefore, in order to serve the various needs or preferences of different end users, multiple video streams of the same video content but with varying frame rate should be offered by a service provider. Temporal scalability is a technique proposed to fulfill this requirement.

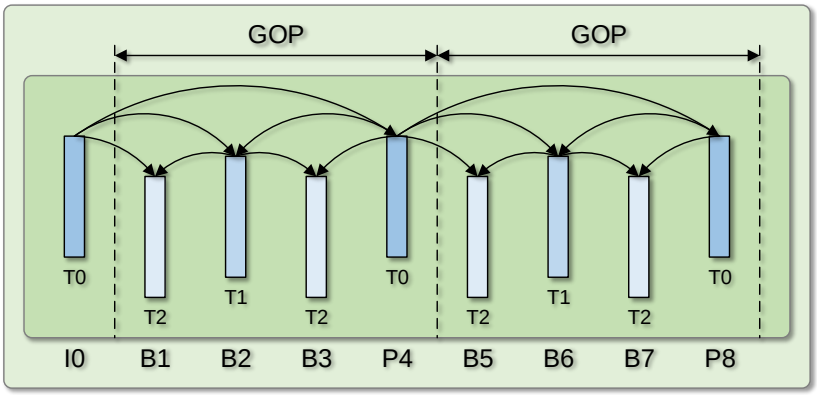


Fig. 2-3 Temporal scalability with three temporal decomposition levels.

In the early stages of SVC development, the JVT considered using short kernel wavelet filters in the temporal direction to implement temporal scalability. This technique is referred to as Motion-Compensated Temporal Filtering (MCTF). However, it has been shown that the hierarchical B-frame structure already supported in H.264/AVC is more compression-efficient than the MCTF scheme, and the hierarchical B-frame structure was adopted. Fig. 2-3 illustrates the hierarchical B-frame structure for temporal scalability with three temporal decomposition levels. In general, a temporally scalable bitstream comprises one temporal base layer and several temporal enhancement layers. Digits following the letter 'T' under each frame depict the temporal level identifier of the frame and letters with dig-

its at the bottom denote the type and display order of the frame. Letter 'I' indicates an I-frame, and 'P' and 'B' indicate a P-frame and a B-frame, respectively. In Fig. 2-3, frames 0, 4 and 8 are key frames located at temporal level 0. Here, key frames refer to temporal base layer frames which are encoded as I-frames using intra-prediction, or P-frames using inter-prediction with previous key frames as references. Frames in the temporal base layer and all the enhancement layers between two consecutive key frames make up a Group of Pictures (GOP). The first frame of the entire video sequence is coded as an Instantaneous Decoding Refresh (IDR) frame. Apart from the key frame, all other frames in the same GOP are bi-directionally predicted B-frames. A pyramid-like prediction structure is adopted, termed a 'hierarchical B-frame structure'. To support multi-temporal layer coding, frames that lie in lower layers are encoded prior to frames of higher layers, so that the higher layer ones can refer to the reconstructed frames in the lower layers. To take the first GOP of Fig. 2-3 as an example, firstly the key frame (frame 4) is encoded followed by frame 2 at temporal level 1, and finally frames 1 and 3 at temporal level 2. The hierarchical encoding scheme enables temporal scalability in an intrinsic manner. Obviously, video sequences with the coarsest supported temporal resolution are represented entirely by key frames. As the number of encoded frames is increased in coding order, the temporal resolution increases as well.

Temporal scalability provides a desirable solution to video transmission in unstable network environments, as it can work with the unequal error protection scheme. That is, the base layer is transmitted with a stronger protection mechanism than the enhancement layers. When a transmission error occurs, video with graceful degradation can still be received.

Spatial Scalability

Spatial scalability is achieved by supporting a set of frames with reduced spatial resolution but which represent the same underlying content. To perform spatial scalability, SVC fol-

lows a layer-based coding architecture, which is also used in other video coding standards, such as MPEG-2, H.263, and MPEG-4. Specifically, the pictures relating to different spatial layers are obtained by downsampling the high resolution video content. Each spatial layer supports a spatial resolution. The input video for each spatial layer is coded using a set of respective layer-related encoding parameters. Unlike earlier video coding standards, a more sophisticated inter-layer prediction mechanism is employed in SVC. Some encoding results of the lower layer can be used to inform the encoding of the enhancement layer, thus eliminating the redundancy between layers. The encoder determines whether to use inter-layer prediction or intra-layer prediction (inter- and intra-prediction) in a switchable manner [47]. This mechanism improves the coding efficiency of the enhancement layers.

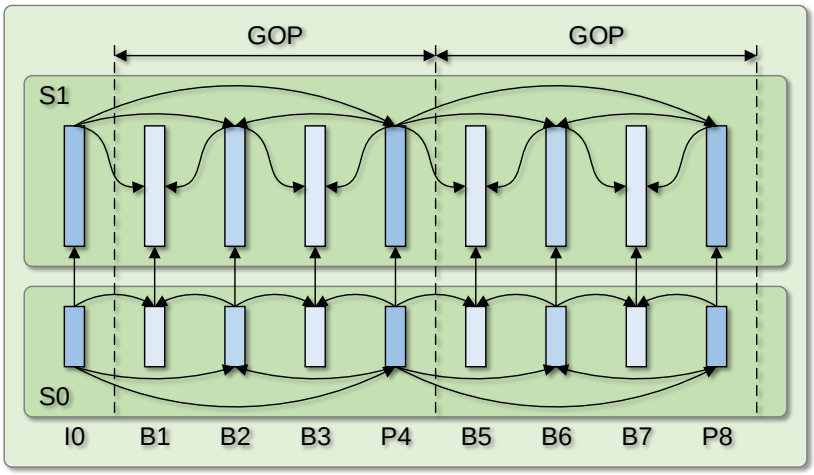


Fig. 2-4 Spatial scalability with two spatial layers.

The spatial layer with the lowest resolution, namely the spatial base layer, is labelled with dependency identifier 0. The dependency identifier increases by one from each spatial layer to the next. Fig. 2-4 shows an example of spatial scalability with two spatial layers. To implement spatially scalable video coding more efficiently than simulcast, the spatial layer with the lowest resolution is encoded first, so that the bitstream of the spatial base layer can be obtained. Then spatial layers with higher resolution are encoded to generate the enhancement bitstream, where the inter-layer information is exploited. The bitstream

of the spatial base layer can be decoded independently to acquire the lower spatial resolution video sequence. Higher resolution video can be obtained by decoding the combination of the spatial base layer bitstream and the enhancement bitstream. In particular, the inter-layer information is exploited by the inter-layer prediction mechanisms comprising inter-layer motion and residual prediction as well as inter-layer intra-prediction. It has been shown that inter-layer prediction is effective in removing the redundancy between layers, by reusing the motion and residual information of the lower layer. However, a penalty for the improvement in coding efficiency is that the computational complexity of the encoder is increased significantly. In order to restrict the increase in complexity and the decoder memory requirement, SVC imposed the constraint that inter-layer prediction is performed only within the same access unit, formed from representations of different spatial resolution at a given time instant, and the frame coding order of all the spatial layers must be the same.

Spatial scalability is very useful for applications such as video broadcasting, since the video may be viewed on different display equipments ranging from smartphones with small screen resolutions to HDTV.

Quality Scalability

Quality scalability is also referred to as SNR scalability or fidelity scalability. The aim is to support video with identical frame rate and spatial resolution but with variable quality. Three types of quality scalability: CGS [43], MGS [44] and FGS [48] are supported.

In CGS, the approach of inter-layer prediction is inherited from spatial scalable coding, but without performing upsampling operations and inter-layer deblocking for intra-coded lower layer macroblocks. Moreover, inter-layer intra-prediction and inter-layer residual prediction are carried out in the transform domain directly. CGS in SVC is achieved by encoding the quality base layer with the coarsest quantisation step size to produce the base layer bitstream, and then refining the residual signals in the enhancement layers with

smaller quantisation step sizes compared to those used in the previous CGS layers. CGS utilises a similar mechanism to spatial scalable coding, therefore it can only support a very limited number of quality levels identical to the number of spatial layers.

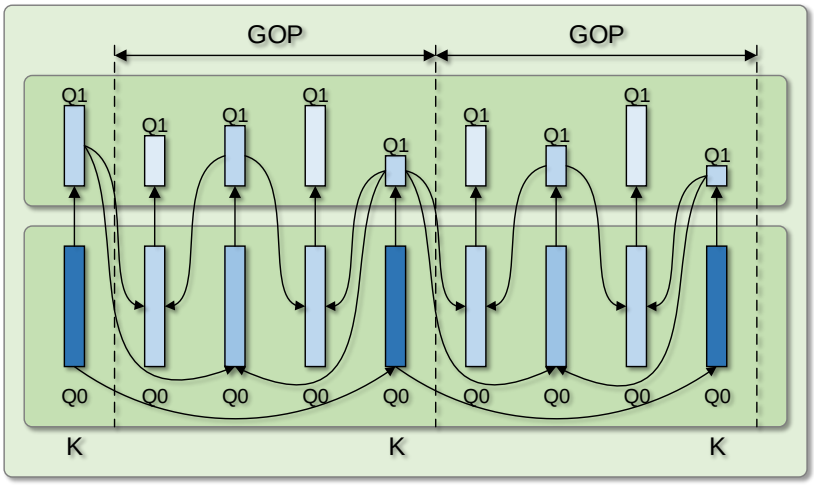


Fig. 2-5 Quality scalable structure in MGS.

The remarkable feature of MGS is that the bit rate can be switched within a certain range. MGS also addresses the problem of how to balance the enhancement layer coding efficiency with drift. MGS introduces the approach of so called key pictures, and stipulates that all of the frames in the coarsest temporal layer are coded as key pictures. As shown in Fig. 2-5, key pictures, which are marked with letter 'K', can be efficiently combined with the hierarchical prediction structure. The reconstruction of key pictures can be achieved using only the base quality layer information, and therefore the coarsest temporal layer frames can be reconstructed without any drift. Consequently, the packets of enhancement layer data can be discarded from a quality scalable bitstream. The resultant drift is thus limited to that between the current key frame and the neighbouring key frame. On the other hand, the temporal enhancement frames use the highest available quality enhancement layer as a reference for motion-compensated prediction, which enables high coding efficiency. To optimise the tradeoff between coding efficiency and drift control, GOP size and/or the number of hierarchical stages can be adjusted relative to the specific

application. In this way, not only is the drift limited to within one single GOP, but also the overall coding efficiency is taken into consideration.

FGS coding has the advantage of enabling a larger degree of flexibility and allows arbitrary bit rate truncation. A base layer is produced using CGS coding and a quality enhancement layer is added as well. For the decoders which do not support FGS, quality base layer frames can be decoded to provide a coarse video sequence. The embedded enhancement layer data is encoded using a bit plane DCT encoder (a detailed explanation of which is presented in subsection 3.1.1). Specifically, the difference between the reconstructed reference and the original picture is processed by the enhancement encoder. After the DCT transform, FGS performs bit plane coding of the transform coefficients in the enhancement layer to produce an embedded bitstream. FGS allows the bit rate to be switched over a wide range, allowing the system to adapt to diverse network conditions.

In summary, CGS has a simple structure and supports only a limited number of quality levels; FGS enables arbitrary bitstream truncation, but with high complexity; MGS is a tradeoff between CGS and FGS and aims to balance coding efficiency and drift.

Quality scalability is highly desirable for storage in video surveillance applications. The quality enhancement parts of the bitstream can be deleted after a certain period of time leaving the low quality part stored for archival purposes [49].

Hybrid Scalability

Temporal, spatial, and quality scalability can be combined to form a hybrid scalability bitstream which supports a variety of temporal and spatial resolutions, and picture quality. This is desirable for video applications involving heterogeneous clients. The clients could request the same video clip but with different frame rates, display resolutions, or picture quality. The video content with the highest requested resolution, frame rate, and picture quality needs only be coded once. The scalable bitstream can then be fully or partially decoded depending on the available network bandwidth or the application require-

ments [50].

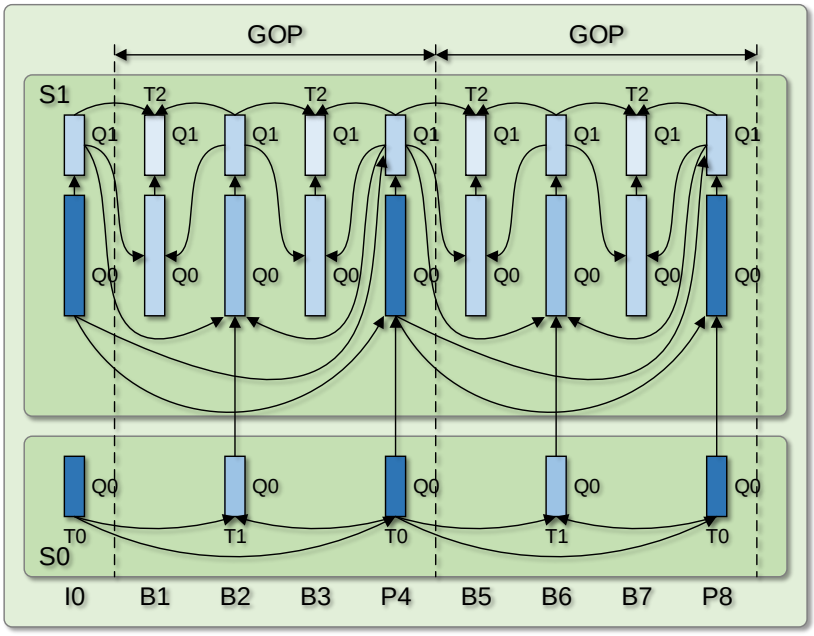


Fig. 2-6 Hybrid scalability with two spatial layers.

An example of combining temporal, spatial and quality scalability is illustrated in Fig. 2-6. In this diagram, the bitstream is composed of two spatial layers S0 and S1, (spatial scalability). The spatial enhancement layer is coded in two quality layers, Q0 and Q1 (quality scalability), generating enhancement bitstreams 1 and 2. Temporal scalability is achieved in the spatial base layer. The base layer bitstream contains a lower frame rate and a smaller picture size, which is a representation at the lowest quality.

2.2 Coding Methods

As an extension of the H.264/AVC standard, SVC reuses all the advanced features of its predecessor. In addition, SVC introduces several new tools to support the desired scalability and to improve the coding efficiency. Inter-layer prediction is one of the new coding tools, which is introduced to exploit lower layer information when encoding the enhancement

layer.

2.2.1 Prediction

The conventional intra-layer prediction tools and inter-layer prediction tools will be described individually in the following subsections, placing particular emphasis on the underlying concepts and the details of the associated techniques.

Directional Spatial Intra-prediction

In modern video codecs, a macroblock is typically predicted from the previously coded data. The prediction is obtained either from the coding results of the current frame or from those of the previously coded frames. The former is usually referred to as intra-prediction and the latter is referred to as inter-prediction.

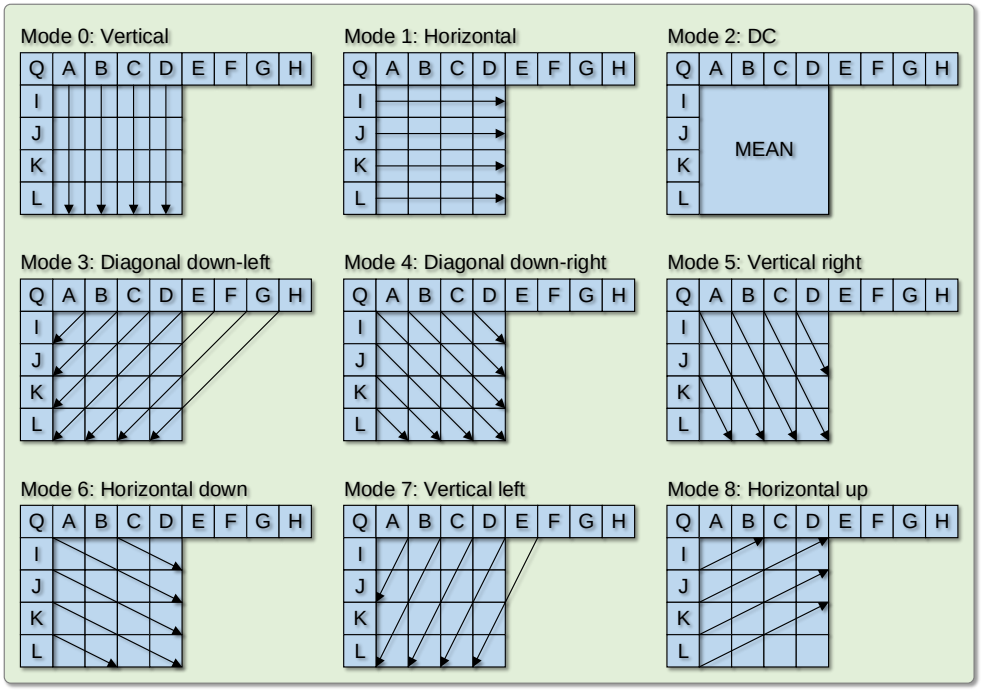


Fig. 2-7 Nine prediction patterns of intra 4×4 type (taken from [30]).

An effective and flexible prediction model is provided in SVC. Novel intra-frame direc-

tional spatial prediction is adopted, in which the previously coded area from the appropriate spatial direction in the same frame is employed to produce an approximation of the current macroblock, so that the residual prediction error is minimised and the coding efficiency is improved.

For the luminance samples, SVC offers three types of intra-prediction mode known as: 16×16, 8×8, and 4×4 [32,51]. In the intra 4×4 mode, there is one DC mode and eight candidate directional modes, as shown in Fig. 2-7. The border pixels A-H in the upper horizontal row and Q-L in the left vertical row of the current 4×4 block come from previously reconstructed blocks and are used as reference blocks. For instance, the vertical mode (mode 0) extrapolates the upper samples in the vertical direction, and the horizontal mode (mode 1) extrapolates a 4×4 block horizontally. The other modes operate in a similar way depending on the respective direction. In contrast to the other prediction modes, DC prediction extrapolates all pixels with the average value of the upper and left-hand samples.

For the intra 16×16 mode, there are four prediction modes available: horizontal, vertical, plane, and DC prediction. Generally, the 4×4 intra mode is often used for regions of a picture containing significant detail, and the 16×16 intra mode is normally appropriate for relatively homogenous areas.

Motion-Compensated Inter-prediction

Similar to the earlier video coding standards, inter-frame prediction and compensation are employed to exploit the temporal redundancies that exist between successive frames. Unlike intra-frame coding, in inter-frame coding the reference blocks obtained from previously reconstructed frames are used to predict the current macroblock. SVC supports a more powerful and flexible inter-frame motion-compensated prediction mechanism than those defined in earlier standards. The distinctive features that enable the significant improvement in coding efficiency include: various block size motion estimation, multi-frame motion compensation, and quarter pixel motion estimation precision [51].

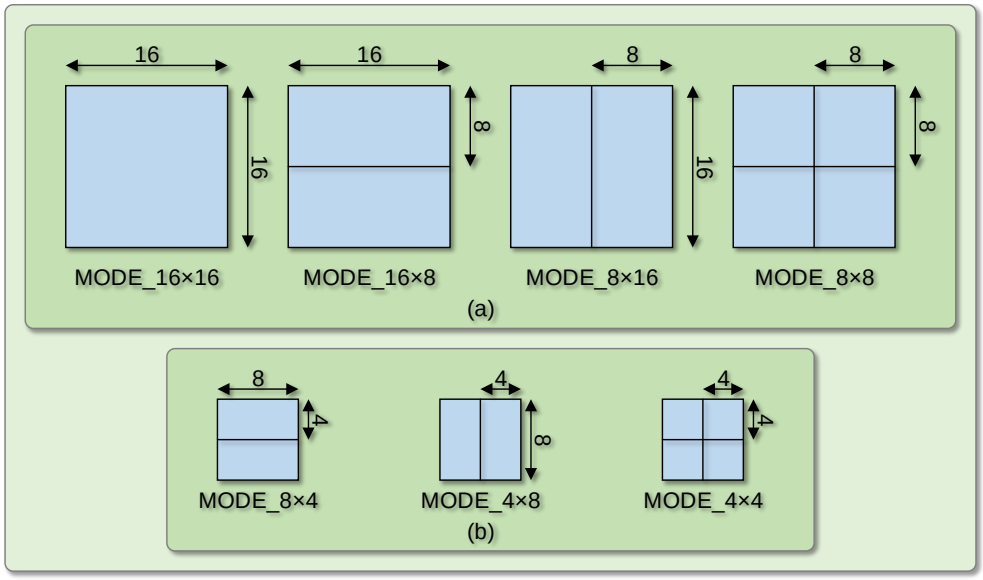


Fig. 2-8 Block modes for inter-frame prediction.

In SVC, seven different block sizes are allowed for motion estimation. A luminance macroblock can be retained as one 16×16 partition, or split into two 16×8 partitions, 8×16 partitions, and four 8×8 partitions, as illustrated in Fig. 2-8(a). If the 8×8 partition is chosen, each of the four 8×8 partitions may be split further into two 8×4 partitions, 4×8 partitions, and four 4×4 partitions, as illustrated in Fig. 2-8(b). Generally, a large partition size is appropriate for homogeneous areas and a small partition size is beneficial for detailed areas.

Inter-layer Prediction

SVC introduced three new prediction methods for spatial scalability and quality scalability to reduce inter-layer redundancy [47].

1. **Inter-layer motion prediction**

   In order to exploit motion correlation between layers and remove the redundancy, a new macroblock mode BL_SKIP, which is referred to as the base layer skip mode, is introduced for the enhancement layers. For the spatial and quality enhancement layer,

when the lower layer macroblock is inter-coded, its motion information including the partition data, reference indices and MVs can be reused for the enhancement layer motion data, as shown in Fig. 2-9. When encoding the spatial enhancement layers, the macroblock partitioning is derived from the upsampled partitioning of the co-located 8×8 block in the lower layer; the reference indices of the enhancement layer are the same as those for the base layer; and the associated MVs are scaled in conformity with the resolution ratios of the enhancement layer and the base layer. These scaled MVs are usually used as Base Layer Motion Vector Predictors (BLMVPs) for conventional motion-compensated macroblock coding types. This type of prediction mode works well in scenes containing fast movement.

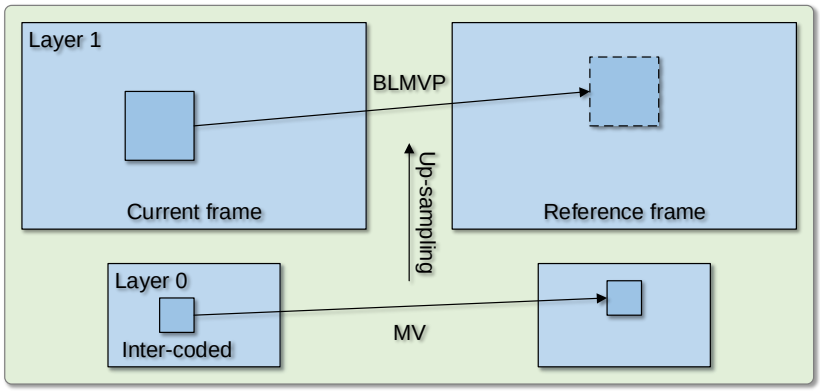


Fig. 2-9 Inter-layer motion prediction in SVC.

2. Inter-layer residual prediction

In order to further reduce the number of bits required for representing the residual of the inter-coded macroblock in the enhancement layers, inter-layer residual prediction is used. When the reference layer macroblock is inter-coded, the residual information of the reference layer can be used to predict the enhancement layer residual signal, as illustrated in Fig. 2-10. This prediction uses the upsampled residual signal of the lower resolution layer. For instance, the upsampled residual signal of the co-located 8×8 sub-macroblock in the base layer is used as the inter-layer predictor of the residual signal

in the enhancement layer macroblock. Therefore, only the corresponding difference (refinement) signal of the residual in the enhancement layer needs to be coded. This type of prediction mode is suited to video sequences comprising rich detail.

The new macroblock mode BL_SKIP represents the case when the enhancement layer macroblock is predicted by "inter-layer motion prediction" or "inter-layer residual prediction".

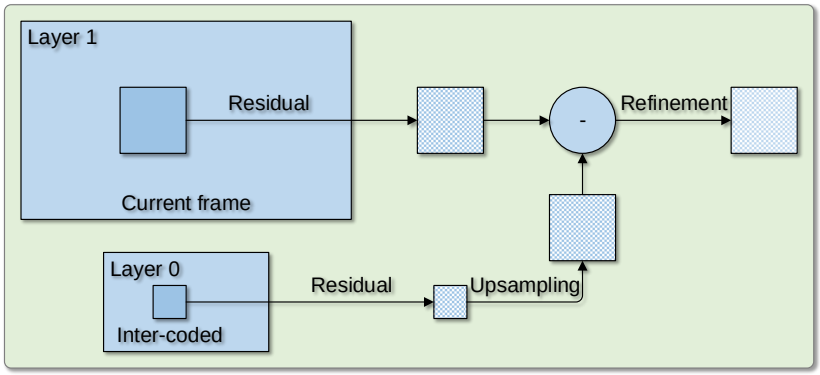


Fig. 2-10 Inter-layer residual prediction in SVC.

3. Inter-layer intra-prediction

In order to further exploit the texture redundancy between layers, an additional macroblock coding mode referred to as inter-layer intra-prediction mode (Intra_BL) is defined. When a submacroblock in the lower layer is intra-coded, the prediction signal for the spatial enhancement layer macroblock can be obtained by upsampling the corresponding reconstructed intra-signal of the lower layer, as shown in Fig. 2-11. This type of prediction mode performs well when encoding I-frames in the enhancement layers. To perform this prediction, the lower layer needs to be completely reconstructed which involves the computationally complex operations of motion-compensated prediction and deblocking. To prevent complete decoding of all lower layers, inter-layer intra-prediction is restricted to macroblocks whose co-located reference layer macroblocks are intra-coded. In addition, only constrained intra-prediction is allowed in the reference layer to ensure that the referred intra-coded macroblock in the reference layer can

only be predicted from another intra-coded macroblock. With these restrictions, each supported layer can be decoded with a single motion compensation loop.

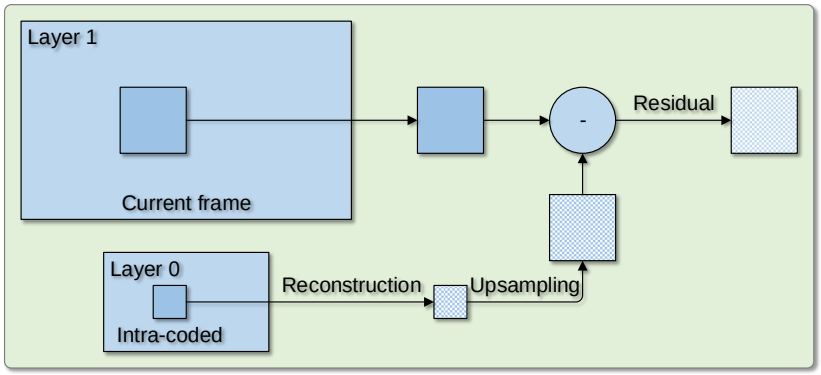


Fig. 2-11 Inter-layer intra-prediction in SVC.

In addition to the aforementioned prediction modes, special modes including I_PCM, so-called DIRECT modes in B-frames and MODE_SKIP modes in P- and B-frames are also supported in SVC. The definition and primary features of each of these modes are summarised in Table 2-1.

Table 2-1
Other macroblock prediction modes in SVC

Mode	Definition and features
I_PCM	Another intra-prediction mode. Prediction and transformation operations are bypassed. This mode is primarily intended to prevent data expansion when encoding at very high quality [30].
MODE_SKIP	Another inter-prediction mode. Coding data such as motion vectors and reference indices are derived from previously transmitted information [34].
DIRECT	Another inter-prediction mode. Its only difference from MODE_SKIP is that the coding data can be derived from the two distinct lists of reference pictures.

2.2.2 DCT Transform and Quantisation

As with previous video coding standards such as MPEG-1/2 and H.261/263 [30], SVC uses the DCT as its transform tool. After inter-frame prediction, intra-frame prediction and inter-layer prediction, the residual is transformed by an integer DCT. Unlike the early video coding standards, SVC is based on a 4×4 block size rather than an 8×8 block size. The integer DCT used in SVC has several superior features compared with earlier DCT implementations [52]:

1. Integer transform (all operations can be completed in integer arithmetic, without any change in accuracy between the forward and inverse transforms).

2. The core part of the transform can be implemented using only addition and shift operations.

3. Scaling multiplication is removed from the transform and incorporated in the quantiser, thus reducing the number of multiplications in the transform procedure.

The forward integer DCT transform is given by

$$\begin{aligned}\mathbf{Y} &= \mathbf{C}_f\mathbf{X}\mathbf{C}_f^{\mathbf{t}} \otimes \mathbf{E}_f \\ &= \left(\begin{bmatrix} 1 & 1 & 1 & 1 \\ 2 & 1 & -1 & -2 \\ 1 & -1 & -1 & 1 \\ 1 & -2 & 2 & -1 \end{bmatrix} \begin{bmatrix} \mathbf{X} \end{bmatrix} \begin{bmatrix} 1 & 2 & 1 & 1 \\ 1 & 1 & -1 & -2 \\ 1 & -1 & -1 & 2 \\ 1 & -2 & 1 & -1 \end{bmatrix} \right) \otimes \begin{bmatrix} a^2 & ab/2 & a^2 & ab/2 \\ ab/2 & b^2 4 & ab/2 & b^2/4 \\ a^2 & ab/2 & a^2 & ab/2 \\ ab/2 & b^2/4 & ab/2 & b^2/4 \end{bmatrix}\end{aligned} \tag{2.1}$$

where **X** and **Y** refer to a 4×4 residual coefficient matrix and the DCT coefficient matrix; $\mathbf{C}_f\mathbf{X}\mathbf{C}_f^{\mathbf{t}}$ denotes the transform kernel; $\mathbf{C}_f$ refers to the forward transform matrix and $\mathbf{C}_f^{\mathbf{t}}$ is the transpose matrix of $\mathbf{C}_f$; $\mathbf{E}_f$ indicates the scaling factor matrix, and the operator $\otimes$ denotes element-by-element multiplication rather than normal matrix multiplication. The coefficients in $\mathbf{E}_f$ are $a = \frac{1}{2}$, $b = \sqrt{\frac{2}{5}}$, $d = \frac{1}{2}$.

Equation (2.1) is an integer approximation of the DCT, but achieves almost the same compression efficiency. Furthermore, the integer DCT possesses many advantages. It can be observed from the transform kernel $\mathbf{C}_f\mathbf{X}\mathbf{C}_f^{\mathbf{t}}$ that each of the elements in $\mathbf{C}_f$ and $\mathbf{C}_f^{\mathbf{t}}$

equals ±1 or ±2, which means that the multiplications can be implemented using only addition and left shift operations. As the scaling factor matrix $\mathbf{E}_f$ scales only the result of the kernel transform, it can be incorporated in the quantiser. Multiplication operations are therefore completely avoided in the transform process. Consequently, the integer DCT achieves a very significant reduction in computational complexity compared to the conventional DCT.

To perform lossy coding, a uniform scalar quantiser is employed after the transform. A total of 52 quantisation step sizes are designed to achieve an accurate and flexible tradeoff between bit rate and picture quality. Each quantisation step size is indexed by a unique Quantisation Parameter (Qp). Each Qp and its corresponding quantisation step size are shown in Table 2-2. It can be seen that the quantisation step size exactly doubles when the Qp is increased by six.

Table 2-2
Qp and its corresponding quantisation step size Q_{step}

Qp	0	1	2	3	4	5
Q_{step}	0.625	0.6875	0.8125	0.875	1	1.125
Qp	6	7	8	9	10	11
Q_{step}	1.25	1.375	1.625	1.75	2	2.25
Qp	...	...	...	...	...	...
Q_{step}	...	...	...	...	...	...
Qp	48	49	50	51		
Q_{step}	160	176	208	224		

In order that division operations are not required in the quantiser, scaling and quantisation are performed as follows,

$$\begin{aligned} |Z_{ij}| &= (|W_{ij}| \cdot MF + f) \gg qbits \\ \mathbf{sign}(Z_{ij}) &= \mathbf{sign}(W_{ij}) \end{aligned} \tag{2.2}$$

where W_{ij} and Z_{ij} denote the DCT coefficients and quantised coefficients respectively; f defines a dead zone control parameter which is either $2^{qbits}/3$ for the intra-coded blocks

or $2^{qbits}/6$ for the inter-coded blocks; the $\gg$ symbol indicates a binary right-shift; MF and $qbits$ are defined as

$$\frac{MF}{2^{qbits}} = \frac{PF}{Q_{\text{step}}}$$
$$qbits = 15 + \lfloor \text{Qp}/6 \rfloor \tag{2.3}$$

where the $\lfloor \cdot \rfloor$ symbol is the floor operator; PF is a^2, $ab/2$, or $b^2/4$ depending on the coefficient's position (i, j) in the 4×4 DCT coefficient matrix. As a result, MF is predefined as a periodic table, as shown in Table 2-3.

Table 2-3
Multiplication factor MF for scaling function

Qp%6	Position (i, j) of the DCT coefficient		
	(0,0),(0,2),(2,0),(2,2)	(1,1),(1,3),(3,1),(3,3)	Others
0	13107	5243	8066
1	11916	4660	7490
2	10082	4194	6554
3	9362	3647	5825
4	8192	3355	5243
5	7282	2893	4559

2.2.3 Entropy Coding

Two types of entropy coding are specified in SVC, Context-Adaptive Variable Length Coding (CAVLC) and Context-Adaptive Binary Arithmetic Coding (CABAC). The former adaptively selects a set of Variable Length Coding (VLC) tables based on the context of past symbols and the latter adjusts the probability model of the arithmetic coder according to the context.

CAVLC

CAVLC [53] is designed to encode the blocks of zig-zag ordered quantised coefficients. Generally, quantised coefficients after zig-zag scanning are sparse, containing a large num-

ber of zero coefficients. Furthermore, most non-zero coefficients are concentrated in the low frequency bands. Consequently the adjacent blocks are highly correlated. With the above properties, quantised coefficients can be coded efficiently, benefitting from the statistics of the previously coded neighbouring blocks as well as the statistics of the previously coded coefficients in the current block. The block diagram of a CAVLC encoder is shown in Fig. 2-12. As illustrated, CAVLC encoding of a block of quantised coefficients proceeds in the following order: 1) the number of coefficients and trailing ones (**coeff_token**), 2) the sign of each trailing one (**trailing_ones_sign_flag**), 3) the levels of the remaining non-zero coefficients (**level**), 4) the total number of zeros before the last coefficient (**total_zeros**), 5) each run of zeros (**run_before**).

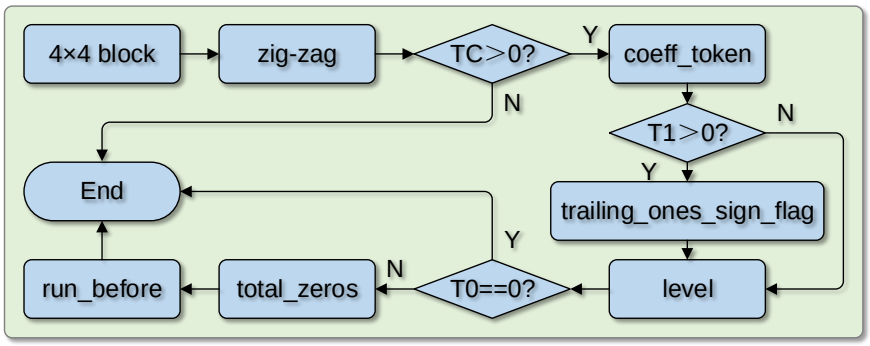


Fig. 2-12 Block diagram of a CAVLC encoder.

In Fig. 2-12, TC, T1, and T0 denote the number of non-zero coefficients, the number of trailing ones, and the total number of zeros before the last non-zero coefficient, respectively.

CABAC

CABAC [54] can be used to further improve the efficiency of entropy coding. The adaptive binary arithmetic coding technique and the context modelling are ingeniously integrated to achieve a high degree of adaptation and a significant reduction in redundancy. Adaptive binary arithmetic coding encodes the binary symbols to meet the target bit rate. Con-

text modelling is performed for the probability model estimation. The block diagram of a CABAC encoder is shown in Fig. 2-13.

The CABAC encoding of a symbol consists of the following steps:

1. **Binarisation**: The non-binary valued symbol is converted into a binary code prior to processing the data symbol in the arithmetic coder.
2. **Context modelling**: Select the appropriate probability model depending on the statistical characteristics of recently coded symbols.
3. **Binary arithmetic coding**: The arithmetic coder compresses the binary symbols based on the selected context model.
4. **Probability update**: Update the selected context model according to the immediately coded data.

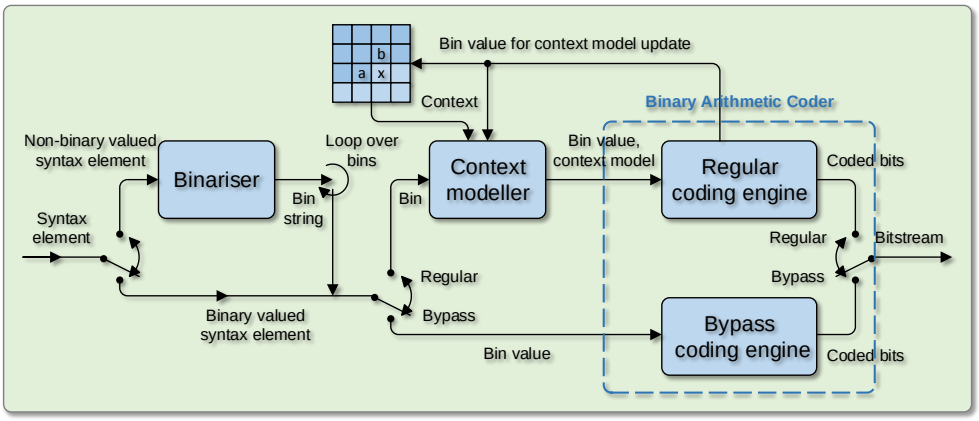


Fig. 2-13 Block diagram of a CABAC encoder.

CABAC provides a better coding gain in terms of compression performance, namely a reduction in bit rate of between 10%-15% compared with CAVLC. However this is at the expense of greater computational complexity.

2.3 Coding Mode Decisions

The mode decision process in the enhancement layer of SVC requires a large amount of computation, and dominates the encoding time. This is due to the utilisation of many time-consuming encoding tools, for example, Rate Distortion Optimisation (RDO) and inter-layer prediction. Evaluation results show that the mode decision process in the enhancement layers accounts for 90% of the total computational requirement [55], and the encoding time of the enhancement layers is 10 times that of the base layer. The next two subsections review the RDO technique and analyse the computational complexity of inter-layer prediction.

2.3.1 Rate Distortion Optimisation

Rate distortion theory provides the theoretical foundation for source coding and forms a major branch of information theory. The objective of rate distortion theory is to formulate the optimal trade-off between [bit] rate (R) and distortion (D). The source model determines the rate distortion function R(D), and different assumptions about the source model result in diverse rate distortion functions [56,57]. R(D) can be represented by a continuous, monotonically decreasing convex function of D, as shown in Fig. 2-14.

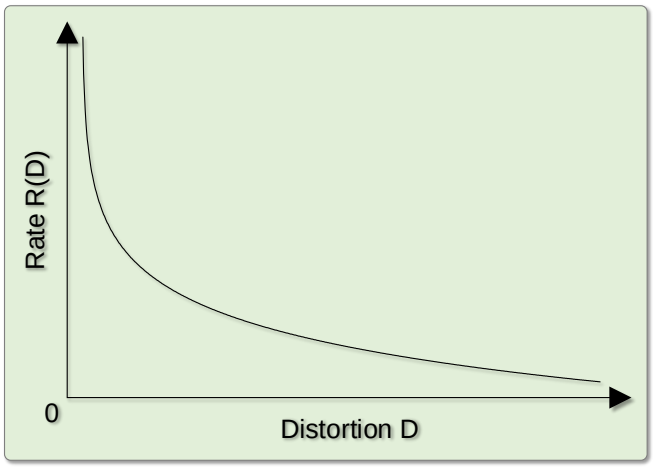


Fig. 2-14 Rate distortion function.

RDO is a coding tool used for selecting the best coding mode, and it attempts to minimise the bit rate subject to a given picture distortion $D_{\max}$,

$$\min\ R,\ \text{s.t.}\ D \leq D_{\max} \tag{2.4}$$

or to maximise picture fidelity under the transmission bit rate constraint $R_{\max}$.

$$\min\ D,\ \text{s.t.}\ R \leq R_{\max} \tag{2.5}$$

The constrained optimisation problems in equations (2.4) and (2.5) are difficult to solve in a straightforward manner. The Lagrangian multiplier method is therefore used to transfer the constraint into a Lagrangian cost function J to be minimised. The Lagrangian cost function J, with Lagrange multiplier λ is defined as follows:

$$\min\{J\} = \min\{D + \lambda \times R\} \tag{2.6}$$

For a given value of λ, the task is to find a set of appropriate coding parameters that produces the minimal value of J.

In SVC, both the number of bits generated by a macroblock partition mode and the degree of distortion are taken into account when choosing the optimal coding mode for each macroblock. The mode with the minimum RD cost is selected as the best mode for the current macroblock. In similarity with equation (2.6), the RD function for a macroblock ω_k is given by

$$J_{\text{MODE}}(\text{Qp}) = D_{\text{MODE}}(\omega_k, \tilde{\omega}_k|\text{Qp}) + \lambda_{\text{MODE}}(\text{Qp}) \cdot R_{\text{MODE}}(\text{Qp}) \tag{2.7}$$

where ω_k is an original macroblock at time k, and $\tilde{\omega}_k$ is the corresponding reconstructed block. R represents the number of bits, D denotes a distortion measure, and λ is empirically defined as [58]

$$\lambda_{\text{MODE}}(\text{Qp}) = 0.85 \times 2^{(\text{Qp}-12)/3} \tag{2.8}$$

In SVC, more candidate partition modes than any previous video coding standard are involved in the rate distortion optimised mode decision process. As discussed in the last section, SVC has inherited all the encoding tools of H.264/AVC. These are supplemented by additional tools to support scalability. Therefore when encoding the enhancement layers in SVC, all the modes concerned with inter-frame prediction, intra-frame prediction and inter-layer prediction are evaluated. Consequently, RDO comprises a major part of the encoder complexity.

2.3.2 Computational Complexity Analysis

The superior coding performance of SVC is achieved at the cost of significantly increased computational complexity. A set of time-consuming encoding tools are incorporated in both H.264/AVC and SVC, for example, bi-directional motion prediction, quarter pixel precision motion estimation, and motion compensation using multiple reference blocks. In addition, the SVC inter-layer prediction tools also incur excessive computational cost. In particular, inter-layer residual prediction doubles the computational complexity of the mode decision process [59]. Inter-layer motion prediction also results in a significant increase in the computational requirement.

In the Joint Scalable Video Model (JSVM), which is the reference software of SVC, inter-layer prediction can be performed in two different ways: an adaptive manner or a forced manner [60]. If the adaptive manner is used, the additional inter-layer prediction modes are required to compete with the regular H.264/AVC modes and the mode with the minimum RD cost is selected as the best mode for the current macroblock. Alternatively, if the forced manner is enabled, all of the macroblocks in the enhancement layer are forced to be encoded as BL_Skip or Intra_BL mode, that is, all the encoding results from the base layer are reused directly.

