Low Complexity Multiview Video Coding

Shadan Khattak

Faculty of Technology

De Montfort University

A thesis submitted for the degree of

Doctor of Philosophy

April 2014

To my family.

Abstract

3D video is a technology that has seen a tremendous attention in the recent years. Multiview Video Coding (MVC) is an extension of the popular H.264 video coding standard and is commonly used to compress 3D videos. It offers an improvement of 20% to 50% in compression efficiency over simulcast encoding of multiview videos using the conventional H.264 video coding standard. However, there are two important problems associated with it: (i) its superior compression performance comes at the cost of significantly higher computational complexity which hampers the real-world realization of MVC encoder in applications such as 3D live broadcasting and interactive Free Viewpoint Television (FTV), and (ii) compressed 3D videos can suffer from packet loss during transmission, which can degrade the viewing quality of the 3D video at the decoder. This thesis aims to solve these problems by presenting techniques to reduce the computational complexity of the MVC encoder and by proposing a consistent error concealment technique for frame losses in 3D video transmission.

The thesis first analyses the complexity of the MVC encoder. It then proposes two novel techniques to reduce the complexity of motion and disparity estimation. The first method achieves complexity reduction in the disparity estimation process by exploiting the relationship between temporal levels, type of macroblocks and search ranges while the second method achieves it by exploiting the geometrical relationship between motion and disparity vectors in stereo frames. These two methods are then combined with other state-of-the-art methods in a unique framework where gains add up. Experimental results show that the proposed low-complexity framework can reduce the encoding

time of the standard MVC encoder by over 93% while maintaining similar compression efficiency performance.

The addition of new View Synthesis Prediction (VSP) modes to the MVC encoding framework improve the compression efficiency of MVC. However, testing additional modes comes at the cost of increased encoding complexity. In order to reduce the encoding complexity, the thesis, next, proposes a bayesian early mode decision technique for a VSP enhanced MVC coder. It exploits the statistical similarities between the RD costs of the VSP SKIP mode in neighbouring views to terminate the mode decision process early. Results indicate that the proposed technique can reduce the encoding time of the enhanced MVC coder by over 33% at similar compression efficiency levels.

Finally, compressed 3D videos are usually required to be broadcasted to a large number of users where transmission errors can lead to frame losses which can degrade the video quality at the decoder. A simple reconstruction of the lost frames can lead to inconsistent reconstruction of the 3D scene which may negatively affect the viewing experience of a user. In order to solve this problem, the thesis proposes, at the end, a consistency model for recovering frames lost during transmission. The proposed consistency model is used to evaluate inter-view and temporal consistencies while selecting candidate blocks for concealment. Experimental results show that the proposed technique is able to recover the lost frames with high consistency and better quality than two standard error concealment methods and a baseline technique based on the boundary matching algorithm.

Declaration

I declare that the material presented in this thesis consists of original work undertaken solely by myself. Information derived from the published and unpublished work of others has been properly referenced. The material has not been submitted in substantially the same form for the award of a higher degree elsewhere.

Shadan Khattak

April 2014

Acknowledgements

I would like to thank my academic supervisors Professor Raouf Hamzaoui, Professor Pascal Frossard and Dr. Shakeel Ahmad for providing me with the best supervision and guidance that a PhD student can wish for. Their constructive criticism of my work always resulted in improving it. Apart from their technical expertise, I learnt a great deal from their uprightness, honesty, and dedication to their work.

I would also like to thank Dr. Thomas Maugey for the very fruitful discussions throughout the course of my PhD and for taking time out of his busy schedule to discuss and share ideas with me during my visits to EPFL.

I am also indebted to De Montfort University for awarding me bursary for three years and for the award of Laxton Bequest Travel Award which allowed me to present my work at the Picture Coding Symposium (PCS 2012) in Krakow, Poland.

I thank Poznan University of Technology for providing the "Poznan_Hall2" and "Poznan_Street" test sequences.

Last, but not the least, I am very thankful to my wife, my son, my parents, my brother and my sisters for their support and patience throughout the duration of my PhD.

Contents

List of Figures

List of Tables

List of Publications

[1] S. Khattak, T. Maugey, R. Hamzaoui, S. Ahmad, and P. Frossard, "Consistent error concealment for multiview plus depth video broadcasting," *IEEE Transactions on Circuits and Systems for Video Technology*, submitted.

[2] S. Khattak, R. Hamzaoui, T. Maugey, S. Ahmad, and P. Frossard, "Bayesian early mode decision technique for view synthesis prediction-enhanced multiview video coding," *IEEE Signal Processing Letters*, vol. 20, no. 11, pp. 1126–1129, Nov. 2013.

[3] S. Khattak, R. Hamzaoui, S. Ahmad, and P. Frossard, "Fast encoding techniques for multiview video coding," *Signal Processing: Image Communication*, vol. 28, no. 6, pp. 569–580, July 2013.

[4] S. Khattak, R. Hamzaoui, S. Ahmad, and P. Frossard, "Low-complexity multiview video coding," in *Proc. Picture Coding Symposium (PCS 2012)*, Krakow, Poland, May 2012, pp. 97–100.

[5] S. Khattak, "Framework for low-complexity multiview video coding," in *14th annual post graduate symposium on the convergence of Telecommunications, Networking, and Broadcasting (PGNet2013)*, Liverpool, UK, June 2013.

List of Abbreviations

AU	Access Unit.
BMA	Boundary Matching Algorithm.
CABAC	Context Adaptive Binary Arithmetic Code.
CAVLC	Context Adaptive Variable Length Code.
CLCMVC	Complete Low Complexity Multiview Video Coding.
CSV	Conventional Stereo Video.
DC	Disparity Compensation.
DCT	Discrete Cosine Transform.
DE	Disparity Estimation.
DES	Depth Enhanced Stereo.
DMS	Depth Motion Vector Sharing.
DV	Disparity Vector.
DVP	Disparity Vector Predictor.
FEC	Forward Error Correction.
FVV	Free Viewpoint Video.
GDV	Global Disparity Vector.
GOP	Group of Pictures.
HD	High Definition.

HEVC	High Efficiency Video Coding.
HVS	Human Visual System.
ICF	Inconsistency Cost Function.
IMVS	Inter-view Motion Vector Sharing.
IVI	Inter-view Inconsistency.
JMVC	Joint Multiview Video Coding.
JMVM	Joint Multiview Video Model.
JVT	Joint Video Team.
LDV	Layered Depth Video.
MB	Macroblock.
MC	Motion Compensation.
ME	Motion Estimation.
MOS	Mean Opinion Score.
MRS	Mixed Resolution Stereo.
MV	Motion Vector.
MVC	Multiview Video Coding.
MVD	Multiview Video plus Depth.
MVE	Motion Vector Extrapolation.
MVP	Motion Vector Predictor.
MVV	Multiview Video.
PDF	Probability Density Function.
PDV	Previous Disparity Vector.
PDV-DE	Previous Disparity Vector Disparity Estimation.
PLR	Packet Loss Rate.
PSNR	Peak Signal-to-Noise Ratio.
QP	Quantization Parameter.

RD	Rate-Distortion.
RDO-MD	Rate Distortion Optimized Mode Decision.
SMCC	Stereo Motion Consistency Constraint.
SMCC-MDE	Stereo Motion Consistency Constraint Motion and Disparity Estimation.
SR	Search Range.
SVC	Scalable Video Coding.
TI	Temporal Inconsistency.
TL	Temporal Level.
UHD	Ultra High Definition.
V+D	Video Plus Depth.
VSC	View Synthesis Concealment.
VSP	View Synthesis Prediction.
VSRS	View Synthesis Reference Software.

List of Notations

A	Intrinsic Matrix.
D	Distortion.
E	Extrinsic Matrix.
J	Rate-Distortion Cost.
R	Bit Rate.
Y	Pixel Value of a Depth Image.
Z	Physical Depth Value of a Pixel.
Z_{far}	Maximum Distance of a 3D Point from Camera.
Z_{near}	Minimum Distance of a 3D Point from Camera.
λ	Lagrange Multiplier.
e	Tolerance Threshold.
f_x	Focal Length in x-axis.
f_y	Focal Length in y-axis.
l	Length of a GOP.
o_x	x-Component of the Principal Point Offset.
o_y	y-Component of the Principal Point Offset.

Chapter 1

Introduction

1.1 Motivation

Video data has seen a rapid growth in the last few years. Its share in the global Internet and mobile traffic is expected to rise to unprecedented levels in the coming years [1]. Fig. 1.1 shows the forecast for global consumer internet traffic by segment. The figure clearly shows that a huge proportion of the overall internet traffic is covered by internet video. It is interesting to note that while both internet video and online gaming involve video data, the share of the former is far greater than that of the latter. This is because unlike internet video where actual transfer of the pixel data takes place, in online gaming, once the virtual world is downloaded, only the position information of gamers is transferred. The share of video data in the overall internet traffic is predicted to rise from 60% today to 70% in the next four years (Fig. 1.1). Thus, video has become one of the most popular and fast growing medium for information, entertainment and communication.

One of the most important advancements in video technology has been the introduction of 3D video. Its popularity is reflected in both the rising global sales of 3D enabled consumer electronics devices (See Fig. 1.2 for market forecast of 3D displays) and the rising number of digital 3D screens in cinemas.

Stereoscopic displays [3] are commonly used to watch 3D videos. While glasses are required to experience 3D sensation, prolonged use of glasses is widely re-

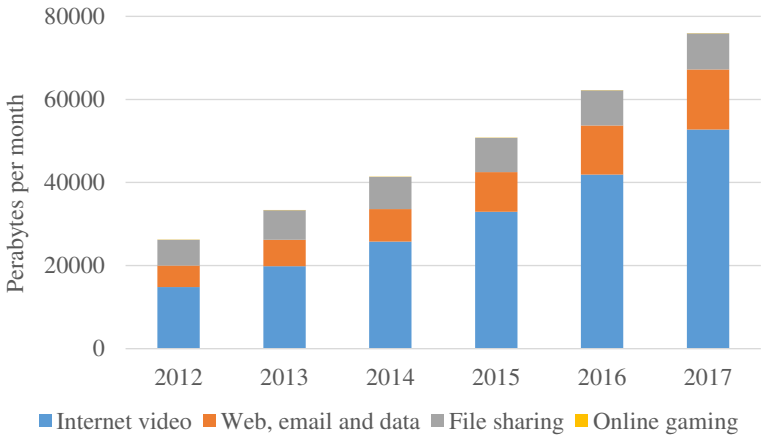


Figure 1.1: Global consumer internet traffic (Cisco Visual Networking Index, 2013 [1]).

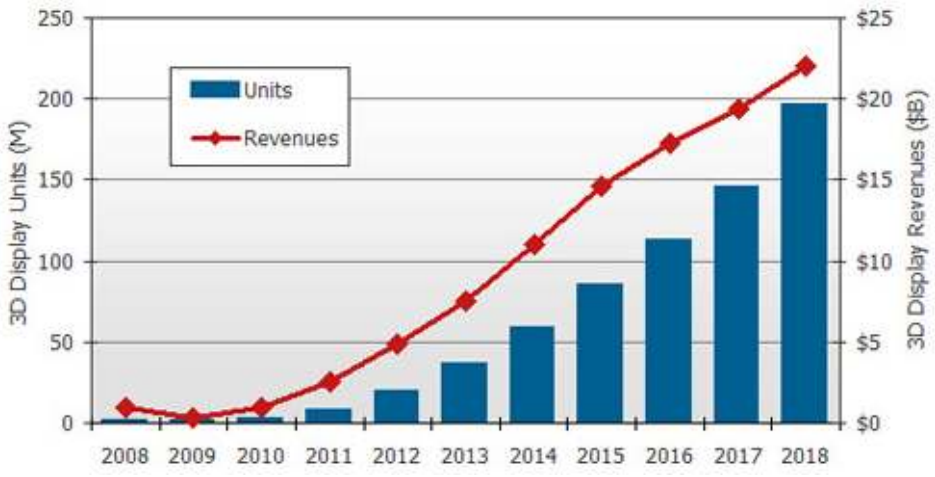


Figure 1.2: 3D display market forecast (DisplaySearch [2])

ported to cause visual discomfort and visual fatigue [4]. A more flexible and efficient alternative is to use autostereoscopic displays which allow the viewer to experience 3D sensation without wearing glasses [5]. These displays require the availability of a large number of views (e.g. Dimenco's Autostereoscopic 3D display with 28 views [6]) which dramatically increases the transmission bandwidth requirement [7].

Similar to 3D video, Free Viewpoint Video (FVV) [8] is another exciting video technology. Both these technologies expand the user's sensation beyond the classical 2D video. While 3D video enhances the visual experience by providing depth impression, FVV allows interactive viewpoint selection or free navigation in real scenes. These technologies do not exclude each other but rather can be combined in a single system to provide both depth impression and free navigation.

Multiview Video (MVV) is generally used as the underlying video format in both 3D video and FVV. The amount of MVV data increases proportionally with the number of views. Thus, for applications requiring a large number of views the resultant MVV data can be huge. Luckily the MVV data is highly redundant as each view represents information about the same scene albeit from different viewpoints (see Fig. 1.3).