Fig. 2-15 shows the average encoding time of different coding options for each video sequence. The statistical data was collected from processing the sequences: 'Foreman',

'Crew', 'Park' and 'Flower'. The first 150 frames of each sequence were processed and the same Qp values, i.e. Qp=24, 28, 32, 36, 40 were used for both the base layer and the enhancement layer.

The histograms in Fig. 2-15 show that when the base layer and the enhancement layer are encoded separately, namely simulcast is employed, about 4-5 times more encoding time than that of single-layer coding is required.

When forced inter-layer prediction is enabled, 1.1 times more encoding time than single-layer coding, or 1/4 the encoding time of simulcast is required. This is because most coding parameters, including partition mode, MVs, etc., for the enhancement layer are deduced from the coding results of the base layer. Thus, the most time-consuming operations, such as motion estimation and RDO, are skipped.

In addition, when adaptive inter-layer prediction is enabled in encoding the enhancement layer, a SVC encoder spends 7-10 times more encoding time than that of a single-layer encoder. Thus, a SVC encoder requires 1.5-2 times more computation than simulcast. This significant increase in encoding time is caused by the fact that motion estimation and RDO are performed twice for each macroblock in the enhancement layer (with and without inter-layer residual prediction). Furthermore, due to inter-layer motion prediction, motion estimation with an additional MVP upscaled from the best MV of the corresponding block in the base layer is required. This further increases the computational complexity.

Inter-layer prediction contains a set of very useful encoding tools which demonstrate an average 3dB coding gain over simulcast [61]. Therefore, a SVC encoder benefits substantially from the use of inter-layer prediction. However, if inter-layer prediction is to be used extensively, a means of reducing the computational complexity is required.

This thesis develops two fast mode decision algorithms for SVC, in chapters 4 and 5. The ultimate goal is to reduce the complexity requirement of the encoders without sacrificing the RD performance.

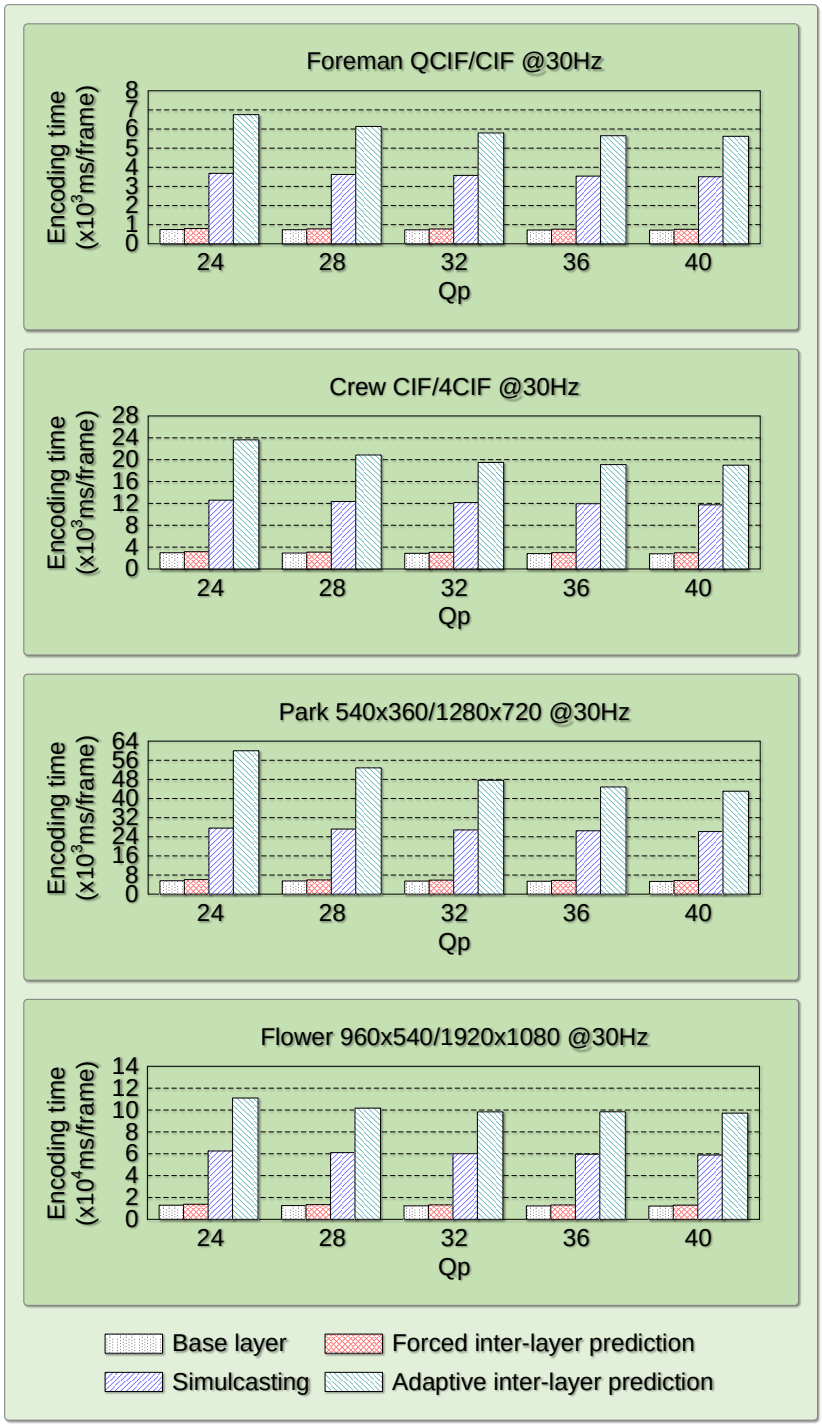


Fig. 2-15 Encoding time comparison of different encoding options.

2.4 Rate Control

The purpose of rate control is to maximise the coding performance (i.e. achieve high compression ratio and/or good image quality) under the constraints of a total bit rate budget and buffer size, by the appropriate allocation of bits.

In practical video transmission systems, compressed video streams need to be transmitted over networks and often there are situations in which the bandwidth is insufficient or the network is unstable. Sometimes due to network congestion or insufficient network bandwidth, the bitstream cannot be transmitted completely, resulting in frame skipping. On the other hand, blindly reducing the bit rate of a video stream will result in quality degradation and waste of available bandwidth resources.

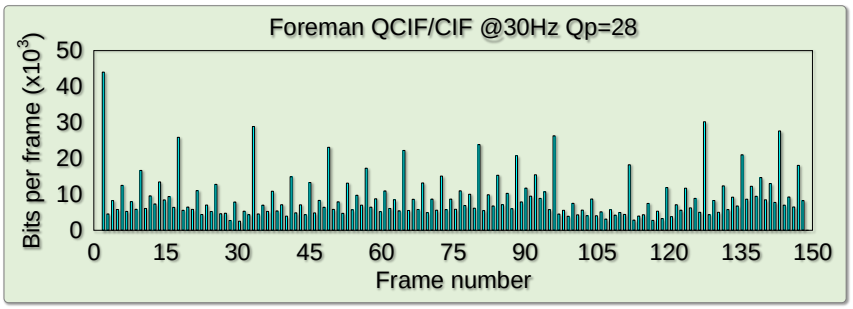


Fig. 2-16 Bit rate fluctuation.

When constant encoding parameters are used during the encoding process, the number of encoded bits fluctuates significantly from frame to frame. Generally, regions comprising fast movement or complex detail need more bits to code and those containing slow motion or little detail require fewer bits. Fig. 2-16 shows the number of encoded bits for each frame of the 'Foreman' sequence with a constant Qp of 28. The first frame is coded as an I-frame, which requires the most bits, and successive frames are coded as P- and B-frames. The number of bits for each frame varies between 2776 and 48416, namely the bit rate varies between 81.33 kbits/s and 1418.44 kbits/s, for a frame rate of 30 fps. The heavy fluctuation in bit rate could cause problems for practical video transmission and storage

systems. For instance, a constant bit rate channel cannot be used for a variable rate bitstream. To guarantee successful transmission and to make the best use of the available network resources, an effective rate control mechanism is essential. With a proper rate control scheme, frame skipping and the wastage of channel resources can be avoided.

In terms of bit rate, video coding approaches can be categorised into two classes: Constant Bit Rate (CBR) encoding and Variable Bit Rate (VBR) encoding. CBR encoding enables the encoder to produce the output bitstream at a constant rate, a feasible solution for streaming multimedia content via limited capacity channels. However, the better and more flexible choice is to allocate more bits for complex pictures to maintain a good picture quality, while avoiding wasting bits on simple pictures. That is the motivation for VBR encoding. In general, rate control consists of three important parts: bit allocation, RD control, and update of the control models [62]. Firstly, a target number of bits for each coding unit, such as a GOP, frame, or macroblock, is allocated depending on the target bit rate or the target picture quality. Secondly, based on the complexity of the current coding unit, a Qp value is calculated using the RD model, and then the Qp is adjusted slightly according to the fullness of the buffer. The derived Qp is used in RDO to produce the target bit rate. Lastly, after encoding the current coding unit, the coefficients of the RD model are updated according to the actual encoded bits and the complexity of the current video unit, allowing the target bits to be allocated for the next coding unit.

Several rate control algorithms have been proposed for video coding standards, such as the Test Model 5 (TM5) [63] for MPEG-2, Test Model Near-term 8 (TMN8) [64] for H.263, Verification Model 8 (VM8) [65] for MPEG4, JVT-G012 for H.264/AVC, and JVT-W043 for SVC. A precise RD model is key to the development of an efficient rate control scheme for each of these video coding standards. Some empirical models, such as the linear model [63], second-order model [64,65], ρ domain linear model [66], and Logarithmic model [67], have been used in previous video coding standards. In SVC, the classical quadratic RD model is employed to describe the relationship between the target number of bits and the

quantisation step size, as follows [68],

$$R_{\text{txt}} = \frac{X_1 \times MAD_{\text{pred}}}{Q_{\text{step}}^2} + \frac{X_2 \times MAD_{\text{pred}}}{Q_{\text{step}}} \tag{2.9}$$

where R_{txt} is the target number of bits assigned to code the texture information of a coding unit; MAD indicates the Mean Absolute Difference of the residual component; Q_{step} is the quantisation step size to be calculated, and X_1 and X_2 are model coefficients. X_1 and X_2 are updated using a linear regression method after the coding of each coding unit, i.e.

$$\begin{aligned} X_2 &= \frac{n\sum_{i=1}^{n} R_{\text{txt}}^i - \left(\sum_{i=1}^{n}\left(Q_{\text{step}}^i\right)^{-1}\right)\left(\sum_{i=1}^{n} Q_{\text{step}}^i \cdot R_{\text{txt}}^i\right)}{n\sum_{i=1}^{n}\left(Q_{\text{step}}^i\right)^{-2} - \left(\sum_{i=1}^{n}\left(Q_{\text{step}}^i\right)^{-1}\right)^2} \\ X_1 &= \frac{\sum_{i=1}^{n}\left(Q_{\text{step}}^i \cdot R_{\text{txt}}^i - X_2\left(Q_{\text{step}}^i\right)^{-1}\right)}{n} \end{aligned} \tag{2.10}$$

where n is the window size excluding the outliers; R_{txt}^i is the actual number of bits generated, and Q_{step}^i is the actual quantisation step size that has been used.

As most rate control algorithms were proposed for non-scalable video coders, rate control algorithms that address the properties of the enhancement layers in SVC need to be developed. Chapter 6 contributes to this field by suggesting an improvement to the JVT-W043 algorithm, which is the default rate control technique in the JSVM reference software.

2.5 Summary

This chapter initially reviewed the scalable extension of the H.264/AVC standard, placing particular emphasis on the multi-layer coding structure. A multi-layer coder generates a single bitstream which contains one base layer and several enhancement layers. The base layer carries the most essential information and the enhancement layers contain complementary information to enhance the perceptual quality. Three basic types of scalability are supported in SVC, namely, temporal, spatial, and quality scalability. In section 2.1, each

type of scalability was discussed in detail. Spatial and quality scalability are realised by a layer-based scheme and temporal scalability is achieved by a hierarchical B-frame structure. At the end of this section, the combination of the three types of scalability, so called hybrid scalability was briefly introduced.

Apart from the sophisticated prediction tools that are inherited from its predecessor H.264/AVC, SVC also introduces several new features to improve the coding efficiency of a multi-layer structure. These include three inter-layer prediction tools. The motion, residual, and texture redundancy that exists between layers is reduced by a set of inter-layer prediction tools. They are inter-layer motion prediction, inter-layer residual prediction, and inter-layer intra-prediction. The features reused from H.264/AVC include spatial intra-prediction with directional extrapolation, and temporal motion-compensated prediction with variable block sizes ranging from 4×4 to 16×16. The encoder determines whether to use inter-layer prediction or intra-layer prediction in a switchable manner, thus maintaining the best compromise between rate and distortion. In addition to the above prediction tools, the integer DCT transform and the entropy coding methods of CAVLC and CABAC were discussed in section 2.2.

Section 2.3 discussed the RD optimised mode decision process and analysed the computational complexity of the enhancement layer in SVC. It is shown that mode selection in the enhancement layers accounts for most of the total computational requirement. Fast mode decisions for the enhancement layer therefore need to be developed.

A brief introduction of rate control techniques was presented in section 2.4. Some rate control algorithms have been proposed for SVC, however they lack consideration of the properties of the enhancement layers. Consequently, an efficient and precise rate control scheme needs to be designed.

Chapter 3

Performance Evaluation of Advanced Scalable Video Coding Schemes

In order to gain a deeper understanding of scalable video coding systems, this chapter reviews and compares three representative scalable coding algorithms. The three schemes are the SVC standard [44], Motion JPEG2000 [69–72], and WSVC [73]. SVC is the most recent international scalable video coding standard, Motion JPEG2000 is part 3 of the image coding standard JPEG2000, and WSVC is a strong competitor in the scalable video coding field. All three algorithms have been designed to produce scalable video bitstreams. SVC employs a multi-layer coding structure, Motion JPEG2000 independently codes each video frame using JPEG2000, and WSVC uses Motion-Compensated Temporal Filtering (MCTF) and the Discrete Wavelet Transform (DWT).

These coding algorithms each contain three key coding modules: transform, quantisation, and entropy coding, however each coding scheme implements the coding modules in a different way. This chapter focuses primarily on the differences between each scheme and on their performance in terms of coding efficiency.

The reminder of this chapter is organised as follows. Section 3.1 briefly describes the algorithms employed in Motion JPEG2000 and WSVC coding schemes, comparison condi-

tions are specified in section 3.2, and extensive evaluation results are presented and analysed in section 3.3. Finally, section 3.4 provides a summary of the chapter.

3.1 Introduction

In order to ascertain the strengths and weaknesses of SVC and gain an insight into its application scenarios, a study comparing it with other scalable coding schemes is presented. Motion JPEG2000 and WSVC are chosen as the closest candidates, as they both include encoding tools to support different modes of scalability.

Motion JPEG2000 is the leading digital cinema standard and is specified in part 3 of the image coding standard JPEG2000. Motion JPEG2000 independently encodes each frame using either lossy or lossless JPEG2000. JPEG2000 was the first wavelet-based international still image compression standard, and was developed by the JPEG, namely ISO/IEC JTC1/SC29/WG1 and ITU-T Study Group 16 (SG16). The DWT was adopted for JPEG2000 because of the 'blocking' artefacts caused by the DCT 8×8 block transform in its predecessor JPEG. Two types of wavelet basis functions are used, the Daubechies 9/7 wavelet is employed for lossy coding and the Le Gall 5/3 wavelet for lossless coding [74]. The core encoding tools of JPEG2000 include the DWT and the entropy coding algorithm, Embedded Block Coding with Optimised Truncation (EBCOT). With these tools, JPEG2000 outperforms JPEG in terms of compression performance by nearly 30% [75]. JPEG2000 also offers several features that did not exist in its predecessor, including an intrinsic multi-resolution characteristic which is useful in multimedia applications.

The Barbell-lifting 3D wavelet coding scheme was proposed by Microsoft Research Asia (MSRA) in response to the call for technology during the development of the SVC standard. The MSRA solution, named WSVC, contains the core technologies of a spatial 2D DWT, MCTF and, for entropy coding, the arithmetic coding algorithm termed Embedded Subband Coding with Optimal Truncation (ESCOT). The term '3D wavelet' means that

the wavelet decomposition consists of a temporal decomposition performed by applying MCTF in the temporal direction and a spatial decomposition performed by applying the DWT spatially in both the horizontal and vertical directions. First, a multi-level MCTF decomposes the video frames into several temporal subbands, then a spatial decomposition is applied to each temporal subband to further decompose the frames. WSVC provides a very convenient platform for continuing research on wavelet-based video technologies.

Table 3-1
Core algorithms in each coding scheme

Coding scheme	Prediction	Transform	Quantisation	Entropy coding
SVC	Intra-frame Inter-frame Inter-layer	2D DCT	Scalar quantisation	CAVLC CABAC
Motion JPEG2000	Intra-frame	2D DWT	Scalar quantisation	EBCOT
WSVC	Intra-frame Inter-frame	2D DWT	Embedded quantisation	ESCOT

The core algorithms of the three coding schemes are summarised in Table 3-1. Three common coding modules: transform, quantisation, and entropy coding are employed in each of the schemes. However a significant difference between SVC and both Motion JPEG2000 and WSVC is that SVC is a hybrid coding scheme based on the DCT, whereas Motion JPEG2000 and WSVC adopt the wavelet transform. Furthermore, each coding scheme contains a distinctive entropy coding method to encode the quantised coefficients. Apart from SVC, which has been introduced in chapter 2, the Motion JPEG2000 and WSVC coding algorithms will be discussed in detail in the following subsections with particular focus on their distinctive features and associated encoding tools.

3.1.1 Motion JPEG2000

Motion JPEG2000 is defined in part 3 of the JPEG2000 image coding standard. As a successor of JPEG, JPEG2000 possesses some new features desirable for interactive multimedia

applications, wired and wireless environments and internet applications. A fundamental difference in JPEG2000 is the use of the DWT rather than the DCT, thus providing a number of advantages. JPEG2000 not only achieves a significant improvement in compression performance, but more importantly, it also provides an entirely new representation of the image. These improvements rely on the incorporation of a number of relatively recent techniques, such as the DWT and the entropy coder using the EBCOT algorithm. The DWT possesses an intrinsic multi-resolution property resulting in an inherent scalability function. JPEG2000 achieves a higher compression ratio than JPEG and can also handle a much larger range of image sizes. The basic framework of a JPEG2000 coder is shown in Fig. 3-1. The JPEG2000 coding procedure can be briefly described as follows [75]:

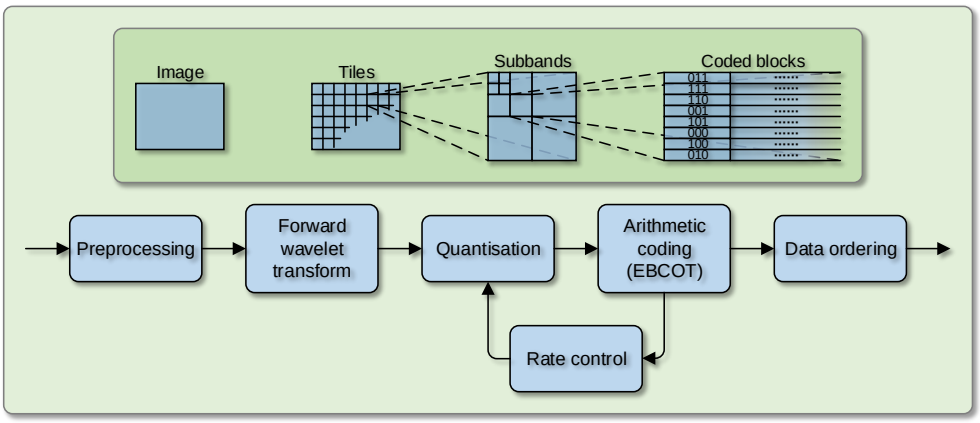


Fig. 3-1 General framework for a JPEG2000 encoder.

1. **Data pre-processing**: Segments the original image into rectangular non-overlapping blocks (tiles).
2. **DWT**: Decomposes the tile components into different wavelet decomposition levels.
3. **Quantisation of transform coefficients**: The wavelet subband coefficients are quantised and collected into 'code blocks'.
4. **Entropy coding**: The quantised coefficients are processed by the context-based binary arithmetic encoder resulting in further compression.
5. **Rate control**: The quantisation step size is adjusted independently for each subband

and each tile to meet the target number of bits.

DWT Transform and Quantisation

One of the primary differences between JPEG2000 and JPEG is the adoption of the DWT. Compared with the DCT, the DWT has a better energy compaction capability as well as a multi-resolution representation. Furthermore, the DWT produces almost no blocking artefacts as with the DCT, and the artefacts are less visible. With the purpose of reducing the correlation between pixels, the DWT decomposes the spatial image into a number of frequency subbands which represent the directional frequency components of the original image. Then the coefficients in each wavelet subband are quantised and coded independently with a different coding strategy. By comparison with the DCT, the DWT achieves a better time frequency localisation. Thus, wavelet-based image coding achieves a superior compression performance compared with its predecessors.

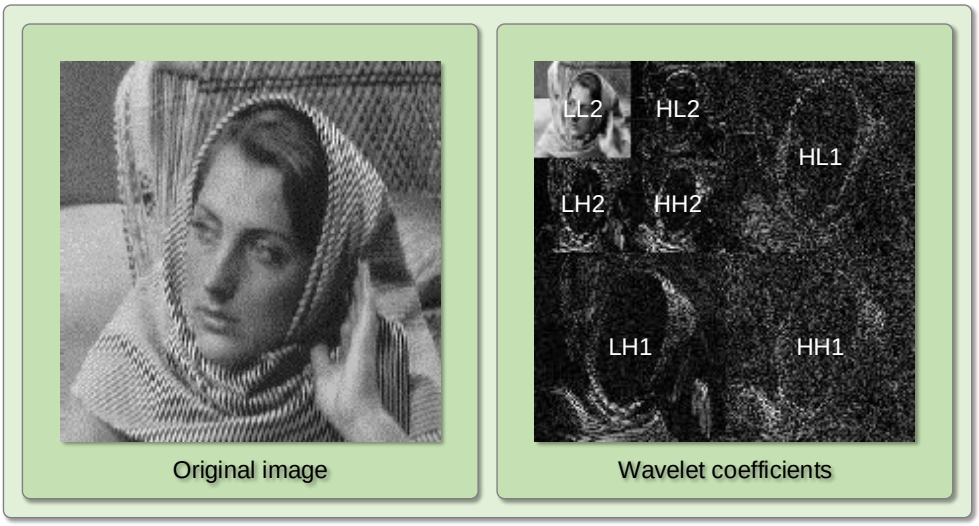


Fig. 3-2 Two level 2D wavelet decomposition of image 'Woman'.

Fig. 3-2 illustrates a two level 2D wavelet decomposition of the natural image 'Woman'. In the first level of decomposition, four distinct wavelet subbands LL1, HL1, LH1, and HH1 are yielded when the DWT low-pass and high-pass filters in both the horizontal and the

vertical directions are applied. The LL1 subband represents an approximation of the original image but at a lower resolution, HL1, LH1, and HH1 are the detail subbands in the horizontal, vertical and diagonal directions, respectively. The LL1 subband is then decomposed into four separate wavelet subbands to obtain the second level DWT coefficients. This procedure can be repeated to further decompose the image.

Table 3-2
Daubechies 9/7 LPF and HPF coefficients for analysis and synthesis filters

n	Analysis filter coefficients	
	LPF	HPF
0	0.6029490182363579	1.11508705245699
±1	0.2668641184428723	-0.5912717631142470
±2	-0.07822326652898785	-0.05754352622849957
±3	-0.01686411844287495	0.09127176311424948
±4	0.02674875741080976	
n	Synthesis filter coefficients	
	LPF	HPF
0	1.115087052456994	0.6029490182363579
±1	0.5912717631142470	-0.2668641184428723
±2	-0.05754352622849957	-0.07822326652898785
±3	0.09127176311424948	0.01686411844287495
±4		0.02674875741080976

Table 3-3
Le Gall 5/3 LPF and HPF coefficients for analysis and synthesis filters

n	Analysis filter coefficients	
	LPF	HPF
0	6/8	1
±1	2/8	-1/2
±2	-1/8	
n	Synthesis filter coefficients	
	LPF	HPF
0	1	6/8
±1	1/2	-2/8
±2		-1/8

Two kinds of coding schemes: lossy compression and lossless compression are supported in JPEG2000. Lossy compression utilises a Daubechies 9/7 filter bank for the irreversible transform, and lossless compression uses a Le Gall 5/3 filter bank for the reversible transform. The analysis and synthesis coefficients of Daubechies 9/7 and Le Gall 5/3 are listed in Tables 3-2 and 3-3, respectively [75]. LPF and HPF stand for Low-Pass FIR Filter and High-Pass FIR Filter, respectively.

The wavelet decomposition in the JPEG2000 standard can be implemented in two ways: a convolution-based approach and a lifting-based approach. In the convolution-based implementation, two sets of functions are utilised: a scaling function and a wavelet function, which correspond to the low-pass and the high-pass filters, respectively. The output subband samples are obtained by convolving the input samples with the scaling and wavelet functions, as defined by the following equations [71].

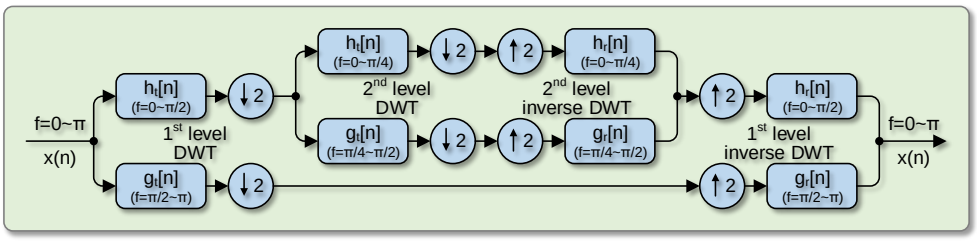


Fig. 3-3 Convolution implementation of the wavelet transform.

$$\begin{aligned} \tilde{y}_{\mathrm{l}}[k] &= \sum_{n} \tilde{x}[n]\mathrm{h}_{\mathrm{t}}[2k-n] \\ \tilde{y}_{\mathrm{h}}[k] &= \sum_{n} \tilde{x}[n]\mathrm{g}_{\mathrm{t}}[2k-n] \end{aligned} \tag{3.1}$$

and the corresponding inverse transform is

$$\tilde{x}[n] = \sum_{n} \tilde{y}_{\mathrm{l}}[n]\mathrm{h}_{\mathrm{r}}[2k-n] + \sum_{n} \tilde{y}_{\mathrm{h}}[n]\mathrm{g}_{\mathrm{r}}[2k-n] \tag{3.2}$$

where $\tilde{x}[n]$ is the symmetric boundary extension of the input samples $x[n]$, and $\tilde{y}[n]$ denotes the symmetric boundary extension of the output subband samples $y[n]$. $\mathrm{h}_{\mathrm{t}}[n]$ and $\mathrm{g}_{\mathrm{t}}[n]$ represent the analysis wavelet kernels, and $\mathrm{h}_{\mathrm{r}}[n]$ and $\mathrm{g}_{\mathrm{r}}[n]$ represent the synthesis

wavelet kernels, respectively. The analysis and synthesis operations are depicted in Fig. 3-3. Note that for simplicity, the wavelet transform is given in one dimensional form.

In order to reduce the memory requirements and to accelerate the processing speed, the lifting-based DWT, often known as the second generation wavelet, has been proposed. The implementation of the lifting-based DWT consists of three steps: split, predict and update. In the split step the original samples $X[n]$ are firstly separated into two non-overlapping parts, an even part $X_e[n]$ and an odd part $X_o[n]$, described as follows [76].

$$\begin{aligned} X_e[n] &= X[2n] \\ X_o[n] &= X[2n+1] \end{aligned} \tag{3.3}$$

In the prediction step, the odd part is used to predict the even one. The derived prediction errors $d[n]$ represent the high frequency subband in the wavelet domain.

$$d[n] = X_e[n] - P(X_o[n]) \tag{3.4}$$

where P represents the prediction operator.

Finally, the odd part is updated by using the prediction errors to obtain the approximation samples $c[n]$, which are referred to as the low frequency subband in the update step.

$$c[n] = X_o[n] + U(d[n]) \tag{3.5}$$

where U denotes the update operator.

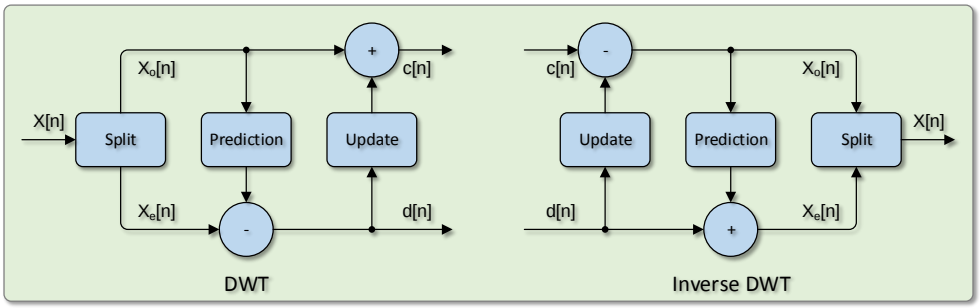


Fig. 3-4 Lifting implementation of the wavelet transform.

In order to achieve several levels of wavelet decomposition, the same processing procedure can be applied to the low frequency subband iteratively. The lifting scheme of the DWT is shown in Fig. 3-4.

After the wavelet transform, a uniform scalar quantisation with dead zone is applied to the transform coefficients in the JPEG2000 lossy compression mode. For each subband, the quantisation step size $\triangle_b$ is determined depending on the dynamic range of the values of the subband coefficients or by some other consideration, such as the perceptual importance or the bit rate budget. The obtained quantisation step size is used to quantise all the wavelet coefficients within the corresponding subband. The wavelet coefficients $y_b(i,j)$ in subband b are mapped to quantised indices $q_b(i,j)$. The scalar quantiser with quantisation step size $\triangle_b$ and a $2\triangle_b$ wide dead zone is shown in Fig. 3-5.

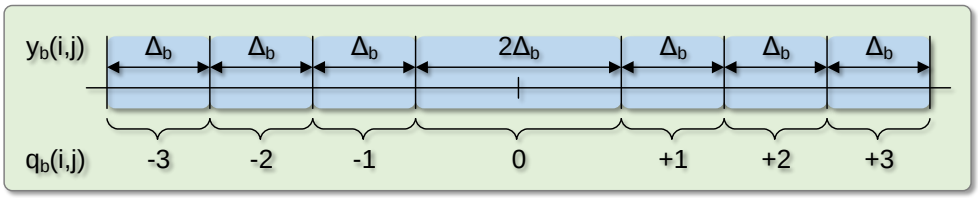


Fig. 3-5 Scalar quantiser with quantisation step size $\triangle_b$ and a $2\triangle_b$ wide dead zone.

The uniform scalar quantisation process with a dead zone is described as

$$q_b(i,j) = \mathbf{sign}\left[y_b(i,j)\right]\left\lfloor \frac{|y_b(i,j)|}{\triangle_b} \right\rfloor \tag{3.6}$$

where the $\lfloor\cdot\rfloor$ symbol is the floor operator.

Entropy Coding

EBCOT [72, 77, 78] is employed as the entropy coding algorithm in JPEG2000. In EBCOT, each subband is partitioned into relatively small rectangular blocks called code blocks. The typical size of a code block is 32×32 or 64×64 pixels, and each code block is coded independently. The EBCOT entropy coding of a subband consists of two steps: tier-1 coding and tier-2 coding. Tier-1 encoder comprises a bit plane coder and a binary arithmetic

coder, named MQ-coder, and it encodes each code block independently. Tier-2 reorders the compressed data and creates the compressed bitstream.

1. Tier-1 coding

After quantisation, each code block is divided into several bit planes according to the quantisation precision. The bit planes starting from the Most Significant Bit (MSB) plane to the Least Significant Bit (LSB) plane are coded progressively, as shown in Fig. 3-6.

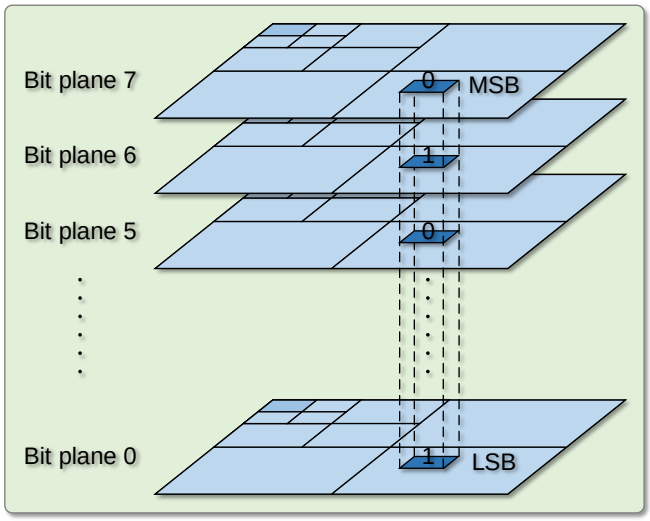


Fig. 3-6 An 8 bit image that is composed of 8 bit planes ranging from LSB to MSB.

1.1. Bit Plane Coder (BPC)

Each bit in a bit plane is encoded through one of the three non-overlapping coding passes, namely, Significance Propagation Pass (SPP), Magnitude Refinement Pass (MRP) and Cleanup Pass (CUP). The type of coding pass to be applied on a bit is determined by the state of the current bit and the context information of its eight adjacent neighbours. The state information contains three state variables σ, σ', and η. Initially, all state variables are initialised to zero. When the first non-zero bit of a sample has already been processed, the state of σ is updated to one, otherwise it is equal to 0. If a Magnitude Refinement Coding (MRC) operation has been

applied to the sample, σ' is set to one; otherwise, it is zero. If a bit of a sample has been processed using the Zero Coding (ZC) operation in the SPP pass, η is set to one; otherwise, it is equal to zero. Depending on the generated state context, one of the three passes (SPP, MRP, and CUP) is then performed. This results in a flexible truncation of the bitstream at the end of each pass enabling the target bit rate to be approached.

1.2. Binary Arithmetic Coder (BAC)

A context-based adaptive binary arithmetic coder, called MQ-coder, is adopted in JPEG2000. The MQ-coder selects a probability value from a predetermined lookup table depending on the generated context in the Bit Plane Coder (BPC). The probability value is used to adjust the intervals and progressively generate the compressed code stream. The block diagram of an EBCOT tier-1 encoder is illustrated in Fig. 3-7.

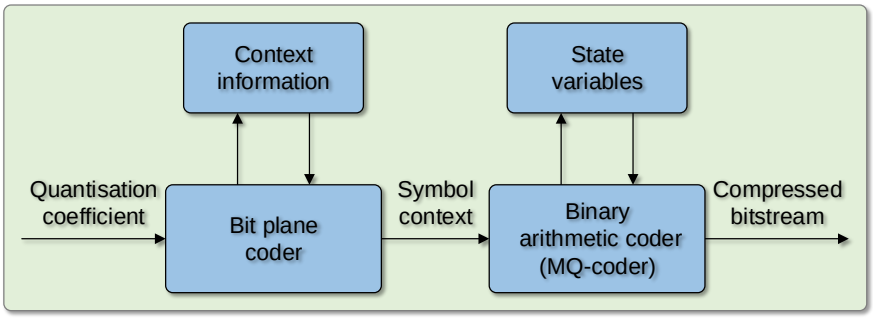


Fig. 3-7 Block diagram of an EBCOT tier-1 encoder.

2. Tier-2 coding

The bitstream generated from each code block needs to be reorganised to facilitate a specific functionality of JPEG2000. This process is often referred to as packetisation. Tier-2 coding is used to represent the layer and block summary information for each code block.

3.1.2 Wavelet Scalable Video Coding

With the intention of developing an efficient SVC standard, MPEG called for proposals targeted at new SVC technologies in October 2003. In response to the call, 15 proposals were submitted in March 2004, the majority of which were wavelet-based. However after the extensive evaluation stage, it was found that the H.264-based proposal suggested by the Fraunhofer Heinrich Hertz Institute (HHI) achieved a better coding performance than the other wavelet-based proposals. Consequently, MPEG selected both the H.264-based SVC proposed by HHI and the Barbell-lifting wavelet-based SVC proposed by MSRA for further improvement and comparison. Six months later, MPEG adopted the HHI proposal as the reference software, however the MSRA WSVC system provides a very convenient platform for continued research on wavelet-based video coding technologies.

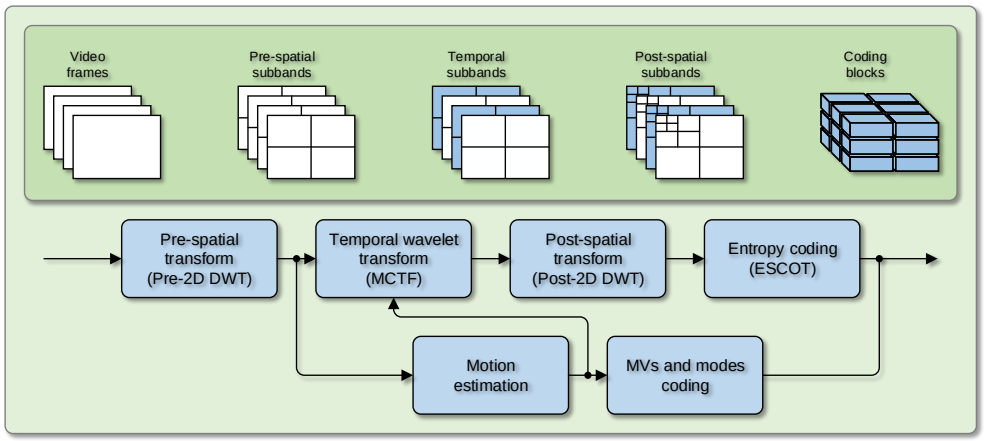


Fig. 3-8 Fundamental framework of WSVC.

Several coding technologies are incorporated in WSVC. These include a spatial 2D DWT which enables spatial scalability, MCTF which supports temporal scalability [79], and ESCOT, which inherits the ideas from EBCOT, and is used to support quality scalability [80]. The fundamental framework of WSVC is presented in Fig. 3-8.

MCTF

Unlike SVC, temporal scalability in WSVC is enabled by a MCTF which exploits temporal correlation [79]. The block diagram of a three level MCTF implementation based on the Haar wavelet basis is illustrated in Fig. 3-9. In this scheme, the input video sequence is first split into even and odd pictures. The prediction operation is then performed between two adjacent pictures to obtain a high-pass picture H and the original odd picture is overwritten by the obtained high-pass picture H. Subsequently, the low-pass picture L is generated from the update operation on the high-pass picture H and the original even picture.

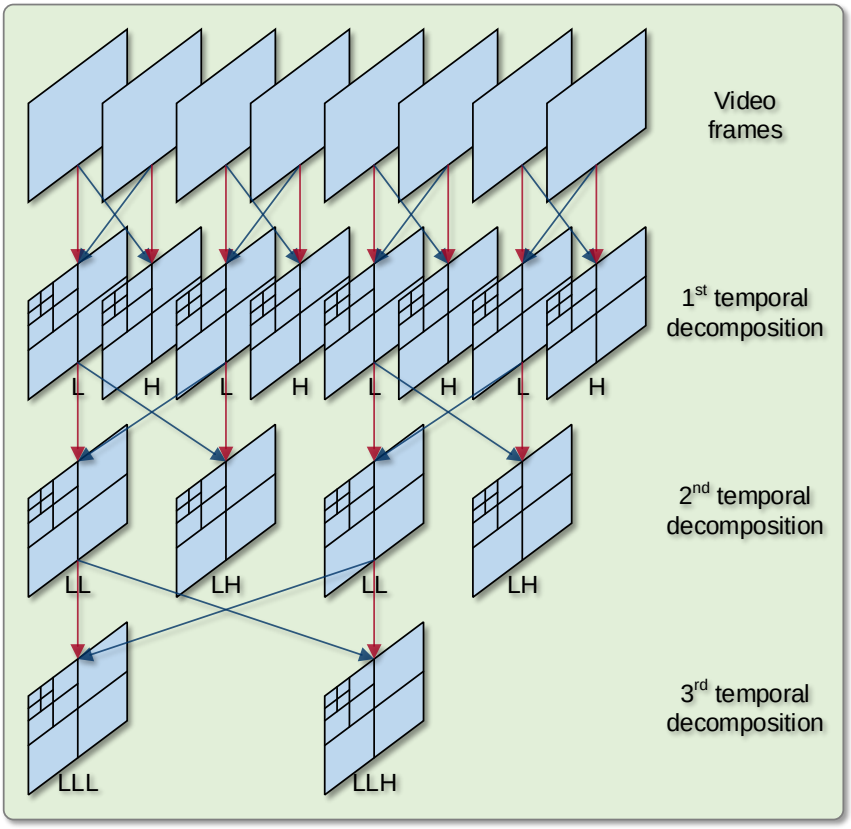


Fig. 3-9 Motion-compensated temporal decomposition using Haar wavelet.

The prediction operation and the update operation are explained by the following equa-

tions:

$$\begin{aligned} H_i &= \frac{1}{\sqrt{2}}(x_{2i+1} - P(x_{2i})) \\ L_i &= H_i + \sqrt{2}U(x_{2i}) \end{aligned} \tag{3.7}$$

where x_{2i+1} and x_{2i} denote the odd and even pictures, respectively, and P and U are the prediction and update operators.

DWT Transform and Quantisation

As in JPEG2000, the Cohen-Daubechies-Feauveau (CDF) 9/7 tap filter, which is most commonly used for lossy compression of images, is adopted by WSVC for the spatial DWT transform. In order to reduce the memory requirements and achieve a computationally efficient implementation of the DWT, the lifting scheme of the CDF 9/7 wavelet transform is employed. The coefficients of the low-pass and the high-pass analysis and synthesis filters of the CDF 9/7 wavelet are shown in Table 3-2. The lifting scheme of the 9/7 wavelet transform can be factorised into a sequence of alternating upper and lower triangular matrices and a diagonal matrix. The factorisation is expressed as follows:

$$\boldsymbol{P}(z) = \begin{bmatrix} 1 & \alpha\left(1+\frac{1}{z}\right) \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ \beta(1+z) & 1 \end{bmatrix} \begin{bmatrix} 1 & \gamma\left(1+\frac{1}{z}\right) \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ \delta(1+z) & 1 \end{bmatrix} \begin{bmatrix} \zeta & 0 \\ 0 & \frac{1}{\zeta} \end{bmatrix} \tag{3.8}$$

where z represents the z-transform; $\alpha\left(1+\frac{1}{z}\right)$ and $\gamma\left(1+\frac{1}{z}\right)$ indicate the prediction operations; $\beta(1+z)$ and $\delta(1+z)$ denote the lifting operations, respectively; ζ and $\frac{1}{\zeta}$ are the scaling coefficients which ensure the orthonormality of the transform. The CDF 9/7 wavelet coefficients of the lifting scheme are: α = -1.586134342, β = -0.05298011854, γ = 0.8829110762, δ = -0.4435068522, ζ = 1.149604398. The block diagram corresponding to equation (3.8) is shown in Fig. 3-10.