(a) Viewpoint 1 (b) Viewpoint 2 (c) Viewpoint 3

Figure 1.3: An example of multiview video: frames from the Breakdancers sequence depicting a scene from three different viewpoints. The horizontal offsets between the three viewpoints can be clearly observed in the portions of the frames identified by the red rectangles.

A more efficient and promising format for 3D video is the Multiview Video

plus Depth (MVD) format [9]. It consists of multiple texture views and their associated depth maps. The availability of depth maps help reduce the transmission bandwidth requirement by transmitting only a subset of the required views to the decoder and generating the remaining views at the decoder by using depth image based rendering/view-synthesis [10][11] techniques (See Fig. 1.4).

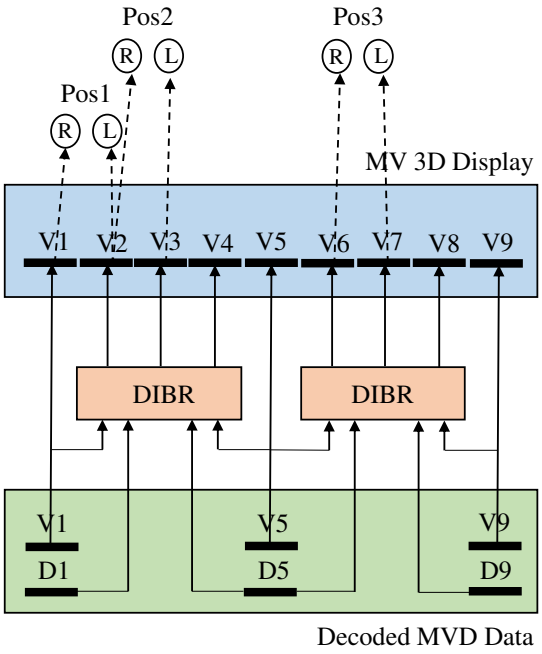


Figure 1.4: Advanced 3DTV concept based on MVD

While the advances in video technology have resulted in an exponential growth of the volume of video data, unfortunately, the underlying transmission and storage technologies have not evolved at the same speed. In order to allow efficient and cost-effective deployment of the new technologies using legacy transmission and storage facilities, it is necessary to efficiently compress the video data. Compression schemes are developed to achieve this. State-of-the-art video compression techniques can reduce the size of raw video by a factor of about 100 without any noticeable reduction in visual quality [12]. In order to allow inter-operability between a variety of devices, it is important to standardize these compression schemes. To this end, two organizations, ITU-T and ISO/IEC have developed

many compression standards over the last few decades (Fig. 1.5). These include the ITU-T H.26L family of standards including H.261 [13] (ratified in 1988, not shown in Fig. 1.5), H.263 [14], H.263+ [15] and H.263++ [16] and the ISO/IEC MPEG family of standards including the MPEG-1 [17] and MPEG-4 Visual [18]. Most modern standards have been a result of a joint effort of the two organizations. These include the H.262/MPEG-2 [19], H.264/MPEG-4 AVC [20, 21] including its two extensions: Scalable Video Coding (SVC) [22] and Multiview Video Coding (MVC) [23, 24], and High Efficiency Video Coding (HEVC) [25]. These standards were developed keeping in view the technologies at that time. Over time, newer standards, with better compression efficiency, have replaced the older ones (e.g., the latest HEVC standard provides approximately a 50% bit rate saving for equivalent perceptual quality compared to its predecessor H.264) [25].

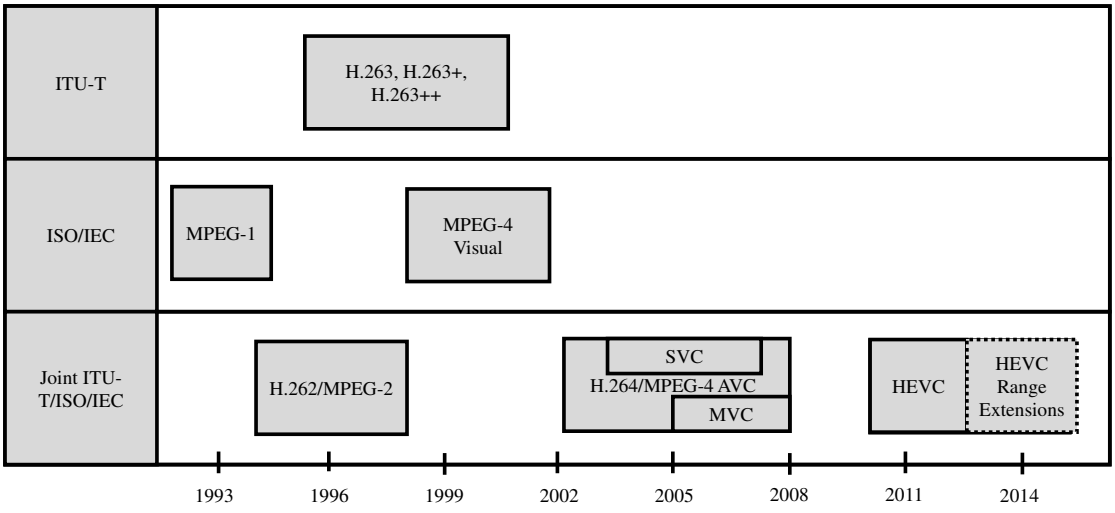


Figure 1.5: Evolution of video compression standards

For 3D video, the additional views required to provide flexible 3D sensation drastically increases the transmission bandwidth requirement. Using the conventional 2D video compression schemes to compress each view independently before transmission reduces the volume of data but only to a certain limit. This posed a new challenge to the video compression community. For efficient compression of 3D video, new methods had to be developed. The success of 3D markets globally

gave a new impetus to the video coding research. Standards aimed at 3D video started emerging.

Multiview Video Coding (MVC) [23, 24], developed as an extension of H.264, is aimed at efficient compression of multiview videos. MVC exploits the redundancy in MVV by using inter-view prediction methods. It allows a reduction of 20% to 50% in bit rates compared to simulcast transmission using H.264/MPEG-4 AVC [24]. Because of its superior compression performance and its support for backward compatibility with legacy 2D devices, it was selected by the Blu-Ray Disc Association as the coding format for 3D video with high-definition resolution [26]. In its design, its fundamental difference with the conventional standards is that while the conventional standards target two types of redundancies present in videos: (i) Spatial (similarities within a picture) and (ii) temporal (similarities between successive pictures), MVC additionally targets a third type i.e., inter-view redundancies (similarities between pictures of neighbouring views). While the methods employed for exploiting spatial redundancies (Intra Prediction) use simple computations (as they are based on predictions from the edges of neighbouring blocks of the same frame), temporal and inter-view redundancies are exploited by using computationally intensive methods (Motion Estimation (ME) and Disparity Estimation (DE) respectively) i.e., in these methods, neighbouring frames from the same view as well as from the adjacent views are searched during prediction. This makes the MVC encoder very slow. For example, for 8 encoded views, the coding complexity of MVC is 19× higher than that of the simulcast H.264 [27]. Though MVC allows a significant reduction in bit rates compared to H.264/MPEG-4 AVC, its slow speed makes it less appealing for use in time-constrained 3D applications. Thus, the encoding speed of MVC is vital for its widespread deployment.

Even though, compared to simulcast encoding, the MVV based MVC provides better compression, for most applications such as glasses-free 3D video using autostereoscopic displays, it can only provide a limited reduction in bit rate (typically around 20% [28], which is still far too high [7]). MVD format is a better choice in such applications as it can reduce the bitrate significantly. Apart from reducing the bitrate requirement by transmitting a subset of the available views, it also allows new prediction modes at the encoder which improve the compres-

sion efficiency [29]. The addition of new modes in the already complex encoding framework of modern block based video encoders further increases their computational complexity. Therefore, it is required to introduce novel algorithms that can allow a reduction in transmission bit rate without a significant computational burden on the encoder.

Another problem with the transmission of MVD videos is that in almost all realistic scenarios, the views that make up the 3D video, are transmitted over packet-based unreliable channels such as the internet or a wireless network. Packet loss in such networks is a common phenomenon. Forward Error Correction (FEC) is generally used to recover lost packets. However, FEC due to bandwidth constraints, FEC may not be able to recover all the lost packets. Similarly, TCP can be used to retransmit lost packets but in broadcast applications the lack of a feedback channel might not allow this. Thus in bandwidth limited broadcast applications, it is expected that the receiver may not receive all the video packets. This can degrade the video quality at the decoder. Error concealment techniques are employed to minimize the effect of packet loss on video quality. Many error concealment methods have been proposed recently for 2D and 3D video transmission. An important aspect of 3D video, often ignored in the existing error concealment methods, is the fact that while independently recovering each lost frame might be sufficient for 2D video, 3D videos require a consistent reconstruction of all the frames that make up the 3D scene. Inconsistent reconstruction of the frames can lead to an imperfect reconstruction of the 3D scene which is undesirable. Thus, for a pleasant viewing experience, it is important to introduce new methods for consistent error concealment of 3D videos.

To address the three problems mentioned above, the thesis will aim at:

- Reducing the time complexity of MVV based MVC encoding.
- Reducing the time complexity of MVD based MVC encoding enhanced with new prediction modes such as View Synthesis Prediction.
- Reducing the effect of packet loss on video quality in an MVD broadcast setup.

CPU time is commonly used for evaluating the time complexity of a video

encoder. Another, less common, method is to use the number of points at which the sum of absolute differences (SAD) is evaluated during motion and disparity estimation processes (search points). In this thesis, time complexity is first evaluated using both the CPU time and the number of search points in Chapter 3. It is found that both the methods provide similar results. Thus, in Chapter 4, time complexity is evaluated only using CPU time.

1.2 Contributions of the Thesis

The main contributions of the thesis are as follows:

- A fast disparity estimation method [30, 31] that exploits the correlation between the temporal level of a frame, and the view-neighbourhood of a macroblock to reduce the search range for disparity estimation. It achieves 35.28% reduction of encoding time, on average, compared to the standard JMVM 6.0 [32] reference software implementation. This method is presented in Section 3.4.1.

- A fast motion and disparity estimation method [31] that exploits the geometric consistency between the motion and disparity vectors of two consecutive stereo video frame pairs to reduce the motion and disparity search ranges. It achieves 41.78% reduction of encoding time, on average, compared to the standard JMVM 6.0 [32] reference software implementation. This method is presented in Section 3.4.2.

- A novel global solution for low-complexity multiview video coding [31, 33]. The solution integrates state-of-the-art algorithms with the above two methods into a unique framework to achieve an encoding time saving of over 93% on average, compared to the standard JMVM 6.0 [32] reference software implementation. Compared to state-of-the-art [34] this is an improvement of up to 11%. This method is presented in Section 3.4.3.

- An early mode decision method for View Synthesis Prediction mode enhanced multiview video coding [35]. The method exploits the bayesian decision rule to minimize the number of candidate modes checked during

encoding. It achieves time saving of over 33% compared to the standard JMVC 6.0 [36] reference software implementation. Compared to a baseline method based on the interview correlation technique proposed in [37], this is an improvement of up to 12%. This method is presented in Section 4.4.2.

- A novel consistency model for error concealment of MVD video that allows to maintain a high level of consistency between frames of the same view (temporal consistency) and those of the neighbouring views (inter-view consistency). The proposed technique outperforms two standard error concealment techniques ([38], [39]) and a baseline method based on Boundary Matching Algorithm (BMA) [40] with respect to both reconstruction quality and view consistency. This method is presented in Section 5.3.

1.3 Outline of the Thesis

The dissertation has been organized as follows:

- **Chapter 2:** This chapter begins with an introduction of the H.264 Video coding standard and its major features. It then provides an overview of the Multiview Video Coding extension of H.264, its features and applications. Common 3D video formats are presented next. At the end a brief description of the view synthesis technique is provided.

- **Chapter 3:** This chapter starts with a complexity analysis of the MVC encoder. It then proposes two new low-complexity techniques for MVC: (i) Previous Disparity Vector Disparity Estimation (PDV-DE) and (ii) Stereo Motion Consistency Constraint Motion and Disparity Estimation (SMCC-MDE). This is followed by a description of the proposed Complete Low Complexity Multiview Video Coding (CLCMVC), which is a novel framework that combines the two proposed methods with state-of-the-art methods to provide a global low-complexity solution.

- **Chapter 4:** This chapter proposes an early mode decision method for the View Synthesis Prediction SKIP mode-enhanced MVC coder where MVD

videos are used to enable view synthesis prediction. It starts with an analysis of the optimal coding modes of this coder. It then proposes a novel Bayesian early mode decision method for it. The method uses Bayes' decision theory to eliminate the less probable candidate modes while encoding a macroblock. The chapter concludes with a discussion on the obtained results.