An embedded scalar quantisation with dead zone is used in WSVC due to the characteristics of the wavelet coefficients. A useful property of embedded quantisation is that the

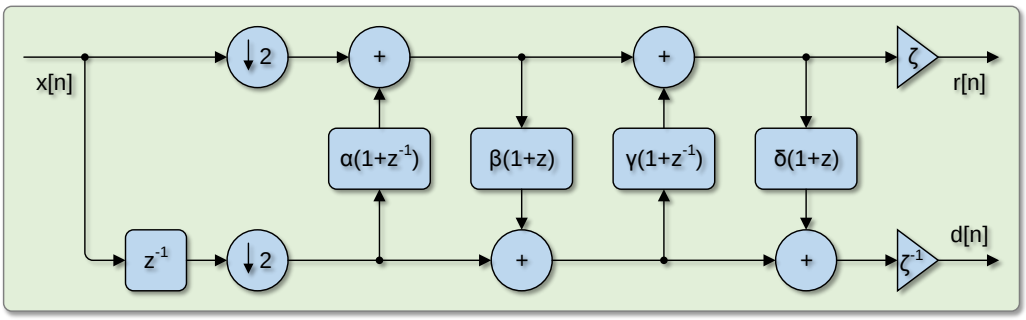


Fig. 3-10 Wavelet transform using the CDF 9/7 lifting scheme.

coded bitstream can be truncated by simply discarding the LSBs to obtain a subset of the compressed bitstream. As more of the compressed bitstream is retained, the reconstruction can be successively refined until full quality reconstruction is obtained. In embedded quantisation, the quantisation intervals of high bit rate quantisers are embedded in the quantisation intervals of all low bit rate quantisers, as shown in Fig. 3-11.

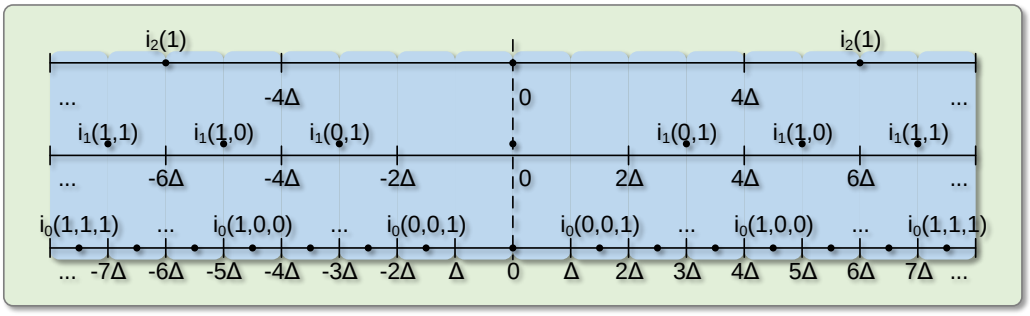


Fig. 3-11 Embedded dead zone uniform scalar quantiser.

The embedded dead zone quantiser quantises each wavelet coefficient into

$$i_p = Q_p(X) = \begin{cases} \mathbf{sign}(X) \cdot \left\lfloor \dfrac{|X|}{2^p \triangle} + \dfrac{\xi}{2^p} \right\rfloor & \textbf{if } \dfrac{|X|}{2^p \triangle} + \dfrac{\xi}{2^p} > 0 \\ 0 & \textbf{otherwise} \end{cases} \tag{3.9}$$

where the $\lfloor \cdot \rfloor$ symbol is the floor operator; $\xi < 1$ determines the width of the dead zone; $\triangle > 0$ is the basic quantisation step size; $p \in \mathbb{Z}_+$ is the quantiser level, which is determined by the dynamic range of the input and a larger p value means a coarser quantiser.

The inverse quantisation is performed as

$$y_i^p = Q_p^{-1}(i_p) = \begin{cases} \mathbf{sign}(i_p) \cdot \left(|i_p| - \frac{\xi}{2^p} + \delta \right) 2^p \Delta & \textbf{if } i_p \neq 0 \\ 0 & \textbf{otherwise} \end{cases} \tag{3.10}$$

where δ indicates the location of y_i^p in the quantisation interval. From equation (3.9), it can be seen that all the dead zone quantisers with quantisation step sizes of $2^p\Delta$, $p \in \mathbb{Z}_+$ are embedded in the quantiser with quantisation step size Δ. The quantisation results of a quantiser with quantisation step sizes of $2^p\Delta$, $p \in \mathbb{Z}_+$ are equivalent to those of the quantiser with quantisation step sizes of Δ but discarding the p LSBs. In other words, if the p LSBs are unavailable, inverse quantisation may still be performed, but a lower level of quality is obtained.

Entropy Coding

ESCOT [80] is employed as the entropy coding algorithm. It inherits the ideas and coding procedures from EBCOT used in JPEG2000. After temporal wavelet decomposition and application of the 2D spatial transform, the wavelet coefficients are encoded using a bit plane coding scheme. ESCOT is utilised to achieve better coding efficiency and flexibility, and to provide other functionalities for scalable wavelet video compression. Using bit plane coding and context-based arithmetic coding, ESCOT achieves bit rate scalability by independently coding coefficients in individual subbands.

Depending on the binary valued state of a sample in each bit plane, one of three coding operations is performed to code the binary information of the current sample. The three coding operations of ESCOT are described as follows.

1. **Zero Coding (ZC)**: When a sample is not yet significant in previous bit planes, ZC is used to code new information about whether it becomes significant or not in the current bit plane. The ZC uses significance information of the neighbouring samples to encode the current sample.

2. **Sign Coding (SC)**: Once a sample becomes significant in the current bit plane, SC is used to code the sign information.
3. **Magnitude Refinement Coding (MRC)**: If a sample has already become significant in a previous bit plane, MRC is used to code the new information about the current sample.

In order to produce an embedded bitstream, RDO is also performed to decide the number of bits that should be allocated for each code block. Finally, the coded data is packeted to form the bitstream.

3.2 Performance Evaluation Design

To compare the coding efficiency of the three scalable coding algorithms, SVC, Motion JPEG2000 and WSVC, JSVM 9.18 [81] was chosen as the evaluation model for SVC, the Kakadu v7.2 [82] implementation was selected for Motion JPEG2000, and the VidWav (Video Wavelet) [83], (which is the Reference Model and Software (RM/RS) utilised by MPEG) was used for WSVC.

3.2.1 Video Test Sequences

In order to make a comprehensive comparison, four different sets of standard video test sequences were processed. Each set relates to a particular resolution, and each set comprises video sequences featuring different degrees of activity and texture detail.

In the first experiment, the RD performance of the three codecs was evaluated for video sequences of low and medium resolution. Low resolution CIF video (352×288 pixels) is commonly used in video conferencing systems and video streaming applications, and medium resolution 4CIF video (704×576 pixels) is widely used in Closed-Circuit Television (CCTV) monitoring systems and SDTV. The second experiment was devoted to evaluation on high resolution video sequences including 720p High Definition (HD) sequences at 1280×720 pixels and 1080p full HD sequences at 1920×1080 pixels. All the test sequences

were in raw YUV 4:2:0 colour format (see Fig. 1-1), and are listed in Table 3-4.

Table 3-4
Video test sequences used for evaluation

Sequence	Resolution
Bus, Foreman	CIF (352×288)
City, Soccer	4CIF (704×576)
Park, Tree	720p (1280×720)
Flower, Sky	1080p (1920×1080)

3.2.2 Codec Settings

The parameter configurations for SVC were set as listed below, and the other parameters were set to the default values of the JSVM 9.18.

† Main profile was used

† CABAC was used for entropy coding

† RDO was always enabled

† One slice per picture

† Intra-frame coding was always considered for each frame

† GOP structures: IIII

† Inter-layer prediction was enabled.

The Motion JPEG2000 codec was configured with the following settings, and other parameters were set to the default values of the Kakadu v7.2 implementation.

† One tile per frame (no tiling)

† Code block size of 64×64

† 5 levels of wavelet decomposition

† 9/7 tap bi-orthogonal Daubechies wavelet filter kernel.

The following parameters were applied in the WSVC encoder, and all options were set to the default values of VidWav.

† 5 levels of wavelet decomposition

† ESCOT was used for entropy coding.

3.2.3 Evaluation Criteria

In order to evaluate the objective performance, the Peak Signal-to-Noise Ratio (PSNR) of the luminance component averaged over all encoded frames in a sequence was chosen as the measure of visual quality:

$$\mathrm{PSNR} = 10\log_{10}\left(\frac{255^2}{\mathrm{MSE}\left(Y,\tilde{Y}\right)}\right) \tag{3.11}$$

where Y represents the intensity of the luminance component in the original picture and $\tilde{Y}$ is the reconstruction. MSE is the Mean Squared Error between the original picture and the reconstructed picture. In addition, the RD curves for the luminance component, which show the luminance PSNR versus the bit rate, are presented and analysed.

3.3 Results and Discussions

The comparison results are described in the following two subsections. The first subsection presents the coding results of each algorithm on low and medium resolution test sequences. The evaluation results indicate the optimal choice for applications such as video conferencing and video surveillance monitoring. In the second subsection, the performance of the three scalable coding algorithms on HD video sequences is discussed.

3.3.1 Evaluations for Low and Medium Resolution Video

Table 3-5 shows the RD performance of the three scalable video coding algorithms for CIF and 4CIF resolution video sequences. The corresponding RD curves for 'Bus', 'Foreman', 'City' and 'Soccer', which relate to CIF and 4CIF resolutions, are presented in Fig. 3-12. For the CIF video sequences, when the bit rate is low, SVC, Motion JPEG2000, and WSVC each achieve very similar coding efficiency. The differences in coding efficiency become

more significant as the bit rate is increased. It can also be seen from Fig. 3-12 that the RD curves for the CIF video sequences ('Bus' and 'Foreman') are very similar at low bit rates. However, the RD curves become more distinctive at higher bit rates, which means these algorithms result in different coding efficiency. In general, at all bit rates WSVC predominates over the other two video coding algorithms in terms of coding efficiency. SVC takes second place, followed by Motion JPEG2000. One possible explanation is that the use of the embedded scalar quantiser and the embedded subband coding tools significantly improves the coding efficiency of WSVC, whereas SVC sacrifices coding efficiency by about 10% compared with the H.264/AVC encoder in order to support various forms of scalability. Consequently, WSVC shows better performance than SVC, a gain in PSNR of up to 3.1dB.

Table 3-5
RD performance for low and medium resolution video sequences

Sequence	SVC(IIII)		JPEG2000		WSVC	
	Bit rate	PSNR	Bit rate	PSNR	Bit rate	PSNR
Bus (CIF)	603.68	23.60	589.22	23.69	603.05	24.75
	1205.91	26.78	1177.44	25.41	1205.30	27.90
	2413.44	30.65	2356.84	29.20	2410.90	32.09
	4820.23	35.85	4707.81	34.76	4819.30	37.96
	9697.84	42.70	9411.40	41.15	9636.20	45.82
Foreman (CIF)	606.89	30.06	592.39	30.21	606.36	32.18
	1208.44	33.93	1179.52	32.91	1207.60	35.30
	2408.25	37.87	2351.16	36.00	2407.20	39.29
	4840.43	42.14	4708.50	39.84	4821.20	44.19
	9678.49	47.77	9405.07	46.03	9630.80	50.21
City (4CIF)	2412.32	27.75	2355.44	27.96	2411.00	29.24
	4821.92	30.86	4708.66	30.47	4821.70	32.26
	9644.43	34.80	9418.30	33.67	9642.90	36.37
	19273.17	39.55	18822.79	39.36	19274.00	42.20
	38805.09	46.18	37642.70	46.36	38544.00	49.40
Soccer (4CIF)	2412.28	32.48	2355.94	32.61	2410.30	33.36
	4823.52	35.68	4710.57	35.11	4821.30	36.60
	9640.67	39.42	9411.57	38.71	9642.20	40.88
	19379.58	44.12	18827.62	44.09	19273.00	46.21
	36332.15	49.62	35309.97	50.90	36155.00	52.26

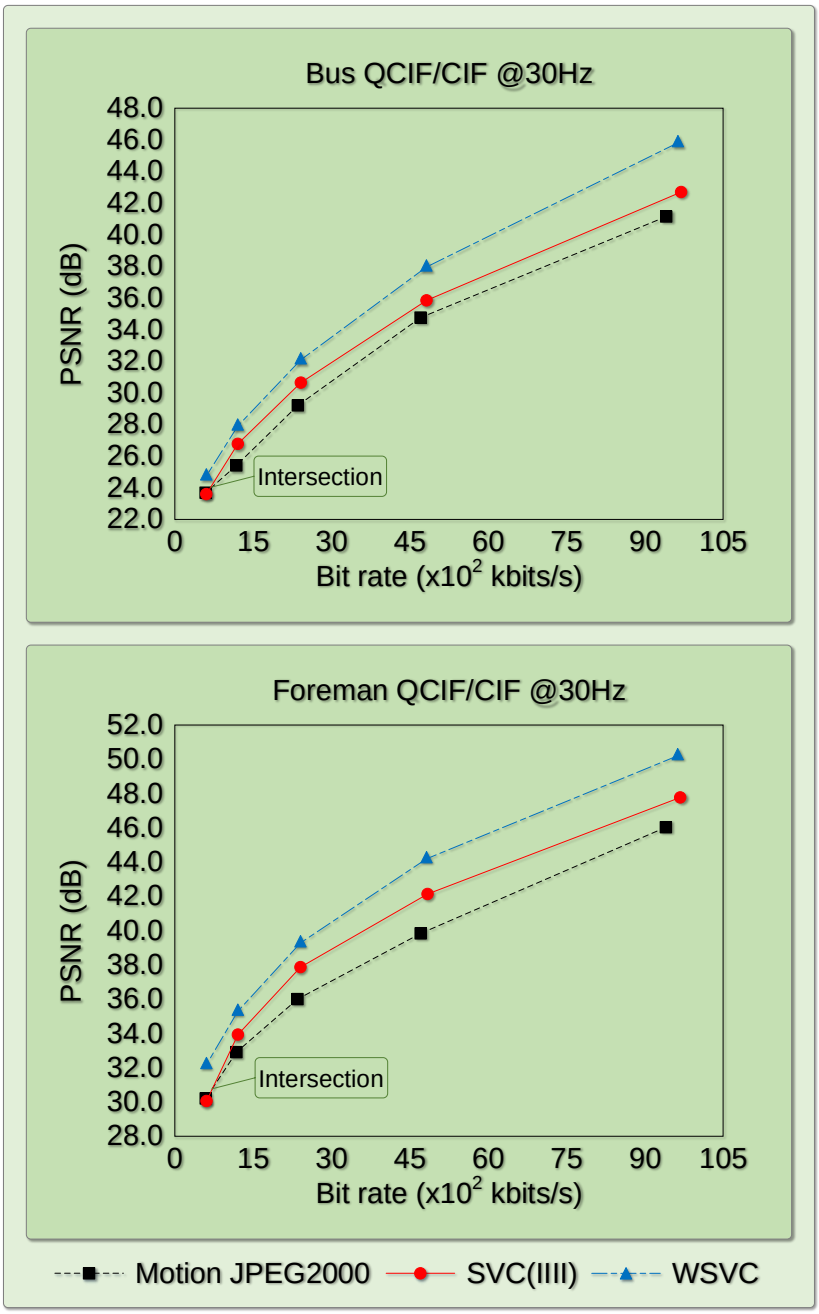


Fig. 3-12 RD performance for low and medium resolution video sequences (continued on next page).

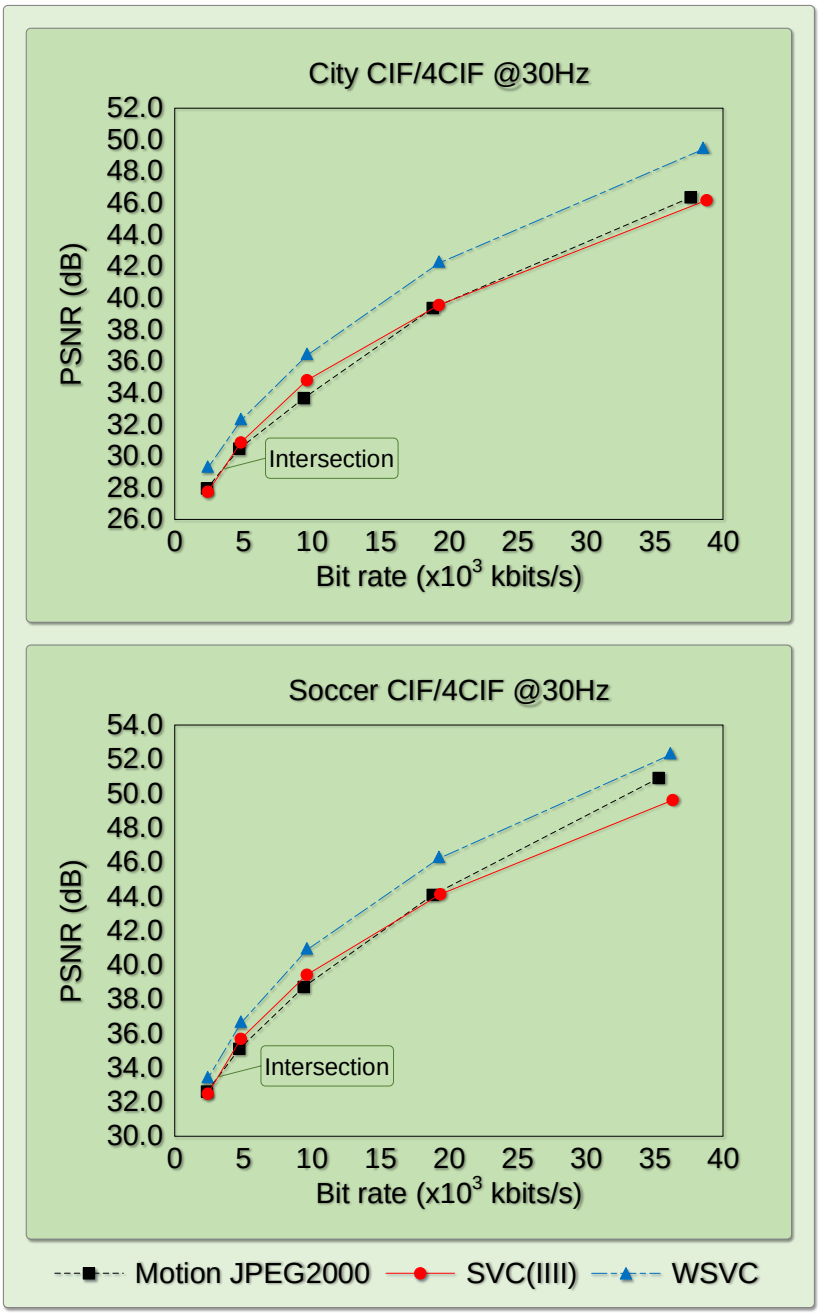


Fig. 3-12 RD performance for low and medium resolution video sequences (continued from last page).

Since Motion JPEG2000 independently codes each video frame using JPEG2000, the coding efficiency of Motion JPEG2000 is exactly the same as that of JPEG2000. Shi et al. [84] and Jiao [85] have shown that H.264/AVC outperforms JPEG2000 at low compression ratios, and vice versa. As demonstrated in Fig. 3-12, SVC consistently outperforms Motion JPEG2000 over a wide range of bit rates. This indicates that the additional coding tools in SVC intra-coding result in extra coding gains. The difference between SVC intra-coding and H.264/AVC intra-coding is the utilisation of inter-layer intra-prediction in SVC. Therefore, it seems reasonable that this inter-layer intra-prediction mechanism is responsible for the improved coding efficiency of SVC and why SVC outperforms Motion JPEG2000.

Similar experimental results are observed for the 4CIF video sequences. It can be seen that Motion JPEG2000 performs better for higher resolution video sequences, but stays competitive with SVC. However, WSVC still produces the best coding performance of the three coding algorithms.

3.3.2 Evaluations for High Resolution Video

The coding performance of SVC, Motion JPEG2000, and WSVC, when encoding high resolution video, namely 720p and 1080p video sequences, is also investigated. The resolutions are the most commonly used HD video display formats.

The RD performance of the three scalable coding algorithms for HD video sequences is presented in Table 3-6. Fig. 3-13 shows the corresponding RD curves for 'Duck', 'Park', 'Flower' and 'Sky', which relate to resolutions of 720p and 1080p. As with the evaluation results for low and medium resolution video, WSVC yields the best coding performance. When the bit rate is high, SVC outperforms Motion JPEG2000 in terms of RD performance, but when the bit rate is low, the inverse is true. It can be seen from Fig. 3-13, that there is an intersection between the RD curve of SVC and that of Motion JPEG2000. When the bit rate is less than the operating point indicated by the intersection, Motion JPEG2000 produces better performance than SVC. However, when the bit rate surpasses this point,

SVC predominates. The position of the intersection is related to the resolution and the content of the picture. When encoding the 720p video sequence, the intersection is located in a relatively low bit rate region. This shows that Motion JPEG2000 only performs better than SVC for a small range of low bit rates. As the resolution is increased, the intersection moves to the right, which means Motion JPEG2000 produces better coding efficiency than SVC in a wider range of lower bit rates. For the video sequences that contain rich detail, such as 'Flower' and 'Duck', the intersection of the RD curves for Motion JPEG2000 and SVC is located at a higher bit rate. The simulation results show that Motion JPEG2000 is the better coding choice for high resolution sequences containing complex detail at low bit rates.

Table 3-6
RD performance for high resolution video sequences

Sequence	SVC(IIII)		JPEG2000		WSVC	
	Bit rate	PSNR	Bit rate	PSNR	Bit rate	PSNR
Park (720p)	5481.09	23.67	5351.56	24.32	5477.80	24.65
	10960.54	26.13	10702.68	25.83	10957.00	27.07
	21910.25	29.44	21395.74	27.84	21910.00	30.49
	43837.56	33.96	42808.68	31.61	43794.00	35.57
	87666.84	40.11	85655.50	35.72	87682.00	42.45
Tree (720p)	5481.52	32.39	5352.00	32.19	5478.70	33.35
	10962.51	34.27	10704.58	33.87	10958.00	35.32
	21905.35	36.57	21390.71	35.27	21907.00	37.75
	43803.78	39.64	42776.20	37.59	43796.00	41.05
	88072.57	44.63	85664.39	41.07	87680.00	46.46
Flower (1080p)	12755.86	40.00	12025.82	43.30	12313.85	43.43
	25295.07	43.44	24059.77	45.37	24636.90	45.57
	47374.14	45.70	48056.31	46.89	49209.56	47.29
	97336.86	49.40	96082.65	48.96	98377.94	49.97
	151692.29	52.36	154619.26	53.21	158278.81	53.53
Sky (1080p)	11992.73	32.90	12042.49	35.63	12314.32	35.78
	23095.69	37.42	24095.58	38.52	24630.34	39.56
	48462.95	42.38	48205.11	41.74	49204.60	43.69
	93754.43	46.55	96499.00	45.78	98810.61	47.47
	154356.05	50.32	188416.13	52.46	192917.98	53.27

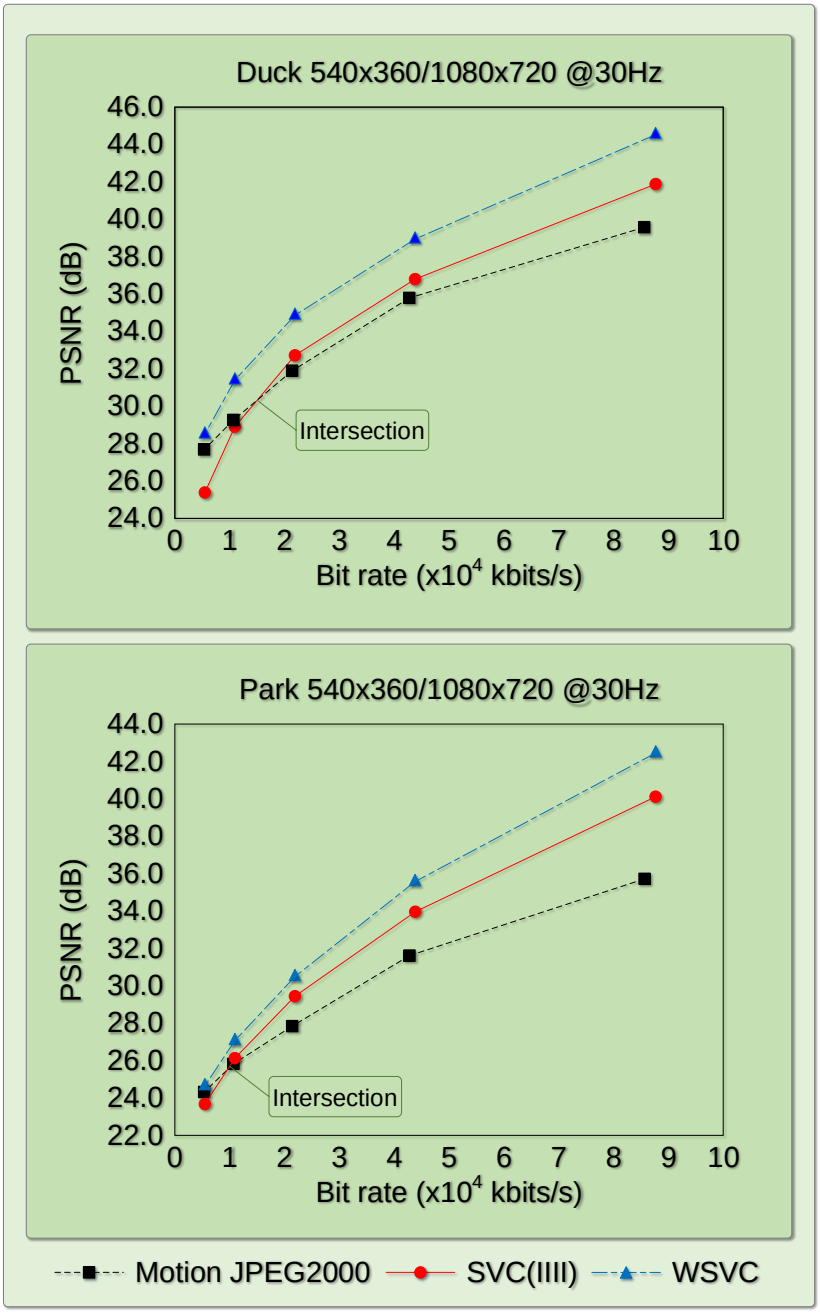


Fig. 3-13 RD performance for high resolution video sequences (continued on next page).

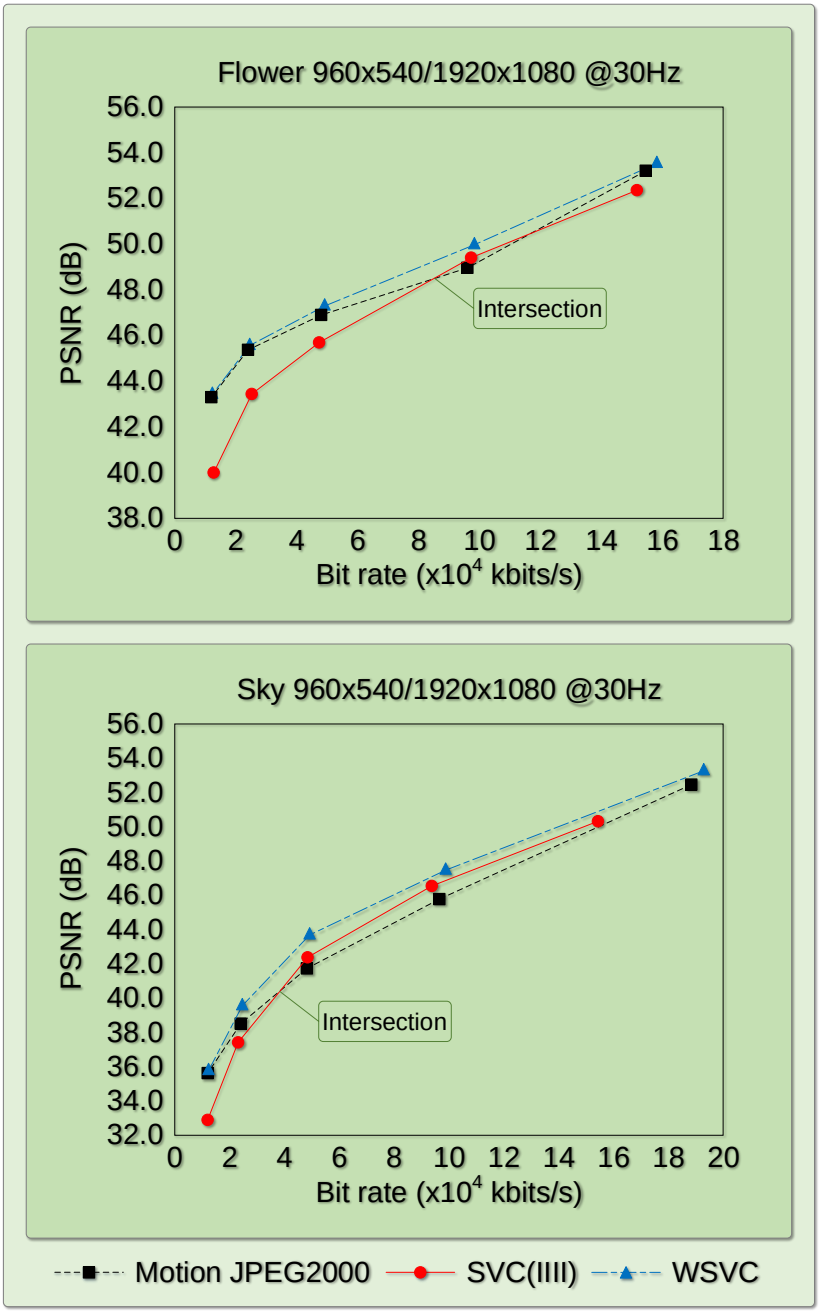


Fig. 3-13 RD performance for high resolution video sequences (continued from last page).

3.4 Summary

In this chapter, a comparative investigation of SVC, JPEG2000 and WSVC was presented. The objective was to obtain a better understanding of scalable coding systems and to gain an insight into the most suitable application scenarios for each scheme. The chapter initially described the key algorithms behind the Motion JPEG2000 and WSVC coding schemes. Firstly, the DWT, uniform dead zone scalar quantisation, and the EBCOT entropy coding method of Motion JPEG2000 were briefly described. Thereafter, the MCTF, the spatial 2D DWT, embedded quantisation and the ESCOT entropy coding algorithm of WSVC were introduced.

As all three video coding algorithms have been designed to produce scalable video bitstreams, the coding efficiency in terms of RD performance was compared and analysed. Experimental results were obtained for a wide range of standard video test sequences with resolutions ranging from CIF to 1080p.

The results show that, of the three codecs, WSVC produces the best coding performance for low and medium resolution video applications, whereas SVC produces the next best performance. When HD video sequences are coded, WSVC still produces the best performance. The relative coding performance of SVC and Motion JPEG2000 depends on the resolution of the video sequence and the content of the pictures.

Chapter 4

Fast Mode Decisions Based on Motion Activity

The scalable extension of H.264/AVC provides an increased degree of supported scalability and demonstrates significant improved coding efficiency relative to the scalable profiles of previous video coding standards. However, the superior coding performance is achieved at the cost of significantly increased computational complexity. The computational requirements of the SVC encoder is a limiting factor for its application. If SVC is to become widely adopted, fast mode decision algorithms to alleviate the computational complexity of the SVC encoder without any significant loss in compression performance and coding efficiency are highly desirable.

As a consequence, fast mode decision algorithms for intra- and inter-frame coding and inter-layer coding, are a major topic in this thesis. In this chapter, a fast inter-frame and inter-layer mode decision algorithm based on motion activity is described. The next chapter extends the work into a hierarchical scheme.

The remainder of this chapter is organised as follows: A literature review of fast mode decision algorithms for SVC is discussed in the next section. Section 4.2 gives a detailed formulation of the proposed fast mode decision algorithm. Experimental results are pre-

sented in section 4.3 and some conclusions are given in section 4.4.

4.1 Existing SVC Fast Algorithms

Compared with the scalable profiles of earlier video coding standards, SVC provides much improved coding efficiency, but at the cost of significantly increased computational complexity. In the encoder, the coding mode decision is the most time-consuming process. Additionally, the use of extra inter-layer prediction tools imposes a considerable computational burden. As the simulation results of chapter 2 show, inter-layer residual prediction doubles the computational complexity of the mode decision process. Inter-layer motion prediction also increases the complexity of the motion estimation process.

Easing the computational burden, and hence reducing the encoding time of the SVC encoder, has been the subject of several studies. Generally, most fast mode decision approaches for SVC detailed in the literature can be categorised as 'fast algorithms extended from single layer coding' and 'solutions targeting inter-layer prediction'. As SVC is an extension of H.264/AVC, the base layer employs the methodology of H.264/AVC, and the enhancement layers are supplemented by additional tools to support the scalability. A common strategy is to use and extend the fast algorithms of H.264/AVC to reduce the computational complexity of SVC. These algorithms employ various features of the current coded macroblock, such as texture and edge, to predict the most likely modes and/or to delete unlikely modes. In addition, the temporal dependency, spatial homogeneity, and the mode correlation are also considered. By contrast, solutions targeting inter-layer prediction exploit some of the coding results of the base layer to relieve the mode decision process in the enhancement layers. The next two subsections review existing methods that fit within each of these categories.

4.1.1 Fast Algorithms Extended from Single Layer Coding

In [86], Yu et al. proposed a fast mode decision algorithm to alleviate inter-frame encoder complexity. This algorithm considers the energy of the transform's AC coefficients as a spatial complexity measurement of a macroblock to reduce the number of candidate modes. Yu showed that when the total energy of the AC coefficients in the macroblock is less than a fixed threshold, the macroblock is best categorised as containing simple texture, and for this to be considered when choosing a reduced subset of modes for evaluation. Although this algorithm demonstrates a significant time saving, the threshold was determined empirically, though little attempt was made to optimise the threshold value. When determining the optimal threshold, both the RD performance and the computational complexity reduction must be taken into consideration.

Based on the assumption that large block partitions are appropriate for homogeneous and stationary areas, Wu et al. [87] described a fast algorithm which performs RDO with only a subset of candidate modes in highly homogeneous and stationary regions. A Sobel filter was used to determine the homogeneity of a macroblock and the Sum of Absolute Differences (SAD) of the macroblock was used to predict the temporal stationarity. However, all the pixels in the whole frame have to be evaluated, which results in additional computational complexity. Consequently, for video sequences with complex motion, this algorithm demonstrated only a limited saving in encoding time.

Based on the assumption that a large partition is beneficial for a macroblock with high motion continuity, Shen et al. [88] employed the spatial continuity of the motion field to reduce the number of mode candidates to be examined. This fast algorithm used a Sobel operator to measure the motion continuity of each macroblock and it showed that motion information can be used for speeding up the encoding process. Due to the bottom-up coding structure of SVC, for each frame, the base layer is encoded before the enhancement layers. Consequently, when encoding the enhancement layers, the motion information of the base layer can be used in a more straight-forward manner, thus improved coding

performance can be expected.

In [89], Yu et al. proposed a three level hierarchical fast mode decision process. The first level considered the SKIP mode, the second level considered the large partitions (INTER_16×16, INTER_16×8, and INTER_8×16 modes), and the last level considered the remaining submacroblock modes. At each level, different strategies were implemented and an early termination of the mode selection process was also triggered to avoid a full cycle of RDO. The hierarchical structure employed in Yu's work demonstrated superior performance for a wide range of video signals. However, there is no inter-layer information in H.264/AVC and, in contrast, SVC contains more context information, which could contribute to the speeding up of the process.

Although the aforementioned algorithms achieve good coding performance, all of them were developed for single layer video coding. It is obvious that as the inter-layer information was not taken into account, there must be a large amount of computational redundancy that remains in the encoding.

4.1.2 Solutions Targeting Inter-layer Prediction

Inter-layer prediction in the SVC encoder requires a significant amount of computation. Fast algorithms need to consider the inter-layer correlation if they are to demonstrate improved performance in computational complexity reduction.

In [90], Kim et al. used a macroblock's RD cost of the BL_SKIP mode to categorise the macroblock complexity. Consequently, the complexity of a macroblock is used to determine the mode candidates that need to be examined. The algorithm then reduces the number of candidate modes for the enhancement layer. Consequently, the RD cost of the BL_SKIP mode is the only indicator of the estimation of a macroblock's complexity in the enhancement layer, which means only a little information of the base layer is reused in Kim's method. In addition, the algorithm's performance is highly dependent on a constant K which determines the time saving and reconstructed picture quality. It cannot guaran-

tee the constant K is widely-applicable to video sequences with different motion activity and picture detail.

In [91], depending on the mode distribution relationship between the base layer and the enhancement layers, Li et al. reduce the number of candidate modes in the enhancement layer according to the best mode in the corresponding position in the base layer. However, this method provides poor results when the correlation between base layer and enhancement layer is weak.

Goh et al. [92] proposed an algorithm that makes use of the relationship between the current macroblock and its neighbours to decrease the encoding time. Nevertheless, for fast changing video sequences, as expected it results in bit rate degradation.

In [93], Zhao et al. attempts to predict the most likely modes and then orders them. However, if a better coding performance is to be achieved, not only should unlikely modes be deleted but also the most probable ones retained. Furthermore, the mode relationship between the macroblock and its neighbours is not investigated, and the time saving could be further improved.

In Lee's work [94], the transform coefficients and partition information of the corresponding macroblock in the base layer are employed to reduce the number of mode candidates in the enhancement layer. However, the relationship between partition information and both spatial detail and temporal similarity does not always exist. Therefore, Lee's method is apt to miss out the best matching mode in the reduced subset of candidates. In order to avoid missing the optimal mode, only when the co-located macroblock is MODE_SKIP or INTRA, can appropriate strategies be applied to find the optimal mode.

4.2 The Proposed Fast Mode Decision Algorithm

In SVC, motion-compensated prediction of the enhancement layer is performed in both the same resolution layer and the corresponding lower layer, which results in greatly in-

creased computational complexity. In [44], the authors pointed out that even if the reconstructed lower layer is essentially lossless, the data may not be optimum for inter-layer prediction. For some video sequences with slow motion and low picture detail, temporal prediction generally achieves a better approximation than the upsampled lower layer reconstruction. Therefore, an inter-layer and inter-frame mode decision algorithm using information from key frames is proposed. Statistics obtained empirically reveal that a macroblock with slow movement is more likely to be best matched by one in the same resolution layer. However for a macroblock with fast movement, block matching between layers is required. This priori knowledge forms the basis for the mode decision method proposed.

4.2.1 Observations and Algorithm Formulation

To improve coding efficiency, besides performing inter- and intra-prediction within each layer, as in single layer coding, inter-layer prediction is also employed in SVC. This exploits the reconstructed data of the lower layers, so in the enhancement layer, the prediction signal is obtained either by conventional motion-compensated temporal prediction or by upsampling the reconstructed lower layer information.

Motion Information from P-Frames

Although inter-layer prediction effectively improves the coding efficiency of SVC, not all lower layer upsampled data is suitable for inter-layer prediction, especially for video sequences with slow motion or simple texture, such as the 'Mother-daughter' sequence. The reason for this is that, in this case, the best matching macroblock can usually be found in the temporal reference frames. Therefore, an approach is proposed to determine whether the current macroblock is more suitable for inter-frame prediction or inter-layer prediction, in other words, to determine whether the current macroblock represents slow or fast motion. It can then be decided which prediction mode should be applied to the current

macroblock, inter-frame prediction or inter-layer prediction. The Motion Vector Difference (MVD) between P-frames in each GOP is chosen as the measure of motion. This is motivated by the hierarchical coding structure of SVC and the fact that MVD is a good measure with which to categorise video motion activity.

In order to realise temporal scalability, SVC adopts a hierarchical coding structure to partition a video sequence into a number of temporal layers. Fig. 4-1 illustrates a typical hierarchical coding structure. The first frame is encoded as an I-frame, and the encoder inserts a key frame at regular intervals, the key frame being encoded either as an I-frame or a P-frame, and serves as a reference for subsequent frames. In a hierarchical coding structure, an I-frame is firstly coded without reference to any other frames. Subsequently each P-frame uses the previous key frame as a reference for prediction. Consequently, the remaining frames of a GOP are hierarchically predicted and coded as B-frames. In other words, the B-frames in a GOP are encoded after the I- and P-frames. Therefore, some encoding results from the P-frames can be used to eliminate the computational cost of the B-frames.

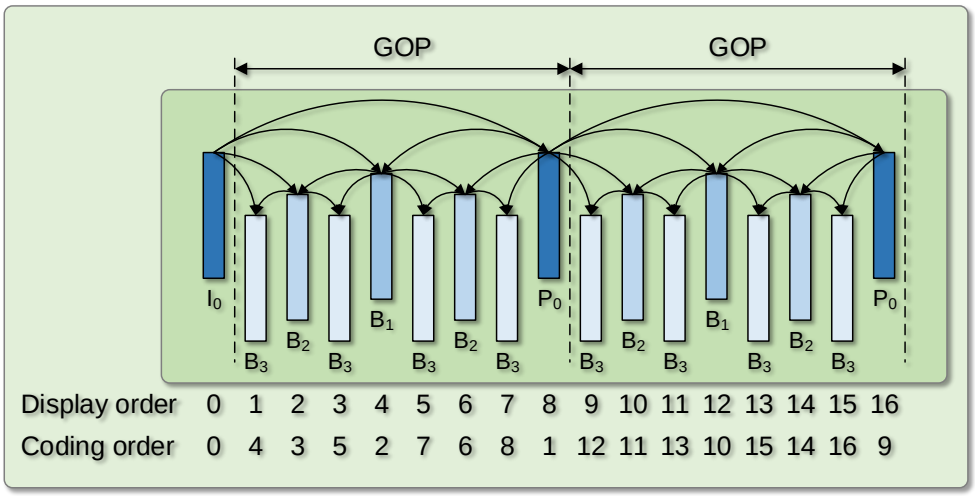


Fig. 4-1 A typical hierarchical B-frame coding structure.

In a natural video clip, a video object usually contains complex movements, such as

translation, rotation, and scaling. A sophisticated model is required to represent the motion in a video sequence, as the trajectory of a video object can be arbitrary [95]. Only considering the simplest case, namely the trajectory, motion is assumed to be linear. In Fig. 4-2, assuming that the truck is moving from the left to the traffic lights on the right at a constant velocity of $v_t(x)$ between $t = t_{k-1}$ and τ ($\tau > t$). The trajectory of the truck can be described using a linear model as follows:

$$x(\tau) = x(t) + v_t(x)(\tau - t) = x(t) + d_{t,\tau}(x) \tag{4.1}$$

where $d_{t,\tau}(x) = v_t(x)(\tau - t)$ is a displacement vector measured from t to τ.

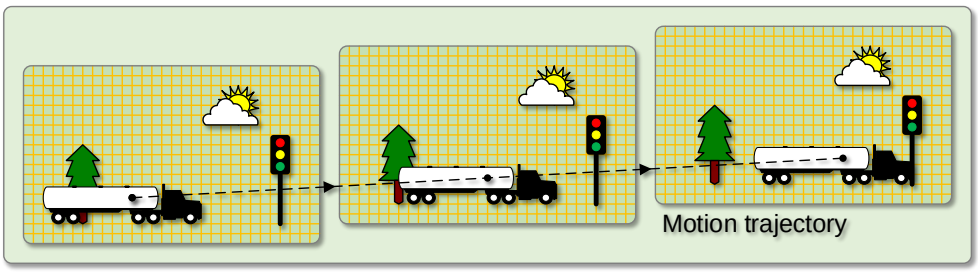


Fig. 4-2 An example of a linear motion trajectory.

In SVC, a block-based search algorithm is used to estimate the displacement of all pixels in a block. The obtained displacement is represented by a MV, which is determined by a predefined matching criterion, such as Cross-Correlation Function (CCF), Mean Squared Error (MSE), and Mean Absolute Error (MAE). As a result, the MV can be expressed as

$$\text{MV} = v_t(x)(\tau - t) \tag{4.2}$$

It can be seen from equation (4.2) that, at a given period of time, a larger MV corresponds to faster motion, and vice versa. Therefore, the MV can be used as a measure to categorise video motion activity.

In SVC, differential coding is applied to MVs to further reduce the motion information overhead. That is, only the difference between the actual MV and the MVP is encoded and

transmitted, instead of coding the actual MV directly. In SVC, the MVP of a macroblock in the enhancement layer is taken either from its spatially surrounding macroblocks in the same layer or from the corresponding macroblock in the previous layer.

In the same layer, the MVs of adjacent macroblocks tend to be very similar, consequently the current MV can be predicted from the three MVs which are located to the left, above, and above-right, as shown in Fig. 4-3.

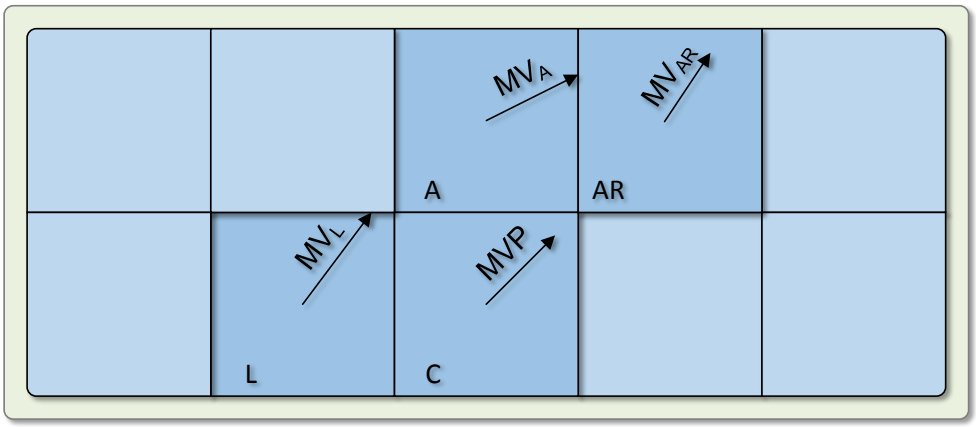


Fig. 4-3 MVs of a current block and its neighbours.

The horizontal and vertical components of the current macroblock's MVP are calculated separately and each of the components is the median value of the three neighbouring MVs.

$$\begin{aligned} \mathrm{MVP}^x &= \textbf{Median}\left(\mathrm{MV}_\mathrm{L}^x, \mathrm{MV}_\mathrm{A}^x, \mathrm{MV}_\mathrm{AR}^x\right) \\ \mathrm{MVP}^y &= \textbf{Median}\left(\mathrm{MV}_\mathrm{L}^y, \mathrm{MV}_\mathrm{A}^y, \mathrm{MV}_\mathrm{AR}^y\right) \end{aligned} \tag{4.3}$$

where x and y denote the horizontal and vertical components; subscripts L, A, and AR stand for the macroblocks to the left, to the above, and the above-right.

In the enhancement layers of SVC, the MVP can be obtained by the conventional median-based approach, or the scaled MV of the corresponding block in the previous layer can be used as the MVP, as shown in Fig. 4-4.

The MVP determines the centre of the searching area. Thereafter, a block matching

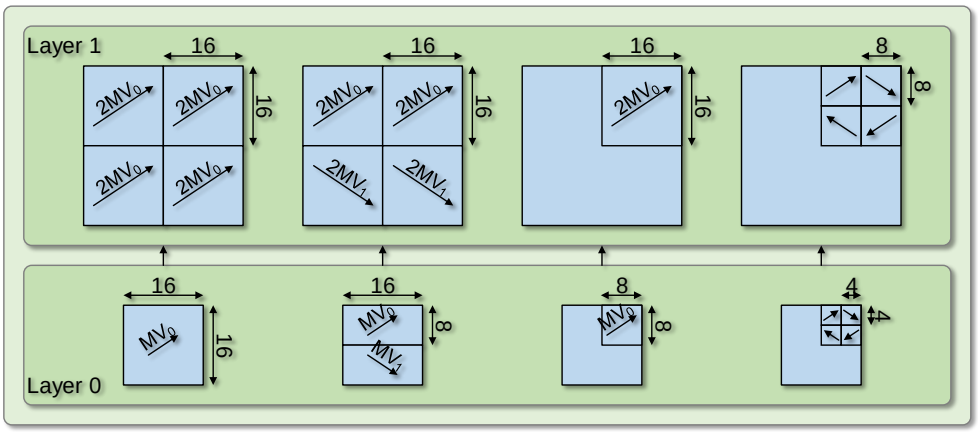


Fig. 4-4 Inter-layer MV prediction with various block sizes.

process is performed to find the actual MV within a defined search range under a conventional matching criterion. Finally, the MVD between the actual MV and the MVP, which is defined as

$$|\mathrm{MVD}| = |\mathrm{MV}_{\mathrm{actual}} - \mathrm{MVP}| \tag{4.4}$$

is encoded and transmitted. The relationship between the actual MV and the MVP is illustrated in Fig. 4-5.

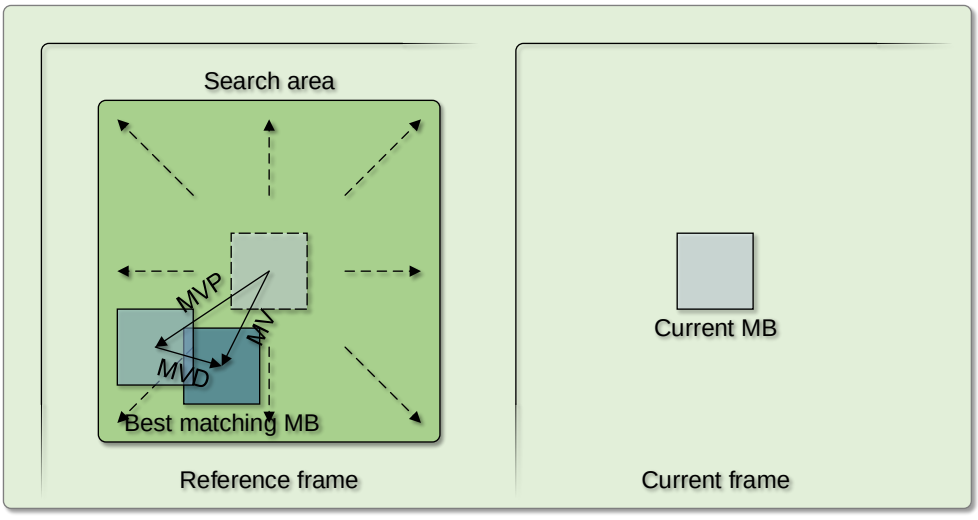


Fig. 4-5 Relationship between MV, MVP and MVD.

Based on the above analysis, it can be concluded that sequences with small motion generally tend to have small MVDs, and vice versa. Therefore, the MVD can be chosen as a measure of video motion activity. Using the MVD also satisfies the main objective of reducing the computational complexity, as the MVD is easy to extract from the coded data.

4.2.2 Algorithm Description

Macroblocks with small MVD are apt to be coded using SKIP_MODE, INTER_16×16 or other large block modes rather than by using subblocks. SKIP_MODE is normally assigned to a macroblock that comprises almost identical pixel information to that of the corresponding macroblock in the same position in the reference frame. The INTER_16×16 mode usually means that a well matching macroblock can be found in the reference frame and there is a small residual component. Consequently, the proposed fast mode decision algorithm is defined as follows:

1. Perform motion estimation between key frames within a GOP in the base layer.
2. Extract the MVD from the P-frames. When a new GOP is processed, update the MVD value.
3. From the MVD, decide whether the macroblock contains significant motion.
4. If the macroblock contains moving features, conduct inter-layer prediction, intra-prediction and subblock prediction modes. Otherwise perform inter-frame prediction in the same spatial layer without subblock modes.
5. Calculate RD cost, and decide the optimal final mode.

A flowchart of the overall process of the proposed algorithm is shown in Fig. 4-6.

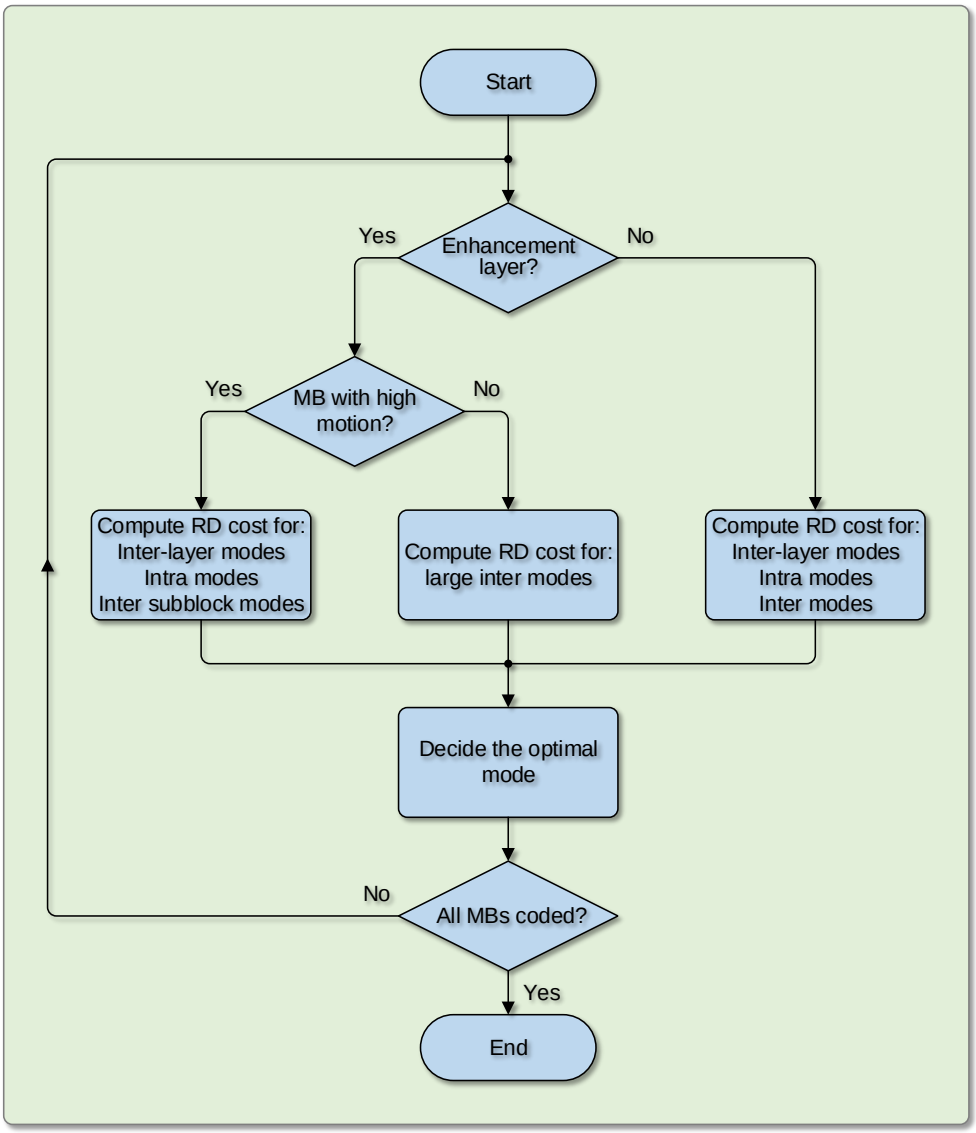


Fig. 4-6 Overall flowchart of the proposed algorithm.

Table 4-1
% Percentage of SKIP_MODE decisions made for different P-frame MVD values

Sequence	MVD (pixels)							
	1/4	1/2	3/4	1	5/4	3/2	7/4	2
Bus	57.01	15.00	4.02	2.72	2.44	1.84	2.19	1.72
Foreman	69.91	11.70	3.83	2.72	2.25	1.50	1.23	1.23
Mobile	76.97	8.80	3.23	1.80	2.17	1.49	1.17	0.87
Mother-daughter	75.11	13.51	4.05	2.28	1.20	0.74	0.47	0.45

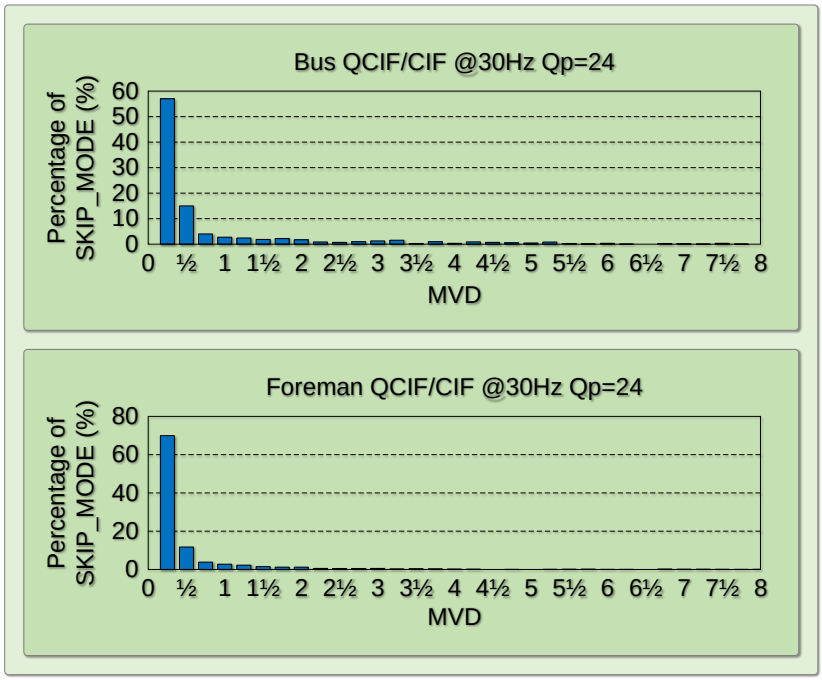


Fig. 4-7 Relationship between P-frame MVD values and the percentage of SKIP_MODE decisions when an exhaustive evaluation is conducted (continued on next page).

Table 4-1 shows the percentage of SKIP_MODE decisions of B-frames made following full RDO in a conventional SVC encoder, for different P-frame MVD values. Note that motion estimation is performed to quarter pixel precision. From observing a large number of video sequences, it is found that when the MVD of a macroblock in a P-frame is below 1 pixel, there is a 57%-75% probability that the corresponding macroblock located in the same position of the B-frame is coded as a SKIP_MODE macroblock, as shown in Fig. 4-7.

This means that a best matching macroblock can be found in the reference frame by temporal prediction. Consequently, a MVD of 1 pixel was chosen as the decision threshold.

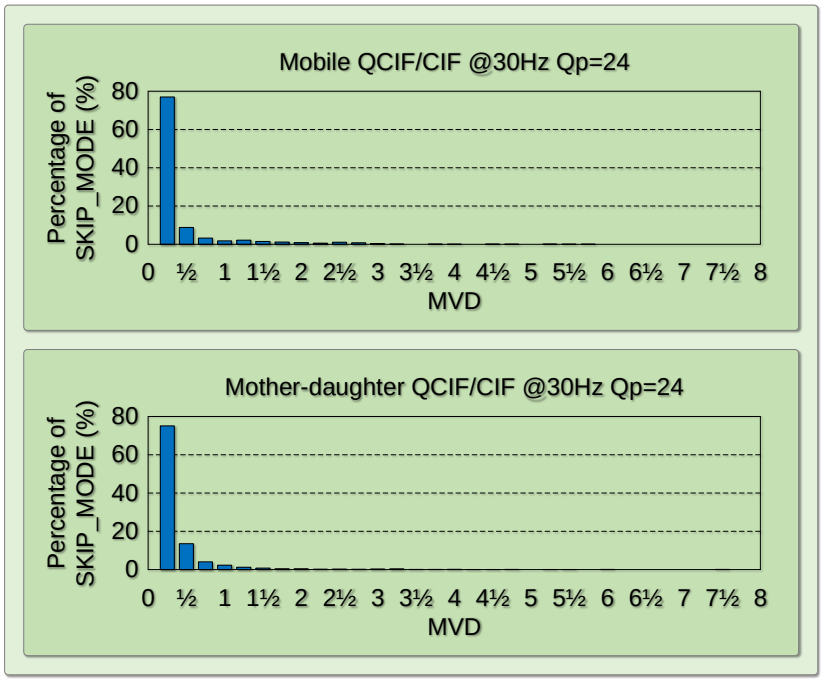


Fig. 4-7 Relationship between P-frame MVD values and the percentage of SKIP_MODE decisions when an exhaustive evaluation is conducted (continued from last page).

4.3 Simulations, Comparisons, and Discussion

The proposed algorithm was implemented and evaluated using the JSVM 9.18 software [81]. Four standard video test sequences with diverse motion content were processed, with Qp values ranging from 24 to 40. The GOP size for a hierarchical B-frame structure was set to 8, and 150 frames were coded to generate a statistically significant result. Only the two layer case was considered. The same Qp was used for both base layer and enhancement layer. RDO was always enabled, and CABAC entropy coding was used. Exhaustive search motion estimation was employed with a search range of ±32 pixels and quarter pixel precision.

In order to perform fair comparisons with the standard JSVM implementation, the following parameters were measured to evaluate the coding performance.

1. Encoding Time Reduction (TR, %), which is calculated as

$$\mathrm{TR} = \left(\mathrm{T}_{\mathrm{JSVM}} - \mathrm{T}_{\mathrm{Proposed}}\right)/\mathrm{T}_{\mathrm{JSVM}} \times 100\% \tag{4.5}$$

where $\mathrm{T}_{\mathrm{JSVM}}$ and $\mathrm{T}_{\mathrm{Proposed}}$ denote the encoding time of the JSVM reference software and the proposed algorithm, respectively;

2. ΔPSNR (dB) is computed according to

$$\Delta\mathrm{PSNR} = \mathrm{PSNR}_{\mathrm{Proposed}} - \mathrm{PSNR}_{\mathrm{JSVM}} \tag{4.6}$$

where $\mathrm{PSNR}_{\mathrm{Proposed}}$ and $\mathrm{PSNR}_{\mathrm{JSVM}}$ denote the PSNR resulting from the proposed algorithm and the JSVM reference software, and ΔBit Rate (BR, %) is computed as

$$\Delta\mathrm{BR} = (\mathrm{BR}_{\mathrm{Proposed}} - \mathrm{BR}_{\mathrm{JSVM}})/\mathrm{BR}_{\mathrm{JSVM}} \times 100\% \tag{4.7}$$

where $\mathrm{BR}_{\mathrm{Proposed}}$ and $\mathrm{BR}_{\mathrm{JSVM}}$ denote the bit rate resulting from the proposed algorithm and the JSVM reference software, respectively;

3. The Bjϕntegaard PSNR (BDPSNR, dB) and Bjϕntegaard Bit Rate (BDBR, %) as defined in [96].

The simulation results are described in the next two subsections. The first subsection demonstrates the results of the proposed fast algorithm with various Qp settings. This is followed by an overall comparison with the JSVM implementation.

4.3.1 Simulation Results for Various Values of Qp

Tables 4-2 to 4-5 show the performance of the encoder incorporating the proposed algorithm compared with the JSVM 9.18 conventional encoder for various Qp values, i.e. Qp= 24, 28, 32, 36, 40.

General trends are identified as follows: the time reduction is affected by the content of

the video test sequences. A greater time reduction is obtained for the sequence which contains slow motion, whereas a relatively smaller time reduction is obtained for the fast moving sequences. This is due to the fact that as the motion activity decreases, SKIP_MODE is used more frequently and fewer other modes are considered in the RDO. Consequently, the time reduction obtained increases. For each individual test sequence, a similar time reduction is obtained regardless of the Qp value chosen.

Table 4-2
Computational performance for 'Bus' sequence

Qp	JSVM		Proposed		TR (%)
	Bit rate (kbits/s)	PSNR (dB)	Bit rate (kbits/s)	PSNR (dB)	
40	276.97	26.75	276.99	26.74	21.69
36	464.71	29.33	465.43	29.31	20.64
32	793.41	32.08	796.92	32.06	20.32
28	1375.16	35.09	1381.17	35.07	21.29
24	2323.40	38.05	2332.50	38.03	20.95

Table 4-3
Computational performance for 'Foreman' sequence

Qp	JSVM		Proposed		TR (%)
	Bit rate (kbits/s)	PSNR (dB)	Bit rate (kbits/s)	PSNR (dB)	
40	113.21	30.04	113.17	30.03	27.97
36	177.61	32.44	177.87	32.43	26.22
32	287.83	34.68	288.58	34.67	25.91
28	487.16	37.11	489.93	37.10	25.37
24	865.34	39.45	869.94	39.42	26.50

The lowest time reduction of approximately 20% is shown in Table 4-2. The 'Bus' sequence comprises many macroblocks that contain fast motion and high spatial detail, and as a result the approximation signal predicted from the upsampled lower layer reconstruction or intra-prediction is used. In contrast, the 'Mother-daughter' sequence comprises large areas of static background and slow illumination changes, and as a result the encoding time is reduced by up to 41% (Table 4-5).

Table 4-4
Computational performance for 'Mobile' sequence

Qp	JSVM		Proposed		TR (%)
	Bit rate (kbits/s)	PSNR (dB)	Bit rate (kbits/s)	PSNR (dB)	
40	231.50	25.58	231.25	25.57	31.48
36	406.19	28.32	405.58	28.31	32.12
32	851.50	31.25	850.69	31.23	32.63
28	1788.72	34.51	1789.57	34.48	32.39
24	3299.58	37.73	3305.31	37.69	33.29

Table 4-5
Computational performance for 'Mother-daughter' sequence

Qp	JSVM		Proposed		TR (%)
	Bit rate (kbits/s)	PSNR (dB)	Bit rate (kbits/s)	PSNR (dB)	
40	32.98	31.79	32.98	31.79	40.64
36	56.45	34.19	56.44	34.19	38.65
32	97.68	36.64	97.73	36.63	36.92
28	168.06	39.33	168.09	39.33	36.39
24	300.69	41.66	300.36	41.66	35.82

Fig. 4-8 shows PSNR-bit rate relationship diagrams for the 'Bus', 'Foreman', 'Mobile', and 'Mother-daughter' sequences. Points marked '$\times$' represent the performance of the JSVM 9.18 benchmark and those marked '$\odot$' represent the proposed fast algorithm. In each of the PSNR-bit rate relationship diagrams, the RD curves of the JSVM encoder and the proposed algorithm appear to overlap exactly. There is marginal deviation in rate distortion from that of the JSVM encoder, therefore very similar coding efficiency is achieved. Thus, it can be concluded that in all cases, encoding time is reduced significantly, yet there is negligible degradation in PSNR and insignificant increment in bit rate.

4.3.2 RD Comparison with the JSVM Implementation

An overall performance comparison of the proposed algorithm with the JSVM implementation is detailed in this subsection. Comparisons are given for RD performance in terms

of ΔPSNR (dB), ΔBR (%), BDPSNR (dB), BDBR (%) and encoding TR (%).

The overall performance comparison with the standard JSVM encoder is summarised in Table 4-6. It shows that the time reduction achieved by the proposed scheme is between 21% and 38% on average, at a cost of less than 0.02dB decrease in PSNR and by no more than a 0.26% increase in bit rate. It is widely accepted that human perception cannot recognise a PSNR difference of less than 0.2dB. Thus, the proposed algorithm results in a significant reduction in computational complexity with negligible effect on rate distortion.

Table 4-7 presents the simulation results of the standard JSVM 9.18 implementation and the proposed algorithm for four standard test sequences and Qp values ranging from 28 to 40. The table shows the overall average results of BDPSNR=-0.02dB and BDBR=0.37%. This indicates that 1) with the same bit rate, the proposed scheme decreases the average PSNR by 0.02dB, and 2) with the same video quality (PSNR), the proposed algorithm increases the bit rate by 0.37% compared with the original JSVM implementation. Therefore, it can be concluded that the proposed algorithm reduces the encoding time by 29% on average, while having a negligible impact on rate distortion, with PSNR losses of 0.01dB-0.03dB and increases in bit rate of between 0.11% and 0.66%.

Table 4-6
Overall comparison of proposed algorithm and JSVM implementation

Sequence	Performance	Qp				Average
		40	36	32	28	
Bus	ΔPSNR	-0.01	-0.02	-0.02	-0.02	**-0.02**
	ΔBR	0.01	0.15	0.44	0.44	**0.26**
	TR	21.69	20.64	20.32	21.29	**20.99**
Foreman	ΔPSNR	-0.01	-0.01	-0.01	-0.01	**-0.01**
	ΔBR	-0.04	0.15	0.26	0.57	**0.24**
	TR	27.97	26.22	25.91	25.37	**26.37**
Mobile	ΔPSNR	-0.01	-0.01	-0.02	-0.03	**-0.02**
	ΔBR	-0.11	-0.15	-0.10	0.05	**-0.08**
	TR	31.48	32.12	32.63	32.39	**32.16**
Mother-daughter	ΔPSNR	0.00	0.00	-0.01	0.00	**0.00**
	ΔBR	0.00	-0.02	0.05	0.02	**0.01**
	TR	40.64	38.65	36.92	36.39	**38.15**

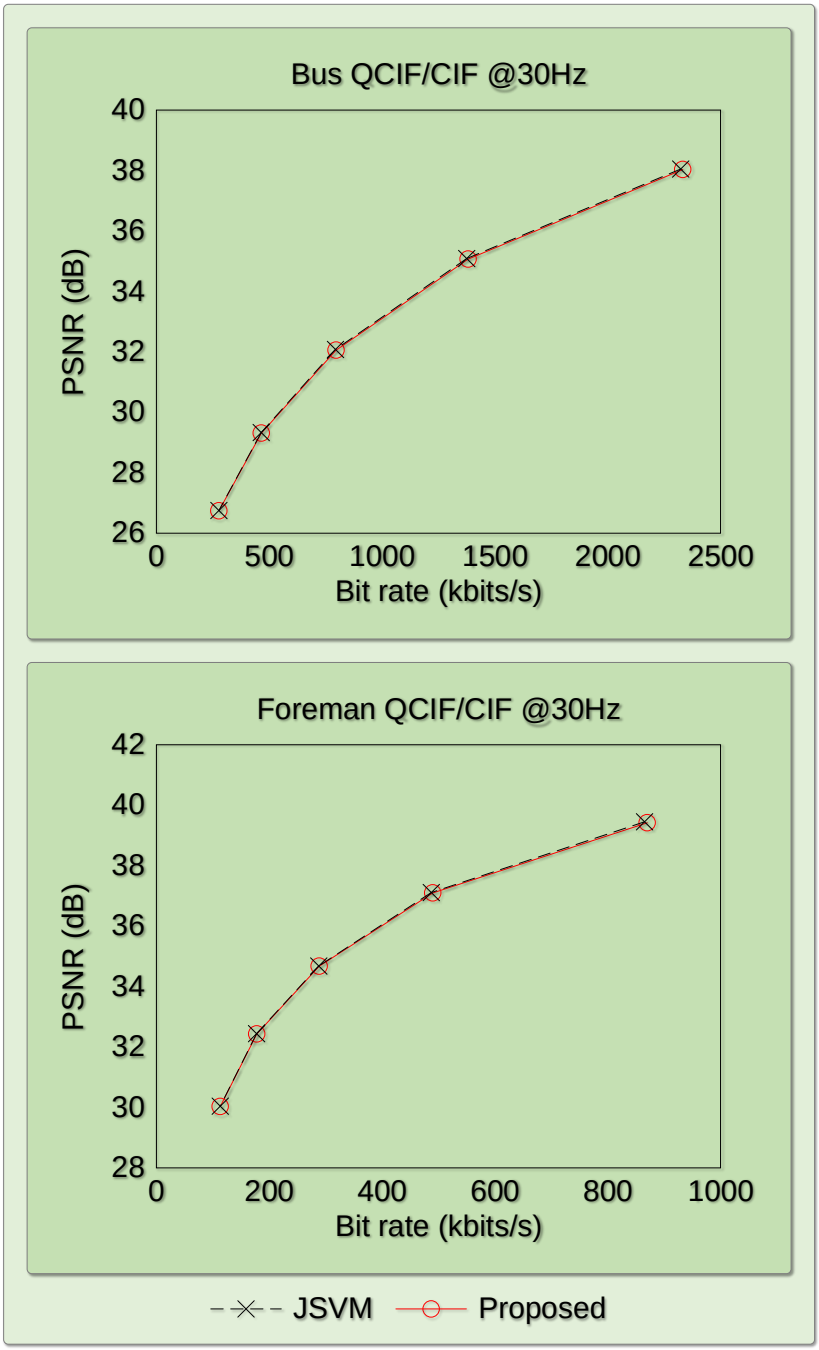


Fig. 4-8 RD performance comparison of JSVM and the proposed algorithm (continued on next page).

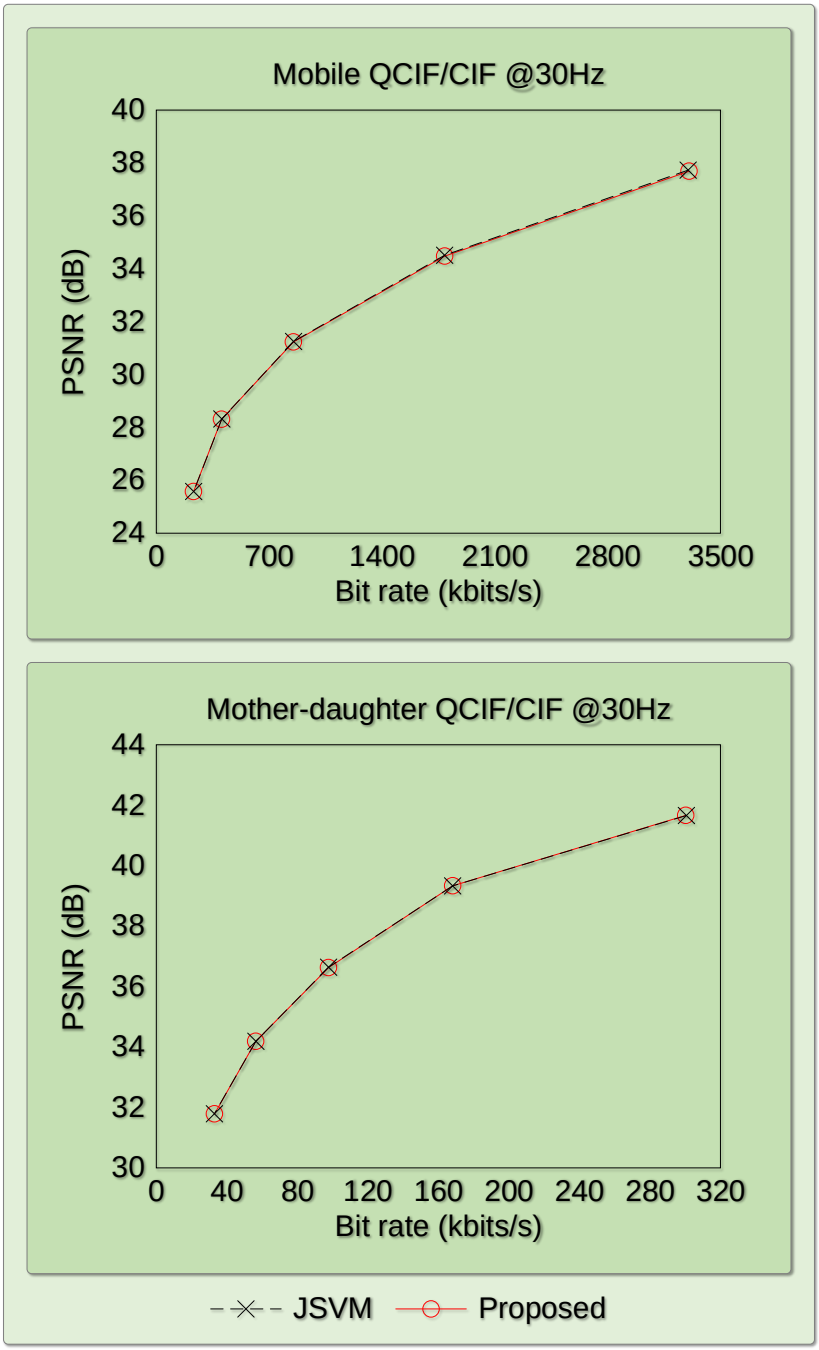


Fig. 4-8 RD performance comparison of JSVM and the proposed algorithm (continued from last page).

Table 4-7
Overall performance when encoding QCIF/CIF sequences

Sequence	Qp	JSVM		Proposed		TR	BDPSNR	BDBR
		Bit rate	PSNR	Bit rate	PSNR			
Bus	40	276.97	26.75	276.99	26.74	20.99	-0.03	0.66
	36	464.71	29.33	465.43	29.31			
	32	793.41	32.08	796.92	32.06			
	28	1375.16	35.09	1381.17	35.07			
Foreman	40	113.21	30.04	113.17	30.03	26.37	-0.02	0.43
	36	177.61	32.44	177.87	32.43			
	32	287.83	34.68	288.58	34.67			
	28	487.16	37.11	489.93	37.10			
Mobile	40	231.50	25.58	231.25	25.57	32.16	-0.01	0.29
	36	406.19	28.32	405.58	28.31			
	32	851.50	31.25	850.69	31.23			
	28	1788.72	34.51	1789.57	34.48			
Mother-daughter	40	32.98	31.79	32.98	31.79	38.15	-0.01	0.11
	36	56.45	34.19	56.44	34.19			
	32	97.68	36.64	97.73	36.63			
	28	168.06	39.33	168.09	39.33			
Average						**29.42**	**-0.02**	**0.37**

4.4 Summary

In this chapter, one of the obstacles that SVC is required to overcome is identified. Due to the many time-consuming encoding tools of SVC, for instance, the conventional intra- and inter-frame prediction, inter-layer prediction, bi-directional motion prediction, quarter pixel precision motion estimation, and motion compensation using multiple reference blocks, the computational requirement of the SVC encoder is far more significant than any existing video coding algorithms. If SVC is to become widely adopted, the computational complexity of the SVC encoder needs to be significantly reduced.

Several studies have been proposed for efficient implementation of intra- and inter-frame coding, and inter-layer coding. In particular, various pixel domain and frequency

domain features, dependency between spatial and temporal neighbouring blocks, and correlation between adjacent layers have been widely studied. Some of the representative fast mode decision algorithms were briefly described.

Unlike the algorithms detailed in the literature, a fast algorithm is proposed that employs MVD as a measure of motion activity to determine the subset of mode candidates to be evaluated. The proposed algorithm is based on the priori knowledge that a macroblock with slow movement is more likely to be best matched by one in the same resolution layer and a block matching process between layers is required for a macroblock with fast movement. Fewer selected candidate modes are required to undergo time-consuming Lagrangian RDO.

Performance comparison of the proposed algorithm with the conventional mode selection method was presented in section 4.3. The proposed algorithm reduces the encoding time, while maintaining high coding efficiency. Evaluation results show that the proposed algorithm achieves up to 40% reduction in coding time with negligible degradation of picture quality and bit rate.

In the proposed algorithm, as MVD was the only parameter used to decide which block modes should be examined, only a modest time reduction was achieved, given that picture quality had to be maintained. As the MVD threshold required for categorisation of the video motion activity is determined empirically, some attempts need to be made to optimise the MVD threshold. This will be discussed in detail in the next chapter.

Chapter 5

Hierarchical Scheme for Fast Mode Decisions

A fast inter-frame and inter-layer mode decision algorithm was discussed in the previous chapter. It was shown to produce a modest reduction in computation time and it also demonstrated the potential to achieve more effective and robust performance. This chapter extends the method proposed in the last chapter, and discusses an efficient implementation of fast coding for SVC.

Unlike existing solutions, the proposed algorithm not only considers inter-layer correlation but also fully exploits both spatial and temporal correlation as well as macroblock texture. All of these factors are organised within a hierarchical structure in the mode decision process. At each level of the structure, different strategies are implemented to eliminate inappropriate candidate modes. Simulation results show that the proposed algorithm reduces encoding time by up to 84% compared with the JSVM 9.18 implementation. This is achieved without any noticeable degradation in RD performance.

The remainder of this chapter is organised as follows. The next section presents the motivation for the proposed algorithm. Section 5.2 discusses the formulation of the proposed algorithm, and the overall structure is also presented in the same section. Extensive

experimental results are presented in section 5.3 and conclusions are given in section 5.4.

5.1 Introduction

In chapter 4, a fast inter-frame and inter-layer mode decision algorithm for the enhancement layer of SVC was discussed. In that algorithm, the average Mean Absolute Difference (MVD) extracted from the key frames in the base layer was chosen as a measure of motion activity. The average MVD is used as a rough estimate of the motion activity between key frames, and as this was the only parameter used to decide which block modes should be examined, only a modest time reduction was achieved, given that picture quality had to be maintained. Due to the effectiveness of the MVD in categorising motion activity, it continues to be used as one of the bases for the proposed algorithm, and is incorporated in the composite fast encoding process. However, in this chapter, MVD is extracted from the co-located macroblock in the base layer, and it is more accurate than that proposed in chapter 4. In addition, a constant MVD threshold was used in the previously proposed approach and it was determined empirically. This chapter optimises the MVD threshold value. When determining the optimal threshold, both the coding efficiency in terms of RD performance and the computational complexity reduction are taken into account.

In [89], the energy of the transformed AC coefficients was used as a measure of a macroblock's spatial complexity to reduce the choice of candidate modes. AC energy is shown to be a good measure by which to classify the homogeneity of a macroblock, and is therefore chosen as a factor to be considered in the mode decision process. Unlike the fixed empirical threshold used in [89], the AC energy threshold in the proposed algorithm is determined from consideration of a large number of training samples and many different types of picture content. Consequently it is more reliable and widely-applicable.

Another consideration is that the encoding results of the base layer can be used to inform the coding of the enhancement layers. In other words, the time of the mode decision

process in the enhancement layer can be reduced significantly by exploiting the available information in the base layer.

The information from both the base layer (partition mode, MVD of co-located macroblock) and the enhancement layer (mode of spatially neighbouring macroblock, texture measurement) are used together in the proposed algorithm. The factors are organised as a hierarchical structure comprising four levels, and different criteria are designed for each level making the proposed method effective and robust. Consequently, the proposed algorithm produces superior results in terms of the computational requirement, regardless of the video content and the coding conditions.

5.2 The Proposed Hierarchical Mode Decision Scheme

The proposed algorithm uses information from both the base layer and the enhancement layers together. In particular, the partition mode and MVD of the co-located macroblock in the base layer are used to refine the list of candidate modes, while the mode of the spatially neighbouring macroblock and the texture measurement of the current macroblock are used to further eliminate unlikely modes. The following subsections introduce a detailed formulation of each level in the hierarchical structure.

5.2.1 Observations, Analysis, and Algorithm Formulation

The following general characteristics form the basis for the proposed fast encoding algorithm. 1) Strong mode dependencies exist between base layer and enhancement layer, and also between a macroblock and its neighbours. 2) Larger partition sizes are more suitable for homogeneous regions, and smaller partition sizes are more beneficial for detailed areas. 3) Regions that contain slow motion and high spatial detail are apt to have the best matching mode involve inter-frame prediction. Each of these characteristics will be examined in the following sections.

Mode Correlation Between Base Layer and Enhancement Layer

In SVC, the base layer is first encoded independently, followed by the enhancement layers. As the input video of the different layers is generated from the same original video source but at different spatial resolutions, the picture content is highly correlated [97]. Specifically, the prediction mode, reference picture indices, and MVs of the enhancement layers are strongly correlated with those of the base layer. Consequently, the MV and block mode can be inferred from the data in the lower spatial layer. In practice, the scaled MVs from the lower layer and the refined macroblock partition mode can be used as predictors. Consequently, by exploiting the mode information of the co-located macroblock in the base layer, the computational complexity of the mode decision in the enhancement layer can be reduced significantly.

To illustrate the mode relationship between the base layer and the enhancement layers and to justify the proposed algorithm, the probability of macroblocks in the enhancement layer being encoded as MODE_SKIP when the mode of the co-located macroblock in the base layer is also MODE_SKIP is analysed. Of all macroblock modes, MODE_SKIP is the most efficient in terms of computational complexity. The second most efficient is the larger block-MODE_16×16. Provided the candidate modes can be narrowed down to MODE_SKIP and MODE_16×16 in advance using inter-layer and neighbouring macroblock correlation information, the motion estimation cost will be reduced significantly. As natural real-world video sequences comprise large areas of homogeneous background or regions with little motion activity, MODE_SKIP is dominant amongst the prediction modes selected. If MODE_SKIP is predicted early enough, a large amount of processing can be saved.

Table 5-1 shows the skip mode correlation between the base layer and the enhancement layer as defined in equation (5.1). Four video sequences with different degrees of activity and detail were examined. The statistics were collected from the first 90 frames of each video sequence. QCIF sequences were used for the base layer and CIF was used for

the enhancement layer.

$$MC_{\mathrm{IL}} = \frac{MB_{\mathrm{B\&E\_SKIP}}}{MB_{\mathrm{E\_SKIP}}} \times 100\% \tag{5.1}$$

where $MB_{\mathrm{B\&E\_SKIP}}$ is the number of macroblocks predicted as MODE_SKIP in the base layer when the corresponding macroblock in the enhancement layer is MODE_SKIP too. $MB_{\mathrm{E\_SKIP}}$ is the number of MODE_SKIP macroblocks in the enhancement layer. MC_{IL} is the mode correlation between the coding layer and reference layers.

Table 5-1
% Mode correlation between base layer and corresponding enhancement layer

Sequence	Qp				
	24	28	32	36	40
Bus	40.09	47.30	53.16	58.61	65.71
Foreman	42.90	53.97	61.04	69.14	79.94
Mobile	49.92	57.70	63.57	65.44	70.35
Mother-daughter	78.81	85.40	90.37	95.03	97.64

From Table 5-1, it can be deduced that if the best mode for the macroblock in the base layer is MODE_SKIP, the corresponding macroblock mode in the enhancement layer is very likely to be MODE_SKIP as well. This is largely true regardless of the video sequence examined.

Mode Correlation Between Macroblock and Its Neighbours

As well as the correlation between layers, there is also a significant dependency between neighbouring macroblocks in the enhancement layer. For a majority of video sequences, MODE_SKIP macroblocks tend to occur in clusters, such as a patch of static background. Consequently there is a high possibility that the best mode for the current macroblock is similar to that of its neighbours, that is, coded macroblocks which are located immediately above and to the left of the current macroblock.

In order to reveal the mode dependency between macroblocks, the probability that

the current macroblock is coded as MODE_SKIP, if one or both of the neighbouring macroblocks are also coded MODE_SKIP is measured.