- **Chapter 5:** This chapter proposes a scene-consistent error concealment method for whole frame losses in multiview video transmission. It starts with an introduction to the error concealment problem in multiview plus depth video transmission. State-of-the-art methods for error concealment are reviewed next. This is followed by the description of the proposed scene-consistent error concealment method. The results are presented and discussed at the end of the chapter.

- **Chapter 6:** This chapter concludes the thesis with a summary of its major findings and recommendations for future work.

Chapter 2

Fundamentals

2.1 Introduction

This chapter presents some fundamental concepts in the field of video coding especially with reference to the H.264 standard, its multiview video coding (MVC) extension, and depth based view synthesis. It starts with an overview of the H.264 video coding standard in Section 2.2. This includes an introduction of the building blocks of the H.264 video coding standard which include the colour space, the macroblock and slice types. Prediction tools used in the standard are presented next. This is followed by an overview of the H.264 encoder. Section 2.3 presents an overview of MVC. 3D video representation formats are introduced in Section 2.4. At the end an analysis of View Synthesis technique is presented in Section 2.5 which starts by describing its pre-requisites like the different coordinate systems used, the camera parameters, and the depth maps. The fundamental view synthesis steps like forward warping, mapping competition, hole filling and merging are presented at the end of the chapter.

2.2 The H.264 Video Coding Standard

This section introduces the H.264 video compression standard. Section 2.2.1 presents an overview of its buildings blocks while Section 2.2.2 describes the different prediction types supported by H.264. Finally, Section 2.2.3 presents and

discusses the H.264 encoder.

2.2.1 Building blocks

- **Color Space and Sampling:** The H.264/AVC standard uses the YCbCr color space where Y, Cb and Cr represent the Luma and Chroma color components. The luma component Y represents the brightness, the chroma component Cb represents the measure by which the colour deviates from gray towards blue and the chroma component Cr represents the measure by which the colour deviates from gray towards red.

  The H.264/AVC standard uses a sampling structure in which the size of the chroma component is half the size of the luma component. This is called the 4:2:0 sampling with 8 bits of precision per sample.

- **Macroblock:** A Macroblock (MB) is the basic coding unit adopted in the H.264/AVC standard. Each frame of a video sequence is divided into fixed-sized MBs of size 16x16 luma samples and 8x8 chroma samples of each chroma component.

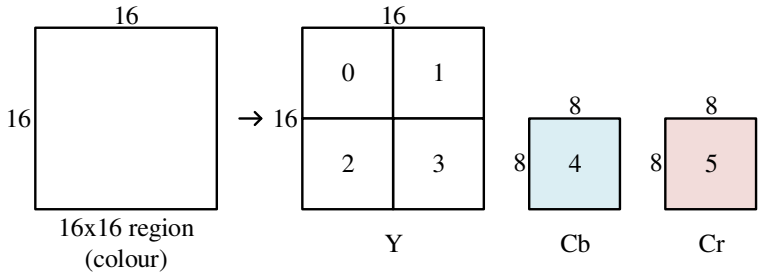


Figure 2.1: A Macroblock

- **Slice:** A sequence of MBs make up a slice. A frame is a collection of one or more slices. A slice can be of type I, P, or B. The type of slice determines the types of prediction that can be used for coding a macroblock in the slice. The H.264/AVC standard defines three types of slices:

- **I slice:** Only intra prediction can be used in an I-slice.
- **P slice:** Both intra prediction and inter prediction are allowed in P slices. An inter predicted macroblock in a P-slice can use at most one prediction signal per prediction block.
- **B slice:** Both intra prediction and inter prediction are allowed in B slices. An inter prediction MB in a B-slice can use at most two prediction signals per prediction block.

2.2.2 Prediction Types

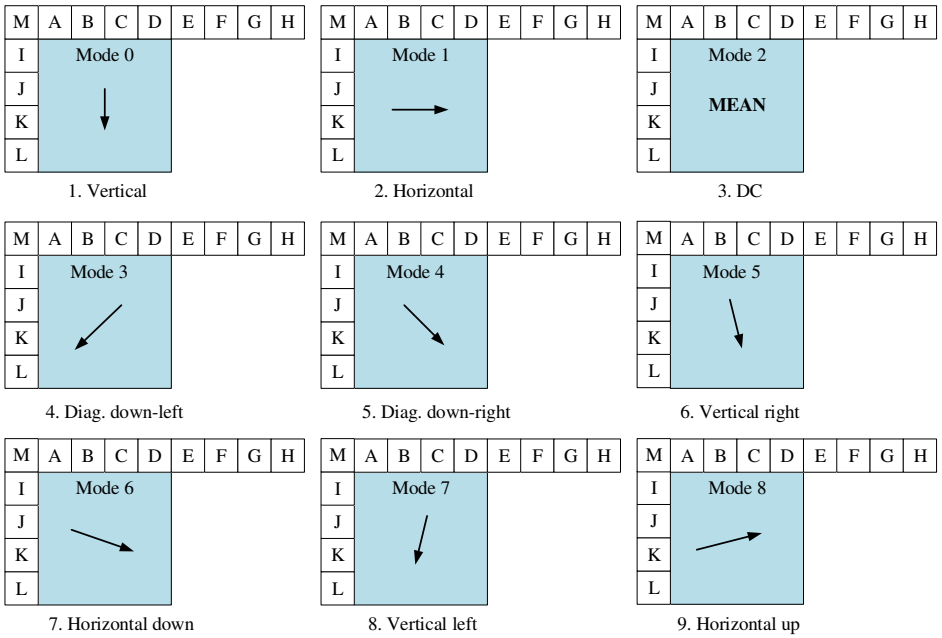
Each macroblock in a slice can be encoded using either intra prediction or inter prediction.

- **Intra Frame Prediction:** In Intra prediction, samples of the macroblock are predicted from within the same slice. Two intra prediction modes are supported: $Intra\_4x4$ and $Intra\_16x16$.

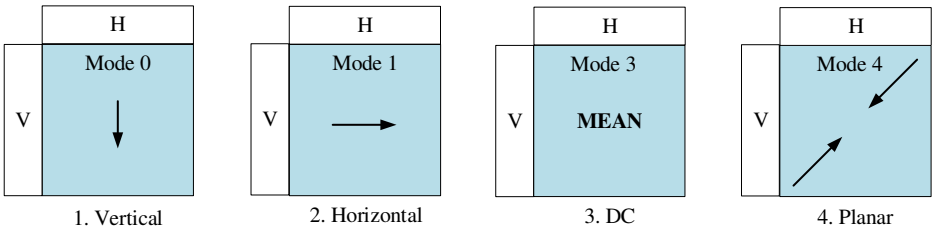
  In the $Intra\_4x4$ mode, the macroblock is divided into $4x4$ partitions, each of which can use one of the 9 $intra\_4x4$ prediction modes (See Fig. 2.2(a)). These include a 'DC' mode in which the whole 4x4 block is predicted from one value. The remaining eight are 'directional' modes each of which allows prediction for a certain direction. This type of prediction is well suited for regions of a frame with significant detail [20].

  In the $Intra\_16x16$ mode, the macroblock is predicted as a whole using one of the four $intra\_16x16$ prediction modes (See Fig. 2.2(b)). It supports four $Intra\_16x16$ modes. These include the 'DC', the 'Planar' and the vertical, and horizontal directional modes. This type of prediction is more suited for smooth areas of a picture.

- **Inter Frame Prediction:**
  - **in P Slices:** When a macroblock in a P slice is coded using inter prediction, it can either be coded as a 16x16 macroblock or can be partitioned in smaller block sizes of 16x8, 8x16 and 8x8 samples each



(a) Intra_4x4 prediction modes



(b) Intra_16x16 prediction modes

Figure 2.2: Intra Prediction Modes.

of which can be independently coded using motion compensated prediction from a reference frame (See Fig. 2.3). If 8x8 partition size is chosen, it can be further split into blocks of sizes 8x4, 4x8 or 4x4 samples. Fig. 2.4 illustrates the different types of macroblock partitions.

– **in B Slices:** B slices utilize a similar macroblock partitioning as P slices. In addition, it allows prediction from two different reference pictures.

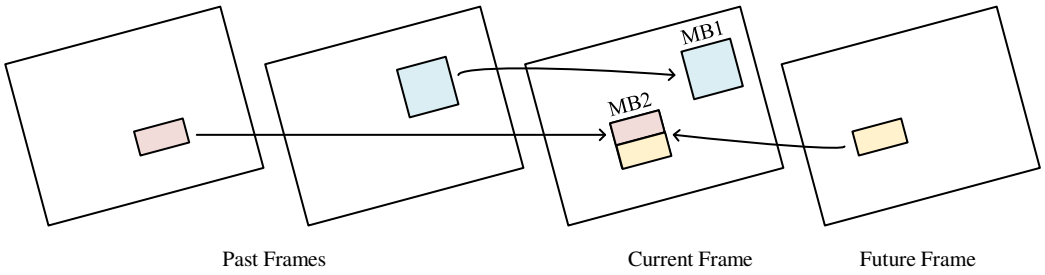


Figure 2.3: Inter Prediction

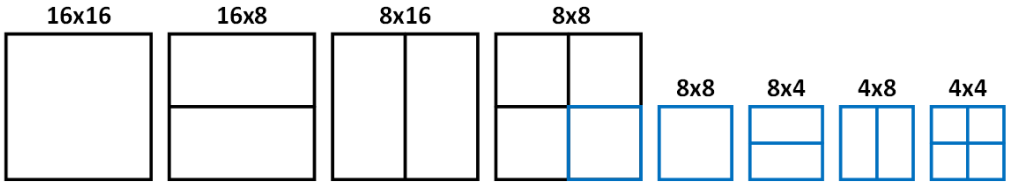


Figure 2.4: Macroblock Partitioning

2.2.3 The H.264 encoder

The H.264 encoder consists of two paths: (i) the forward path for encoding (shown in black lines in Fig. 2.5) and (ii) the reverse path for decoding and reconstruction of the current picture (shown in blue lines in Fig. 2.5).

In the forward path, a macroblock is first processed as either inter coded macroblock or an intra coded macroblock. In the former case, it goes through motion estimation and compensation to generate a prediction signal. In the latter, it uses prediction from spatially neighbouring samples of the current slice that have already been encoded and reconstructed by the encoder. The prediction signal is then subtracted from the original macroblock to obtain a residual macroblock. The residual macroblock is transformed using a separable integer transform with similar properties as those of a 4x4 Discrete Cosine Transform (DCT), quantized and entropy coded. Two entropy coders are defined in the standard: (i) Context Adaptive Binary Arithmetic Code (CABAC) and (ii) Context Adaptive Variable Length Code (CAVLC). The entropy coder also processes other information such as the motion vectors, reference frame index(es), macroblock partition modes (if

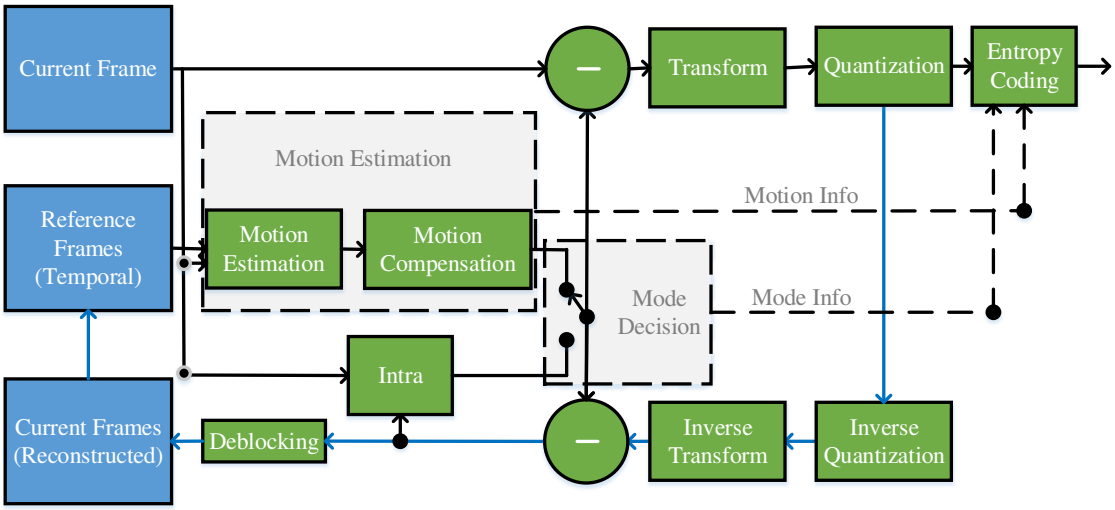


Figure 2.5: Block diagram of the H.264 encoder

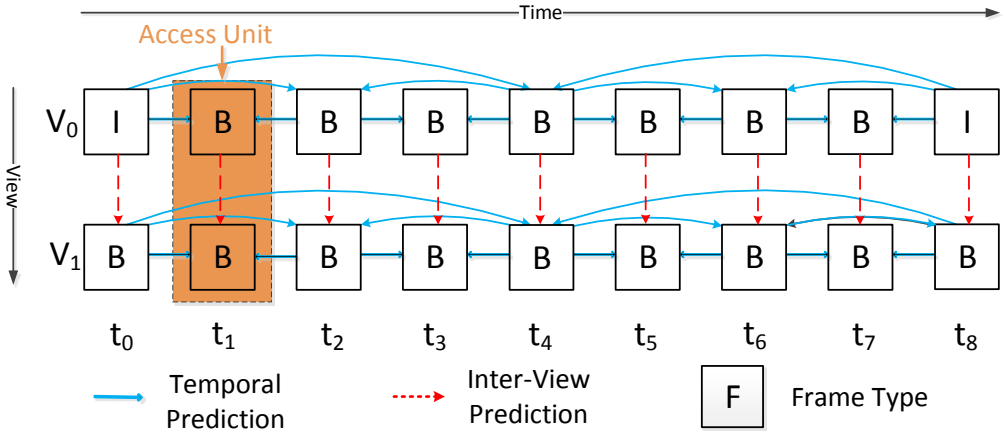
inter prediction used) and intra modes (if intra prediction used).