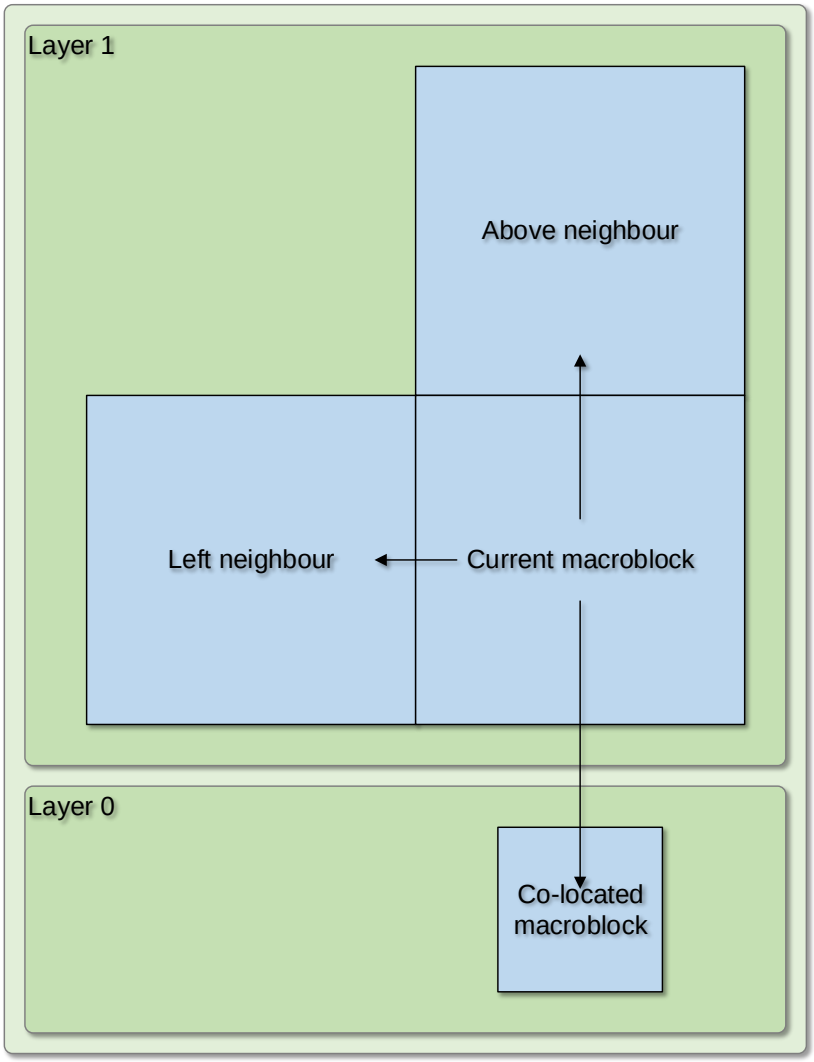


Fig. 5-1 Spatial locations of neighbouring macroblocks in the same layer and co-located macroblock in the base layer.

Equation (5.2) denotes the mode correlation between the current macroblock and its neighbours.

$$MC_{SN} = \frac{MB_{C\&N\_SKIP}}{MB_{C\_SKIP}} \times 100\% \tag{5.2}$$

where $MB_{\text{C\&N\_SKIP}}$ is the number of macroblocks which are predicted MODE_SKIP when at least one of the macroblock's neighbours in the enhancement layer is MODE_SKIP as well. $MB_{\text{C\_SKIP}}$ is the number of MODE_SKIP macroblocks in the enhancement layer. MC_{SN} is the mode correlation between a macroblock and its spatial neighbours.

Table 5-2 shows the mode correlation between a macroblock and its neighbours as defined in equation (5.2). The same coding conditions as in the last subsubsection are applied.

Table 5-2
% Mode correlation between macroblock and its neighbours

Sequence	Qp				
	24	28	32	36	40
Bus	45.24	52.18	56.95	62.52	66.88
Foreman	45.00	55.24	61.93	69.26	77.53
Mobile	46.06	52.43	57.78	62.16	66.27
Mother-daughter	71.73	82.05	87.07	92.86	96.16

From the observations above, it can be inferred that if the best mode for the co-located macroblock in the base layer or the neighbouring macroblocks in the enhancement layer is MODE_SKIP, there is a high probability that the current macroblock in the enhancement layer is also MODE_SKIP, because of the strong dependency that exists. The spatial locations of macroblocks in which information is reused are illustrated in Fig. 5-1.

DCT Coefficients and Picture Content

The energy distribution property of the DCT coefficients is employed to evaluate the homogeneity of a macroblock. In a smooth region of an image, the DCT energy generally tends to be concentrated in the low frequency components, whereas in a block comprising high detail, the frequency domain energy is distributed more in the AC coefficients.

Given the energy conservation principle, for a 16×16 macroblock, $f(x, y)$, the energy

of the AC coefficients can be calculated as

$$E_{\mathrm{AC}} = \mathbf{E}\left[f^2(x,y)\right] - \mathbf{E}\left[f(x,y)\right]^2 \tag{5.3}$$

where **E** is the expectation. When the total energy of the AC coefficients in a macroblock is greater than a predefined threshold, the macroblock is best categorised as containing high spatial detail, and for this to be considered when choosing a reduced subset of modes for evaluation. An appropriate AC energy threshold can speed up the mode selection algorithm to the maximum extent while still maintaining the reconstructed picture quality. A rule for choosing a proper AC threshold can be deduced as in equation (5.4).

$$J(Thr) = PA(Thr) + TR(PA|Thr) \tag{5.4}$$

where TR represents the computational time reduction. Thr is the AC threshold parameter and PA denotes the degree of prediction accuracy, defined as

$$PA(Thr) = \frac{\sum_{i=0}^{T-1}\left(P_{\mathrm{mode}}(i) \bigcap F_{\mathrm{mode}}(i)\right)}{T} \times 100\% \tag{5.5}$$

in which $P_{\mathrm{mode}}(i)$ represents the best mode of the reduced subset of prediction modes, and $F_{\mathrm{mode}}(i)$ is the prediction mode chosen if an exhaustive evaluation is conducted. T is the total number of macroblocks for which a reduced mode evaluation is performed. When the decision made by a partial evaluation is identical to that of the full evaluation, $P_{\mathrm{mode}}(i) \bigcap F_{\mathrm{mode}}(i)$ equals 1, otherwise it is zero. Thus equation (5.5) represents the proportion of the same prediction decisions made by a partial evaluation as that of a full evaluation.

The goal is to find the AC threshold value that maximises the sum of the prediction accuracy and the computational time reduction, while maintaining the prediction accuracy constraint that $PA \geq PA_{\mathrm{MIN}}$. In order to find the most desirable AC energy threshold,

equation (5.4) leads to the optimisation problem of equation (5.6).

$$\arg\max_{(Thr)} J = PA(Thr) + TR(PA|Thr) \tag{5.6}$$

For which the solution is given as

$$\begin{aligned} \frac{\partial PA(Thr)}{\partial (Thr)} + \frac{\partial TR(PA|Thr)}{\partial (Thr)} &= 0 \\ \frac{\partial PA(Thr)}{\partial (Thr)} &= -\frac{\partial TR(PA|Thr)}{\partial (Thr)} \end{aligned} \tag{5.7}$$

From equation (5.7), the optimum AC threshold is that at which the slope of the prediction accuracy is equal to the inverse of the slope of the proportion of decisions satisfied by a reduced set evaluation. A threshold value of less than optimal makes the prediction accuracy increase, however the computational cost is increased accordingly. On the other hand, a threshold value larger than optimal reduces the computational cost, but the picture quality is degraded.

In order to obtain consistent performance on a wide range of video sequences with varying picture detail and motion activity, the statistical data that collected from processing four video sequences: 'Bus', 'Foreman', 'Mobile' and 'Mother-daughter' was averaged. The first 80 frames of each sequence were processed, and QCIF sequences were used for the base layer and CIF were used for the enhancement layer. Thus, a large number of training samples (4×80×22×18=126720) were used to obtain the optimal threshold. Furthermore, the video sequences contained widely different texture content, resulting in a reliable and widely-applicable threshold value.

Table 5-3 shows the relationship between AC energy threshold and both prediction accuracy and computational time reduction. The corresponding relationship curves are shown in Fig. 5-2.

From the composite results using the four video test sequences, an AC energy threshold of 125000 was chosen.

Table 5-3
% Prediction accuracy and time reduction corresponding to different AC energy thresholds

AC threshold	50000	75000	100000	150000	200000
PA	87.38	84.69	83.95	81.56	80.66
TR	36.47	42.00	46.17	53.33	59.26

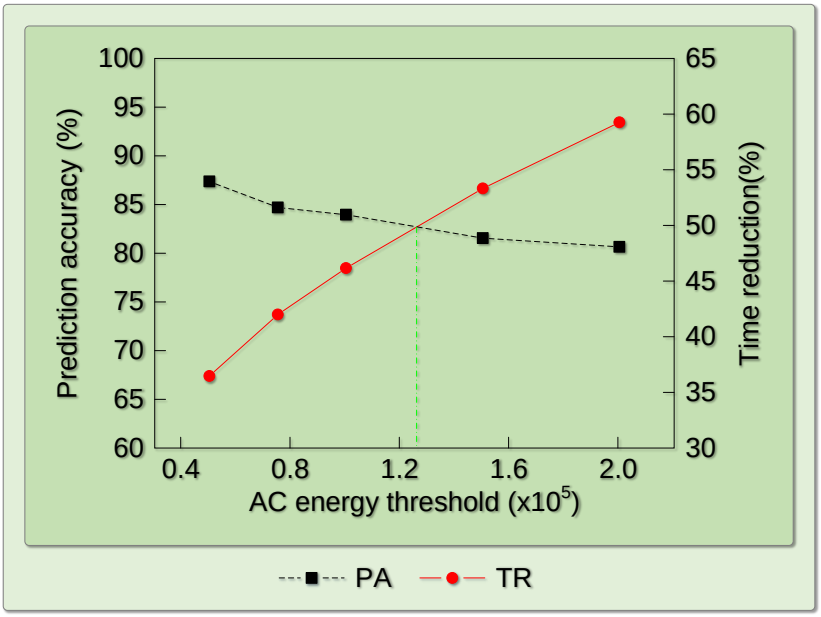


Fig. 5-2 Relationship between AC energy threshold and both prediction accuracy and computational time reduction.

Detection of Motion Activity

As mentioned in chapter 4, not all lower layer upsampling data is suitable for inter-layer prediction, especially when the video sequence contains slow motion and high spatial detail. Under such conditions, more precise prediction is generally obtained by inter-frame prediction. Therefore it is desirable to identify the amount of motion in the sequence as well as the spatial detail.

MVD from the co-located macroblock in the base layer is chosen as the measure of motion activity, as defined in equation (4.4).

In SVC, each 4×4 block is assigned one MV. For a 16×16 macroblock, the mean sum of the squares of 16 MVs is calculated as follows,

$$\mathrm{MVD}_{\mathrm{sum}} = \sum_{i=0}^{15} \frac{\sqrt{\mathrm{MVD}_x^2(i) + \mathrm{MVD}_y^2(i)}}{16} \tag{5.8}$$

where $\mathrm{MVD}_{\mathrm{sum}}$ represents the overall MVD for the macroblock in the base layer. MVD_x and MVD_y denote the horizontal and vertical components of a MV. Generally, macroblocks containing little motion tend to result in small MVD values and vice versa. The MVD is easy to extract from the coded data, and this supports the goal of the algorithm, namely to reduce computational complexity.

In chapter 4, MVD was used as the basis for a fast inter-frame and inter-layer mode decision algorithm. As MVD was the only parameter used to decide which block modes should be examined, only a modest time reduction was achieved, given that picture quality had to be maintained. Even so, MVD was shown to be a good measure by which to categorise video motion activity.

Choice of an appropriate MVD threshold can result in a good trade-off between prediction accuracy and speed of mode decision. This plays a crucial role in the fast mode selection algorithm. The most favourable threshold should fulfill the following two requirements:

1. Exclude unnecessary mode candidates as much as possible, thus maximising the time saving;

2. Maintain a prediction accuracy as high as that of an exhaustive evaluation, thus minimising picture quality degradation.

A similar methodology to that used to determine the AC energy threshold can be employed to choose the MVD threshold.

Table 5-4 shows the relationship between MVD threshold and both prediction accuracy and computational time reduction. The corresponding relationship curves are shown in Fig. 5-3, where PA denotes the prediction accuracy and TR represents the computational

time reduction. Again, the results represent the average performance of four video test sequences comprising varying degrees of motion activity and spatial content.

Table 5-4
% Prediction accuracy and time reduction corresponding to different MVD thresholds

MVD threshold	0.1	0.2	0.5	1.0	2.0
PA	91.41	91.07	87.44	85.76	83.94
TR	54.29	55.14	68.25	75.50	81.46

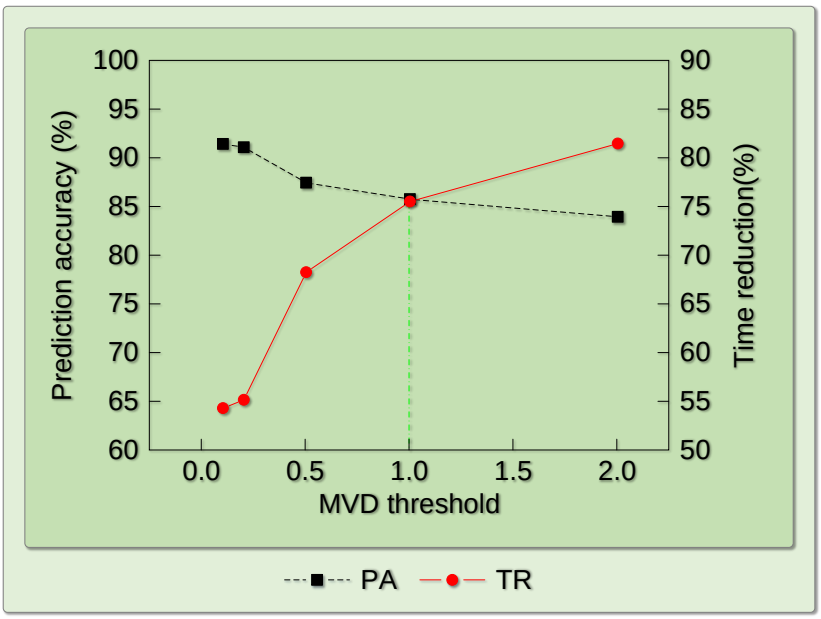


Fig. 5-3 Relationship between MVD threshold and both prediction accuracy and computational time reduction.

In accordance with the methodology explained in the last subsubsection, a MVD of 1.0 was chosen as the threshold.

5.2.2 The Structure of the Proposed Algorithm

The framework of the proposed fast mode selection algorithm is illustrated in Fig. 5-4. The basic motivation is to reduce the number of mode candidates in the enhancement layer by exploiting the information in co-located macroblocks in the base layer and in neigh-

bouring macroblocks. As the coded data of the base layer will be reused either directly or indirectly, its efficacy influences the performance of the encoder. The mode chosen for a macroblock in the base layer is estimated using an exhaustive evaluation. As to the enhancement layer, the proposed algorithm is described as follows:

1. Check the mode of the co-located macroblock in the base layer. If it is intra-coded, it means that a matching macroblock via inter-frame prediction in the reference frames could not be found. Usually, such macroblocks contain fast changing or highly detailed information. For such macroblocks, compare the RD cost of all the modes (including inter-layer prediction) and select the mode with minimum RD cost as the best mode for the current macroblock. Otherwise, proceed to step 2.

2. Check the co-located macroblock in the base layer and the neighbouring macroblocks. If at least one of these macroblocks is encoded as MODE_SKIP, evaluate the RD cost of employing either MODE_SKIP or MODE_16×16. If mode MODE_SKIP has the least RD cost, it is chosen as the best mode for the current macroblock. Otherwise, proceed to step 3.

3. Measure the homogeneity of the macroblock content. In the case of high homogeneity, i.e. $E_{AC} \leq 125000$, large block partition sizes (MODE_16×16, MODE_16×8 and MODE_8×16) require evaluation. Otherwise, proceed to step 4.

4. Discover the MVD which is easily extracted from the coded bitstream. If MVD<1.0, the macroblock contains little motion, and motion estimation can be performed with fewer candidates. Otherwise, more candidates are chosen corresponding to a larger search range.

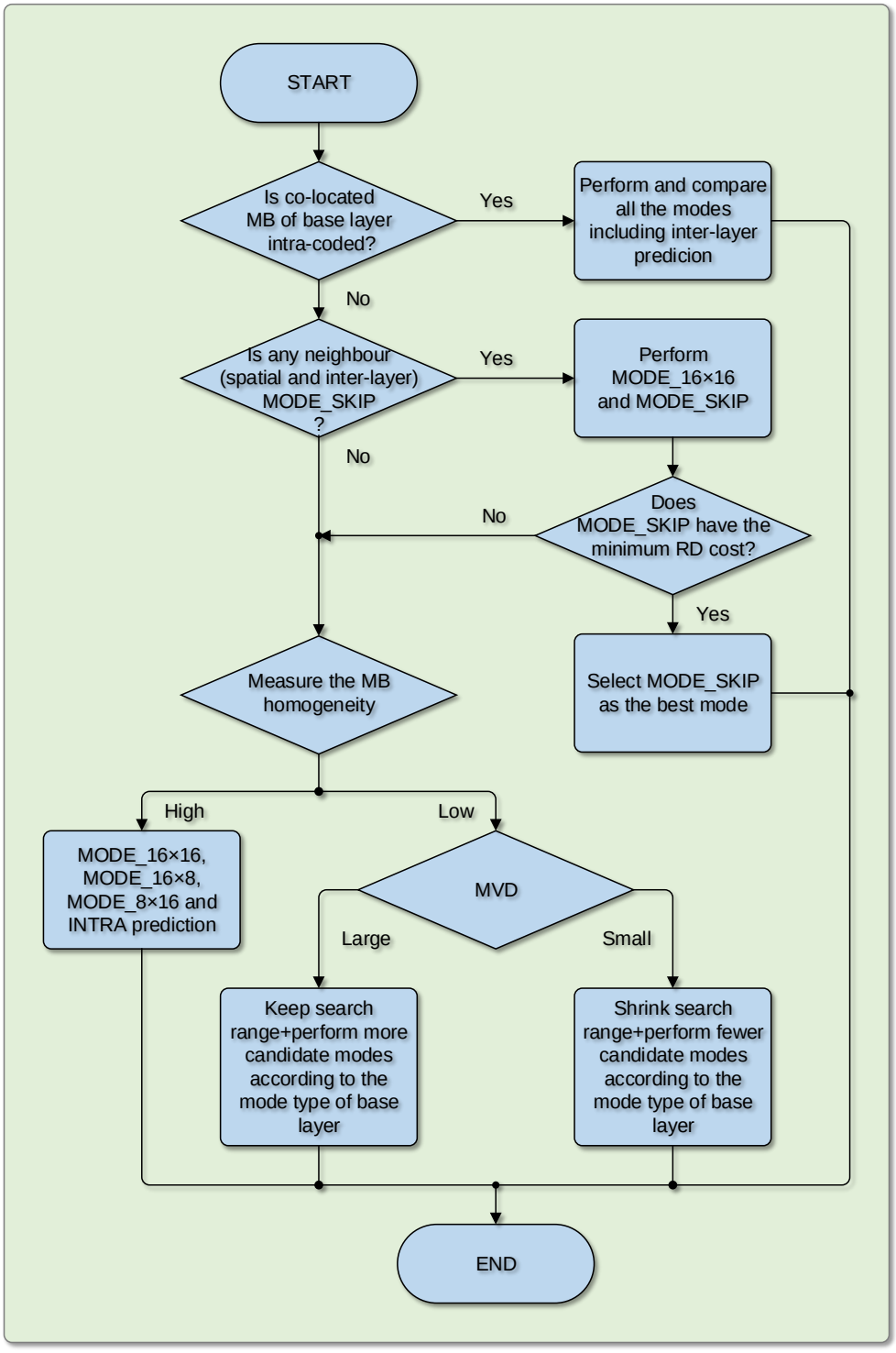


Fig. 5-4 Overall scheme of proposed hierarchical algorithm.

5.3 Simulations, Comparisons, and Discussion

The proposed algorithm was implemented using the JSVM 9.18 reference software [81]. Initially, four standard video test sequences were processed, with different Qp factors ranging from 28 to 40. RDO was always enabled, and CABAC entropy coding was used. Exhaustive search motion estimation was employed with a search range of ±32 pixels and quarter pixel precision. The GOP size for the hierarchical B-frame structure was set to 16, and 150 frames were coded to generate a reliable result. The same Qp was used for both base layer and enhancement layers.

In order to validate the effectiveness of the proposed algorithm in different scenarios, video sequences with different motion activity and picture detail were selected. The 'Bus' sequence contains a fast moving object and high spatial detail; 'Foreman' features moderately complex motion and picture detail; 'Mobile' contains complex picture detail and regular motion; 'Mother-daughter' comprises low motion activity and little detail.

In order to perform fair comparisons with the standard JSVM implementation and the fast implementations recently proposed in [90], [93] and [94], the following parameters were measured to evaluate the coding performance. 1) Time Reduction (TR, %), 2) BDPSNR (dB) and BDBR (%), 3) ΔPSNR (dB) and ΔBR (%), as defined in section 4.3 .

5.3.1 Simulation Results for Various Values of Qp

Table 5-5 shows the simulation results of the standard JSVM 9.18 implementation and the proposed algorithm for four standard test sequences and Qp values ranging from 28 to 40. It can be seen that the proposed algorithm reduces the encoding time significantly with negligible bit rate incrementation and quality loss over a wide range of bit rates. Evaluation results show that the proposed scheme reduces the encoding time by 61%-84% relative to the JSVM encoder. There is negligible impact on rate distortion, with PSNR losses of 0.02dB-0.10dB and increases in bit rate of between 0.43% and 1.96%.

Table 5-5
Performance when encoding QCIF/CIF sequences

Sequence	Qp	JSVM		Proposed		TR	BDPSNR	BDBR
		Bit rate	PSNR	Bit rate	PSNR			
Bus	40	344.28	26.93	345.58	26.89	67.19	-0.10	1.96
	36	565.78	29.44	570.21	29.39	64.46		
	32	942.03	32.15	952.50	32.09	62.39		
	28	1579.67	35.07	1594.53	34.99	61.29		
Foreman	40	157.93	30.28	157.74	30.25	73.71	-0.07	1.44
	36	246.51	32.68	246.99	32.63	69.35		
	32	390.35	34.96	392.27	34.91	65.81		
	28	635.55	37.32	643.72	37.24	62.31		
Mobile	40	469.64	25.86	470.31	25.83	71.52	-0.06	1.09
	36	752.69	28.44	754.69	28.41	70.09		
	32	1314.27	31.25	1319.59	31.21	69.20		
	28	2342.70	34.48	2359.71	34.42	68.72		
Mother-daughter	40	66.83	32.24	66.69	32.22	84.16	-0.02	0.43
	36	107.42	34.62	107.52	34.61	82.19		
	32	175.36	37.08	175.47	37.06	78.93		
	28	284.31	39.59	284.90	39.55	75.51		

Fig. 5-5 shows the encoding time of the JSVM 9.18 implementation and the proposed algorithm for four standard test sequences and Qp values ranging from 24 to 40. It is apparent that the encoding time reduction increases as the value of Qp increases. The proposed scheme performs best on video sequences containing smooth movement and low texture detail, such as 'Mother-daughter', but also shows a considerable time reduction for video sequences with complex motion and high spatial detail, such as 'Bus' and 'Mobile'. For instance, in the case of the 'Bus' sequence with fast motion activity and high spatial detail, as the percentage of MODE_SKIP macroblocks in the base layer is small, the time reduction is lower than that of the other test sequences. Even so, the computation time is reduced by over 61%. For the 'Mother-daughter' sequence comprising little motion, an average time saving of 78% is achieved. The maximum time saved is 84% for the sequence containing slow motion. The improvement in coding time reduction is achieved by discarding the least possible modes to be selected. For the low spatial detail and low motion activity

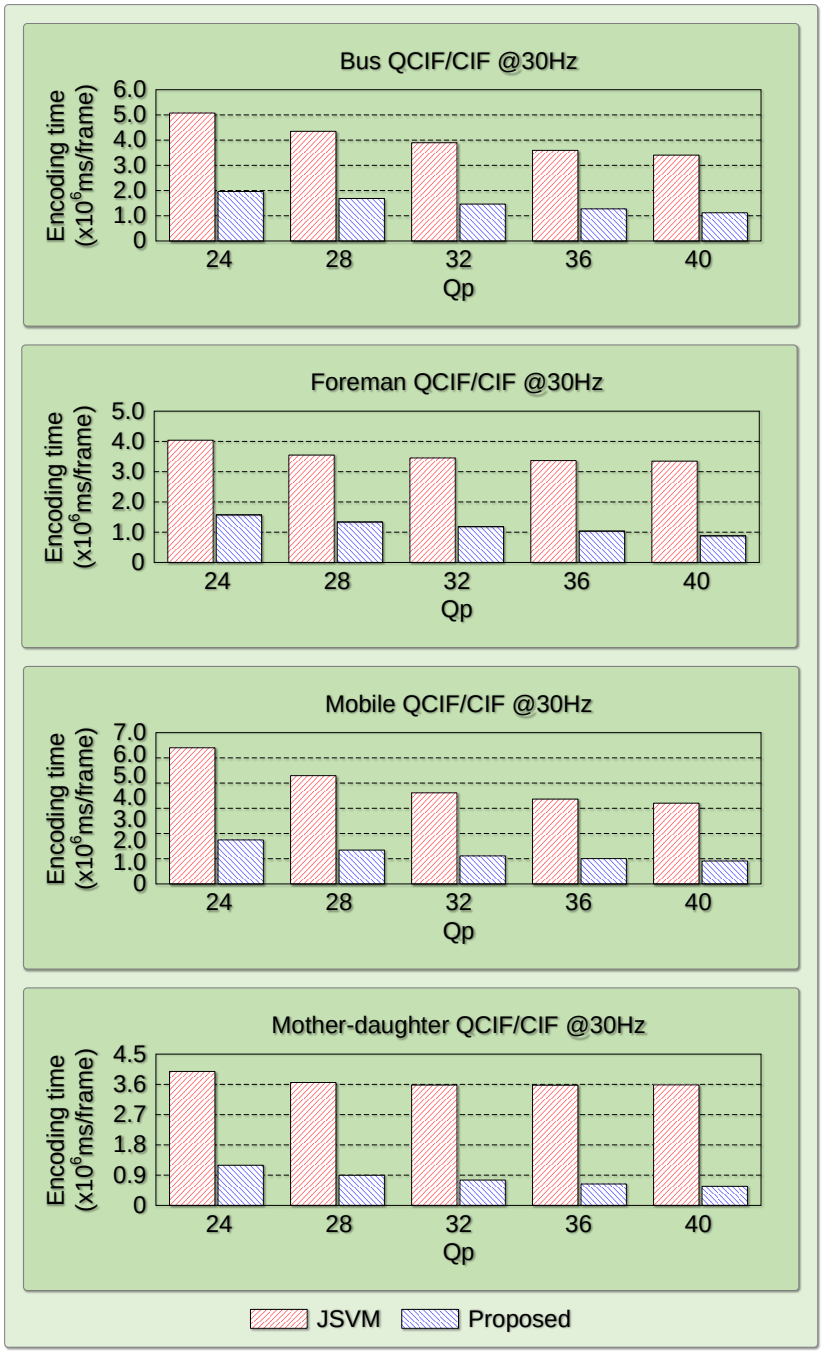


Fig. 5-5 Encoding time for video sequences for various Qp values.

sequence, 'Mother-daughter', the majority of the mode candidates in the enhancement layer are reduced to MODE_SKIP and MODE_16×16, resulting in a significant reduction in computation. While the high motion and detailed sequences require somewhat more prediction modes, the advantages become less significant.

5.3.2 Overall Comparison with the JSVM Implementation

The overall performance compared with the standard JSVM encoder is summarised in Table 5-6. It shows that the time reduction achieved by the proposed scheme is between 64% and 80% on average, at a cost of less than 0.08 dB decrease in PSNR and by no more than a 1.29% increase in bit rate. The proposed algorithm results in a significant reduction in computational complexity with negligible effect on rate distortion.

Table 5-6
Overall comparison of proposed algorithm and JSVM implementation

Sequence	Performance	Qp				Average
		40	36	32	28	
Bus	ΔPSNR	-0.04	-0.05	-0.06	-0.08	**-0.06**
	ΔBR	0.38	0.78	1.11	0.94	**0.80**
	TR	67.19	64.46	62.39	61.26	**63.83**
Foreman	ΔPSNR	-0.03	-0.05	-0.05	-0.07	**-0.05**
	ΔBR	-0.12	0.20	0.49	1.29	**0.47**
	TR	73.71	69.35	65.81	62.31	**67.80**
Mobile	ΔPSNR	-0.03	-0.04	-0.05	-0.05	**-0.04**
	ΔBR	0.14	0.27	0.40	0.73	**0.39**
	TR	71.52	70.09	69.20	68.72	**69.88**
Mother-daughter	ΔPSNR	-0.01	-0.01	-0.02	-0.04	**-0.02**
	ΔBR	-0.22	0.09	0.06	0.21	**0.04**
	TR	84.16	82.19	78.93	75.51	**80.20**

Fig. 5-6 shows the RD curves for each of the four video sequences under a variety of Qp values. The time reduction curves are also shown on the same diagrams. There is marginal deviation in rate distortion from that of the JSVM encoder, therefore very similar coding efficiency is achieved. From Fig. 5-6, it can be seen the RD curves of the proposed algorithm

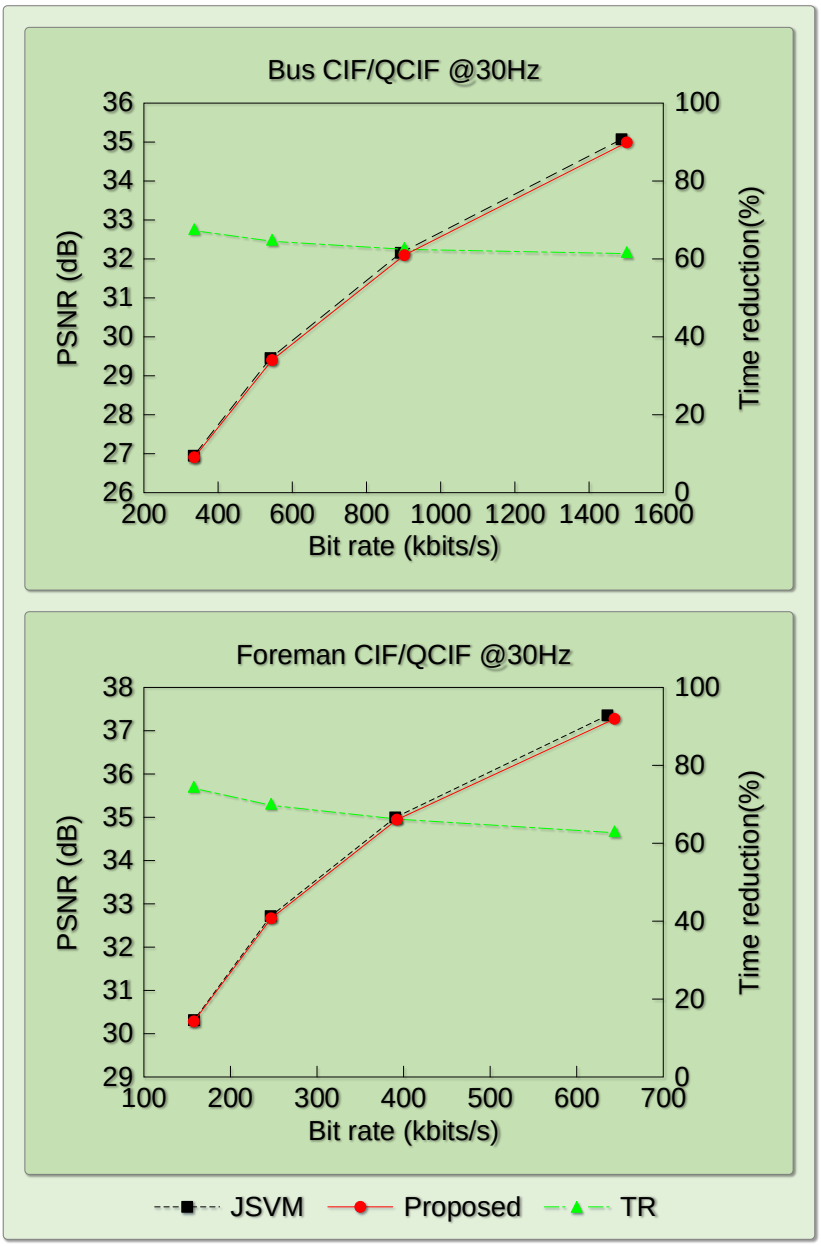


Fig. 5-6 RD performance for different video sequences (continued on next page).

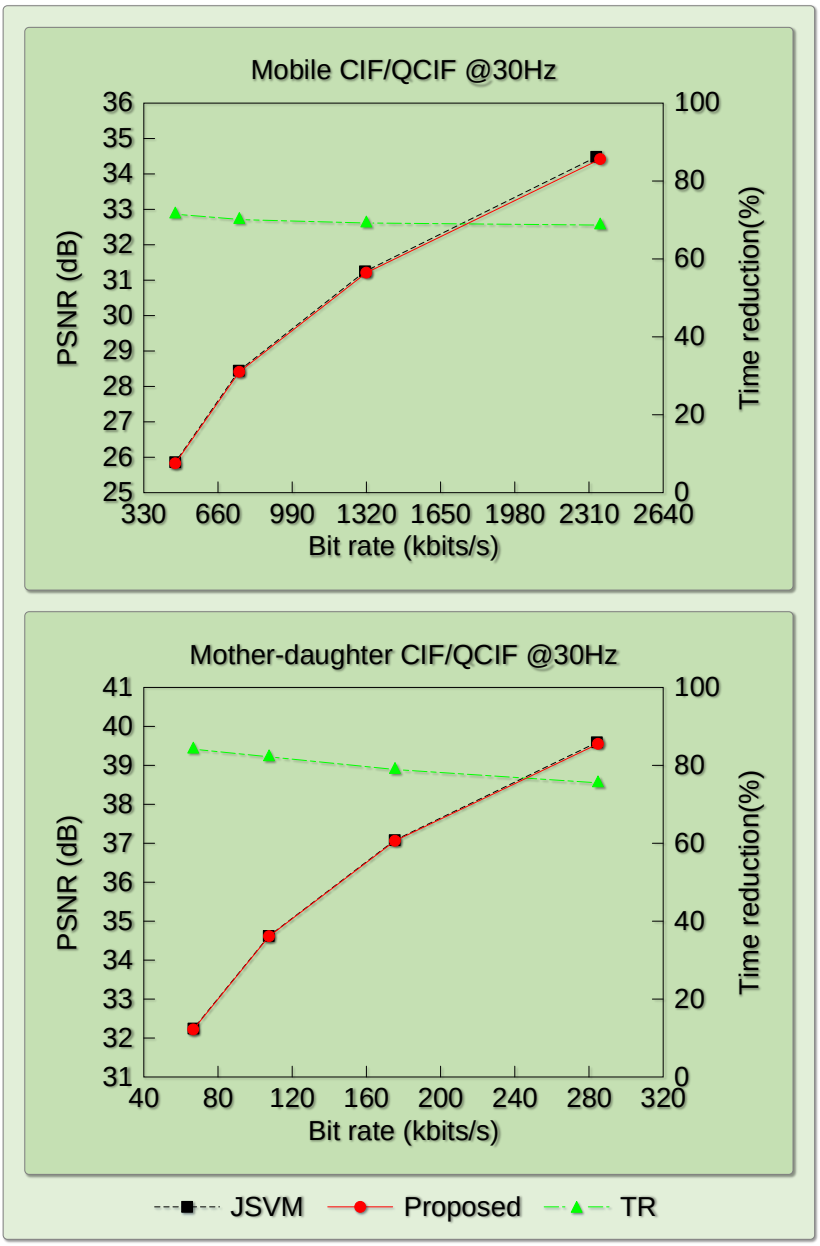


Fig. 5-6 RD performance for different video sequences (continued from last page).

are almost superimposed on those of the JSVM benchmark, which means the proposed algorithm achieves very similar coding efficiency to that of the JSVM encoder. The time reduction curves show that the proposed algorithm consistently provides a time reduction of greater than 60% for different video sequences under a wide range of bit rates.

In order to test the robustness of the proposed algorithm, the algorithm was applied to video sequences of larger spatial resolution and with multiple enhancement layers.

Table 5-7 compares the coding performance with the standard JSVM encoder when higher resolution images are coded: CIF for the base layer and 4CIF for the enhancement layer. Table 5-8 summarises the performance for three spatial layers: QCIF for the base layer, CIF for the first enhancement layer and 4CIF for the second enhancement layer. Four video test sequences were coded: 'City' which contains slow motion and relatively simple detail, 'Crew' with both moderate motion and detail, 'Harbor' that features complex detail but slow motion, and 'Soccer' that comprises very fast motion and considerable detail.

Table 5-7
Performance when encoding CIF/4CIF sequences

Sequence	Performance	Qp				BDPSNR	BDBR	TR
		40	36	32	28			
City	ΔPSNR	-0.02	-0.01	-0.02	-0.02	-0.02	0.38	59.13
	ΔBR	-0.28	0.05	0.04	0.24			
Crew	ΔPSNR	-0.05	-0.04	-0.03	-0.04	-0.02	0.60	59.39
	ΔBR	-0.85	-0.46	-0.18	-0.08			
Harbor	ΔPSNR	-0.02	-0.03	-0.04	-0.10	-0.04	0.97	57.21
	ΔBR	-0.44	-0.21	0.07	0.34			
Soccer	ΔPSNR	-0.01	-0.02	-0.05	-0.09	-0.04	1.05	59.85
	ΔBR	-0.09	-0.02	0.12	0.36			

The results in Tables 5-7 and 5-8, show that the proposed algorithm consistently achieves a significant reduction in computational complexity yet maintains a similar RD performance with that of the standard JSVM encoder. Specifically, the computational requirement is reduced by 57-59% when encoding the CIF/4CIF video sequences, while incurring at most a 1.05% increase in bit rate and a 0.04dB loss in PSNR. This is true even for the com-

Table 5-8
Performance when encoding QCIF/CIF/4CIF sequences

Sequence	EL	Performance	Qp 40	Qp 36	Qp 32	Qp 28	BDPSNR	BDBR	TR
City	1st	ΔPSNR	-0.01	-0.02	-0.03	-0.03	-0.03	0.55	68.89
		ΔBR	-0.10	-0.16	0.22	0.60			
	2nd	ΔPSNR	-0.01	-0.02	-0.02	-0.02	-0.02	0.41	
		ΔBR	-0.10	-0.11	0.32	0.46			
Crew	1st	ΔPSNR	-0.02	-0.03	-0.04	-0.06	-0.02	0.90	68.68
		ΔBR	-0.16	-0.51	-0.27	-0.09			
	2nd	ΔPSNR	-0.03	-0.03	-0.03	-0.04	-0.02	0.97	
		ΔBR	-0.51	-0.34	-0.19	0.11			
Harbor	1st	ΔPSNR	-0.03	-0.03	-0.03	-0.04	-0.04	0.46	67.28
		ΔBR	-0.08	0.04	0.32	0.42			
	2nd	ΔPSNR	-0.02	-0.03	-0.07	-0.13	-0.07	0.59	
		ΔBR	-0.21	0.23	0.37	0.57			
Soccer	1st	ΔPSNR	-0.02	-0.03	-0.04	-0.04	-0.05	1.66	69.77
		ΔBR	-0.38	0.14	0.29	0.63			
	2nd	ΔPSNR	-0.02	-0.02	-0.08	-0.19	-0.90	2.01	
		ΔBR	-0.23	0.32	0.75	0.77			

plex sequence 'Soccer'. When coding three spatial layers, the time reduction is 67-69% and, again, there is minimal reduction in rate distortion.

5.3.3 Comparisons with Other Algorithms

The proposed algorithm is compared also with three state-of-the-art SVC fast mode decision algorithms, namely those proposed by Kim [90], Zhao [93], and Lee [94]. The same video sequences as used by each of the authors were processed, and identical test conditions were employed.

The comparisons are shown in Tables 5-9 to 5-11. The proposed algorithm produced a better computational time reduction in each case. Compared with each benchmark, the proposed algorithm produced a negligible increase in bit rate and visually imperceptible decrease in PSNR. These results demonstrate that the proposed method outperforms the algorithms previously proposed.

Table 5-9
Comparison of proposed algorithm with Kim's algorithm

Sequence	Qp (BL/EL)	Kim [90]			Proposed		
		TR	△PSNR	△BR	TR	△PSNR	△BR
Bus	30/25	57.56	-0.012	1.43	62.25	-0.12	1.51
	30/30	56.72	-0.054	0.60	62.86	-0.09	1.08
	30/35	56.14	-0.110	-1.37	63.77	-0.05	0.45
City	30/25	63.48	-0.009	1.14	67.84	-0.04	1.86
	30/30	62.37	-0.047	0.09	69.04	-0.03	1.03
	30/35	61.76	-0.053	-1.12	70.47	-0.02	0.12
Crew	30/25	54.79	-0.011	0.76	60.58	-0.31	1.96
	30/30	53.78	-0.044	-0.27	61.29	-0.11	1.02
	30/35	54.12	-0.058	-1.66	62.25	-0.03	0.68
Football	30/25	52.20	-0.005	0.95	57.79	-0.53	0.92
	30/30	51.63	-0.045	-0.05	58.29	-0.48	0.48
	30/35	51.34	-0.063	-1.47	59.22	-0.33	0.17
Foreman	30/25	61.71	-0.017	1.06	63.00	-0.14	1.87
	30/30	61.18	-0.054	-0.24	64.23	-0.07	0.61
	30/35	61.09	-0.090	-1.84	65.22	-0.03	-0.24
Harbor	30/25	58.57	-0.004	0.81	68.19	-0.16	0.81
	30/30	57.87	-0.030	0.30	68.77	-0.07	0.60
	30/35	57.48	-0.065	-0.30	69.95	-0.04	0.14
Average		**57.43**	**-0.043**	**-0.07**	**64.17**	**-0.15**	**0.84**

Table 5-10
Comparison of proposed algorithm with Zhao's algorithm

Sequence	Qp (BL/EL)	Zhao [93]			Proposed		
		TR	BDPSNR	BDBR	TR	BDPSNR	BDBR
Bus	20/14 26/20 32/26 38/32	58.21	-0.073	1.29	63.32	-0.091	1.77
City		60.15	-0.072	1.82	69.91	-0.039	0.73
Coastguard		62.66	-0.040	0.77	71.71	-0.041	1.10
Crew		62.84	-0.077	1.68	66.56	-0.073	1.71
Football		59.37	-0.052	0.83	56.25	-0.087	1.82
Foreman		60.46	-0.093	2.39	66.46	-0.058	1.18
Harbor		61.01	-0.044	0.80	69.74	-0.075	1.72
Mobile		59.55	-0.065	1.14	69.44	-0.056	1.08
News		64.70	-0.076	1.66	77.38	-0.068	1.01
Silent		60.66	-0.080	1.63	75.38	-0.034	0.75
Average		**60.96**	**-0.070**	**1.40**	**68.61**	**-0.062**	**1.29**

Table 5-11
Comparison of proposed algorithm with Lee's algorithm

Sequence	Qp (BL/EL)	Lee [94]			Proposed		
		TR	△PSNR	△BR	TR	△PSNR	△BR
Container	28/28	68.7	-0.02	0.21	80.22	-0.01	-0.05
	32/32	69.3	-0.03	0.02	81.57	-0.01	-0.06
	36/36	69.6	-0.02	-0.13	82.68	0	-0.04
	40/40	69.6	-0.01	0.05	83.57	0	0.02
Crew	28/28	45.6	-0.08	0.50	62.42	-0.09	1.14
	32/32	54.6	-0.10	0.34	65.84	-0.07	0.15
	36/36	60.0	-0.16	-0.08	69.34	-0.06	-0.08
	40/40	63.1	-0.15	-0.30	74.48	-0.06	-0.31
Foreman	28/28	53.7	-0.14	0.09	62.71	-0.06	1.17
	32/32	59.4	-0.18	0.45	65.07	-0.05	0.54
	36/36	63.3	-0.09	-0.49	68.61	-0.04	0.33
	40/40	65.2	-0.21	-1.22	73.19	-0.03	-0.09
Harbor	28/28	34.2	-0.04	0.36	67.77	-0.11	0.70
	32/32	46.1	-0.07	0.20	68.36	-0.06	0.50
	36/36	56.3	-0.09	-0.10	70.50	-0.05	0.49
	40/40	63.0	-0.11	-0.22	73.43	-0.04	-0.18
Mother-daughter	28/28	66.0	-0.06	0.13	75.51	-0.04	0.21
	32/32	67.7	-0.07	-0.16	78.93	-0.02	0.06
	36/36	68.0	-0.06	-0.36	82.19	-0.01	0.09
	40/40	68.1	-0.04	-0.45	84.16	-0.01	-0.22
Silent	28/28	62.1	-0.04	0.48	72.49	-0.06	0.93
	32/32	64.7	-0.04	0.22	74.95	-0.02	0.32
	36/36	66.4	-0.05	0.01	78.17	-0.03	0.07
	40/40	67.4	-0.05	-0.48	81.03	-0.01	-0.02
Tempete	28/28	54.0	-0.06	0.71	68.43	-0.06	0.15
	32/32	61.0	-0.08	0.50	69.62	-0.07	0.03
	36/36	65.6	-0.09	0.33	71.77	-0.04	-0.33
	40/40	67.8	-0.09	0.38	75.10	-0.02	-0.29
Average		**61.5**	**-0.08**	**0.04**	**73.07**	**-0.04**	**0.19**

5.4 Summary

In this chapter, a fast mode decision algorithm for efficient implementation of the SVC extension of H.264/AVC is described. It aims to achieve an equivalent RD performance and a reduced computational requirement for the enhancement layer in SVC.