In the reverse path is included to reconstruct the coded picture exactly as by the decoder. These pictures are then used as references for intra and inter prediction. In order to reconstruct a macroblock, the inverse quantized and inverse transformed residual is added to the prediction signal from the forward path. A deblocking filter is then applied to it to reduce the blocking artefacts. The reconstructed pictures are added to the Reference Picture Buffer and are subsequently available to be used as references.

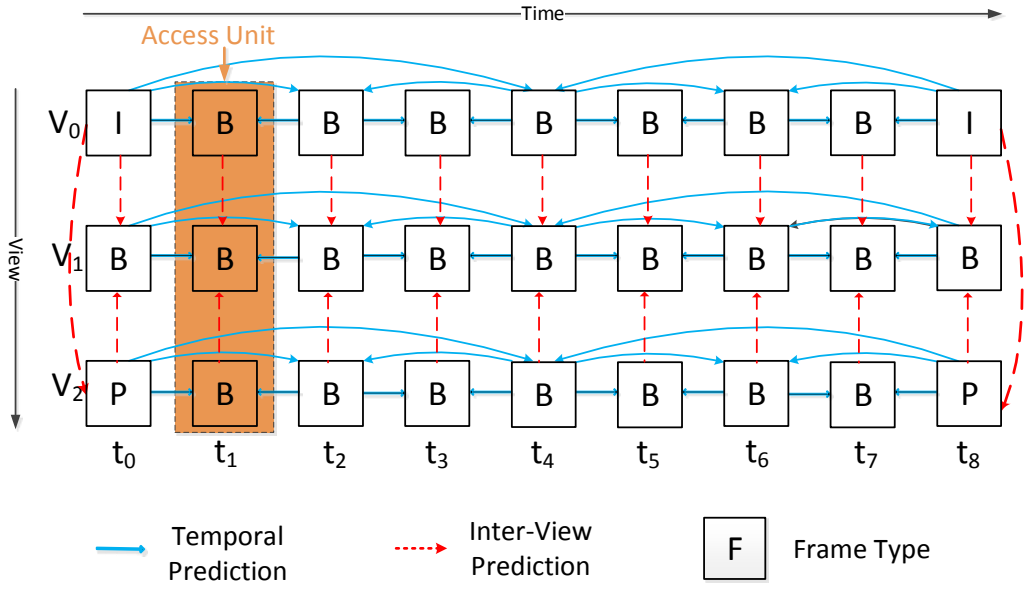
2.3 Multiview Video Coding

This section introduces the Mulitiview Video Coding (MVC) extension of H.264 video compression standard. Section 2.3.1 presents an overview of inter-view prediction in MVC while Section 2.3.2 presents and discussed the MVC encoder. Finally, Section 2.3.3 gives an overview of the various scenarios and applications in which MVC is used.

2.3.1 Enabling Inter-view Prediction



(a)



(b)

Figure 2.6: Typical MVC prediction structures for the (a) two-view stereo high profile and (b) three-view multiview high profile. An Access Unit (AU) refers to a set of frames corresponding to the same time instance but in different views

Unlike inter prediction in H.264 which only exploits the temporal redundancy between frames of the same view, inter-view prediction exploits both the temporal redundancy between frames of the same view as well as the spatial redundancy between frames of neighbouring views. This is made possible by extending the capability of reference frame lists of H.264/AVC (which only contained indices of frames from the same view) to include indices of frames from neighbouring views as well. Hence, frames from other views are also available during inter prediction process. The only limitation is that the inter-view frames must be contained within the same Access Unit (AU) as the current frame (See Fig. 2.6). An AU is defined by a set of frames corresponding to the same time instance but in different views.

The MVC specification [21] defines two profiles: (i) the stereo high profile and (ii) the multiview high profile (Fig. 2.6). The stereo high profile is limited to two views while the multiview high profile supports multiple views.

2.3.2 The MVC encoder

An MVC encoder (Fig. 2.7) is largely similar to the H.264 encoder but uses an enhanced prediction mechanism. That is, on top of Motion Estimation (ME) and Motion Compensation (MC), the MVC encoder uses Disparity Estimation (DE) and Disparity Compensation (DC) which allows prediction from inter-view reference frames. Hence the mode decision in MVC decides between Inter, Inter-view and Intra prediction, unlike in H.264 where the decision has to be between Inter and Intra modes. JMVC [41] (and the earlier Joint Multiview Video Model (JMVM) [32]), which is the standard reference software for MVC, uses the Rate Distortion Optimized Mode Decision (RDO-MD) for mode selection. RDO-MD is based on the rate and distortion costs i.e.,

$$J = D + \lambda R \tag{2.1}$$

where J is the RD cost, λ is the Lagrange multiplier, and R and D are the bitrate and distortion costs respectively. During the mode decision process for a macroblock, J is evaluated for all possible combinations of block sizes using both ME and DE in all available reference frames and the best combination in RD

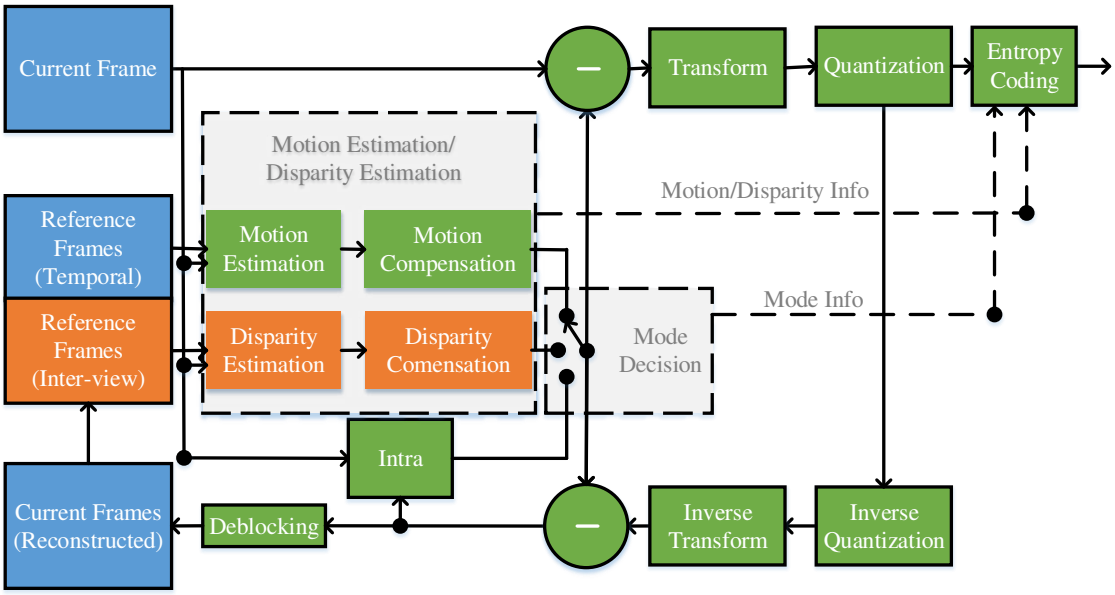


Figure 2.7: Block diagram of the MVC encoder

sense (i.e., which minimizes J) is finally selected to encode the macroblock.

2.3.3 Scenarios and Applications

There are two main applications of MVC as identified by the MVC standardization project [42]. These are: 3D video and Free view-point video.

A 2D video provides a very limited viewing sensation compared to what the Human Visual System (HVS) is capable of. Two important cues missing from 2D video compared to the HVS are:

- seeing a different image with each eye **(stereo parallax)**, and
- seeing a different image as the head is moved **(motion parallax)**.

The aim of new video technologies like 3D video and free view-point video is to provide the missing cues in 2D video such as the stereo and/or motion parallax.

- **3D video:** A basic 3D video is required to provide at least stereo parallax. Stereoscopic displays can be for displaying such simple 3D videos.
- **Free view-point video:** A basic free view-point video is required to provide at least motion parallax. In order to provide this kind of ability, the display is required to be able to emit more than two views. The viewing range in which motion parallax can be experienced is limited by the number of views that the display can emit.

  Both 3D and free view-point videos can also be combined in one application such as in the advanced multiview autostereoscopic displays which provide both stereo and motion parallax.

2.4 3D Video Formats

This sections presents an overview of the different video formats available for 3D video. The texture based video formats Conventional Stereo Video (CSV) and Multiview Video (MVV) are discussed in Section 2.4.1 and Section 2.4.2 respectively while the depth based formats Video plus Depth (V+D) and Multiview Video plus Depth (MVD) are presented in Section 2.4.3 and Section 2.4.4. Other, less common, formats are presented in Section 2.4.5.

2.4.1 Conventional Stereo Video

Conventional Stereo Video (CSV) [43] is the most common format for 3D video and is used in basic 3D applications such as 3D video with glasses. It is based on two texture views (Fig. 2.8(a)). In order to reduce the amount of data associated with two full resolution texture videos, a simple variant is the Mixed Resolution Stereo (MRS) [44] which replaces one of the full resolution texture videos with a half resolution one.

2.4.2 Multiview video

Multiview Video (MVV) is based on multiple texture views (Fig. 2.8(b)). It is an extension of CSV to more than 2 views. MVV provides flexible 3D viewing

(a) Conventional Stereo Video (CSV)

(b) Multiview Video (MVV)

Figure 2.8: Typical examples of texture based video formats

options such as view-point switching but using a large amount of video data. A problem with MVV is the linear relationship between the data rate and the number of views.

2.4.3 Video plus depth

Video Plus Depth (V+D) [45] format is very similar to the CSV format with the difference that one of the texture views of CSV is replaced with a depth map (Fig. 2.9(a), See. Section 2.5.3 for an explanation of depth maps). The second texture view required to provide 3D sensation, is generated at the decoder using the depth image and view synthesis techniques (explained in the next section). This approach is more efficient compared to CSV as it can provide comparable visual quality to that of CSV but using a smaller amount of video data [43]. This

is because a depth image contains less information compared to the texture data and can be compressed more efficiently [11]. The depth maps can be compressed at only 10% to 20% of the bitrate required for encoding of texture maps [45]. There are two reasons for this: (i) the depth map is represented by a single signal while the texture map usually contains three signals corresponding to one luma and two chroma components and (ii) the depth signal mainly consists of large homogeneous areas, inside scene objects, which are easy to compress [26]. While the depth maps are generally encoded using conventional video codecs such as H.264/AVC, MVC or HEVC, alternative approaches such as platelet coding [46] are being studied which may outperform the conventional codecs. The V+D format was specified by MPEG as the representation format for 3D video in its MPEG-C Part 3 [47]. While CSV can be readily used for 3D videos, enabling 3D using V+D format is relatively complex as it involves view synthesis process at the decoder.

2.4.4 Multiview plus depth

Multiview plus depth (MVD) [43] is based on multiple texture views with their associated depth maps (Fig. 2.9(b)). It can be considered as a combination of MVV and V+D and can be used to provide motion parallax at significantly lower bit rates compared to MVV [43]. It can be used to synthesize a very large number of views at the decoder compared the V+D format which can generate views in a very limited range around the available original view [43]. The recent MVC+D [48, 49] 3D video coding standard specifies MVD as the underlying video representation format [48, 49].

2.4.5 Other formats

Layered Depth Video (LDV) and Depth Enhanced Stereo (DES) [43] are two other promising formats for 3D video. LDV is based on one texture view with its associated depth map and one or more enhancement texture and or depth layers. These layers can for example contain only the occluded/dis-occluded regions. DES is an advanced stereo video format that provides backward compatibility

(a) Video plus Depth (V+D)

(b) Multiview Video plus Depth (MVD)

Figure 2.9: Typical examples of depth based video formats

with applications based on CSV but can use additional depth and/or enhance-

ment layers when used in more advanced 3D applications.