This chapter initially analysed the previously proposed algorithms and identified several factors which may be used for fast video coding. A more sophisticated hierarchical structure was therefore proposed to achieve a further complexity reduction.

Firstly, the proposed algorithm measured the correlation of a macroblock in the enhancement layer and both the co-located macroblocks in the base layer and neighbouring macroblocks in the enhancement layer. The base layer information and the context information of neighbouring macroblocks are then exploited to reduce the number of mode candidates that need to be evaluated. The homogeneity of the macroblock is then measured and, where appropriate, the candidate modes are further reduced. Finally, the motion activity in the base layer is exploited. Each of these factors is considered in the mode selection process, making the proposed method effective and robust.

Extensive evaluations have been performed to compare the hierarchical algorithm with the standard JSVM implementation and the state-of-the-art SVC fast mode decision algorithms with a wide range of Qp values for a variety of video sequences. Compared with the standard JSVM implementation, simulation results show that the algorithm achieves a reduction in encoding time of up to 84% with negligible reduction in coding efficiency and reconstructed video quality. Compared with state-of-the-art SVC fast mode decision algorithms, the proposed algorithm achieves a superior computational time reduction under very similar RD performance conditions.

Chapter 6

Improved Rate Control Scheme for SVC

Rate control plays an important role in video coding. In real-time applications, rate control enables the output bit rate to adjust quickly to the available channel bandwidth. With a proper control scheme, 'overflow' and 'underflow' of the buffer are prevented, which means that frame skipping and wastage of channel resources can be avoided. Furthermore, rate control appropriately allocates the available bits according to the complexity of the image content, so that the quality of the video is maximised.

However the rate control scheme in the latest JSVM reference software lacks consideration of the implications of inter-layer prediction as it was designed for single layer video encoders. In this chapter, a novel rate control algorithm is proposed for when inter-layer prediction is employed. Firstly, a RD model for inter-layer prediction coding of the spatial enhancement layers is developed. Secondly, an optimised MAD prediction model for spatial enhancement layers is described. This considers the MAD from previous temporal frames and base layer frames together. Simulation results show that the proposed MAD prediction model reduces the MAD prediction error by up to 79% relative to the linear prediction model used in the default rate control algorithm of SVC, namely the JVT-W043 rate

control scheme. Compared with JVT-W043 the proposed algorithm improves the average PSNR by up to 0.34dB or produces an average saving in bit rate of up to 7.78%.

The remainder of this chapter is organised as follows: A brief review of rate control algorithms for SVC is discussed in the next section. Section 6.2 discusses the formulation of the proposed RD models and the proposed MAD prediction scheme. The overall structure is also presented in the same section. Extensive experimental results are presented in section 6.3 and conclusions are given in section 6.4.

6.1 Existing Rate Control Algorithms

Rate control algorithms for video coding were widely studied prior to the release of the SVC standard, the aim being to ensure successful transmission of an encoded bitstream and to make full use of the limited bandwidth. Consequently, rate control plays an important role, as it directly influences the coding efficiency of the video encoder. Whether the rate control is efficient or not largely depends on the accuracy of the rate control model and the effectiveness of the rate control algorithm. It affects not only the stability of the bit rate, but also the picture quality of the entire video sequence.

Several rate control algorithms for scalable video coders have been proposed. These include the JVT-W043 rate control algorithm, which has been incorporated in the latest JSVM reference software, and several improved algorithms. JVT-W043 and a number of representative rate control algorithms suggested for SVC will be discussed in the following subsections.

6.1.1 Default Implementation in JSVM

JVT-W043 was suggested by JVT and has become the default rate control scheme for the base layer of SVC. JVT-W043 follows closely the rate control implementation that was proposed in JVT-G012, called the adaptive basic unit layer rate control algorithm. JVT-G012 is

the default rate control scheme of the H.264/AVC standard, and achieves a good balance between algorithm complexity and rate control performance. JVT-G012 introduces a new concept of 'basic unit' and a linear MAD prediction model to solve the 'chicken and egg paradox' that exists. It involves three steps: GOP level rate control, frame level rate control and basic unit level rate control. A basic unit usually comprises a number of consecutive macroblocks; when it contains only one macroblock then it is considered as macroblock level rate control algorithm. When it contains all the macroblocks in a frame then it becomes frame level rate control.

1. **GOP level rate control**

   In this step, the target number of bits for each GOP is allocated according to the target bit rate and the current capacity of the virtual buffer. When starting to encode the i^{th} GOP, the target number of bits $T_r\left(n_{i,0}\right)$ for this GOP are determined as

   $$T_r\left(n_{i,0}\right) = \frac{u\left(n_{i,1}\right)}{f} \times N_{\text{gop}} - \left[\frac{B_s}{8} - B_c\left(n_{i-1,N_{\text{gop}}}\right)\right] \tag{6.1}$$

   where $u\left(n_{i,1}\right)$ is the instantaneous target bit rate when the 1^{st} frame of the i^{th} GOP is being coded; f denotes the predefined frame rate; N_{gop} refers to the number of frames in each GOP; B_s is the buffer size and $B_c(n_{i-1,N_{\text{gop}}})$ is the current capacity of the virtual buffer after coding the $(i-1)^{\text{th}}$ GOP.

   As the channel bandwidth, namely the target bit rate, may vary with time, $T_r\left(n_{i,j}\right)$ is updated after each frame is coded as

   $$T_r\left(n_{i,j}\right) = T_r\left(n_{i,j-1}\right) + \frac{u\left(n_{i,j}\right) - u\left(n_{i,j-1}\right)}{f}\left(N_{\text{gop}} - j\right) - T_p\left(n_{i,j-1}\right) \tag{6.2}$$

   where $T_p\left(n_{i,j-1}\right)$ refers to the actual number of bits generated by the $\left(j-1\right)^{\text{th}}$ frame of the i^{th} GOP.

2. **Frame level rate control**

   The bit budget $\tilde{f}\left(n_{i,j}\right)$ for the j^{th} frame of the i^{th} GOP is calculated according to the

target buffer level, the frame rate, the target bit rate, and the capacity of the buffer.

$$\tilde{f}\left(n_{i,j}\right)=\frac{u\left(n_{i,j}\right)}{f}+\gamma\left[Tbl\left(n_{i,j}\right)-B_c\left(n_{i,j}\right)\right] \tag{6.3}$$

where γ is a constant, and its typical value is 0.75 when there is no B-frame, and is 0.25 otherwise. $Tbl\left(n_{i,j}\right)$ is the target buffer level.

When the available bits remaining are also considered, the bit budget for the j^{th} frame is calculated as

$$\hat{f}\left(n_{i,j}\right)=\frac{W_p\left(n_{i,j-1}\right)T_r\left(n_{i,j}\right)}{W_p\left(n_{i,j-1}\right)N_{p,r}\left(j-1\right)+W_b\left(n_{i,j-1}\right)N_{b,r}\left(j-1\right)} \tag{6.4}$$

where $N_{p,r}\left(j-1\right)$ and $N_{b,r}\left(j-1\right)$ refer to the remaining number of P- and B-frames in the GOP respectively; $W_p\left(n_{i,j-1}\right)$ and $W_b\left(n_{i,j-1}\right)$ denote the average picture complexity of the P-frames and B-frames respectively.

The final target number of bits for the j^{th} frame is determined as a weighted combination of $\hat{f}\left(n_{i,j}\right)$ and $\tilde{f}\left(n_{i,j}\right)$

$$f\left(n_{i,j}\right)=\beta\times\hat{f}\left(n_{i,j}\right)+\left(1-\beta\right)\times\tilde{f}\left(n_{i,j}\right) \tag{6.5}$$

After encoding the j^{th} frame, the model coefficients are updated according to the actual generated bits, the Qp value used, and the MAD of the residual component obtained.

3. Basic unit level rate control

If the basic unit is not a frame, basic unit level rate control needs to be performed. As for frame level rate control, basic unit level rate control comprises the following steps: 1) Target bit allocation for each basic unit. 2) Linear prediction of MAD. 3) Calculation of Qp. 4) Update of the model coefficients.

6.1.2 Other Improved Algorithms

So far, most rate control algorithms have been developed for single layer video coding, however several rate control algorithms have been suggested for SVC [98–104]. They con-

sider either precise target bit allocation or the optimisation of the RD model.

The rate control algorithm in [98] operates in each spatial layer individually, that is the dependency between spatial layers is not exploited explicitly. If the spatial correlation and other new features introduced by the layer-based structure of SVC can be adequately exploited, improved coding performance is expected.

Later, Xu et al. proposed a rate control algorithm for spatial and CGS scalable coding in SVC [99]. This method employs the improved TMN8 model for Qp estimation based on the mode analysis of I-, P-, and B-frames. The TMN8 rate control algorithm was developed for the H.263 video coding standard and is primarily used in low bit rate and low delay environments.

In [100] and [101], Liu et al. proposed that the MAD can be predicted from either the previous frame in the same layer or the corresponding frame in the base layer, through a switching law. It is shown that the previous temporal frames and the reference frame in the base layer can provide useful information for MAD prediction in the enhancement layer. In order to make full use of the available information, an improved MAD model needs to be developed, by which information from previous temporal frames and the reference frame in the base layer is considered together.

Hu et al. [102] proposed a frame level rate control algorithm for temporal scalability in SVC by developing a set of weighting factors for bit allocation. Their work focused on bit allocation among different temporal layers and lacked the support for other scalability features. Subsequently, Hu et al. developed a Cauchy distribution-based Rate-Quantisation (R-Q) model for each spatial layer in [103]. However, it is arguable whether the Cauchy distribution-based R-Q model outperforms the classic Laplace distribution-based model.

In [104], Liu et al. proposed a bit allocation algorithm for SVC where the inter-layer dependency is taken into consideration. Although Liu et al.'s work provides a reasonable bit allocation mechanism, inter-layer correlation is not explicitly exploited in optimisation of the RD model.

6.2 The Proposed Rate Control Algorithm

The JVT-W043 rate control algorithm is implemented only in the base layer of the latest JSVM reference software, and it does not support the enhancement layers which provide the scalability functions. For the enhancement layers, the target bit rate is controlled by a FixedQp tool, where a logarithmic search algorithm is applied to find the appropriate Qp value required to meet the target bit rate. Rate control algorithms that address the properties of the enhancement layers need to be developed.

As SVC uses a multi-layer structure to support scalability, the new features mean that rate control models for single layer video coding are not suitable for precisely describing the RD behaviour. The coding efficiency in terms of RD performance is not guaranteed to be optimal if rate control schemes developed for single layer coding are incorporated in a multi-layer video system. When a rate control scheme is developed for scalable video coding systems, the inter-layer correlation between the enhancement layer and the corresponding base layer needs to be considered.

6.2.1 RD Models for Prediction Modes

In single layer coding schemes, such as H.264/AVC, only intra-layer prediction (including intra-prediction and inter-prediction) is applied. However, due to the multi-layer structure of SVC, it employs the inter-layer prediction mechanism to reuse the coded lower layer data for encoding the corresponding enhancement layer. Consequently, it is observed that macroblocks coded using inter-layer prediction and those coded by intra-layer prediction have dissimilar statistical properties. The reasons for this are summarised as follows:

1. **Data to be coded**: The inter-layer prediction residual is different from that of intra-layer prediction. For instance, when performing inter-layer residual prediction, the upsampled residual signal of the co-located 8×8 submacroblock in the base layer is used as the inter-layer prediction of the residual signal in the enhancement layer mac-

roblock. Therefore, only the corresponding difference (refinement) signal of the residual needs to be coded in the enhancement layer. On the other hand, the residual obtained from intra-layer prediction is the difference between the original block and its matching block.

2. **Picture content**: The coding efficiency of inter-layer prediction tools relies heavily on the content of the video sequences to be coded. For example, inter-layer prediction is not efficient for coding sequences containing homogeneous texture or slow motion, since the high frequency components in the enhancement layer cannot be reconstructed well by upsampling information from the base layer. Macroblocks with little detail and slow motion are more likely to be best matched with a block by inter-frame prediction in the same layer. That is to say, the temporal correlation within the same layer is generally higher than the correlation between two layers in the same frame, except where very fast motion activity is present [105]. However, macroblocks with fast movement usually need more bits to encode.
3. **Picture quality of base layer**: The quality of information referred from the base layer has a significant impact on the efficiency of inter-layer prediction tools [106]. Specifically, when the quality of the base layer is poor, only a rough prediction is obtained and more bits have to be spent to compensate the prediction error. In such cases, intra-layer prediction is more efficient than inter-layer prediction. By contrast, if the base layer is of relatively high quality, the performance of inter-layer prediction improves and fewer bits are needed.

Based on the above analysis, it can be deduced that the average number of bits for inter-layer predicted macroblocks in the enhancement layers is significantly different from that for macroblocks which have been coded using temporal and spatial prediction within the same enhancement layer.

Existing rate control strategies assume that the statistical property of a video source is fixed [107], and then they derive a precise RD model, but with low computational com-

plexity [108]. The analysis shows that the use of inter-layer prediction in the enhancement layer results in a divergence from existing RD models.

The significant difference in the number of bits generated for inter-layer predicted macroblocks and intra-layer predicted macroblocks leads to serious prediction errors in RD model coefficients. Having improper coefficients in the model means that accurate Qp prediction cannot be achieved and the rate control mechanism must be suboptimal. As it is not possible to model a unified relationship between Qp and the number of bits for both inter-layer prediction and intra-layer prediction, the relationship between Qp and the number of bits for inter-layer prediction needs to be considered separately.

To illustrate the mathematical relationship between Qp and the number of bits for both inter-layer predicted macroblocks and intra-layer predicted macroblocks, and to justify the proposed algorithm, the simulation results from processing four sequences with different degrees of activity and detail are analysed. The statistics were collected from the first 150 frames of each video sequence. QCIF sequences were used for the base layer and CIF were used for the enhancement layer.

Table 6-1
Average number of bits per macroblock for both inter-layer predicted macroblocks and intra-layer predicted macroblocks

Sequence	Prediction mode	Qp				
		28	32	36	40	44
Bus	Intra-layer	98.98	47.91	18.53	6.26	1.66
	Inter-layer	98.68	52.22	28.05	15.28	7.94
Football	Intra-layer	77.89	43.92	22.07	9.41	3.79
	Inter-layer	100.3	56.63	32.12	20.31	11.76
Foreman	Intra-layer	18.61	6.52	2.40	0.86	0.39
	Inter-layer	28.81	14.27	7.59	4.72	3.20
Mobile	Intra-layer	185.2	77.89	22.57	6.76	2.23
	Inter-layer	162.4	96.88	46.41	22.33	12.79

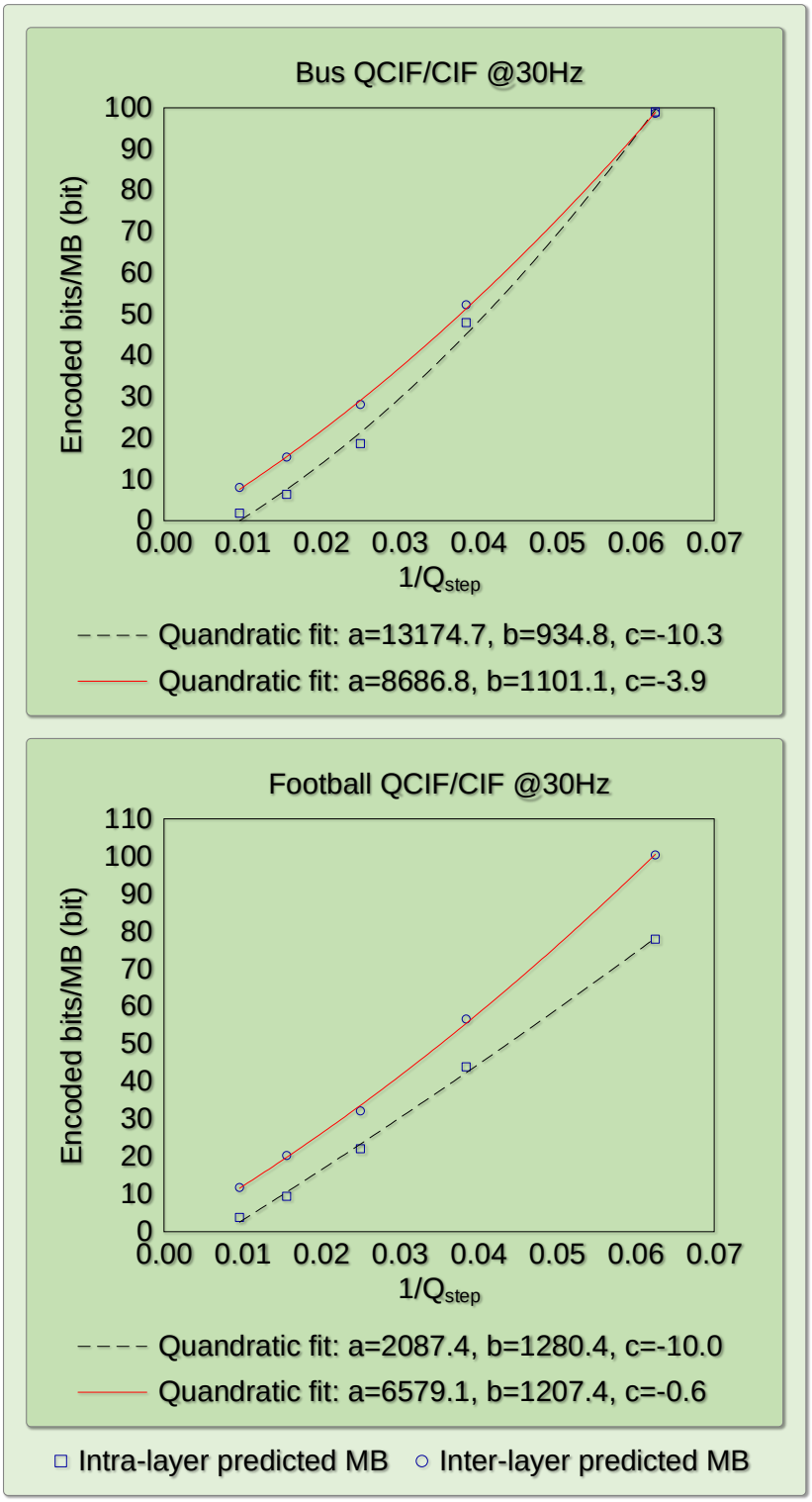


Fig. 6-1 Relationship between average number of bits and Q_{step} for both inter-layer coding and intra-layer coding. Points are actual data; curves are fitted to the data (continued on next page).

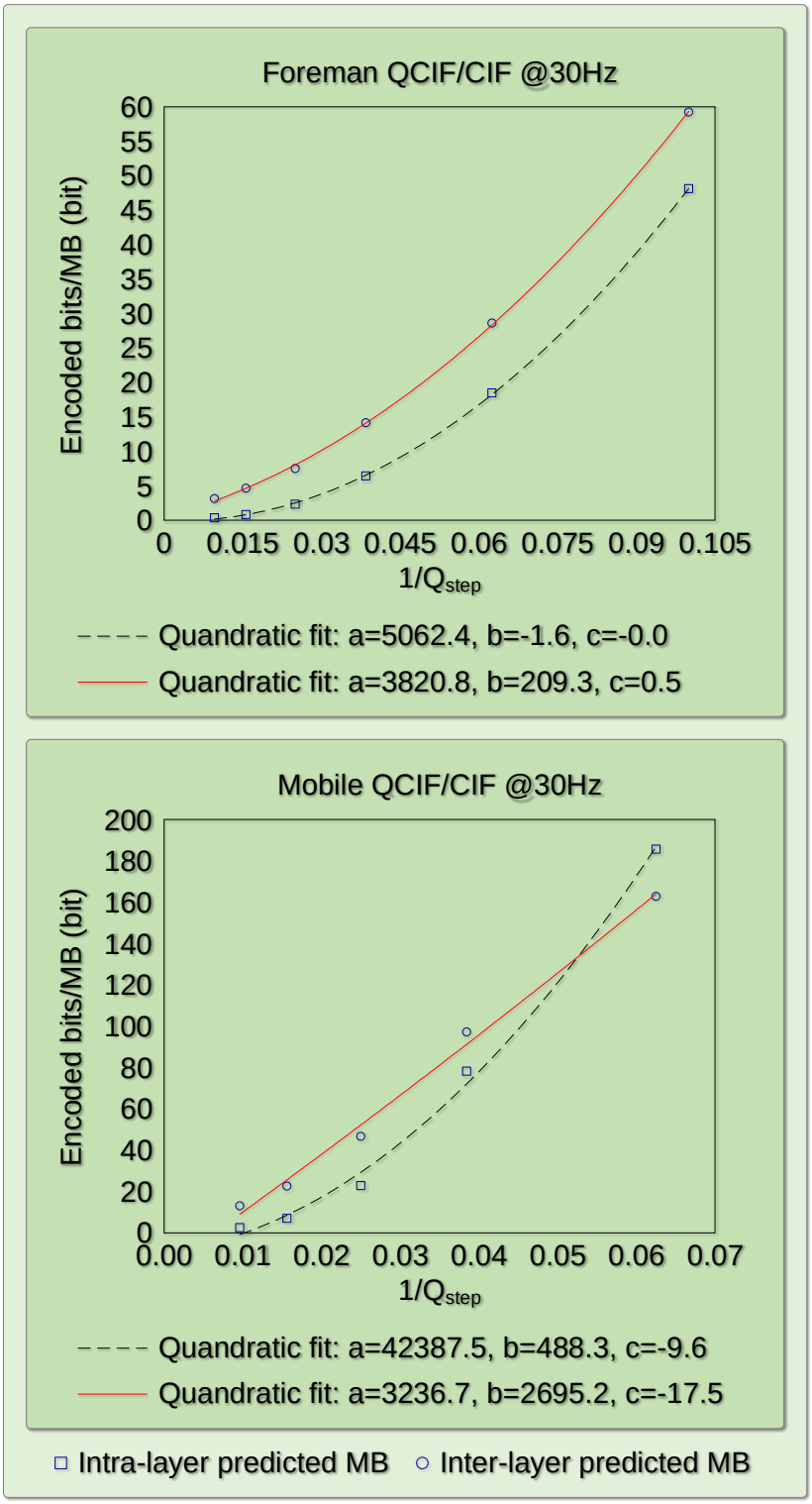


Fig. 6-1 Relationship between average number of bits and Q_{step} for both inter-layer coding and intra-layer coding. Points are actual data; curves are fitted to the data (continued from last page).

Table 6-1 shows the average number of bits for inter-layer predicted macroblocks in the enhancement layer and those for intra-layer predicted macroblocks in the same enhancement layer, under a variety of Qp values. It is observed that the average number of bits used for encoding inter-layer predicted macroblocks is significantly different from that required for intra-layer predicted macroblocks. In general, as Qp increases, significantly more bits are consumed by inter-layer prediction coding than by intra-layer prediction.

As discussed in section 2.2, the model relationship between Qp and Q_{step} is

$$\text{Qp} = 2^{\frac{Q_{\text{step}}}{6}} \zeta(Q_{\text{step}} \% 6) \tag{6.6}$$

where $\zeta(0)$=0.675; $\zeta(1)$=0.6875; $\zeta(2)$=0.8125; $\zeta(3)$=0.875; $\zeta(4)$=1.0; $\zeta(5)$=1.125 [33].

Fig. 6-1 illustrates the relationship between Q_{step} values and the obtained number of bits R_{txt} for both inter-layer predicted macroblocks and intra-layer predicted macroblocks. The quadratic curves fitting the measured data are also presented. It can be seen that the measured data can be represented by quadratic functions very well.

$$R_{\text{txt}} = \frac{a}{Q_{\text{step}}^2} + \frac{b}{Q_{\text{step}}} + c \tag{6.7}$$

where c≈0. The coefficients of the quadratic model are obtained by finding the minimal fitting error. Although the observed $R_{\text{txt}} - Q_{\text{step}}$ relationship can be represented by quadratic models, the model coefficients are significantly different. From these observations, it is concluded that for optimised rate control within the enhancement layers, separate, and sufficiently accurate, models must be used for inter-layer prediction and intra-layer prediction. The relationship between $Q_{\text{step}}^{\text{inter}}$ and R_{txt} for inter-layer prediction needs to be represented by

$$R_{\text{txt}} = \frac{a^{\text{inter}}}{\left(Q_{\text{step}}^{\text{inter}}\right)^2} + \frac{b^{\text{inter}}}{Q_{\text{step}}^{\text{inter}}} + c^{\text{inter}} \tag{6.8}$$

Similarly, the model for intra-layer prediction needs to be represented separately as

$$R_{\text{txt}} = \frac{a^{\text{intra}}}{\left(Q_{\text{step}}^{\text{intra}}\right)^2} + \frac{b^{\text{intra}}}{Q_{\text{step}}^{\text{intra}}} + c^{\text{intra}} \tag{6.9}$$

Consequently, similar to the classic quadratic RD model described in equation (2.9), $Q_{\text{step}}^{\text{inter}}$ and $Q_{\text{step}}^{\text{intra}}$ can be modelled accurately by quadratic functions with their respective model coefficients.

$$\begin{aligned} R_{\text{txt}} &= \frac{X_1^{\text{inter}} \times \text{MAD}_{\text{pred}}}{\left(Q_{\text{step}}^{\text{inter}}\right)^2} + \frac{X_2^{\text{inter}} \times \text{MAD}_{\text{pred}}}{Q_{\text{step}}^{\text{inter}}} \\ R_{\text{txt}} &= \frac{X_1^{\text{intra}} \times \text{MAD}_{\text{pred}}}{\left(Q_{\text{step}}^{\text{intra}}\right)^2} + \frac{X_2^{\text{intra}} \times \text{MAD}_{\text{pred}}}{Q_{\text{step}}^{\text{intra}}} \end{aligned} \tag{6.10}$$

where R_{txt} is the target number of bits for the current macroblock; MAD_{pred} is the MAD predicted from the previous coding results, $Q_{\text{step}}^{\text{inter}}$ and $Q_{\text{step}}^{\text{intra}}$ are the desired quantisation step sizes for inter-layer prediction and intra-layer prediction respectively. X_1^{inter} and X_2^{inter}, X_1^{intra} and X_2^{intra}, are the model coefficients for inter-layer prediction and intra-layer prediction, each updated after the coding of a macroblock using inter-layer prediction and intra-layer prediction respectively.

6.2.2 Optimisation of the MAD Prediction Model

From equations (6.10) it can be seen that the quantisation step sizes $Q_{\text{step}}^{\text{inter}}$ and $Q_{\text{step}}^{\text{intra}}$ depend on the model coefficients, the target number of bits R_{txt} for the current macroblock, and the predicted MAD value of the current macroblock. However, the MAD value is unknown before RDO. The MAD value can be obtained only after coding the current macroblock using the quantisation step size, but the model needs the MAD value to calculate the quantisation step size. This is a 'chicken and egg situation'. The JVT-W043 rate control algorithm overcomes this problem by using the MAD value of the macroblock in the same position of the previous frame to predict the MAD value of the current macroblock, thus permitting the quantisation step size to be calculated. A linear MAD model is adopted here

as [109]:

$$\mathrm{MAD}_j = a_1 \times \mathrm{MAD}_{j-1} + a_2 \tag{6.11}$$

where a_1 and a_2 are model coefficients, updated after the coding of each frame. MAD_j denotes the predicted MAD of the current macroblock and MAD_{j-1} denotes the actual MAD of the macroblock in the corresponding position of the previous temporal frame.

In the quadratic RD model, MAD prediction is very important as it directly affects the allocation of bits. As the prediction is not always accurate, there is always some small error in bit allocation, and this cannot be avoided in the JVT-W043 algorithm. As shown in Fig. 6-2, if MAD fluctuates due to fast motion or scene changes in the video sequence, the linear model performs poorly, and there is always a delay. In the example, this phenomenon is particularly obvious at frames 9, 34 and 44 in the 'Football' sequence. What is more, the prediction model coefficients have to be updated using the rapidly changing MAD values, which results in a significant prediction error. The prediction errors then propagate to future predictions and have a negative effect on the performance of the rate control mechanism.

In the SVC encoding process, for each frame, the base layer is encoded first before the enhancement layers. Furthermore, the content of the base layer and the enhancement layers are highly correlated. As shown in Fig. 6-3, even though the MAD values of the two layers are not the same, they have a similar tendency in the presence of abrupt changes. This leads to the idea that some encoding results of the base layer can be used to inform the coding of the enhancement layers, thus benefitting from the bottom-up coding structure of the standard.

In order to examine the correlation between the actual MAD values in the base layer and those in the enhancement layer, the Pearson Correlation Coefficient (PCC) is employed as a measure. The PCC of two random variables X and Y is defined as [110]:

$$r = \frac{\mathbf{E}\big[(X-\mu_X)(Y-\mu_Y)\big]}{\sigma_X \sigma_Y} \tag{6.12}$$

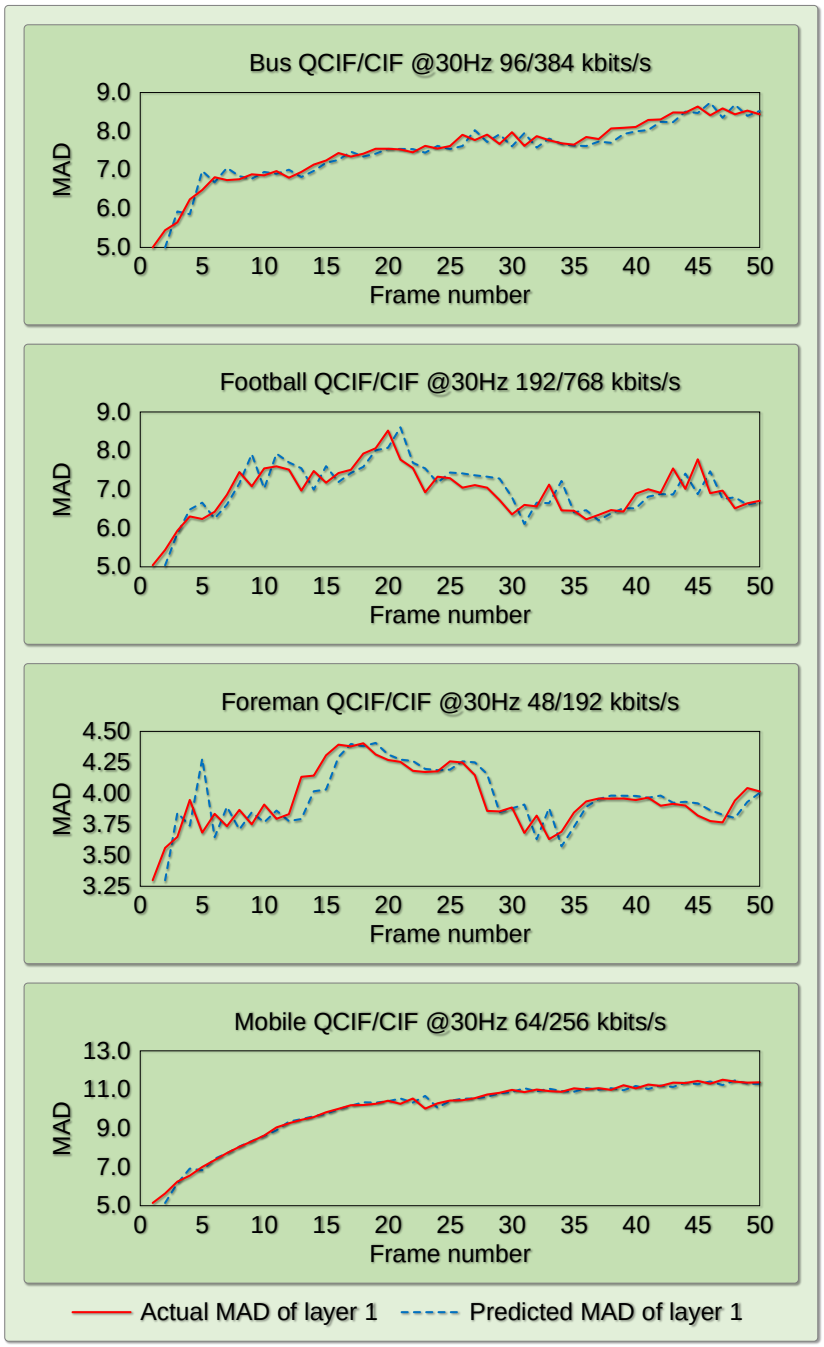


Fig. 6-2 Relationship between predicted and actual MAD values.

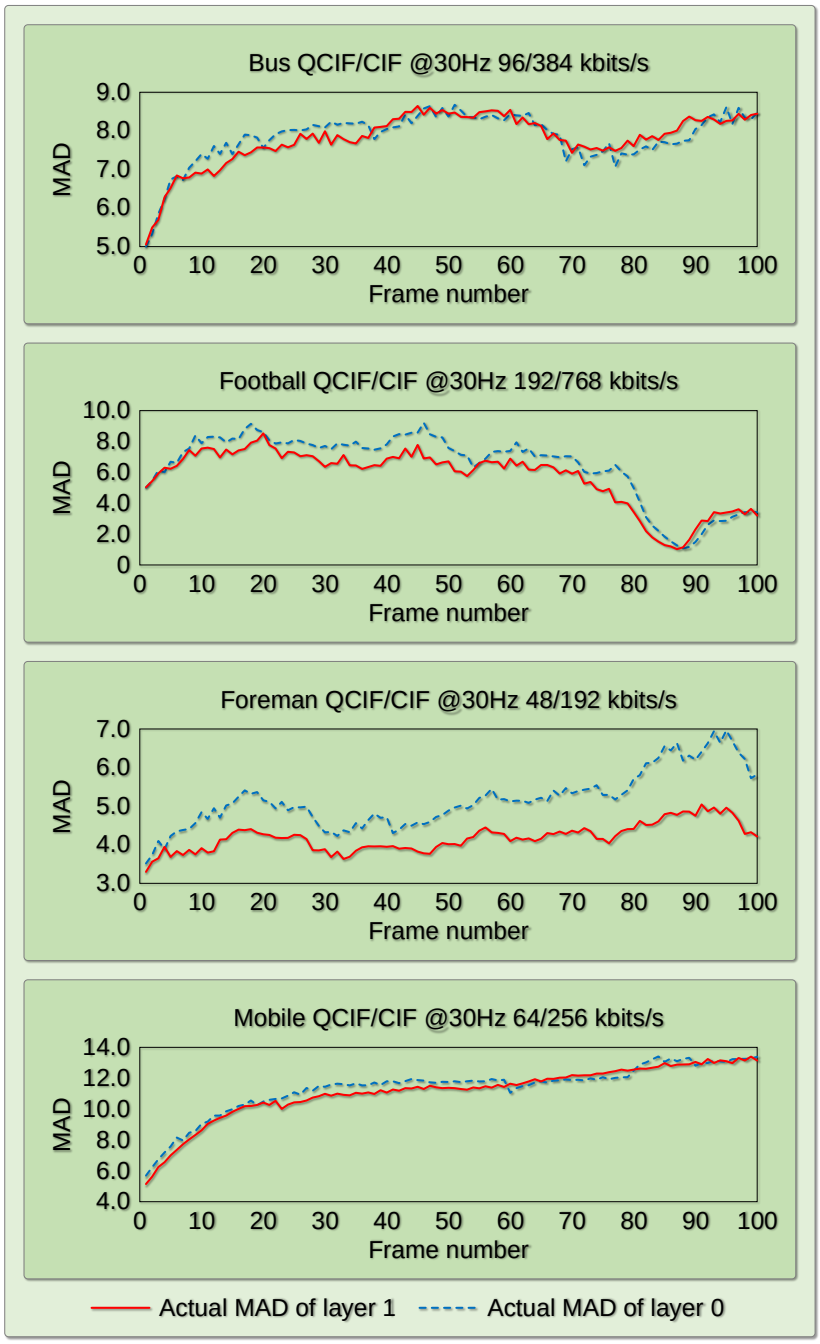


Fig. 6-3 Relationship between actual MAD values of base layer and those of the spatial enhancement layer.

where r ($r \leq 1$) indicates the strength of the linear correlation between two random variables X and Y. **E** is the expectation, σ_X and σ_Y are the standard deviations of X and Y, and μ_X and μ_X are the means of X and Y. The closer the absolute value of r is to 1, the stronger the correlation between the two variables. A positive r indicates the two variables have a positive correlation, and vice versa.

Four 30Hz video sequences were tested under a variety of bit rates ranging from 48 kbits/s to 2560 kbits/s. The statistics were collected from the first 150 frames of each video sequence. QCIF sequences were used for the base layer and CIF for the enhancement layer.

Table 6-2
MAD correlation between base layer and corresponding enhancement layer

Sequence	BR_{tgt} (BL/EL)	PCC	Sequence	BR_{tgt} (BL/EL)	PCC
Bus	96/384	0.91	Football	192/768	0.96
	128/512	0.86		256/1024	0.96
	192/768	0.87		384/1536	0.91
	320/1280	0.78		640/2560	0.88
Foreman	48/192	0.84	Mobile	64/256	0.98
	64/256	0.93		96/384	0.93
	96/384	0.89		128/512	0.82
	160/640	0.90		192/768	0.78

As seen in Table 6-2, most of the PCCs are greater than 0.85, therefore the actual MAD of the base layer and that of the enhancement layer are regarded as high correlated. For the 'Football' sequence, in which sudden changes occasionally occur, as it contains very fast moving objects, all of the PCCs are greater than 0.88 for a wide range of bit rates. Such high PCCs show that a strong linear correlation exists in the MAD between adjacent layers. These observations lead to the idea that the MAD of the base layer can be employed to assist the MAD prediction in the enhancement layer. Therefore, a new MAD prediction model for the spatial enhancement layer using the encoding results from both the base layer and the previous temporal frames as factors in the MAD prediction procedure is pro-

posed. The new prediction model is defined as:

$$\mathrm{MAD}^{i}_{\mathrm{el}} = \left(1-\omega^{i}_{\mathrm{er}}\right)\cdot \mathrm{MAD}^{i}_{\mathrm{el,temp}} + \omega^{i}_{\mathrm{er}}\cdot \mathrm{MAD}^{i}_{\mathrm{el,il}} \tag{6.13}$$

where ω_{er} is a weighting factor, which is calculated as

$$\omega^{i}_{\mathrm{er}} = \frac{\left|\mathrm{MAD}^{i}_{\mathrm{bl,pred}} - \mathrm{MAD}^{i}_{\mathrm{bl,act}}\right|}{\mathrm{MAD}^{i}_{\mathrm{bl,act}}} \tag{6.14}$$

where the subscripts 'el' and 'bl' indicate the enhancement layer and the base layer; $\mathrm{MAD}^{i}_{\mathrm{bl,act}}$ and $\mathrm{MAD}^{i}_{\mathrm{bl,pred}}$ refer to the actual and predicted MAD of a macroblock, which is the co-located macroblock of the i^{th} macroblock in the enhancement layer; $\mathrm{MAD}^{i}_{\mathrm{el,temp}}$ and $\mathrm{MAD}^{i}_{\mathrm{el,il}}$ indicate the temporally predicted MAD and the inter-layer predicted MAD of the i^{th} macroblock in the enhancement layer.

The temporally predicted MAD is obtained through the linear prediction model defined in equation (6.11),

$$\mathrm{MAD}^{i}_{\mathrm{el,temp}} = t^{i}_{1} \times \mathrm{MAD}^{i-1}_{\mathrm{el}} + t^{i}_{2} \tag{6.15}$$

where $\mathrm{MAD}^{i-1}_{\mathrm{el}}$ refers to the actual MAD value of the macroblock in the corresponding position of the previous frame; t^{i}_{1} and t^{i}_{2} are model coefficients updated after the coding of each macroblock.

Similar to equation (6.11), a linear prediction model is proposed for the prediction of the MAD of a macroblock in the enhancement layer, using the actual MAD value of its co-located block in the base layer,

$$\mathrm{MAD}^{i}_{\mathrm{el,il}} = s^{i}_{1} \times \mathrm{MAD}^{i}_{\mathrm{bl}} + s^{i}_{2} \tag{6.16}$$

where $\mathrm{MAD}^{i}_{\mathrm{bl}}$ denotes the actual MAD of the block in the co-located position of the base layer; s^{i}_{1} and s^{i}_{2} are model coefficients updated using a linear regression method after the

coding of each macroblock as follows.

$$\tilde{s}_1 = \frac{\sum_{i=1}^{n}\left(\mathrm{MAD}_{\mathrm{bl}}^{i}\cdot\mathrm{MAD}_{\mathrm{el,act}}^{i}\right)-\frac{1}{n}\sum_{i=1}^{n}\mathrm{MAD}_{\mathrm{bl}}^{i}\sum_{i=1}^{n}\mathrm{MAD}_{\mathrm{el,act}}^{i}}{\sum_{i=1}^{n}\left(\mathrm{MAD}_{\mathrm{bl}}^{i}\right)^2-n\sum_{i=1}^{n}\mathrm{MAD}_{\mathrm{bl}}^{i}\sum_{i=1}^{n}\mathrm{MAD}_{\mathrm{bl}}^{i}} \qquad (6.17)$$
$$\tilde{s}_2 = \frac{1}{n}\sum_{i=1}^{n}\mathrm{MAD}_{\mathrm{el,act}}^{i}-\tilde{s}_1\frac{1}{n}\sum_{i=1}^{n}\mathrm{MAD}_{\mathrm{bl}}^{i}$$

where n is the window size excluding the outliers; $\mathrm{MAD}_{\mathrm{el,act}}^{i}$ refers to the actual MAD value of the i^{th} macroblock in the enhancement layer.

It can be seen that the numerator of equation (6.14), namely $\left|\mathrm{MAD}_{\mathrm{bl,pred}}^{i}-\mathrm{MAD}_{\mathrm{bl,act}}^{i}\right|$, denotes the MAD prediction error in the base layer, consequently ω_{er}^{i} indicates the MAD prediction error rate. This parameter is adjusted adaptively according to the MAD prediction accuracy in the base layer. For instance, in the presence of sudden changes, the temporal linear prediction performs poorly, and the MAD prediction error ω_{er}^{i} in the base layer must be large. Therefore, when encoding the enhancement layer, a large ω_{er}^{i} is applied which means that the inter-layer MAD prediction is assigned a greater weight, and temporal MAD prediction has a lower weight. Consequently the enhancement layer is made aware of the abrupt changes of MAD in advance and the proposed MAD prediction model promptly adjusts the weighting factor ω_{er}^{i} to reduce the prediction errors. Therefore, the proposed MAD prediction model is completely adaptive, as the weight of the temporal MAD prediction and that of the inter-layer MAD prediction can be adjusted instantly according to the error rate of the linear MAD prediction in the base layer.

Consequently, when encoding the enhancement layers, the RD model equations (6.10) are rewritten as

$$R_{\mathrm{txt}}^{i} = \frac{X_1^{\mathrm{inter},i}\times\mathrm{MAD}_{\mathrm{el}}^{i}}{\left(Q_{\mathrm{step}}^{\mathrm{inter},i}\right)^2}+\frac{X_2^{\mathrm{inter},i}\times\mathrm{MAD}_{\mathrm{el}}^{i}}{Q_{\mathrm{step}}^{\mathrm{inter},i}} \qquad (6.18)$$

$$R_{\mathrm{txt}}^{i} = \frac{X_1^{\mathrm{intra},i}\times\mathrm{MAD}_{\mathrm{el}}^{i}}{\left(Q_{\mathrm{step}}^{\mathrm{intra},i}\right)^2}+\frac{X_2^{\mathrm{intra},i}\times\mathrm{MAD}_{\mathrm{el}}^{i}}{Q_{\mathrm{step}}^{\mathrm{intra},i}} \qquad (6.19)$$

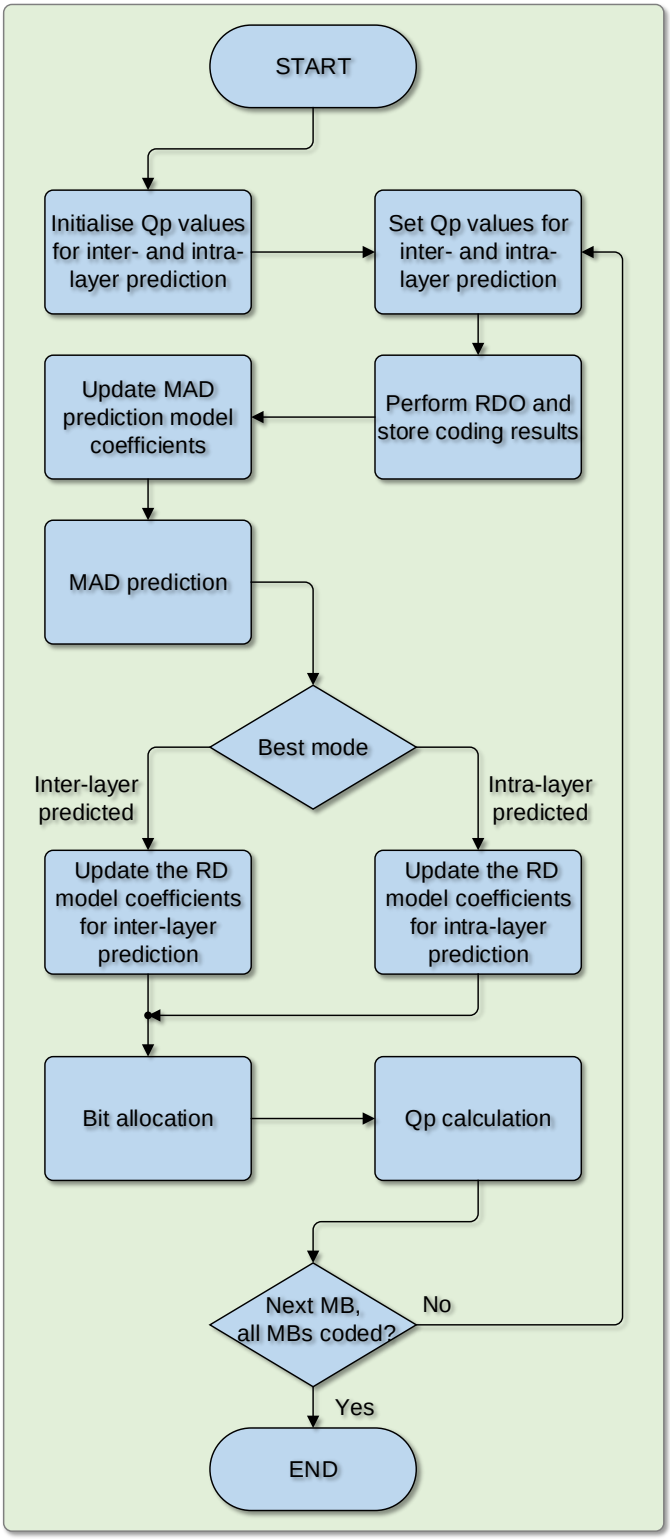


Fig. 6-4 Overall scheme of proposed algorithm.

6.2.3 Overall Structure of the Proposed Algorithm

The framework of the proposed rate control algorithm is illustrated in Fig. 6-4. The proposed macroblock level rate control algorithm is applied only to the spatial enhancement layer. For the base layer, the JVT-W043 rate control algorithm is employed. The proposed rate control algorithm is described as follows.

1. **Initialisation**: Set the Qp for inter-layer prediction, namely $\mathrm{Qp}^{\mathrm{inter}}$ and the Qp for intra-layer prediction, namely $\mathrm{Qp}^{\mathrm{intra}}$ equal to the initial Qp value.
2. **Qp values setting**: When performing inter-layer prediction, $\mathrm{Qp}^{\mathrm{inter}}$ is used, when performing intra-layer prediction, $\mathrm{Qp}^{\mathrm{intra}}$ is used.
3. **RDO**: Perform RDO for the current macroblock. The best mode, the actual number of bits generated ($R_{\mathrm{txt,act}}$), the actual MAD ($\mathrm{MAD}_{\mathrm{el,act}}$), and the used Qp value are stored.
4. **Update of MAD prediction model coefficients**: Use $\mathrm{MAD}_{\mathrm{el,act}}$ and the actual MAD of the macroblock in the corresponding position of the previous temporal frame in the same layer to update the temporal MAD prediction model coefficients in equation (6.15). Use $\mathrm{MAD}_{\mathrm{el,act}}$ and $\mathrm{MAD}_{\mathrm{bl}}$, which is the actual MAD of the co-located macroblock in the base layer, to update the inter-layer MAD prediction model coefficients using equation (6.17).
5. **MAD prediction**: Calculate the predicted MAD value, $\mathrm{MAD}_{\mathrm{el}}$, for the next macroblock according to equation (6.13).
6. **RD model coefficients update**: If the best mode is via inter-layer prediction, proceed to step **6.1.**). Otherwise, proceed to step **6.2.**).
   - **6.1.** Set $\mathrm{Qp}_{\mathrm{act}}^{\mathrm{inter}}$ as the Qp value which was stored in step **3.**), and use $R_{\mathrm{txt,act}}$ and $\mathrm{Qp}_{\mathrm{act}}^{\mathrm{inter}}$ to update the RD model coefficients in equation (6.18).
   - **6.2.** Set $\mathrm{Qp}_{\mathrm{act}}^{\mathrm{intra}}$ as the Qp value which was stored in step **3.**), and use $R_{\mathrm{txt,act}}$ and $\mathrm{Qp}_{\mathrm{act}}^{\mathrm{intra}}$ to update the RD model coefficients in equation (6.19).
7. **Bit allocation**: Calculate the target number of bits, R_{txt}, for the next macroblock according to the default bit allocation implementation in JSVM, as described in subsection

6.1.1.

8. **Qp calculation**: Given the target number of bits, the Qp for inter-layer prediction, namely Qp^{inter}, is calculated based on the RD model in equation (6.18), and the Qp for intra-layer prediction, namely Qp^{intra}, is calculated based on the RD model in equation (6.19)
9. **End of frame check**: Proceed to **2.**) until encoding the current frame is finished, and then process the next frame.

6.3 Simulations, Comparisons, and Discussion

The proposed algorithm was incorporated in the JSVM 9.19.14 reference software [111]. In order to validate the effectiveness of the proposed algorithm, video sequences with different degrees of motion activity and picture detail were coded and the results compared with using only the JVT-W043 algorithm. The first 150 frames of each video sequence were coded to generate a reliable result. Two spatial layers were evaluated and the proposed algorithm was applied to the enhancement layer. RDO was always enabled, and CABAC entropy coding was used. Fast search motion estimation was employed with a search range of ±32 pixels and quarter pixel precision. Adaptive inter-layer prediction was enabled for the enhancement layer. As the proposed algorithm attempts to optimise the RD and MAD prediction models, which involve only P-frames, the GOP structure was set to IPPP. In this work, a macroblock was chosen as the basic unit for rate control. To compare the results with the JVT-W043 algorithm, the initial Qp value is set to 32 for both schemes. For the other coding parameters, the default values as specified in the manual of the JSVM reference software [111] were used.

In order to evaluate the coding performance of the proposed rate control algorithm, the following parameters were measured against the JVT-W043 scheme.

1. The MAD prediction accuracy, which is measured in terms of the MAD prediction error ($Er_{\mathrm{MAD}}, \%$) averaged over all coded macroblocks in a sequence. It is computed accord-

ing to

$$Er_{\mathrm{MAD}} = \frac{1}{N} \cdot \frac{1}{H} \cdot \frac{1}{W} \sum_{i=1}^{N} \sum_{j=1}^{H} \sum_{k=1}^{W} \left| \mathrm{MAD}_{\mathrm{pred}}^{i,j,k} - \mathrm{MAD}_{\mathrm{act}}^{i,j,k} \right| \tag{6.20}$$

where N is the total number of coded frames, H and W refer to the number of macroblocks in the vertical and horizontal direction respectively, $\mathrm{MAD}_{\mathrm{pred}}$ and $\mathrm{MAD}_{\mathrm{act}}$ denote the predicted MAD value and the actual MAD value, respectively;

2. The bit rate mismatch (Mism. %) between the target bit rate and the actual bit rate, which is calculated according to

$$\mathrm{Mism.} = \left| \mathrm{BR}_{\mathrm{tgt}} - \mathrm{BR}_{\mathrm{act}} \right| / \mathrm{BR}_{\mathrm{tgt}} \times 100\% \tag{6.21}$$

where $\mathrm{BR}_{\mathrm{tgt}}$ and $\mathrm{BR}_{\mathrm{act}}$ denote the target bit rate and the actual bit rate, respectively;

3. RD performance in terms of PSNR (dB), Bit Rate (BR, kbits/s), BDPSNR (dB) and BDBR (%) as described in [96].

6.3.1 MAD Prediction Accuracy

Table 6-3 shows the MAD prediction accuracy of the proposed MAD prediction model and the linear prediction model used in JVT-W043, for four standard test sequences. For each sequence, four target bit rates are evaluated. All the test sequences and bit rates used in the experiments are those recommended by the JVT in document JVT-Q205 [112].

It can be seen that the proposed MAD prediction model significantly improves the MAD prediction accuracy compared with the JVT-W043 prediction model. Evaluation results show that the proposed MAD prediction model reduces the MAD prediction error by 15%-79% relative to the linear prediction model.

From Table 6-3, it is apparent that the MAD prediction error reduction relates to the content of the video sequence. The proposed model performs best on video sequences containing irregular motion or scene changes, such as 'Football', but also shows a considerable MAD prediction error reduction for video sequences with moderate movement,

such as 'Bus' and 'Foreman'. For instance, in the case of the 'Foreman' sequence with moderately complex motion and picture detail, the MAD prediction error is reduced by over 37%. For the 'Football' sequence comprising globally fast movement and complex detail, an average reduction of 64% in MAD prediction error is achieved. The maximum prediction error reduction is 79% for the sequence containing fast and irregular motion. For the sequence containing complex picture detail and regular motion, such as 'Mobile', the proposed model provides a prediction error reduction of greater than 15%.

Table 6-3
MAD prediction accuracy comparison for spatial enhancement layer

Sequence	BR_{tgt} (BL/EL)	Er_{MAD}		Reduction (%)	Average (%)
		JVT-W043	Proposed		
Bus	96/384	1.12	0.81	-27.68	-36.00
	128/512	1.04	0.71	-31.73	
	192/768	0.94	0.59	-37.23	
	320/1280	0.95	0.50	-47.37	
Football	192/768	1.58	0.78	-50.63	-64.24
	256/1024	1.59	0.66	-58.49	
	384/1536	1.61	0.50	-68.94	
	640/2560	1.80	0.38	-78.89	
Foreman	48/192	0.56	0.39	-30.36	-37.82
	64/256	0.54	0.35	-35.19	
	96/384	0.53	0.31	-41.51	
	160/640	0.52	0.29	-44.23	
Mobile	64//256	1.06	0.83	-21.70	-17.26
	96/384	1.02	0.84	-17.65	
	128/512	1.02	0.87	-14.71	
	192/768	1.00	0.85	-15.00	
Average		**1.06**	**0.60**	**-38.83**	**-38.83**

This improvement in MAD prediction accuracy is important. For some sequences, such as 'Football', the JVT-W043 linear model using temporal correlation performs poorly, but the proposed MAD prediction model provides an acceptable solution. With the proposed model, the coding results from the base layer are used to assist the MAD prediction in the enhancement layers. When encoding the enhancement layers, the encoder is made

aware of the abrupt changes of MAD in advance and promptly adjusts the MAD prediction to reduce the prediction errors. Due to the improvement in the MAD prediction accuracy, the bits are allocated more appropriately, and not only is there an improvement in the rate control accuracy but also an increase in the quality of the reconstructed video.

To conclude, the proposed MAD prediction model improves the accuracy of MAD prediction for all types of video sequence, especially for sequences with fast motion or scene changes.

6.3.2 Rate Control Accuracy and RD Performance

The encoding results of the spatial enhancement layer using the proposed rate control mechanism are compared with JVT-W043. The rate control accuracy and the comparative RD performance results for four target bit rates are shown in Table 6-4. QCIF sequences were used for the base layer and CIF were used for the enhancement layer. Again, all the test sequences and bit rates used in the experiments are those recommended in document JVT-Q205 [112].

It can be seen that both the proposed algorithm and the JVT-W043 scheme work well in terms of the rate control accuracy at various target bit rates. Although both methods produce the target bit rates, the accuracy of the proposed algorithm is better than JVT-W043 in most cases. This is because the proposed optimised MAD prediction model results in a smaller prediction error when fast motion occurs. Most of the mismatch errors are less than 0.1% and the maximum error is 0.47%. The overall average absolute mismatch error is 0.14%. Consequently, it can be concluded that bit rate is precisely controlled using the proposed algorithm.

The proposed rate control mechanism also achieves better RD performance for the enhancement layers than the JVT-W043 scheme. The comparative performance results are also shown in Table 6-4. The results show that 1) given the same bit rate, the proposed algorithm increases the average PSNR by up to 0.28dB, and 2) given the same video quality

(PSNR), the proposed algorithm produces an average saving in bit rate of up to 5.13%, compared to the JVT-W043 algorithm. In general, the PSNR of each of the four sequences is increased at all ranges of target bit rate. Therefore, the proposed algorithm improves the coding efficiency compared with the JVT-W043 rate control algorithm, and this is true regardless of target bit rate.

Fig. 6-5 shows the RD curves for each of the four video sequences under a variety of bit rates. From Fig. 6-5, it can be seen the RD curves of the proposed algorithm are always higher than those of the JVT-W043,which means the proposed algorithm achieves better coding efficiency than the JVT-W043.

In order to test the robustness of the proposed algorithm, it was applied to video sequences of larger spatial resolution and with multiple enhancement layers.

Table 6-5 compares the coding performance with JVT-W043 when higher resolution images are coded: CIF for the base layer and 4CIF for the enhancement layer. The corresponding RD curves are presented in Fig. 6-6. Table 6-6 summarises the performance for three spatial layers: QCIF for the base layer, CIF for the first enhancement layer and 4CIF for the second enhancement layer. Fig. 6-7 shows the corresponding RD curves.

The results in Tables 6-5 and 6-6, show that the proposed algorithm consistently results in a low bit rate mismatch error, yet produces higher coding efficiency compared with the JVT-W043 algorithm. Specifically, all the mismatch errors obtained by the proposed algorithm are less than 0.23%, and the proposed algorithm produces coding gains of up to 0.28dB in average PSNR and savings of up to 7.78% bit rate on average.

Table 6-4
Comparison of proposed algorithm with JVT-W043 when encoding QCIF/CIF sequences

Sequence	BR_{tgt} (BL/EL)	JVT-W043 [113]			Proposed			BDBR	BDPSNR
		BR	PSNR	Mism.	BR	PSNR	Mism.		
Bus	96/384	384.4	28.05	0.11	384.5	28.19	0.14	-3.23	0.15
	128/512	512.5	29.37	0.11	512.4	29.49	0.07		
	192/768	768.2	31.22	0.03	768.2	31.39	0.03		
	320/1280	1279.8	33.76	0.02	1280.2	33.92	0.02		
Football	192/768	768.3	32.86	0.04	768.7	33.11	0.08	-5.13	0.28
	256/1024	1024.4	34.27	0.04	1024.5	34.54	0.05		
	384/1536	1536.5	36.41	0.03	1536.5	36.71	0.03		
	640/2560	2556.0	39.31	0.16	2560.1	39.58	0.00		
Foreman	48/192	193.1	32.72	0.57	192.8	32.77	0.44	-4.13	0.17
	64/256	257.4	33.90	0.53	257.2	34.03	0.47		
	96/384	385.2	35.48	0.32	385.3	35.67	0.33		
	160/640	640.4	37.45	0.07	640.4	37.70	0.07		
Mobile	64/256	256.9	23.98	0.36	256.7	24.13	0.29	-2.47	0.10
	96/384	384.9	25.58	0.23	384.7	25.66	0.18		
	128/512	512.1	26.70	0.02	512.3	26.79	0.06		
	192/768	767.2	28.25	0.10	768.2	28.36	0.02		
Average				**0.17**			**0.14**	**-3.74**	**0.18**

Table 6-5
Comparison of proposed algorithm with JVT-W043 when encoding CIF/4CIF sequences

Sequence	BR_{tgt} (BL/EL)	JVT-W043 [113]			Proposed			BDBR	BDPSNR
		BR	PSNR	Mism.	BR	PSNR	Mism.		
City	256/1024	1026.9	32.75	0.28	1025.5	32.79	0.15	-3.13	0.09
	384/1536	1538.9	33.98	0.19	1537.3	34.08	0.08		
	512/2048	2048.7	34.80	0.03	2049.0	34.88	0.05		
	768/3072	3073.5	35.85	0.05	3072.5	35.97	0.02		
Crew	384/1536	1534.6	36.75	0.09	1536.6	36.94	0.04	-7.60	0.20
	512/2048	2046.8	37.53	0.06	2048.2	37.73	0.01		
	768/3072	3067.0	38.57	0.16	3072.3	38.78	0.01		
	1280/5120	5112.0	39.82	0.16	5119.5	40.03	0.01		
Harbor	384/1536	1537.7	31.46	0.11	1536.3	31.58	0.02	-4.25	0.16
	512/2048	2049.3	32.49	0.06	2048.2	32.65	0.01		
	768/3072	3067.0	34.00	0.16	3072.0	34.17	0.00		
	1280/5120	5118.9	35.89	0.02	5120.0	36.07	0.00		
Soccer	384/1536	1536.2	34.29	0.01	1536.9	34.49	0.06	-6.39	0.23
	512/2048	2046.4	35.35	0.08	2048.3	35.62	0.01		
	768/3072	3068.4	36.82	0.12	3072.2	37.06	0.01		
	1280/5120	5110.8	38.54	0.18	5119.5	38.73	0.01		
Average				**0.11**			**0.03**	**-5.34**	**0.17**

Table 6-6
Comparison of proposed algorithm with JVT-W043 when encoding QCIF/CIF/4CIF sequences

Sequence	BR_{tgt} (BL/EL1/EL2)	EL	JVT-W043 [113]			Proposed			BDBR	BDPSNR
			BR	PSNR	Mism.	BR	PSNR	Mism.		
City	64/256/1024	1st	256.7	31.82	0.26	256.6	31.88	0.23	-4.09	0.17
			384.5	33.42	0.12	384.4	33.66	0.10		
			511.6	34.64	0.08	512.4	34.80	0.08		
	96/384/1536		768.3	36.32	0.04	768.3	36.47	0.03		
	128/512/2048	2nd	1026.8	32.76	0.27	1025.8	32.75	0.18	-3.47	0.10
	192/768/3072		1537.4	33.95	0.09	1537.2	34.07	0.08		
			2049.4	34.77	0.07	2049.0	34.88	0.05		
			3074.5	35.83	0.08	3072.6	35.95	0.02		
Crew	96/384/1536	1st	383.7	35.12	0.07	384.3	35.37	0.07	-5.99	0.28
			511.4	36.44	0.12	512.1	36.72	0.03		
			766.7	38.32	0.17	768.1	38.63	0.01		
	128/512/2048		1278.6	40.59	0.11	1279.8	40.84	0.02		
	192/768/3072	2nd	1534.4	36.68	0.10	1536.7	36.88	0.05	-7.78	0.21
	320/1280/5120		2046	37.48	0.10	2048.4	37.67	0.02		
			3068	38.52	0.13	3072.3	38.74	0.01		
			5110.7	39.76	0.18	5119.7	39.99	0.01		

(Continued on next page)

Table 6-6
Comparison of proposed algorithm with JVT-W043 when encoding QCIF/CIF/4CIF sequences

(Continued from last page)

Sequence	BR_{tgt} (BL/EL1/EL2)	EL	JVT-W043 [113]			Proposed			BDBR	BDPSNR
			BR	PSNR	Mism.	BR	PSNR	Mism.		
Harbor	96/384/1536	1st	384.1	28.87	0.02	384.2	29.03	0.05	-2.78	0.12
			510.4	29.99	0.31	512.1	30.13	0.03		
			767.4	31.69	0.07	768.0	31.81	0.00		
	128/512/2048		1277.4	34.02	0.20	1280.0	34.12	0.00		
	192/768/3072	2nd	1534.2	31.39	0.12	1536.4	31.50	0.03	-4.58	0.17
	320/1280/5120		2047.0	32.41	0.05	2048.3	32.58	0.01		
			3071.3	33.91	0.02	3072.2	34.10	0.01		
			5116.5	35.81	0.07	5119.9	35.99	0.00		
Soccer	96/384/1536	1st	384.3	33.37	0.09	384.6	33.63	0.15	-6.82	0.34
			511.9	34.60	0.03	512.2	34.99	0.04		
			767.7	36.56	0.04	767.9	36.91	0.01		
	128/512/2048		1278.0	39.14	0.16	1279.6	39.47	0.03		
	192/768/3072	2nd	1537.6	34.20	0.10	1536.8	34.41	0.05	-6.04	0.22
	320/1280/5120		2047.2	35.30	0.04	2048.4	35.54	0.02		
			3068.5	36.76	0.11	3072.0	36.99	0.00		
			5110.7	38.50	0.18	5119.6	38.69	0.01		
Average					**0.11**			**0.04**	**-5.19**	**0.20**

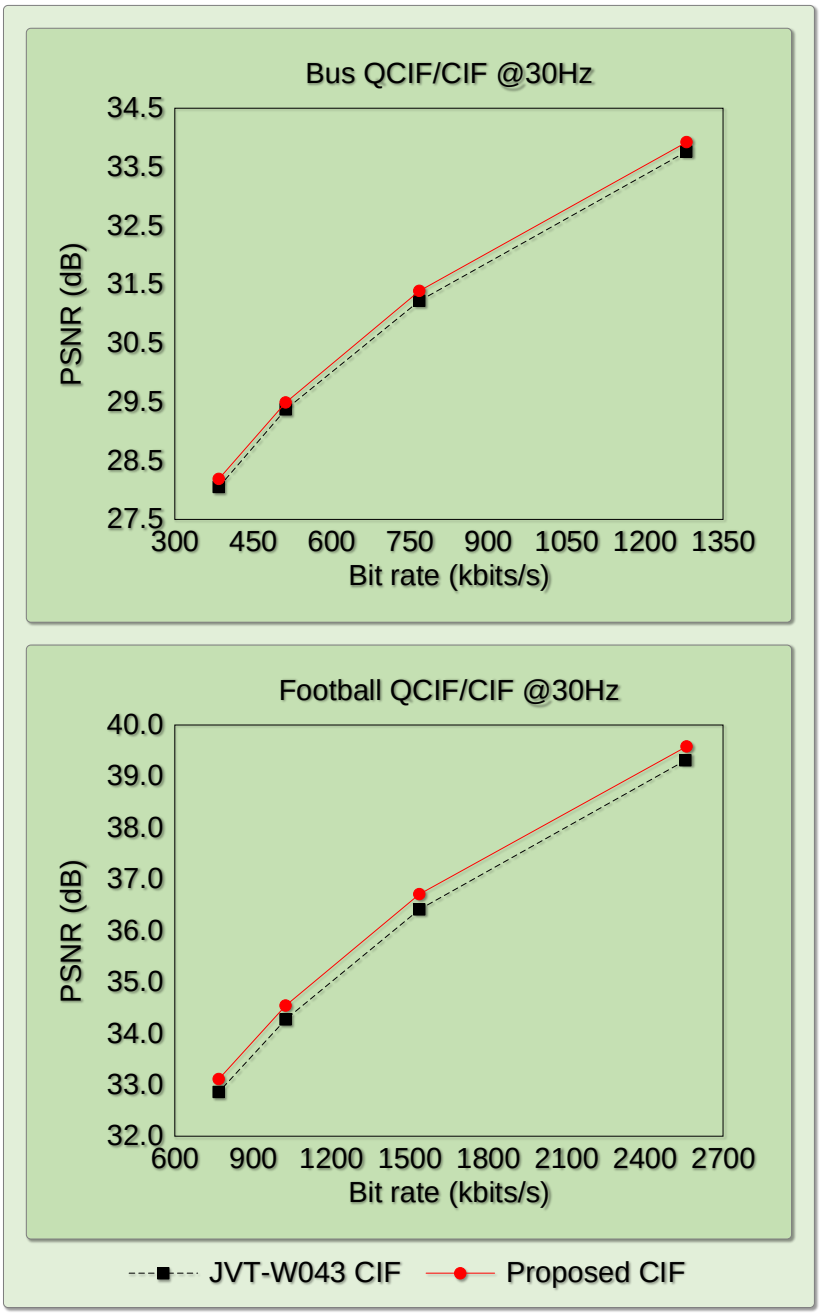


Fig. 6-5 RD performance for QCIF/CIF video sequences (continued on next page).

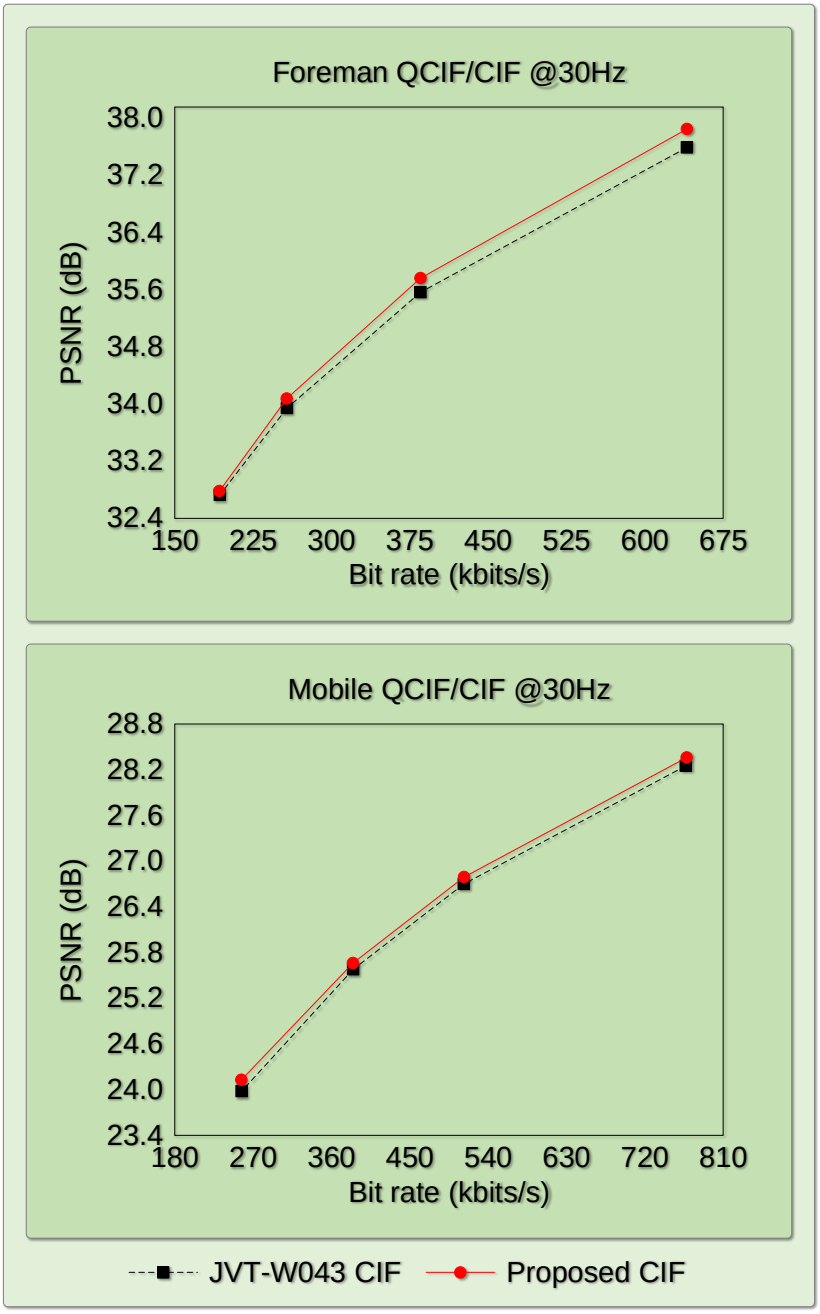


Fig. 6-5 RD performance for QCIF/CIF video sequences (continued from last page).

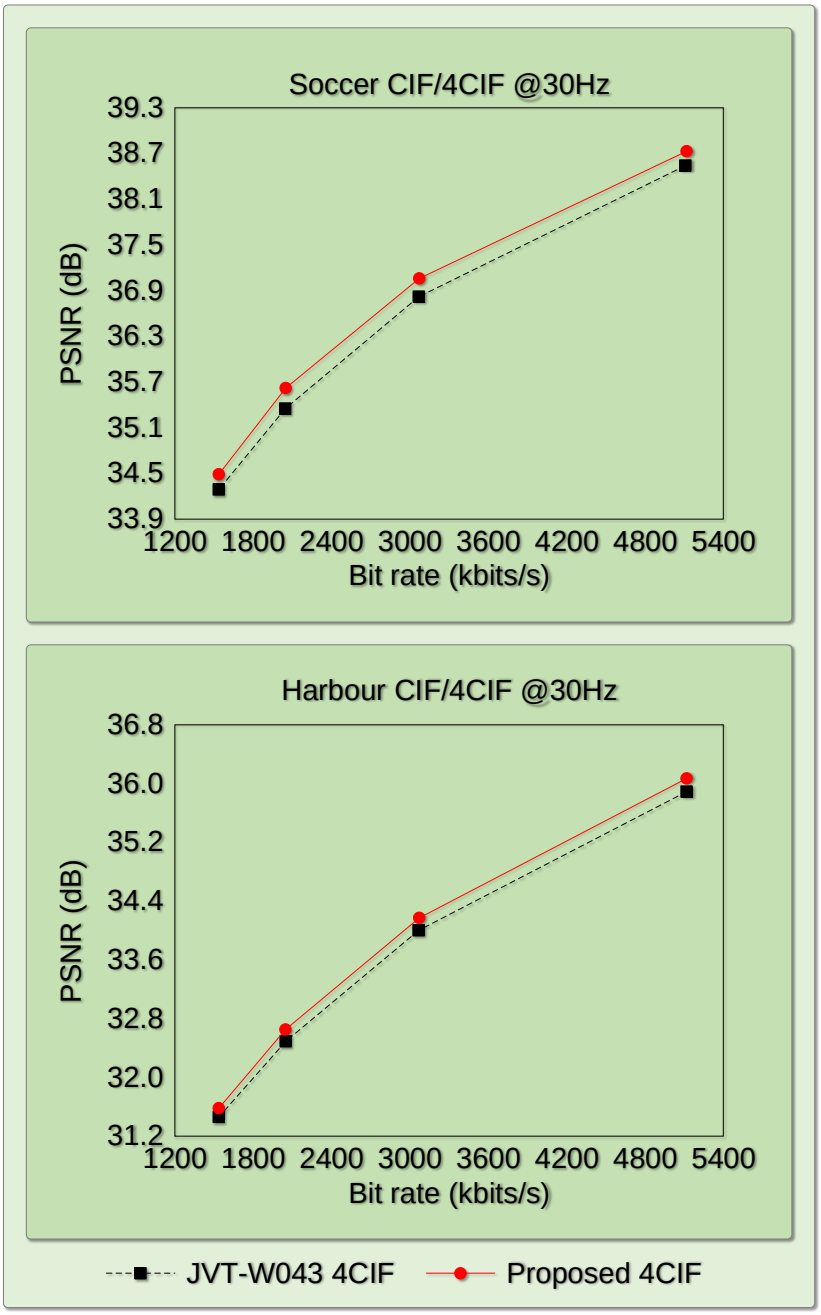


Fig. 6-6 RD performance for CIF/4CIF video sequences (continued on next page).

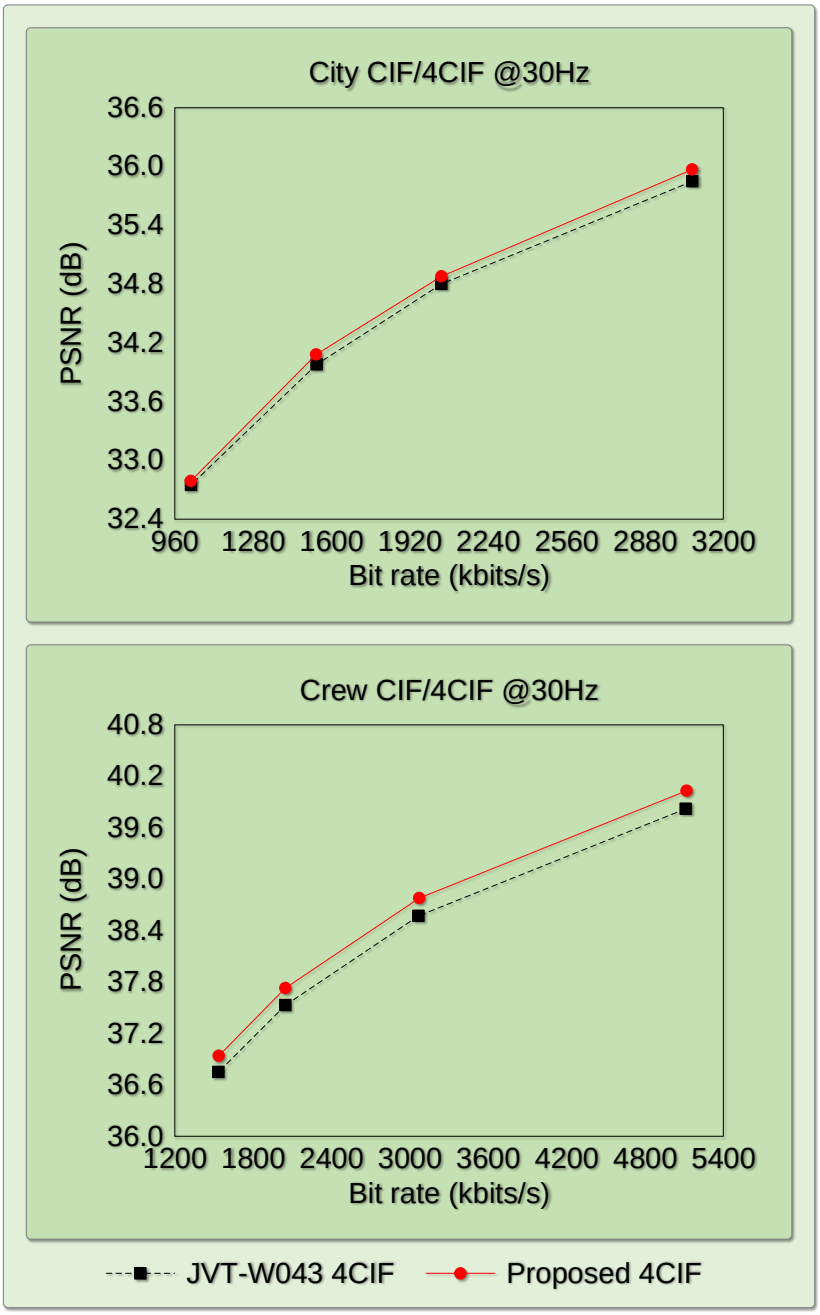


Fig. 6-6 RD performance for CIF/4CIF video sequences (continued from last page).

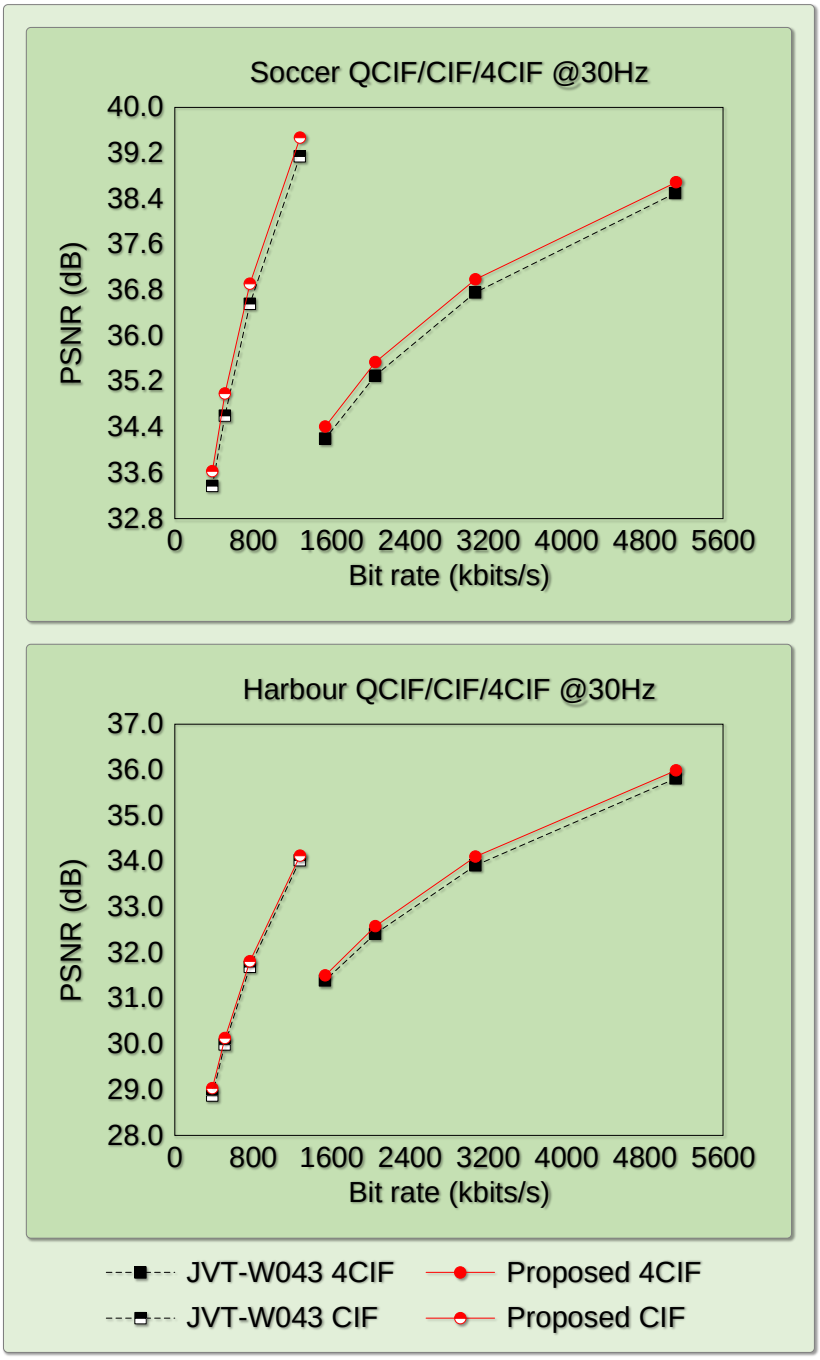


Fig. 6-7 RD performance for QCIF/CIF/4CIF video sequences (continued on next page).