2.5 View Synthesis for depth based video

An important advantage of the depth based 3D video formats is that they allow the creation of synthetic frames at the decoder by using view synthesis. This section briefly overviews the view synthesis process. Section 2.5.1 provides the necessary background information on the world, camera and image coordinate systems which are used during the view synthesis process. Section 2.5.2 describes the different types of parameters associated with a camera. The concept of 'depth maps' is described in Section 2.5.3. Forward warping, which uses camera parameters and depth maps to project a pixel from one view to another, is discussed in Section 2.5.4. Two common problems associated with forward warping are: mapping competition and hole filling. These are discussed in Section 2.5.5. Finally, view synthesis can be performed using two reference views, rather than one. The process and advantages of using two reference views for view synthesis are discussed in Section 2.5.6.

2.5.1 Coordinate Systems

View synthesis uses the well-known pinhole camera model which defines three types of coordinate systems [50]. The model is briefly described here.

The three coordinate systems are the world coordinate system, the camera coordinate system and the image coordinate system (See Fig. 2.10). The world coordinate system has three dimensions ((Xw, Yw, Zw) in Fig. 2.10) and is independent from any particular camera. In contrast, each camera has its own camera and image coordinate systems. In Fig. 2.10, the three dimensional camera coordinates are denoted by (Xc, Yc, Zc) while the two dimensional image coordinate system is denoted by (u, v).

2.5.2 Camera Parameters

Two sets of camera parameters describe the relationship among the coordinate systems. These are the intrinsic matrix A and extrinsic matrix E. The intrinsic

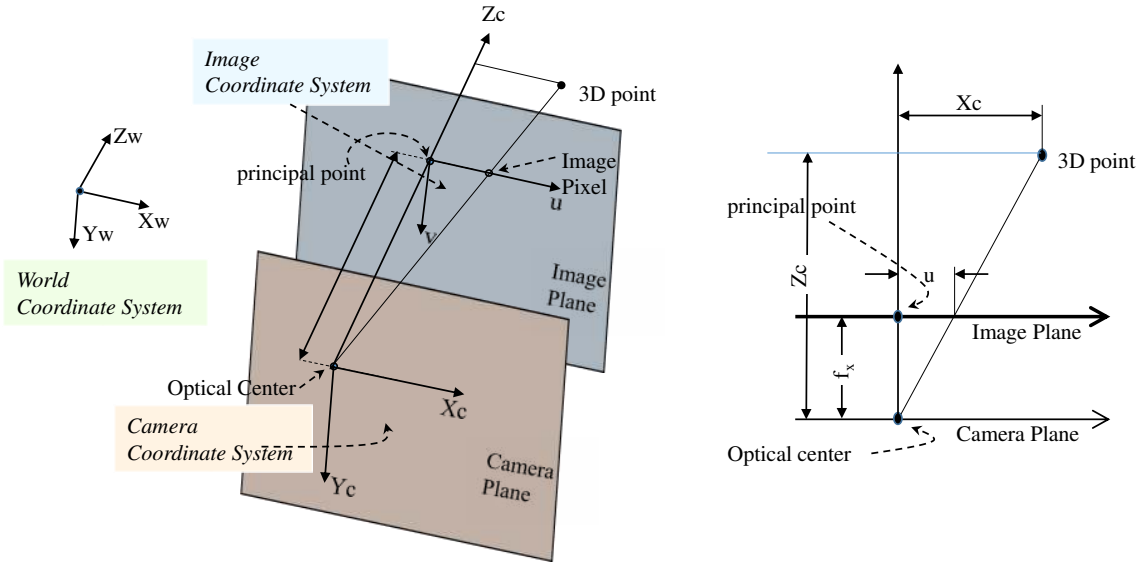


Figure 2.10: Left: 3D coordinate system; Right: Relationship between image and camera coordinates

matrix A represents the transformation from a camera coordinate system to the image coordinate system and is represented as:

$$A = \begin{pmatrix} f_x & 0 & o_x \\ 0 & f_y & o_y \\ 0 & 0 & 1 \end{pmatrix} \tag{2.2}$$

where f_x and f_y represent the focal lengths in x-axis and y-axis respectively, and (o_x, o_y) represents the principal point offset [11].

The extrinsic matrix $E = [R|t]$ is used to obtain a transformation from the world coordinate system to the camera coordinate system. It consists of a 3×3 rotation matrix $R_{3\times3}$ and a 3×1 translation vector $t_{3\times1}$

2.5.3 Depth maps

Depth maps are grayscale (8 bit) images that represent the distances of objects from the camera. The physical depths can be obtained from the corresponding

8-bit pixel values using the following equation:

$$Z = \frac{1}{\frac{Y}{255}\left(\frac{1}{Z_{near}} - \frac{1}{Z_{far}}\right) + \frac{1}{Z_{far}}} \tag{2.3}$$

where Y is the pixel value of the depth image while Z_{near} and Z_{far} indicate the minimum and maximum distance of a 3D point from the camera respectively while Z is the physical depth value for pixel value Y. With 8 bits, it is possible to define $2^8 = 256$ different depth levels. Objects located closer to the camera are represented by higher pixel values while those located farther are represented by lower pixel values. For example, the farthest object is represented by a pixel value of 0 while the nearest object is represented by a pixel value of 255. An example depth map is shown in Figure 2.11 [51].

Figure 2.11: Example of depth maps

While the term 'depth' is generally considered to have the same meaning as that of 'disparity', it is important to note that they refer to different but related aspects of a video. They are different, as depth represents the distance of an object from the camera, while disparity represents the distance between two corresponding points in the left and right image of a stereo pair. They are related, because depth at various scene points can be recovered by knowing the disparities of corresponding image points [52] i.e.,

$$Z = \frac{b.f}{x_l - x_r} \tag{2.4}$$

where b is the baseline distance between the identical left and right cameras, f is the focal length of the cameras while x_l and x_r are the respective corresponding image points in the two cameras.

2.5.4 Forward Warping

This is the first step of the view synthesis process. In this step, the depth map of the reference view and the camera parameters are used to project pixels from a reference view to a virtual view.

For example, a pixel at position (u_r, v_r) in the image coordinate system of the reference view r can be projected at a position (u_v, v_v) in a virtual view v using forward warping. This is done in two steps. First, the pixel is projected into the world coordinates (x_W, y_W, z_W) i.e.,

$$\begin{pmatrix} x_W \\ y_W \\ z_W \end{pmatrix} = R^{-1}_{3\times3,r}. \left(z_{C,r}.A^{-1}_{3\times3,r} \begin{pmatrix} u_r \\ v_r \\ 1 \end{pmatrix} - t_{3\times1,r} \right) \tag{2.5}$$

where $R_{3\times3,r}$, $z_{C,r}$, $A_{3\times3,r}$, and $t_{3\times1,r}$ represent the rotation matrix, the depth value of the pixel at position (u_r, v_r), the intrinsic matrix and translation vector of reference camera r respectively.

In the next step, the 3D point in the world coordinate system is projected to the image coordinate system of the virtual view v i.e.,

$$z_{c,v}. \begin{pmatrix} u_v \\ v_v \\ 1 \end{pmatrix} = A_{3\times3,v}. \left(R_{3\times3,v} \begin{pmatrix} x_W \\ y_W \\ z_W \end{pmatrix} + t_{3\times1,v} \right) \tag{2.6}$$

2.5.5 Mapping competition and hole filling

Two problems can occur during forward warping: (i) more than one pixel in the reference view maps to a particular position in the virtual view, and (ii) no pixel in the reference view maps to a particular position in the virtual view. The former results in mapping competition while the latter results in black holes.

The mapping competition is resolved by analysing the depth value for all the

candidate pixels in the reference view that map to a particular position in the virtual view. The pixel with the largest depth value (closest to the camera) is finally selected.

For hole filling, mainly, three different approaches have been used in the literature:

(i) Extrapolation of neighbouring pixels [11]: As holes originate from disocclusion, a simple approach is to fill them using neighbouring pixels that belong to the background (have smaller depth values). This simple approach is usually sufficient in case of narrow baseline spacing.

(ii) Image Inpainting [53]: Inpainting is a method that is generally used to fill up the missing parts of an image. It first splits the image into two functions with different basic characteristics. It then reconstructs each function independently with structure and texture filling algorithms. An inpainting technique is also adopted in the popular View Synthesis Reference Software (VSRS) [54][55] developed by Nagoya University. In order to prevent the foreground object from inpainting source, it is proposed to replace the foreground boundaries by some background before performing inpainting [56].

(iii) Temporal compensation [57]: The idea behind temporal compensation is that when the foreground object is moving, it may disclose the background behind it. So, if the image sequence is analysed, it may be possible to use some background information from other frames to help fill the holes in the current frame. Compared to the first two methods, this method is more challenging since it requires heavy processing of multiple frames in a sequence.

2.5.6 Merging multiple reference views

The quality of the virtual view is much better when two reference views are used. This is because the virtual view can be synthesised using both the left reference view and the right reference view respectively. The two synthesised views are complementary to each other [11]. Unlike, view synthesis using one reference view where many large holes appear in the virtual view, when two reference views are used, most of the holes can be filled up by merging the virtual views obtained using the left and the right reference views.

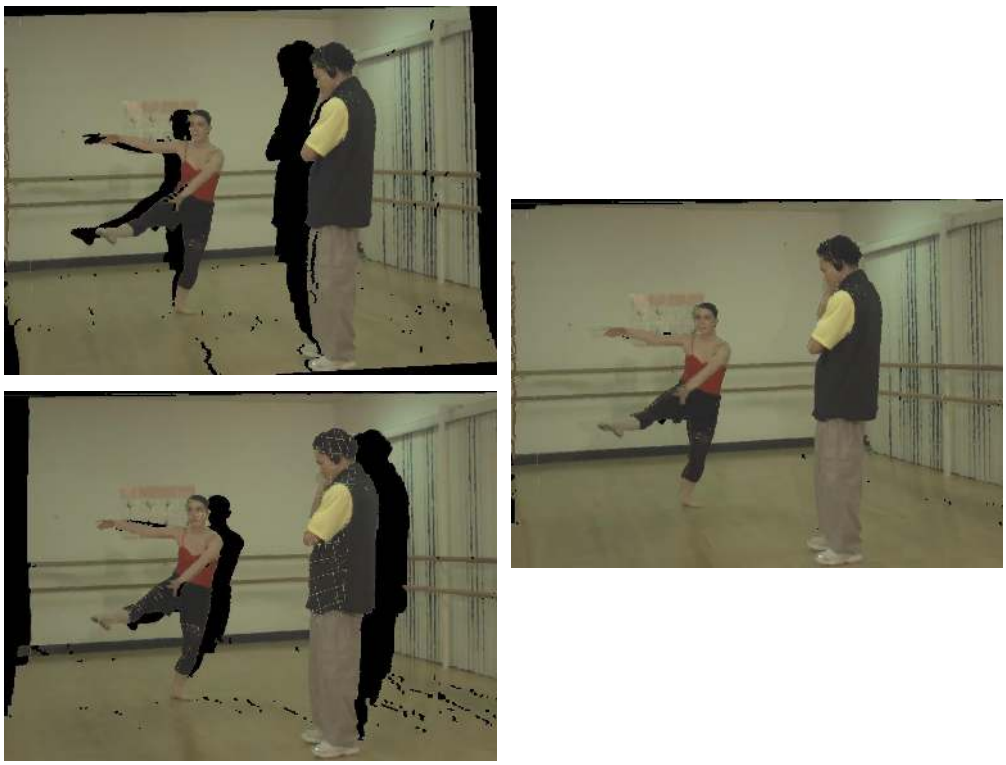


Figure 2.12: View Synthesis using two reference views

Like the mapping competition problem described in the previous section, while merging the two synthesised views, it is possible that more than one candidates will be competing for a position in the final synthesised view. In order to resolve this problem, the mapping competition method described in the previous section can be used here as well. Another option is to average out the results from the two reference views [11]. An example of view synthesis using two reference views is shown in Fig. 2.12.

Chapter 3

Fast Encoding Techniques for Multiview Video Coding

3.1 Introduction

Multi-view Video (MVV) is a technology that uses multiple cameras to simultaneously capture a scene from different view points. It is used in applications such as 3D Television and Free View-point Television (FTV) [8]. While MVV gives a richer viewing experience than conventional video, it produces a huge amount of data. However, since the data from all cameras relates to the same scene, it is highly redundant. This has led to the development of Multiview Video Coding (MVC) [23, 24], the multiview extension of one of the popular video coding standard, H.264/AVC [20, 21]. In MVC, reference frames for block matching are taken from neighbouring views (Disparity Estimation) as well as across the temporal axis (Motion Estimation). A typical prediction structure [28] of MVC is presented in Fig. 3.1.

MVC uses variable block size motion and disparity estimation, which requires an exhaustive search for motion and disparity vectors using all the available block sizes. Unfortunately, this makes the MVC encoder very complex. High computational complexity has been hampering the use of multiview video into real-time realistic media applications (e.g., 3D live broadcasting and interactive FTV) [58]. Therefore, reducing the time complexity of the MVC encoder is very

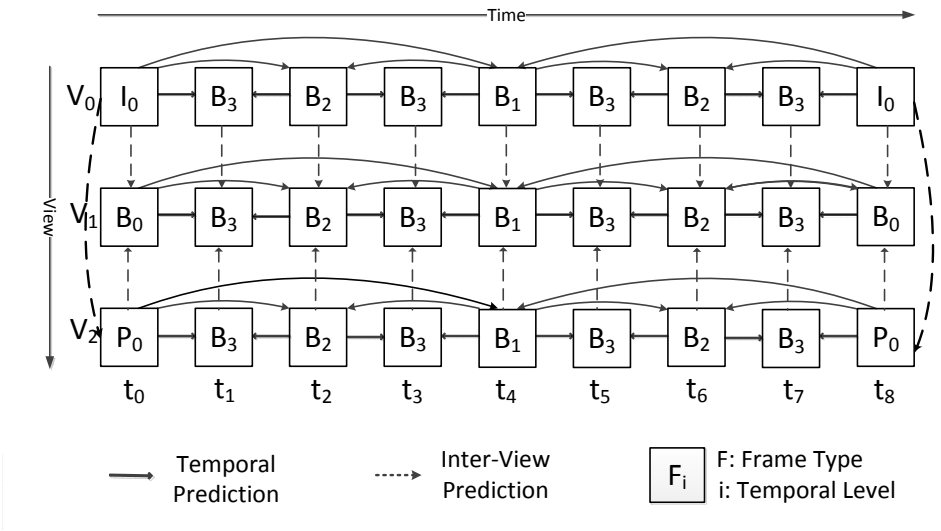


Figure 3.1: Typical MVC prediction structure. V_0, V_1, and V_2 represent three views while $t_0, t_1, \ldots, t_8$ represent nine successive frames. In each view, the first frame of the Group of Pictures (GOP) is said to be at Temporal Level 0 (TL0). All the frames that use frames at TL0 as references belong to Temporal Level 1 (TL1). Similarly, the frames that use frames at TL1 as references belong to Temporal Level 2 (TL2), etc.

important.

Complexity reduction can be achieved in several ways, e.g., by reducing the number of candidate modes, the number of reference frames, the number of directions for prediction, or the search range. Such reductions must, however, be done with smallest penalty on Rate-Distortion (RD) performance. State-of-the-art methods address specific parts of the problem (e.g., number of candidate modes, number of reference frames, number of prediction directions, or the search range) but, there is no global solution yet. Moreover, most of the published work relies only on CPU time saving as the evaluation metric for low complexity encoding methods. This may not necessarily reflect the efficiency of the methods themselves, as it is dependent on the particular implementation and the test platform.

The contributions of this chapter are as follows. It defines four levels of complexity in the MVC encoder and identifies the best previous fast encoding techniques at each level. It then combines these techniques in a unique framework

in which savings in complexity add up. It is observed that the performance of the combination largely depends on motion and disparity in the video sequence as well as the encoding bitrate. The complexity savings are larger for low motion content. They are also larger at low bitrates than at high bitrates. In order to improve the performance for high motion content and high bitrates, two new fast encoding techniques are proposed. These are: (i) Previous Disparity Vector Disparity Estimation (PDV-DE), which exploits the correlation between temporal levels and disparity vectors and (ii) Stereo-Motion Consistency Constraint Motion and Disparity Estimation (SMCC-MDE), which exploits the geometrical relationship of consecutive frames in the multiview video sequence.

The performance of the proposed global solution is analysed using two metrics: CPU time and number of search points. Experimental results show that the proposed solution can save up to 93.7% in encoding time and 96.9% in number of search points compared to JMVM 6.0 [32] using the fast TZ search mode [59], with a negligible cost of 0.05 dB decrease in PSNR and 1.46% increase in bitrate. This is an improvement of over 11% and 7%, respectively, compared to the best published method [34]. This method uses inter-view mode and Motion Vector (MV) correlations to reduce the complexity of the mode decision, the reference frame selection, and the block matching process.

The remainder of the chapter is organized as follows. In Section 3.2, an overview of the encoding complexity framework for MVC is presented and its main bottlenecks are identified. State-of-the-art solutions that have been proposed to address them are presented in Section 3.3. In Section 3.4, the proposed low-complexity solution, which combines PDV-DE and SMCC-MDE with state-of-the-art methods is presented. Section 3.5 contains experimental results. Section 3.6 provides conclusions and suggests future research directions.

3.2 MVC encoding complexity

In order to analyse the complexity of MVC encoding, this chapter focuses on JMVM, which is the reference software for MVC. For efficient compression, JMVM offers multiple ways of encoding a macroblock. These include a choice of different macroblock partition sizes, prediction directions, reference frames and search

Table 3.1: Complexity levels in Multiview Video Coding.

Level 1: Mode	Level 2: Prediction Direction	Level 3: Reference Frame	Level 4: Block Matching
1. SKIP 2. INTER16X16 3. INTER16X8 4. INTER8X16 5. INTER8X8 6. INTER8X4 7. INTER4X8 8. INTER4X4 9. INTRA16, INTRA8, INTRA4	1. Forward 2. Backward 3. Bi-directional	1. ME 2. DE	Search for the best rate-distortion match

window sizes. In the standard implementation, all the possible options are exhaustively checked, and the ones resulting in the lowest rate-distortion cost are finally selected. The following four levels of complexity in JMVM are identified in this chapter (Table 3.1).

- Level 1 - **Mode Selection**: Several modes are checked in sequence to find the best rate-distortion match for the current macroblock. These modes are: (i) SKIP, (ii) INTER16x16, (iii) INTER16x8, (iv) INTER8x16, (v) INTER8x8, (vi) INTER8x4, (vii) INTER4x8, (viii) INTER4x4, (ix) INTRA16, (x) INTRA8, and (xi) INTRA4. When a macroblock is encoded using the SKIP mode, no motion or residual data is transmitted. The macroblock is reconstructed with the help of motion vectors from the spatially neighbouring macroblocks. In INTER16x16 mode, a single motion/disparity vector along with the residual data is transmitted. In INTER16x8 and INTER8x16 modes, a macroblock is partitioned into two partitions of sizes 16x8 and 8x16, respectively, and for each partition, a separate motion/disparity vector is transmitted. In INTER8x8 mode, a macroblock is partitioned into four partitions of size 8x8 and four motion/disparity vectors are transmitted. Each 8x8 size partition can be further divided into three possible sub-macroblock partitions of sizes 8x4, 4x8, and 4x4. For each sub-macroblock partition, a separate motion/disparity vector is transmitted.

- Level 2 - **Prediction Direction Selection**: For each INTER mode in Level 1, a best match is sought in: (i) past frames (forward prediction), (ii)

future frames (backward prediction), and (iii) a combination of one past and one future frame (bi-directional prediction).

- Level 3 - **Reference Frame Selection**: For each prediction direction selected in Level 2, the JMVM reference software searches reference frames from different views to find the best match through block-matching. These frames can be: (i) from the same view (using ME), (ii) from the two neighbouring views (using DE).

- Level 4 - **Block Matching**: For each reference frame, a best match is sought in a search window of size $n \times n$, where n denotes the number of pixels. For good compression efficiency, usually a large window size (($\pm$ 64, $\pm$ 64) in JMVM 6.0) is used. An important element in the search process is the determination of the motion vector predictor. The motion vector predictor determines the starting point for the search process. The more accurate the predictor is, the more probable it is to find the best match in a smaller search area.

3.3 Related Work

Several methods have been proposed to reduce the encoding complexity of MVC. They are briefly reviewed below.

Mode Selection: A fast mode selection method exploiting the correlation between the modes of neighbouring views is proposed in [60]. In order to predict the mode of the current macroblock, the modes of the corresponding macroblock in the neighbouring view and its eight spatially neighbouring macroblocks are taken into consideration. Weights are assigned to each mode and macroblocks are classified according to the average weight. If the average is less than 0.125, then the current macroblock is called *Simple* and only SKIP mode and INTER16x16 modes are considered. If the average is greater than 0.125 and smaller than 0.25, then the current macroblock is called *Normal* and additionally, INTER16x8 and 8x16 are also considered. Finally, if the average weight is greater than 0.25, the macroblock is called *Complex* and all modes are considered. Zeng, Ma, and Cai [61] extend the work of [60] by increasing the number of macroblock types to five.

A fast mode decision method based on rate-distortion costs is presented in [62]. It uses inter-view rate-distortion cost correlation of optimal modes to reduce the number of candidate modes in the current view.

Early detection of SKIP mode reduces the complexity of the encoder significantly as macroblocks encoded in SKIP mode do not require block matching. An early SKIP mode detection method is proposed in [37]. The detection is based on the analysis of SKIP mode decisions of the nine corresponding neighbours in the neighbouring right view.

Prediction Direction Selection: In JMVM, most of the pictures are of B type. For macroblocks in B pictures, motion and disparity estimation are done using forward, backward and bi-directional prediction. Zhang et al. [63] observe that the prediction direction that results in the lowest rate-distortion cost for INTER16x16 is also the one that results in the lowest rate-distortion cost for the other INTER modes. So they propose to save encoding time by selecting for all modes the prediction direction that results in the lowest rate-distortion cost for INTER16x16.

Reference Frame Selection: Zhang et al. [63] restrict block matching to the reference frame that gives lowest rate-distortion cost for INTER16x16. Another fast reference frame selection method is presented in [64]. Frames are divided into regions with homogeneous motion (*homogeneous* regions) and regions with complex motion (*complex* regions). The classification is based on forward motion vectors for 4x4 pixel blocks in four view-neighbouring macroblocks (i.e., corresponding macroblock in the neighbouring right view together with its left, upper, and upper-left macroblocks). The authors observe that in homogeneous regions inter-view prediction is rarely used and thus propose to disable DE in those regions.

Another adaptive disparity estimation method is proposed by Shen et al. [34]. This method enhances the method of [64] by defining a third class of regions, namely *medium homogeneous* regions. Moreover, the classification is refined by involving all nine view-neighbouring macroblocks. DE is disabled in homogeneous regions, as well as in medium regions if the rate-distortion cost of the motion vector predictor (initial prediction of the motion vector) is smaller than that of the disparity vector predictor (initial prediction of the Disparity Vector (DV)).

While the method in [63] reduces the number of reference frames for smaller macroblock partitions to one in each prediction direction, all the reference frames are still checked for the INTER16x16 mode. Similarly, the method in [34] reduces the number of reference frames in each prediction direction to one in homogeneous regions. But in complex regions, two reference frames are checked in each direction.

Block Matching: Shen et al. [34] observe that the best block (for ME and DE) is usually found close to the current macroblock for homogeneous regions, far away from the current macroblock for complex regions, and somewhere in between for medium homogeneous regions. Therefore, the search range for a macroblock is adjusted according to the region type. For homogeneous regions, the search range is limited to a quarter of the full search range. For medium homogeneous regions, it is limited to half the full search range. For complex regions, full search is used. The spatio-temporal correlation of the disparity fields is studied in [65]. The authors find that the search range for disparity estimation can be reduced if multiple candidates are considered as search centres. However, they do not exploit the correlations between disparity vectors at different temporal levels, which can be used to further reduce the search ranges. Similarly, they do not study the effect of the type of macroblock on this correlation.

Deng et al. [66, 67] use Stereo Motion Consistency Constraint (SMCC) to reduce the complexity of motion and disparity estimation for stereo video coding. They use an iterative search strategy in which SMCC geometry is exploited to get base motion and disparity vectors. In order to reduce the effect of macroblock boundary mismatches on the performance of their algorithm, they extend the search around the base motion and disparity vectors iteratively. Because base motion and disparity vectors are not very accurate, a large search region is required around them during the successive iterations.

Combinations: Finally, some combinations of methods at different levels of the encoding scheme have been proposed. Zhang et al. [68] combine a rate-distortion cost threshold fast mode decision technique with the multiple reference frame selection method in [63]. While this algorithm speeds up the mode decision and reference frame selection processes, redundancies still exist in prediction direction selection and block-matching. Shafique, Zatt, and Henkel [69] take into

consideration texture classification and rate-distortion cost of a macroblock to predict the mode and prediction direction. However, the reference frame selection and block matching steps are not modified.

3.4 Proposed framework

In order to achieve maximum reduction in encoding complexity, it is important to simplify the processes involved at all levels of the encoding process. There is no method that simultaneously reduces the complexities present at all these levels. This provides a motivation to reduce the complexity at all levels and thus to present a complete low-complexity solution for MVC encoding. In order to do this, first, state-of-the-art methods ([37], [63] and [34]) that target different levels of the encoding are combined into a novel framework. It is noticed that the gains add up. It is then observed that, while the combination speeds up the overall MVC encoding process, its performance at high bitrates and for content with high motion can still be improved. Thus, two new complexity reduction techniques are proposed. The first one (PDV-DE) reduces the search range by exploiting the correlation of the disparity fields of successive frames at different temporal levels. The second one (SMCC-MDE) exploits the geometric constraint between motion and disparity vectors of two consecutive stereo pairs to reduce the area where a potential best rate-distortion match lies. The details of these two techniques are presented in the next two sub-sections and a summary of the complete framework is presented in Section 3.4.3.