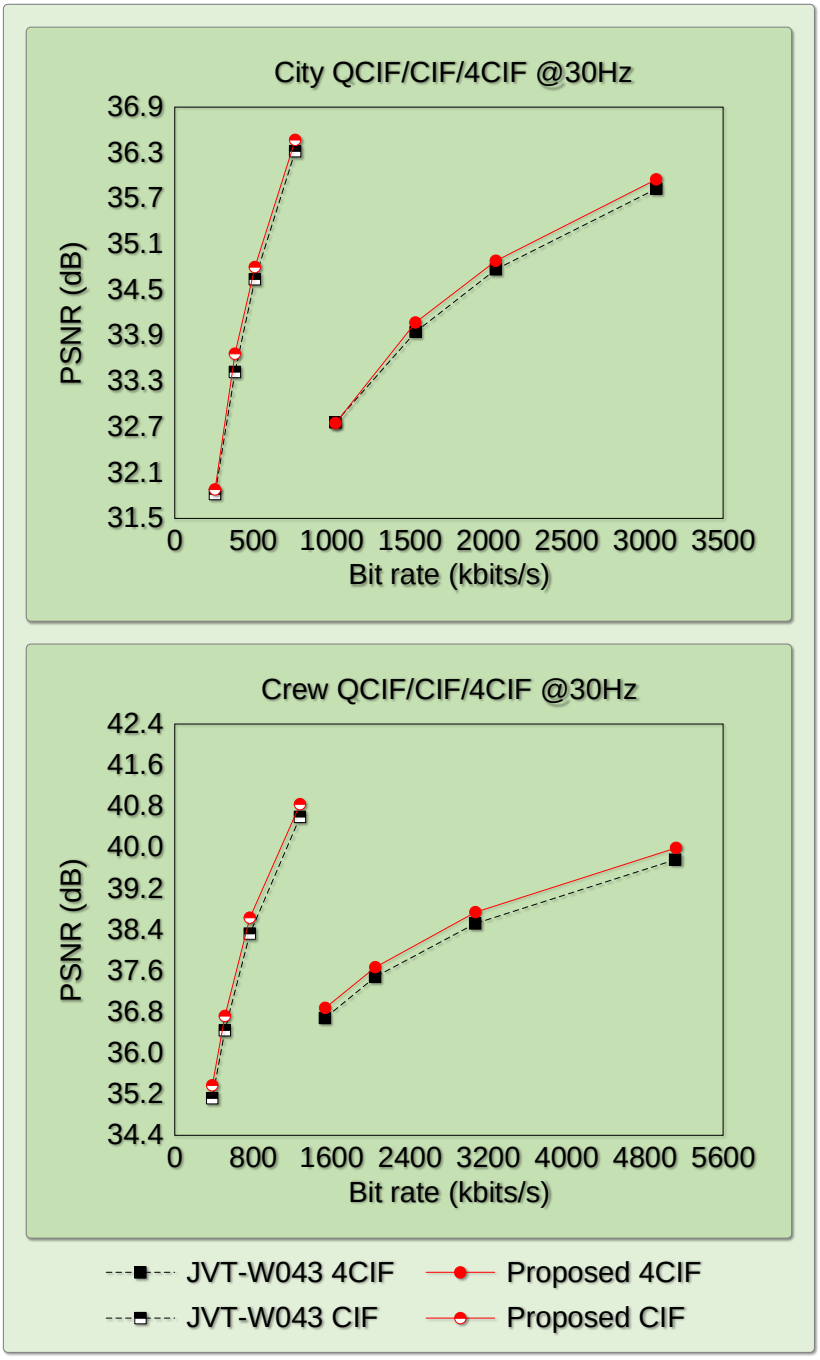


Fig. 6-7 RD performance for QCIF/CIF/4CIF video sequences (continued from last page).

6.4 Summary

Rate control is an important component of a video coder and is employed to make the best use of available network resources. With an efficient rate control scheme, frame skipping and wastage of channel resources can be avoided. Although rate control is not a normative tool in most video coding standards, it has been studied extensively in the application of different video coding standards.

This chapter initially described existing rate control algorithms suggested for SVC. Subsequently, a rate control scheme for the spatial enhancement layer in SVC was described in section 6.2. The scheme introduces a separate RD model for inter-layer prediction coding in the enhancement layer. An improved MAD prediction model was also proposed, where the MAD from previous temporal frames and the base layer are used together to assist the MAD prediction in the enhancement layers.

In applying each of the above techniques, both the target bit rate mismatch is reduced and the coding efficiency is significantly improved. Performance comparisons with JVT-W043 were presented in section 6.3. Simulation results show that the proposed method achieves better rate control accuracy than the JVT-W043 scheme, the average rate control mismatch error being 0.07%. Furthermore, the proposed algorithm accomplishes higher coding efficiency, namely a gain in PSNR of up to 0.34dB and a saving in bit rate of up to 7.78%. It should be noted that although the proposed algorithm does not generate state-of-the-art performance in terms of RD performance, it only exploits the properties of the enhancement layers in SVC. It can be readily combined with the existing rate control algorithms for H.264/AVC, thus achieving further improvements.

Chapter 7

Conclusions and Further Work

The work described in this thesis has been concerned with improvements to the latest scalable video coding standard. The rapid developments in network and multimedia technologies result in an increasing number of video applications in heterogeneous environments. Facing the challenges posed by heterogeneous networks, diverse terminal devices and various user demands, high coding efficiency is not the only goal of video coding systems but also the ability to meet the requirements of these new video coding challenges. SVC is the scalable extension of the H.264/AVC standard, and it is designed to address the needs of applications and to make the video bitstream more adaptable. SVC provides temporal, spatial and quality scalability in a single bitstream, furthermore, it achieves better coding efficiency than the scalable profiles of previous video coding standards. In this thesis, a new fast inter-frame and inter-layer coding mode selection process for SVC based on motion activity is proposed. This process is extended into a hierarchical scheme providing significant improvements in performance. A rate control scheme with RD analysis of prediction modes is also presented. The following three sections summarise the key contributions and draw several conclusions. Section 7.4 provides some directions for further research, and concluding remarks are given in section 7.5.

7.1 Performance Comparison of Advanced Scalable Video Coding Schemes

Three scalable video coding algorithms have been standardised in recent years. SVC, the scalable extension of H.264/MPEG-4 AVC, is the most recent international standard, Motion JPEG2000 is part 3 of the JPEG2000 image coding standard, and WSVC is a strong competitor in the scalable video coding field. All three of the video coding algorithms have been designed to produce scalable video bitstreams. SVC employs a multi-layer coding structure, Motion JPEG2000 independently codes each video frame using JPEG2000, and WSVC uses MCTF and the DWT.

Little research has compared the performance of the three scalable video coding algorithms. This work is timely given that the next generation of scalable video coding, Scalable High-efficiency Video Coding (SHVC) built on H.265/High Efficiency Video Coding (HEVC), is expected to be finalised next year, and there will be a continued drive to further improve coding efficiency and to reduce computational complexity. In chapter 3, focus was placed on the difference between each coding algorithm, and a comparative study of the coding efficiency for high quality video coding using SVC, Motion JPEG2000 and WSVC was conducted.

Chapter 3 began by introducing the core encoding tools in each coding algorithm. Each of the coding schemes was reviewed mainly in terms of its transform, quantisation, and entropy coding tools. SVC utilises the integer DCT, scalar quantisation, and the CAVLC and CABAC entropy coding methods. Motion JPEG2000 employs the DWT, uniform dead zone scalar quantisation, and the EBCOT entropy coding technique. The MCTF, DWT, embedded quantisation and the ESCOT entropy coding algorithm are incorporated in WSVC.

The comparison of each coding scheme concentrates on the performance in terms of coding efficiency. The three video coding schemes were examined when video sequences with different resolutions and picture content are processed. Empirical evaluation re-

sults [114] showed that the optimal choice among the three coding schemes depends on the application scenario. When only intra-coding tools are enabled, the experimental results show that WSVC achieves a better coding performance than Motion JPEG2000 and SVC for video within a wide range of resolutions. When encoding low and medium resolution video sequences, SVC consistently outperforms Motion JPEG2000 over a wide range of bit rates. When high resolution video sequences are coded, Motion JPEG2000 achieves a better performance than SVC at the lower bit rates. However, SVC outperforms Motion JPEG2000 for high resolution video sequences when the bit rate is high.

7.2 Fast Algorithms for SVC

The high degree of scalability and superior coding efficiency of SVC are achieved at the cost of significantly increased computational complexity. Many time-consuming encoding tools are incorporated in SVC and they impose a considerable computational burden on the SVC encoder. Therefore, it is desirable to design fast mode decision algorithms to alleviate the computational complexity of the encoder, while maintaining high compression performance and coding efficiency. In chapters 4 and 5, two fast implementations, namely a fast inter-frame and inter-layer mode decision algorithm and a hierarchical scheme for fast mode selection are described.

7.2.1 Fast Inter-frame and Inter-layer Mode Decisions

SVC employs a multi-layer scheme to support scalabilities. In the spatial and quality enhancement layers, SVC incorporates inter-layer prediction mechanisms to complement the H.264/AVC motion estimation and mode decision processes. Although inter-layer prediction in SVC can significantly improve the coding efficiency of the enhancement layers, the use of additional inter-layer prediction tools also imposes a considerable computational burden. In SVC, both the number of bits generated by a partition mode as well as the

degree of distortion are taken into account when choosing the optimal coding mode for each macroblock. For an enhancement layer, all the modes concerned with inter-frame prediction, intra-frame prediction and inter-layer prediction are evaluated. The mode with the minimum RD cost is selected as the best mode for the current macroblock. The implementation of the mode selection process in the enhancement layers necessitates a large amount of computational resources due to the requirements of motion estimation and RDO. In particular, inter-layer residual prediction doubles the computational complexity of the mode decision process in H.264/AVC. Inter-layer motion prediction also increases the complexity of the motion estimation in mode evaluation process.

In order to reduce the computational requirements of the enhancement layers, several fast mode decision approaches have been developed. In chapter 4, a fast inter-frame and inter-layer mode decision algorithm is proposed for the SVC enhancement layers.

There is a common tendency that a macroblock with slow movement is more likely to be best matched by one in the same resolution layer. However for a macroblock with fast movement, motion estimation between layers is required. This property forms the basis for the proposed simple but effective fast mode decision algorithm. Only a subset of mode candidates is required to be evaluated, provided it can be determined in advance whether the current macroblock is more suitable for inter-layer prediction or inter-frame prediction, in other words, whether the current macroblock represents slow or fast motion. The MVD between P-frames in each GOP is used to determine the video motion activity. It can then be decided which prediction mode should be applied to the current macroblock, inter-layer prediction or inter-frame prediction according to the degree of motion. A full cycle of Lagrangian RDO can be avoided by discarding the least likely candidate modes.

The proposed fast algorithm [115] is implemented into the coding process of the JSVM 9.18 reference software recommended by JVT. The performance is evaluated using a set of standard video test sequences with different degrees of motion activity and picture detail. Each test sequence, comprising 150 frames, is processed with different Qp values ranging

from 28 to 40. The simulation compares the performance with the standard JSVM implementation. The experimental results, shown by the RD relationship curves, indicate that the proposed fast algorithm incurs a negligible PSNR degradation and an insignificant increase in bit rate. In terms of speedup, the proposed fast implementation demonstrates a time reduction over the JSVM of approximately 30% on average. In conclusion, the simulations show that the proposed fast algorithm achieves very similar coding performance in terms of picture degradation and bit rate increase as that of the standard JSVM 9.18 implementation, yet reduces the encoding time by up to 40%.

The proposed fast inter-frame and inter-layer mode decision algorithm exhibits the following strengths:

1. **Simplicity**: MVD is the only parameter used to decide which block modes should be examined, and it is easy to extract this information from the coded data. This supports the goal of the research, namely to reduce computational complexity.
2. **Effectiveness**: The simulations show that, in all cases, encoding time is reduced significantly, yet there is negligible degradation in PSNR and insignificant increment in bit rate.

The proposed algorithm also shows the following limitations:

1. A constant MAD threshold was used to categorise the video motion activity. It is preferable to choose the optimum MAD threshold in an adaptive manner according to the context.
2. Motion activity was employed as the only factor to speed up the encoding process in the proposed fast algorithm. If other available information, such as inter-layer correlation and spatial correlation were taken into consideration, it is expected that the computational requirements can be reduced further.

7.2.2 Hierarchical Scheme for Fast Mode Selection

In chapter 5, a hierarchical fast mode decision process for the SVC extension of H.264/AVC was described. Unlike existing fast algorithms, the proposed algorithm is robust, and it makes full use of inter-layer, temporal and spatial correlation, as well as the texture information of each macroblock. Consequently, it produces better results than existing best-performing methods.

The proposed algorithm [116, 117] employs a coding structure comprising four hierarchical levels. In each level of the structure, different strategies are triggered to discard unlikely modes whilst retaining the most probable ones. In this way, RDO is performed only for the most likely candidate modes, and the unnecessary computation of the unlikely ones is avoided. The first level of the proposed algorithm relies on the strong mode dependencies that exist between the base layer and the enhancement layer. The partition mode of the corresponding macroblock in the base layer is used to refine the candidate modes for the current macroblock in the enhancement layer. In the second level, the reduction of candidate modes depends on the general tendency that the best mode for the current macroblock is very likely to be the same as its neighbours. The common observation that larger partition sizes are more suitable for homogeneous regions, and smaller partition sizes are beneficial for detailed areas implicitly forms the basis for the third level of the proposed coding algorithm. The last level identifies the amount of motion and then chooses the subset of candidate modes to be evaluated. The reason for this is that regions which contain slow motion and high spatial detail are most likely to be coded using an inter-frame prediction mode.

The performance of the proposed fast scheme is compared to that of the standard JSVM implementation. The performance is evaluated with Qp values ranging from 28 to 40 using various standard test sequences at different resolutions. When encoding the QCIF/CIF video sequences, evaluation results show that the proposed scheme reduces the encoding time by an average of 70.4% compared to the JSVM encoder. The coding ef-

ficiency loss is negligible, as the PSNR loss is 0.02dB-0.10dB and the increase in bit rate between 0.43% and 1.96%.

The simulations also compare the coding performance with the standard JSVM encoder when higher resolution images and more spatial layers are coded. The experimental results confirm that the proposed algorithm consistently achieves a significant reduction in computational complexity, while keeping nearly the same RD performance as the JSVM encoder.

In addition to the JSVM encoder, the proposed algorithm is compared also with three best performing fast mode decision algorithms under identical test conditions [90, 93, 94]. The comparisons demonstrate that the proposed algorithm consistently outperforms the approaches previously proposed in terms of encoding time reduction. With regards to the RD performance, the proposed fast algorithm produced a negligible increase in bit rate and visually imperceptible decrease in PSNR.

Although the proposed fast algorithm provides the best results, it also has the following limitations. It performs best for video sequences containing smooth movement and low texture detail. Less significant encoding time reduction is obtained for sequences containing high motion and spatial detail, as the evaluation of somewhat more prediction modes is required. Addressing this problem could form the basis of future research.

To conclude, the proposed scheme uses a considerable amount of information to improve coding speed. Inter-layer correlation is exploited by reusing the partition mode of the corresponding macroblock in the base layer. Spatial correlation is exploited by considering neighbouring macroblocks, and temporal correlation is used by consideration of the MVD. Furthermore, the homogeneity of the macroblock is also measured. Each of these factors is considered in the mode selection process, making the proposed method effective and robust. This is largely true regardless of video sequence and coding conditions.

7.3 Rate Control for SVC with Optimised RD Model

Rate control is a very important part of any video coding system, and SVC is no exception. Rate control enables the encoder to produce the output bitstream at a constant rate. The rate control scheme allocates more bits for complex pictures to maintain acceptable picture quality and fewer bits for simple pictures. In this way, the quality of the video sequence as a whole is maximised. The main purpose of rate control is to regulate the bitstream according to the available bandwidth, a predefined buffer size, or the storage capability, so that the video quality is maintained as high as possible. A well-designed rate control scheme should achieve the target bit rate, transmission delay, and optimal picture quality under various conditions. In chapter 6, the RD properties of the different SVC prediction modes are analysed. A more accurate RD model is then proposed for the spatial enhancement layer in SVC. Due to the bottom-up coding structure of SVC, some encoding results of the base layer can be used to inform the coding of the enhancement layers. Therefore, a new MAD prediction model for the spatial enhancement layer using the encoding results from the base layer as a factor in the MAD prediction procedure is proposed.

In SVC, inter-layer prediction is utilised to improve the coding efficiency of the enhancement layers. However the rate control scheme in the JSVM software lacks consideration of the implications of inter-layer prediction as it was designed for non-scalable video encoders. From observation and analysis, macroblocks coded using inter-layer prediction and those coded by intra-layer prediction have dissimilar statistical properties. This leads to serious prediction errors in the RD model. Considering the influence of inter-layer prediction, a two-part RD model is proposed. By applying the above model, a more accurate Qp estimation is achieved, thus improving the coding efficiency.

In the quadratic RD model, the MAD value is required to estimate the Qp value, but it can only be obtained after RDO where the Qp value is a necessary parameter. In the standard JSVM implementation, the MAD value of the current basic unit is predicted using the

MAD value of the basic unit in the same position of the previous frame. In the encoding process, the base layer is encoded first, followed by the enhancement layers. This leads to the idea that some encoding results of the base layer can be used as a reference for encoding the enhancement layers. A new MAD prediction model is developed that simultaneously considers the MAD from previous temporal frames and the reference frame in the base layer. Due to the high dependency between layers, the encoder is made aware of the abrupt changes of MAD in advance and promptly adjusts the MAD prediction to reduce the prediction errors. In this way, the bits are allocated more appropriately and not only is there an improvement in the rate control accuracy, but also an increase in the quality of the reconstructed video.

The proposed rate control algorithm [118] is examined in terms of the MAD prediction error, bit rate mismatch and RD performance. The simulation results indicate that the proposed MAD prediction model reduces the MAD prediction error by an average of 43% compared with the JSVM implementation. The control accuracy of the proposed algorithm is maintained to within 0.07% on average. Compared with the JSVM rate control scheme, the proposed algorithm improves the average PSNR by up to 0.34dB or produces an average saving in bit rate of up to 7.78%.

In conclusion, the proposed rate control algorithm significantly reduces the MAD prediction error and achieves better coding performance in terms of picture quality and compression ratio compared to the JSVM implementation, yet maintains the bit rate mismatch within a very low level.

7.4 Directions for Further Work

Further work could focus on the scalable extension of H.265/HEVC, more precisely, on fast algorithms and rate control schemes for HEVC. HEVC is the latest video compression standard developed by the JCT-VC. The first version of HEVC was approved in January 2013.

The popularity of HDTV (e.g. 1280×720 or 1920×1080 resolution) and the expected popularity of UHDTV (e.g. 3840×2160 or 7680×4320 resolution) place greater demands on coding efficiency than H.264/AVC. HEVC has been designed to meet all the application requirements of H.264/AVC and with particular emphasis on two key issues: higher picture resolution and the increased use of parallel processing architectures [38]. The next generation of scalable video coding SHVC, built on HEVC, is under development and is expected to be finalised next year. There will be a continued drive to further improve coding efficiency and to reduce computation complexity.

7.4.1 Fast Algorithms for HEVC

HEVC aims to achieve a significant improvement in compression performance relative to the H.264/AVC standard. In particular, it aims to reduce the bit rate by half that of H.264 while achieving equivalent perceptual video quality. This superior coding efficiency is achieved at the expense of higher complexity. It is reported that the complexity of an HEVC encoder is several times higher than that of an H.264/AVC encoder, if all advanced encoding capabilities of HEVC are utilised. These features include a quadtree structure, increased intra-coding directions, sophisticated interpolation filters, various in-loop filters, and enhanced entropy coding schemes.

The significantly increased computational complexity makes real-time applications with reasonable processing capability a critical obstacle to the use of HEVC. Consequently there is a need for a fast HEVC encoder to be developed, ideally one which results in negligible loss in coding efficiency.

It is observed that a substantial encoding complexity increase occurs in the encoding modules of intra picture prediction, motion compensation, and the transform. Computational complexity reduction of these encoding tools is expected to be an active research area for the next few years.

7.4.2 Rate Control for HEVC

Although HEVC was approved as an international standard in early 2013, little work has been done to find a feasible and effective rate control scheme. The HEVC Test Model (HM) is not currently equipped with a rate control module. Encoding HEVC video is different to H.264/AVC because of the influence of the new HEVC coding tools, e.g. additional coding modes, quadtree structure, etc. It is reported that simply transplanting the H.264/AVC rate control algorithm into HEVC results in a severe loss in coding efficiency. Thus a more appropriate rate control scheme for HEVC is required. It may be possible to extend the algorithm proposed in this thesis to make it appropriate for HEVC. However, several technical requirements have to be taken into consideration:

1. Would it improve the picture quality under the bit rate constraint?
2. Would the output bitstream match the target bit rate accurately?
3. Would full use of the encoding buffer be made, so that neither overflow nor underflow takes place?

7.5 Concluding Remarks

This thesis initially compared the coding efficiency in coding high quality video using SVC, Motion JPEG2000 and WSVC. Thus a thorough understanding of scalable video coding was gained. Subsequently, the thesis details several improved algorithms for both coding mode selection and rate control in SVC. In the mode selection chapters, the contribution focuses mainly on the design of fast implementations for the SVC scalable video coding standard. Chapters 4 and 5 analysed the additional computation imposed by the RDO and inter-layer prediction. These two chapters also reviewed the research on fast intra-coding, inter-coding, and inter-layer coding, and improvements for each coding scheme were described. The performance of the proposed algorithms have been evaluated extensively in terms of picture degradation, bit rate increment, rate distortion, and computa-

tional speedup. Overall, the new hierarchical technique demonstrates the same coding performance in terms of the picture quality and compression ratio as that of the JSVM implementation, the SVC standard reference software, yet achieves a saving in encoding time of up to 84%. This is true regardless of the video content and the coding conditions.

In chapter 6 rate control based on RD analysis of prediction modes was developed. The statistical properties of macroblocks coded using inter-layer prediction and those coded by intra-layer prediction were analysed in detail. A two-part RD model was then proposed. In addition, the encoding results from the base layer were used to assist MAD prediction in the enhancement layers, and a new MAD prediction model was developed. Overall, the proposed rate control scheme was shown to achieve higher coding efficiency, namely a coding gain of up to 0.34dB in PSNR and a saving of 7.78% in bit rate compared with the default rate control method of SVC.

Bibliography

[1] Youtube. [online]. http://www.youtube.com/yt/press/statistics.html (Retrieved 22-09-2013).

[2] M. Robin and M. Poulin. *Digital Television Fundamentals: Design and Installation of Video and Audio Systems (2nd Edition).* McGraw-Hill, 2000.

[3] G. Collins. *Fundamentals of Digital Television Transmission.* Wiley, 2000.

[4] P. Cianci. *High Definition Television: The Creation, Development and Implementation of HDTV Technology.* McFarland, 2012.

[5] A. Bock. *Video Compression Systems: From First Principles to Concatenated Codecs.* IET, 2009.

[6] A. Tekalp. *Digital Video Processing.* Prentice Hall, 1995.

[7] P. Kaiser and R. Boynton. *Human Color Vision (2nd Edition).* Optical Society of America, 1996.

[8] Y. Shi and H. Sun. *Image and Video Compression for Multimedia Engineering: Fundamentals, Algorithms, and Standards.* CRC Press, 1999.

[9] A. Netravali and B. Haskell. *Digital Pictures: Representation, Compression and Standards (2nd Edition).* Springer, 1995.

[10] M. Al-Mualla, C. Canagarajah, and D. Bull. *Video Coding for Mobile Communications: Efficiency, Complexity and Resilience.* Academic Press, 2002.

[11] R. Clarke. *Transform Coding of Images.* Academic Press, 1985.

[12] W. Pratt, J. Kane, and H. Andrews. Hadamard transform image coding. *Proc. IEEE*, 57(1):58–68, Jan. 1969.

[13] N. Ahmed, T. Natarajan, and K. Rao. Discrete cosine transform. *IEEE Trans. Comput.*, C-23(1):90–93, Jan. 1974.

[14] K. Rao and P. Yip. *Discrete Cosine Transform: Algorithms, Advantages, Applications.* Academic Press, 1990.

[15] D. Huffman. A method for the construction of minimum-redundancy codes. *Proc. IRE*, 40(9):1098–1101, Sep. 1952.

[16] G. Langdon. An introduction to arithmetic coding. *IBM J. Res. Develop.*, 28(2):135–149, Mar. 1984.

[17] R. Schäfer and T. Sikora. Digital video coding standards and their role in video communications. *Proc. IEEE*, 83(6):907–924, Jun. 1995.

[18] ITU-T. *ITU-T Rec. H.261: Video codec for audio-visual services at p×64 kbits.* Mar. 1993.

[19] ITU-T. *ITU-T Rec. H.263: Video coding for low bit rate communication.* Mar. 1996.

[20] L. Chiariglione. MPEG and multimedia communications. *IEEE Trans. Circuits Syst. Video Technol.*, 7(1):5–18, Feb. 1997.

[21] L. Chiariglione. Impact of MPEG standards on multimedia industry. *Proc. IEEE*, 86(6):1222–1227, Jun. 1998.

[22] ISO/IEC MPEG. *ISO/IEC 11172 Part 2 (Video): Coding of moving pictures and associated audio for digital storage media at up to about 1.5 Mbit/s.* 1993.

[23] A. Puri and A. Eleftheriadis. MPEG-4: an object-based multimedia coding standard supporting mobile applications. *Mobile Netw. Appl.*, 3(1):5–32, Jun. 1998.

[24] ITU-T and ISO/IEC MPEG. *ITU-T Rec. H.262 and ISO/IEC 13818 Part 2 (Video): Generic coding of moving pictures and associated audio information.* 1995.

[25] B. Haskell, A. Puri, and A. Netravali. *Digital Video: An Introduction to MPEG-2.* Springer, 1997.

[26] J. Watkinson. *MPEG-2.* Focal Press, 1999.

[27] ISO/IEC MPEG. *ISO/IEC 14496 Part 2 (Visual): Coding of audio-visual objects.* 1999.

[28] W. Pennebaker and J. Mitchell. *JPEG: Still Image Data Compression Standard.* Springer, 1992.

[29] Y. Wang, J. Ostermann, and Y. Zhang. *Video Processing and Communications.* Prentice Hall, 2002.

[30] M. Ghanbari. *Standard Codecs: Image Compression to Advanced Video Coding (3rd Edition).* IET, 2011.

[31] R. Koenen. *Overview of the MPEG-4 standard.* ISO/IEC JTC1/SC29/WG11, MPEG 40th Meeting, Doc. N1730, Stockholm, Sweden, Jul. 1997.

[32] JVT of ITU-T VCEG and ISO/IEC MPEG. *ITU-T Rec. H.264 and ISO/IEC 14496 MPEG-4 Part 10 (AVC): Advanced video coding for generic audiovisual services.* 2003.

[33] I. Richardson. *H.264 and MPEG-4 Video Compression: Video Coding for Next-generation Multimedia.* Wiley, 2003.

[34] T. Wiegand, G. Sullivan, G. Bjϕntegaard, and A. Luthra. Overview of the H.264/AVC video coding standard. *IEEE Trans. Circuits Syst. Video Technol.*, 13(7):560–576, Jul. 2003.

[35] A. Luthra and P. Topiwala. Overview of the H.264/AVC video coding standard. In *Proc. SPIE Conf. on Applicat. of Digital Image Process. XXVI*, volume 5203, pp. 417–431, 2003.

[36] A. Puri, X. Chen, and A. Luthra. Video coding using the H.264/MPEG-4 AVC compression standard. *Signal Process. Image Commun.*, 19(9):793–849, Oct. 2004.

[37] S. Kwon, A. Tamhankar, and K. Rao. Overview of H.264/MPEG-4 part 10. *J. Vis. Commun. Image Represent.*, 17(2):186–216, Apr. 2006.

[38] G. Sullivan, J. Ohm, W. Han, and T. Wiegand. Overview of the High Efficiency Video Coding (HEVC) standard. *IEEE Trans. Circuits Syst. Video Technol.*, 22(12):1649–1668, Dec. 2012.

[39] JCT-VC of ITU-T VCEG and ISO/IEC MPEG. *ITU-T Rec. H.265 and ISO/IEC 23008 MPEG-H Part 2 (HEVC): High efficiency video coding.* 2013.

[40] J. Ohm and G. Sullivan. High efficiency video coding: the next frontier in video compression [Standards in a Nutshell]. *IEEE Signal Process. Mag.*, 30(1):152–158, Jan. 2013.

[41] H. Hang, W. Peng, C. Chan, and C. Chen. Towards the next video standard: High Efficiency Video Coding. In *Proc. Asia-Pacific Signal and Inform. Process. Assoc. Annu. Summit and Conf.*, pp. 609–618, 2010.

[42] J. Ohm. Advances in scalable video coding. *Proc. IEEE*, 93(1):42–56, Jan. 2005.

[43] H. Huang, W. Peng, T. Chiang, and H. Hang. Advances in the scalable amendment of H.264/AVC. *IEEE Commun. Mag.*, 45(1):68–76, Jan. 2007.

[44] H. Schwarz, D. Marpe, and T. Wiegand. Overview of the scalable video coding extension of the H.264/AVC standard. *IEEE Trans. Circuits Syst. Video Technol.*, 17(9):1103–1120, Sep. 2007.

[45] H. Schwarz, D. Marpe, and T. Wiegand. *Hierarchical B pictures*. JVT of ISO/IEC MPEG & ITU-T VCEG, 16th JVT Meeting, Doc. JVT-P014, Poznań, Poland, Jul. 2005.

[46] H. Schwarz, D. Marpe, and T. Wiegand. Analysis of hierarchical B pictures and MCTF. In *Proc. IEEE Int. Conf. Multimedia Expo*, pp. 1929–1932, 2006.

[47] C. Segall and G. Sullivan. Spatial scalability within the H.264/AVC scalable video coding extension. *IEEE Trans. Circuits Syst. Video Technol.*, 17(9):1121–1135, Sep. 2007.

[48] W. Li. Overview of fine granularity scalability in MPEG-4 video standard. *IEEE Trans. Circuits Syst. Video Technol.*, 11(3):301–317, Mar. 2001.

[49] R. Schäfer, H. Schwarz, D. Marpe, T. Schierl, and T. Wiegand. MCTF and scalability extension of H.264/AVC and its application to video transmission, storage, and surveillance. In *Proc. SPIE Visual Commun. Image Process.*, pp. 1–6, 2005.

[50] H. Schwarz, D. Marpe, T. Schierl, and T. Wiegand. Combined scalability support for the scalable extension of H.264/AVC. In *Proc. IEEE Int. Conf. Multimedia Expo.*, pp. 1–4, 2005.

[51] G. Sullivan and T. Wiegand. Video compression-from concepts to the H.264/AVC standard. *Proc. IEEE*, 93(1):18–31, Jan. 2005.

[52] H. Malvar, A. Hallapuro, M. Karczewicz, and L. Kerofsky. Low-complexity transform and quantization in H.264/AVC. *IEEE Trans. Circuits Syst. Video Technol.*, 13(7):598–603, Jul. 2003.

[53] G. Bjϕntegaard and K. Lillevold. *Context-adaptive VLC (CVLC) coding of coefficients.* JVT of ISO/IEC MPEG & ITU-T VCEG, 3rd JVT Meeting, Doc. JVT-C028, Fairfax, USA, May 2002.

[54] D. Marpe, H. Schwarz, and T. Wiegand. Context-based adaptive binary arithmetic coding in the H.264/AVC video compression standard. *IEEE Trans. Circuits Syst. Video Technol.*, 13(7):620–636, Jul. 2003.

[55] Z. Chen, J. Syu, and P. Chang. Fast inter-layer motion estimation algorithm on spatial scalability in H.264/AVC scalable extension. In *Proc. IEEE Int. Conf. Multimedia Expo.*, pp. 442–446, 2010.

[56] T. Berger. *Rate Distortion Theory: A Mathematical Basis for Data Compression.* Prentice Hall, 1971.

[57] J. Woods. *Multidimensional Signal, Image, and Video Processing and Coding (2nd Edition).* Academic Press, 2011.

[58] T. Wiegand, H. Schwarz, A. Joch, F. Kossentini, and G. Sullivan. Rate-constrained coder control and comparison of video coding standards. *IEEE Trans. Circuits Syst. Video Technol.*, 13(7):688–703, Jul. 2003.

[59] C. Park, S. Baek, M. Yoon, H. Kim, and S. Ko. Selective inter-layer residual prediction for SVC-based video streaming. *IEEE Trans. Consum. Electron.*, 55(1):235–239, Feb. 2009.

[60] M. Wien, H. Schwarz, and T. Oelbaum. Performance analysis of SVC. *IEEE Trans. Circuits Syst. Video Technol.*, 17(9):1194–1203, Sep. 2007.

[61] D. Alfonso, M. Gherardi, A. Vitali, and F. Rovati. Performance analysis of the scalable video coding standard. In *Proc. Int. Conf. Packet Video*, pp. 243–252, 2007.

[62] F. Kelly, A. Maulloo, and D. Tan. Rate control for communication networks: shadow prices, proportional fairness and stability. *J. Oper. Res. Soc.*, 49(3):237–252, Mar. 1998.

[63] Test Model Editing Committee. *MPEG-2 Video Test Model 5 (TM5)*. ISO/IEC JTC1/SC29/WG11, MPEG Meeting, Doc. N0400, Sydney, Australia, Apr. 1993.

[64] T. Gardos. *Video Codec Test Model, Near-Term, Version 8 (TMN8)*. ITU-T VCEG, 1st VCEG Meeting, Doc. Q15-A-59, Portland, USA, Jun. 1997.

[65] Video Group. *MPEG-4 Video Verification Model Version 8.0 (VM8)*. ISO/IEC JTC1/SC29/WG11, MPEG 40th Meeting, Doc. N1796, Stockholm, Sweden, Jul. 1997.

[66] Z. He and S. Mitra. Optimum bit allocation and accurate rate control for video coding via ρ-domain source modeling. *IEEE Trans. Circuits Syst. Video Technol.*, 12(10):840–849, Oct. 2002.

[67] K. Seo, S. Heo, and J. Kim. Adaptive rate control algorithm based on logarithmic R-Q model for MPEG-1 to MPEG-4 transcoding. *Signal Process. Image Commun.*, 17(10):857–875, Nov. 2002.

[68] T. Chiang and Y. Zhang. A new rate control scheme using quadratic rate distortion model. *IEEE Trans. Circuits Syst. Video Technol.*, 7(1):246–250, Feb. 1997.

[69] JPEG of ITU-T SG16 and ISO/IEC JTC1/SC29/WG1. *ITU-T Rec. T.800 and ISO/IEC 15444 Part 1 (Core Coding System): JPEG2000 image coding system.* 2000.

[70] JPEG of ITU-T SG16 and ISO/IEC JTC1/SC29/WG1. *ITU-T Rec. T.802 and ISO/IEC 15444 Part 3 (Motion JPEG2000): JPEG2000 image coding system.* 2002.

[71] D. Taubman and M. Marcellin. *JPEG2000: Image Compression Fundamentals, Standards, and Practice.* Springer, 2002.

[72] T. Acharya and P. Tsai. *JPEG2000 Standard for Image Compression: Concepts, Algorithms and VLSI Architectures.* Wiley, 2005.

[73] R. Xiong, J. Xu, F. Wu, and S. Li. Barbell-lifting based 3-D wavelet coding scheme. *IEEE Trans. Circuits Syst. Video Technol.*, 17(9):1256–1269, Sep. 2007.

[74] C. Christopoulos, A. Skodras, and T. Ebrahimi. The JPEG2000 still image coding system: an overview. *IEEE Trans. Consum. Electron.*, 46(4):1103–1127, Nov. 2000.

[75] A. Skodras, C. Christopoulos, and T. Ebrahimi. The JPEG2000 still image compression standard. *IEEE Signal Process. Mag.*, 18(5):36–58, Sep. 2001.

[76] K. Andra, C. Chakrabarti, and T. Acharya. A VLSI architecture for lifting-based forward and inverse wavelet transform. *IEEE Trans. Signal Process.*, 50(4):966–977, Apr. 2002.

[77] D. Taubman. High performance scalable image compression with EBCOT. In *Proc. IEEE Int. Conf. Image Process.*, pp. 344–348, 1999.

[78] D. Taubman. High performance scalable image compression with EBCOT. *IEEE Trans. Image Process.*, 9(7):1158–1170, Jul. 2000.

[79] N. Adami, A. Signoroni, and R. Leonardi. State-of-the-art and trends in scalable video compression with wavelet-based approaches. *IEEE Trans. Circuits Syst. Video Technol.*, 17(9):1238–1255, Sep. 2007.

[80] J. Xu, Z. Xiong, S. Li, and Y. Zhang. Three-dimensional embedded subband coding with optimized truncation (3-D ESCOT). *Appl. Comput. Harmonic Anal.*, 10(3):290–315, May 2001.

[81] JSVM 9.18 reference software. [online]. CVS server garcon.ient.rwth-aachen.de (Retrieved 22-09-2013).

[82] Kakadu reference software. [online]. http://www.kakadusoftware.com (Retrieved 22-09-2013).

[83] VidWav evaluation software. [online]. CVS server garcon.ient.rwth-aachen.de (Retrieved 22-09-2013).

[84] B. Shi, L. Liu, and C. Xu. Comparison between JPEG2000 and H.264 for digital cinema. In *Proc. IEEE Int. Conf. Multimedia Expo.*, pp. 725–728, 2008.

[85] R. Jiao. Performance comparison between AVC I-frame coding and JPEG2000. In *Proc. Int. Conf. Computer Modeling Simulation*, pp. 315–318, 2010.

[86] A. Yu and G. Martin. Advanced block size selection algorithm for inter frame coding in H.264/MPEG-4 AVC. In *Proc. IEEE Int. Conf. Image Process.*, pp. 95–98, 2004.

[87] D. Wu, F. Pan, K. Lim, S. Wu, Z. Li, X. Lin, S. Rahardja, and C. Ko. Fast intermode decision in H.264/AVC video coding. *IEEE Trans. Circuits Syst. Video Technol.*, 15(7):953–958, Jul. 2005.

[88] L. Shen, Z. Liu, Z. Zhang, and X. Shi. Fast inter mode decision using spatial property of motion field. *IEEE Trans. Multimedia*, 10(6):1208–1214, Oct. 2008.

[89] A. Yu, G. Martin, and H. Park. Fast inter-mode selection in the H.264/AVC standard using a hierarchical decision process. *IEEE Trans. Circuits Syst. Video Technol.*, 18(2):186–195, Feb. 2008.

[90] S. Kim, K. Konda, P. Mah, and S. Ko. Adaptive mode decision algorithm for inter layer coding in scalable video coding. In *Proc. IEEE Int. Conf. Image Process.*, pp. 1297–1300, 2010.

[91] H. Li, Z. Li, and C. Wen. Fast mode decision for coarse grain SNR scalable video coding. In *Proc. IEEE Int. Conf. Acoust. Speech Signal Process*, pp. 545–548, 2006.

[92] G. Goh, J. Kang, M. Cho, and K. Chung. Fast mode decision for scalable video coding based on neighboring macroblock analysis. In *Proc. ACM Symp. Appl. Comput.*, pp. 1845–1846, 2009.

[93] T. Zhao, H. Wang, and S. Kwong. Fast inter-layer mode decision in scalable video coding. In *Proc. IEEE Int. Conf. Image Process.*, pp. 4221–4224, 2010.

[94] B. Lee, M. Kim, S. Hahm, C. Park, and K. Park. A fast mode selection scheme in inter-layer prediction of H.264 scalable extension coding. In *Proc. IEEE Int. Symp. Broadband Multimedia Syst. Broadcast.*, pp. 1–5, 2008.

[95] A. Bovik. *Handbook of Image and Video Processing (2nd Edition)*. Academic Press, 2005.

[96] G. Bjϕntegaard. *Calculation of average PSNR differences between RD-curves*. ITU-T VCEG, 13th VCEG Meeting, Doc. VCEG-M33, Austin, USA, Apr. 2001.

[97] R. Zhang and M. Comer. Efficient inter-layer motion compensation for spatially scalable video coding. *IEEE Trans. Circuits Syst. Video Technol.*, 18(10):1325–1334, Oct. 2008.

[98] L. Xu, S. Ma, D. Zhao, and W. Gao. Rate control for scalable video model. In *Proc. SPIE, Visual Commun. Image Process.*, pp. 525–534, 2005.

[99] L. Xu, W. Gao, X. Ji, D. Zhao, and S. Ma. Rate control for spatial scalable coding in SVC. In *Proc. Picture Coding Symp.*, pp. 1–4, 2007.

[100] Y. Liu, Y. Soh, and Z. Li. Rate control for spatial/CGS scalable extension of H.264/AVC. In *Proc. IEEE Int. Symp. Circ. Syst.*, pp. 1746–1750, 2007.

[101] Y. Liu, Z. Li, and Y. Soh. Rate control of H.264/AVC scalable extension. *IEEE Trans. Circuits Syst. Video Technol.*, 18(1):116–121, Jan. 2008.

[102] S. Hu, H. Wang, S. Kwong, T. Zhao, and C. Kuo. Rate control optimization for temporal-layer scalable video coding. *IEEE Trans. Circuits Syst. Video Technol.*, 21(8):1152–1162, Aug. 2011.

[103] S. Hu, H. Wang, S. Kwong, and C. Kuo. Novel rate-quantization model-based rate control with adaptive initialization for spatial scalable video coding. *IEEE Trans. Ind. Electron.*, 59(3):1673–1684, Mar. 2012.

[104] J. Liu, Y. Cho, Z. Guo, and J. Kuo. Bit allocation for spatial scalability coding of H.264/SVC with dependent rate-distortion analysis. *IEEE Trans. Circuits Syst. Video Technol.*, 20(7):967–981, Jul. 2010.

[105] R. Zhang and M. Comer. Rate distortion analysis for spatially scalable video coding. *IEEE Trans. Image Process.*, 19(11):2947–2957, Nov. 2010.

[106] X. Li, P. Amon, A. Hutter, and A. Kaup. Performance analysis of inter-layer prediction in scalable video coding extension of H.264/AVC. *IEEE Trans. Broadcast.*, 57(1):66–74, Mar. 2011.

[107] J. Xie and L. Chia. Study on the distribution of DCT residues and its application to RD analysis of video coding. *J. Vis. Commun. Image Represent.*, 19(7):411–425, Oct. 2008.

[108] J. Ribas-Corbera and S. Lei. Rate control in DCT video coding for low-delay communications. *IEEE Trans. Circuits Syst. Video Technol.*, 9(1):172–185, Feb. 1999.

[109] Z. Li, F. Pan, K. Lim, G. Feng, X. Lin, and S. Rahardja. *Adaptive basic unit layer rate control for JVT.* JVT of ITU-T VCEG and ISO/IEC MPEG, 7th JVT Meeting, Doc. JVT-G012-r1, Pattaya II, Thailand, Mar. 2003.

[110] J. Rodgers and W. Nicewander. Thirteen ways to look at the correlation coefficient. *Am. Stat.*, 42(1):59–66, Feb. 1988.

[111] JSVM 9.19 reference software. [online]. CVS server garcon.ient.rwth-aachen.de (Retrieved 22-09-2013).

[112] M. Wien and H. Schwarz. *Testing conditions for SVC coding efficiency and JSVM performance evaluation.* JVT of ITU-T VCEG and ISO/IEC MPEG, 16th JVT Meeting, Doc. JVT-Q205, Poznan, Poland, Jul. 2005.

[113] A. Leontaris and A. Tourapis. *Rate control for the Joint Scalable Video Model (JSVM).* JVT of ITU-T VCEG and ISO/IEC MPEG, 24th JVT Meeting, Doc. JVT-W043, San Jose, USA, Apr. 2007.

[114] X. Lu and G. Martin. Performance comparison of the SVC, WSVC, and Motion JPEG2000 advanced scalable video coding schemes. In *Proc. IET Int. Conf. Intelligent Signal Process.*, pp. 1–6, 2013.

[115] X. Lu and G. Martin. Fast H.264/SVC inter-frame and inter-layer mode decisions based on motion activity. *IET Electron. Lett.*, 48(2):84–86, 2012.

[116] X. Lu and G. Martin. A hierarchical mode decision scheme for fast implementation of spatially scalable video coding. In *Proc. IEEE Visual Commun. Image Process.*, pp. 1–6, 2012.

[117] X. Lu and G. Martin. Fast mode decision algorithm for the H.264/AVC scalable video coding extension. *IEEE Trans. Circuits Syst. Video Technol.*, 23(5):846–855, May 2013.

[118] X. Lu and G. Martin. Rate control for scalable video coding with rate-distortion analysis of prediction modes. In *Proc. IEEE Int. Workshop Multimedia Signal Process.*, pp. 289–294, 2013.