Section 3.4.1 and Section 3.4.2 present the two proposed methods while Section 3.4.3 presents the complete low complexity framework in which the two proposed methods are combined with other state-of-the-art methods.

3.4.1 Previous Disparity Vector Disparity Estimation (PDV-DE)

While DE consumes as much time as ME, the probability that it is used for prediction is generally low [64, 34]. In this section, it is proposed to adjust the search range for DE according to the temporal level of the frame and the type of

the macroblock (simple, normal, or complex).

The search for the best match starts in the search centre. If the search centre is close to the best match, it might be possible to reduce the size of the search range and still find the best match. Because it is not known a priori where the best match for the current macroblock will be found, JMVM uses median prediction. In median prediction, the median of the vectors of the top, top-right, and left macroblocks are used as the search centre. The same procedure is used for both ME and DE. However, the nature of disparity is different from that of motion. Indeed, even in the presence of motion, the source of disparity (i.e., the camera arrangement) is usually fixed. Thus, disparity is not as difficult to predict as motion. Moreover, the disparity fields of successive frames are highly correlated [65]. Thus, if the search process is started from the position identified by the disparity vector of the corresponding macroblock in the previous frame, it is expected that the best match will be found very early in the process.

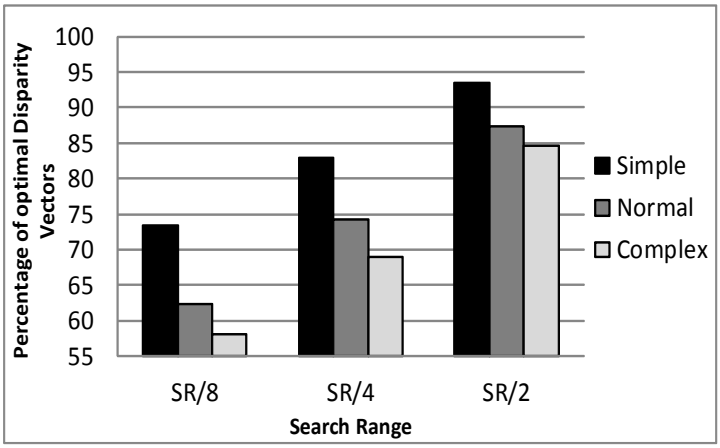


Figure 3.2: Optimal Disparity Vector Distribution at TL3

In order to validate this assumption, the procedure for finding the search centre is modified. The disparity vector of the corresponding macroblock in the temporally preceding frame is used as the search centre, instead of the median prediction. This vector is called Previous Disparity Vector (PDV). The initial Search Range (SR) is set to 64. Then, the proportion of macroblocks that find their best match in various search ranges: 1/8th of the initial search range (SR/8),

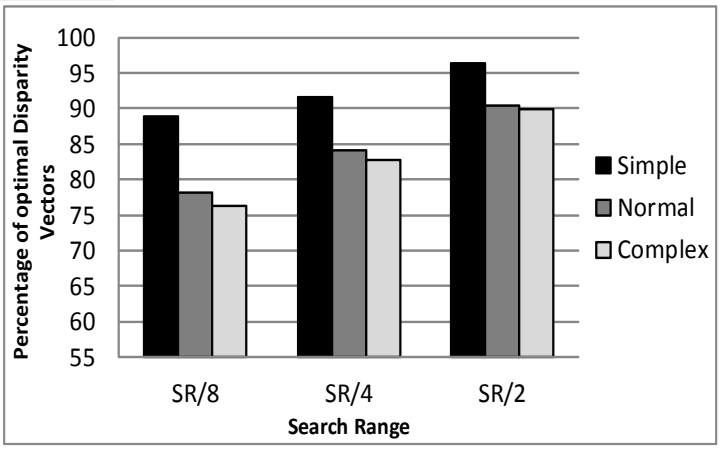


Figure 3.3: Optimal Disparity Vector Distribution at TL4

1/4th of the initial search range (SR/4), and half of the initial search range (SR/2) are determined. It is noted that the way the best match is spread across the search range depends on the temporal level of the frame as well as on the macroblock type. The number of temporal levels depends on the length of a Group of Pictures (GOP). A GOP of length l has $\lceil \log_2(l) \rceil + 1$ Temporal Levels (TLs). In these experiments, the GOP length is 16, so the highest temporal level is 4 (TL4), while the second highest is 3 (TL3). For frames at higher temporal levels, the best match is usually found in a smaller area than for those at lower temporal levels. For example, for frames at TL4, the best match is found in a smaller area than for those at TL3 (Fig. 3.2). Also for simple macroblocks, the best match is found in a smaller area than for normal and complex macroblocks. This indicates that if the previous disparity vector is used as the search centre, the search range can be reduced adaptively according to both temporal level and macroblock type.

Based on the observations and motivations in this subsection, a new search strategy of disparity estimation, called PDV-DE is formulated. During disparity estimation, the search centre is set to PDV (Fig. 3.4) and two conditions are checked: (i) Does the frame belong to TL3 or TL4? (ii) Is the macroblock simple, normal or complex? If the frame belongs to TL4 and the macroblock is of type 'simple', the search range is reduced to 1/8th of the initial search range. At the same temporal level, the search range for macroblocks of type 'normal' is

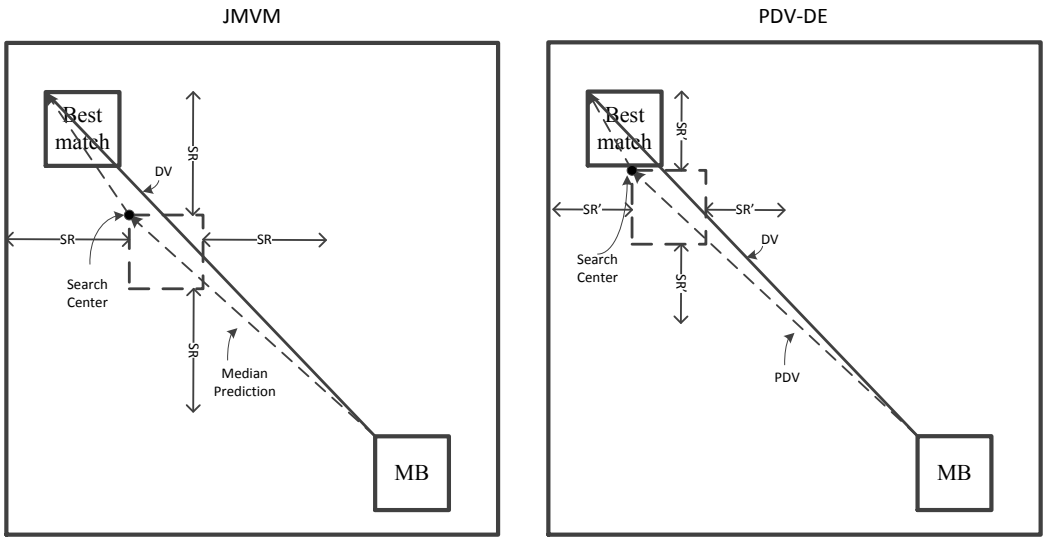


Figure 3.4: PDV-DE

reduced to 1/4th of the initial search range, and for macroblocks of type 'complex', it is reduced to half the initial search range. Since at lower temporal levels, the correlation between disparity vectors decreases, a slightly different search strategy is used to maintain similar rate-distortion performance to that of JMVM. So at TL3, if the macroblock is of type 'simple', the search range is reduced to a quarter of the initial search range, while for 'normal' macroblocks, it is reduced to half the search range. The search range is not reduced for 'complex' macroblocks. The complete search strategy of PDV-DE is shown in Fig. 3.5.

3.4.2 Stereo Motion Consistency Constraint Motion and Disparity Estimation (SMCC-MDE)

Stereo Motion Consistency Constraint (SMCC) is a geometrical constraint between the motion and disparity fields of two stereo pairs of video [70]. It is a pixel-based method where vectors are associated with pixels and denote the difference between the coordinates of corresponding pixels in different frames.

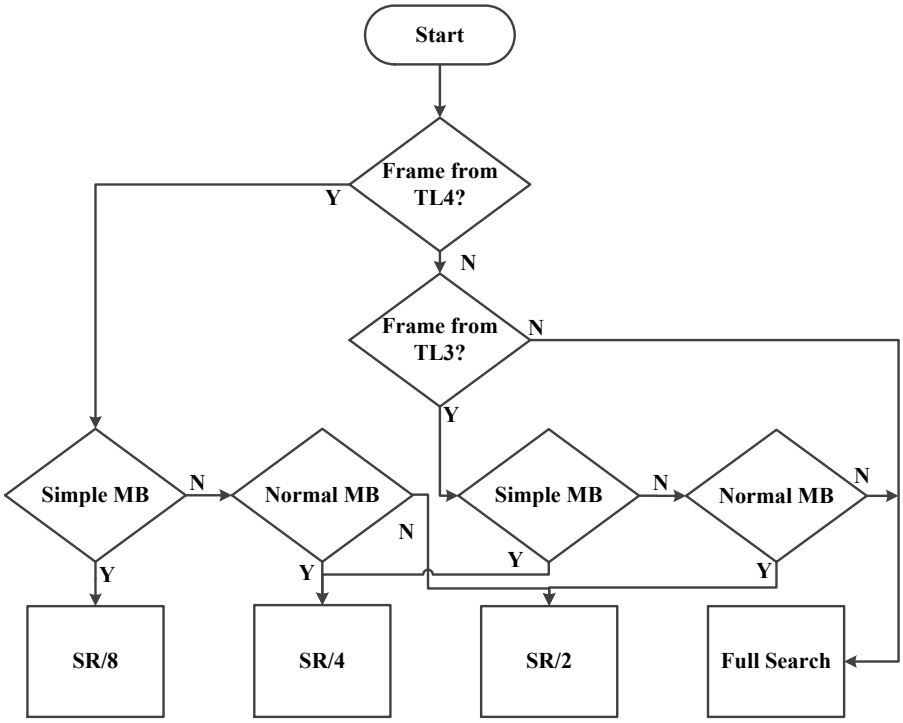


Figure 3.5: PDV-DE Search Strategy

SMCC is used to speed up the pixel matching process by providing a prediction for the optimal motion and disparity vectors.

In this section, SMCC is extended to block-based MVC. Fig. 3.6 and Fig. 3.7 illustrate the proposed method. Four frames ($F_{1,t}, F_{0,t}, F_{1,t-1}$, and $F_{0,t-1}$) from two neighbouring views (V_0, V_1) and two consecutive time instances ($t-1$: previous, t: current) are considered. The goal is to predict the motion and disparity vectors $MV_{1,t}$ and $DV_{1,t}$ for the current macroblock (MB).

$MV_{0,t}$, $MV_{0,t-1}$ and $MV_{1,t-1}$ are defined as the motion vector of the corresponding macroblock in $F_{0,t}$, the motion vector of the corresponding macroblock in $F_{0,t-1}$, and the motion vector of the corresponding macroblock in $F_{1,t-1}$ respectively (Fig. 3.6). The correlation between the motion fields of neighbouring views is exploited and an estimate $Est(MV_{1,t})$ of the motion vector $MV_{1,t}$ is obtained

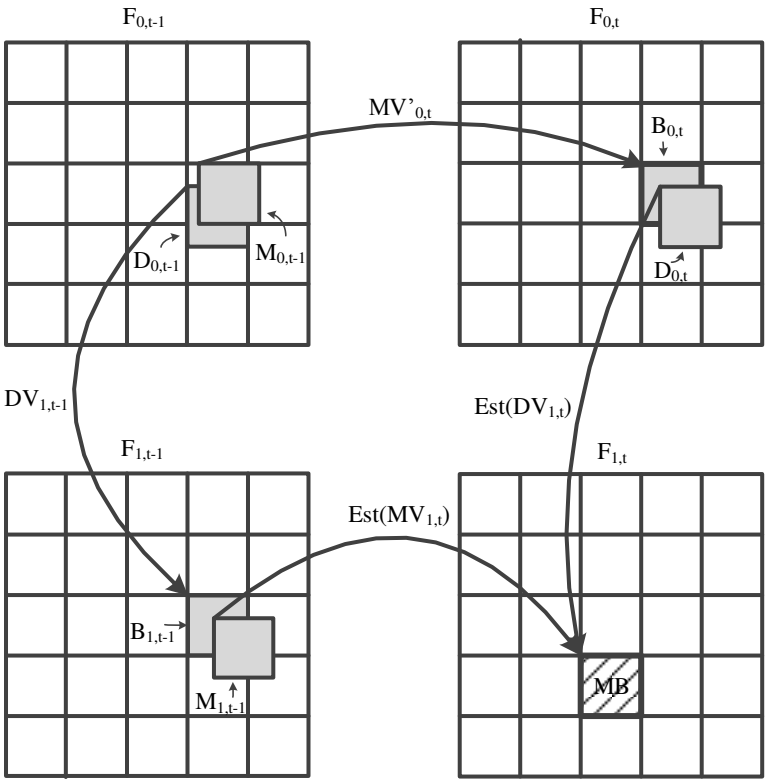


Figure 3.6: SMCC Scheme 1

as

$$Est(MV_{1,t}) = MV_{1,t-1} + MV_{0,t} - MV_{0,t-1} \tag{3.1}$$

In order to find an estimate of $DV_{1,t}$, $B_{1,t-1}$ is defined as the macroblock with maximum overlap with the optimal match in $F_{1,t-1}$ for the current macroblock. Then $DV_{1,t-1}$, the disparity vector of $B_{1,t-1}$, is used to obtain a macroblock $D_{0,t-1}$ in frame $F_{0,t-1}$. Next, the macroblock $B_{0,t}$ in frame $F_{0,t}$ whose motion vector $MV'_{0,t}$ is associated with the macroblock $M_{0,t-1}$ in $F_{0,t-1}$ with maximum overlap with $D_{0,t-1}$ is found (Fig. 3.7).

If the motion and disparity compensated macroblocks in frames $F_{0,t}$, $F_{1,t-1}$,

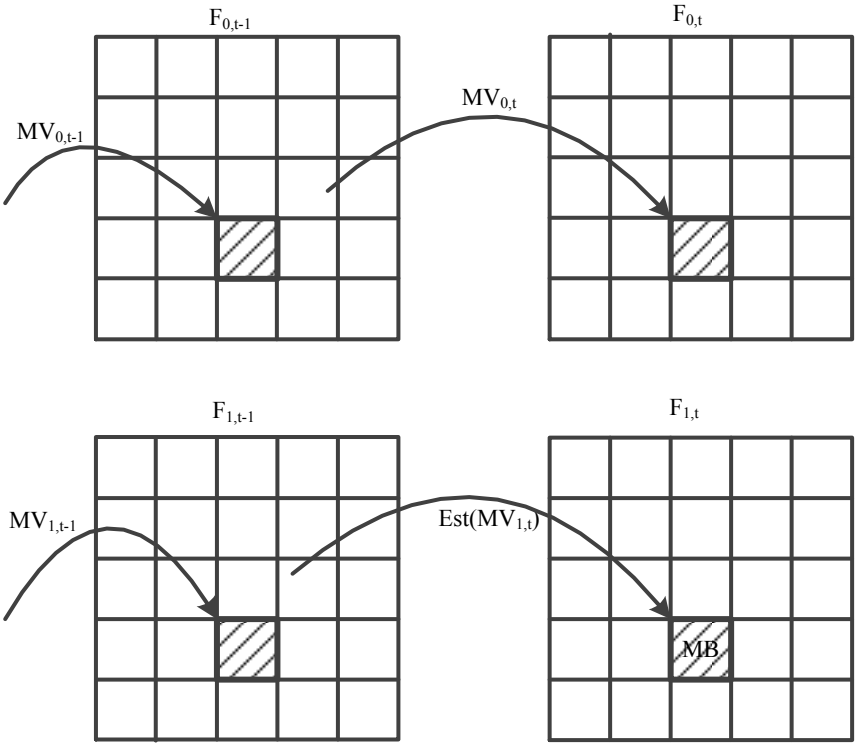


Figure 3.7: SMCC Scheme 2

and $F_{0,t-1}$ are perfectly aligned on macroblock boundaries, then, by analogy with pixel-based stereo motion consistency, we have

$$MV'_{0,t} + Est(DV_{1,t}) = Est(MV_{1,t}) + DV_{1,t-1} \tag{3.2}$$

Thus, given $Est(MV_{1,t})$, $DV_{1,t-1}$, and $MV'_{0,t}$, one can use (3.2) to find an estimate of $DV_{1,t}$.

The next step is to refine the estimated motion vector (Fig. 3.8). This is done by setting the search range to

$$SR = \lceil \max(|Est(MV_{1,t}x)|, |Est(MV_{1,t}y)|) \rceil \tag{3.3}$$

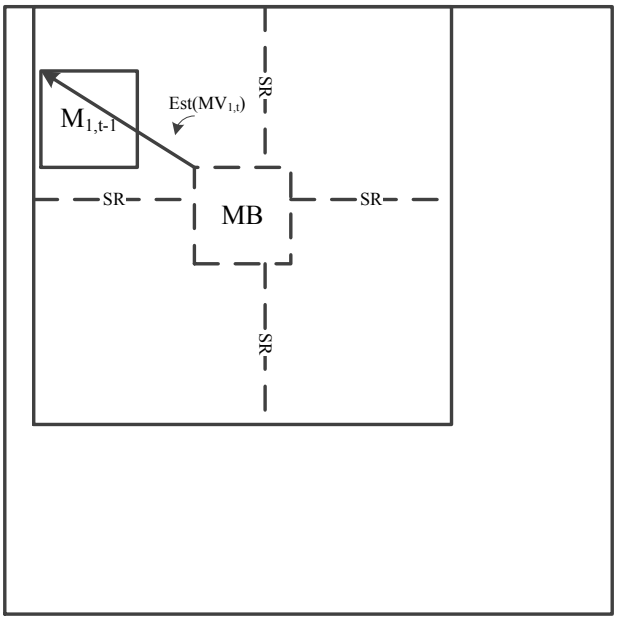


Figure 3.8: SMCC Scheme 3

A similar procedure is applied to refine the estimated disparity vector.

3.4.3 Complete Framework

In this section, a framework is presented that allows to reduce the encoding complexity at all levels. The proposed framework combines the methods in [37], [63], and [34] with the two new methods described in Sections 3.4.1 and 3.4.2. A flowchart is shown in Fig. 3.9.

1. **Input:** Three views V_0, V_1 and V_2. A macroblock MB in V_1 to encode.
2. **Pre-Processing:** Encode V_0 and V_2 using JMVM. Find the macroblock in view V_2 defined by the Global Disparity Vector (GDV) [71]. GDV represents the average disparity in MB units ($\pm$ 16 integer pixel units) between the current frame and the frame of a reference view. Obtain the mode size decisions and forward motion vectors (at 4x4 block level) of this macroblock and its eight spatial neighbours.

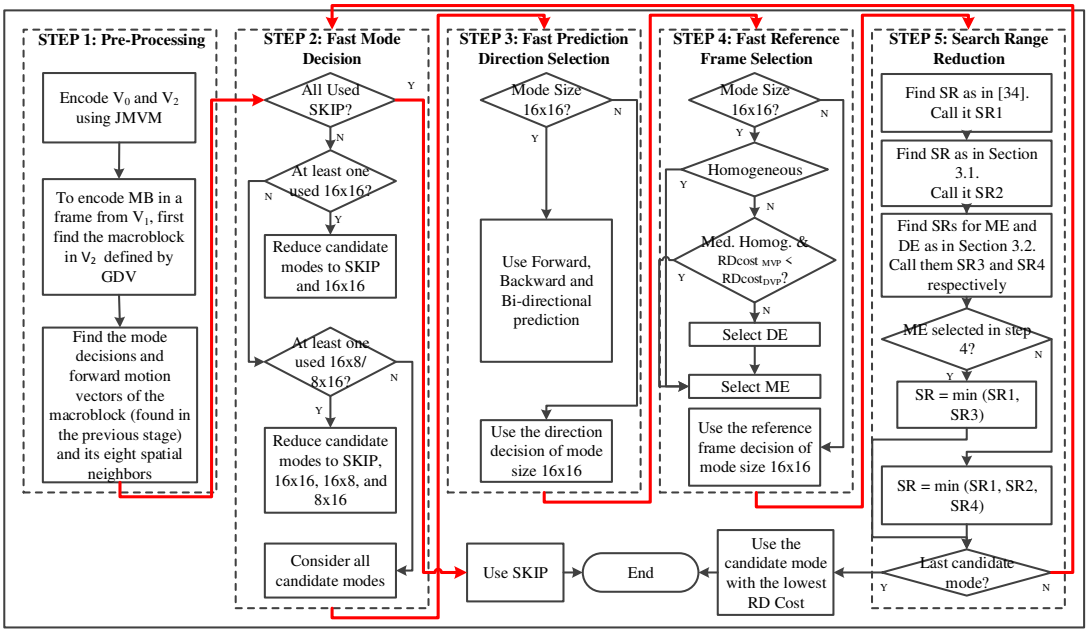


Figure 3.9: Complete Low Complexity Multiview Video Coding

3. **Fast Mode Decision:**

(i) If the nine macroblocks in the view neighbourhood use the SKIP mode, then encode MB with the SKIP mode.

(ii) If at least one of the nine macroblocks in the view neighbourhood is encoded using the 16x16 mode, then check the SKIP and 16x16 modes.

(iii) If at least one of the nine macroblocks in the view neighbourhood is encoded using the 16x8 or 8x16 mode, then check the SKIP, 16x16, 16x8, and 8x16 modes.

(iv) In all other cases, check all the mode sizes.

4. **Fast Prediction Direction Selection:**

(i) If the mode size is 16x16, search using Forward, Backward and Bi-directional Prediction.

(ii) For all other mode sizes, use the prediction direction decision of mode size 16x16.

5. **Fast Reference Frame Selection:**

   (i) If the mode size is 16x16 and

   (a) motion is homogeneous: disable DE and search the remaining (temporal) reference frame(s).

   (b) motion is medium homogeneous and the RD cost of Motion Vector Predictor (MVP) is less than that of Disparity Vector Predictor (DVP), disable DE and search the remaining (temporal) reference frames. MVP and DVP are the initial (basic) motion and disparity vectors during motion and disparity estimation.

   (c) in all other cases, search all the reference frames.

   (ii) For all other mode sizes, use the reference frame decision of mode size 16x16.

6. **Search Range Reduction:** In the identified reference frames, use the following criteria to determine the search range (SR).

   (i) Motion homogeneity: Use the search strategy proposed in [34] to determine the search range and call it SR1.

   (ii) Temporal level of the frame and mode complexity of the MB: Use PDV-DE (Section 3.4.1) to determine the search range and call it SR2.

   (iii) Stereo motion consistency constraint: Use SMCC-MDE (Section 3.4.2) to determine the search range and call it SR3 for ME and SR4 for DE.

   (iv) If both the reference frame and the current frame belong to the same view (Intra-View), then SR = min (SR1, SR3). Otherwise (Inter-View), SR = min (SR1, SR2, SR4)

3.5 Experimental Results and Discussion

This section presents and discusses the results of the experiments performed to evaluate the performance of the proposed methods. The experimental setup is presented in Section 3.5.1, the results of PDV-DE and SMCC-MDE are presented in Section 3.5.2 and Section 3.5.3 respectively while the results of CLCMVC are presented and discussed in Section 3.5.4.

3.5.1 Setup

Four test sequences recommended by the Joint Video Team (JVT) [72] were used in the experiments . These are: Ballroom, Exit, Vassar and Race1 (See Fig. 3.10). For PDV-DE and SMCC-MDE, two test sequences (Ballroom and Exit) were used. Ballroom and Race1 are examples of sequences with high motion content while Vassar and Exit represent sequences with low motion content. Ballroom and Exit are also representative of sequences with large disparities. In the experiments, three views of the test sequences are used for simulations. The second view was encoded using the proposed algorithm, and the first and the third views were used as reference views. For all sequences, for a fair comparison with other methods, the GOP size was set to 16, and the maximum search range was $\pm$ 64. Results are presented for QP values 20, 24, 28, 32, and 36. The simulations were run on a machine with Intel Core i5 dual core 2.67 GHz CPU and 4 GB RAM.

Two indicators of complexity reduction were considered: CPU time saving and number of search points saving compared to the fast TZ search method [59] in JMVM. The number of search points is the number of times the rate-distortion cost is checked during motion and disparity estimation. The following formulas are used to calculate the percentage time saving, the percentage number of search points saving, the percentage additional bitrate and the difference in Peak Signal-to-Noise Ratio (PSNR), respectively:

$$\Delta T(\%) = \frac{T_{JMVM} - T_{METHOD}}{T_{JMVM}} \times 100$$

Figure 3.10: Multiview test dataset: Ballroom (top-left), Exit (top-right), and Vassar (bottom-right) from Mitsubishi Electric Research Lab (MERL), and Race1 (bottom-left) from KDDI

$$\Delta N(\%) = \frac{N_{JMVM} - N_{METHOD}}{N_{JMVM}} \times 100$$

$$\Delta B(\%) = \frac{B_{METHOD} - B_{JMVM}}{B_{JMVM}} \times 100$$

$$\Delta PSNR(dB) = PSNR_{JMVM} - PSNR_{METHO}$$

Here $T_{JMVM}, N_{JMVM}, B_{JMVM}$, and $PSNR_{JMVM}$ represent the encoding time, the number of search points, the bitrate, and the PSNR obtained using the JMVM algorithm, while T_{METHOD}, N_{METHOD}, B_{METHOD}, and $PSNR_{METHOD}$ represent the encoding time, the number of search points, the bitrate, and the PSNR obtained using the proposed method. The number of search points is calculated

by dividing the number of search points for the whole sequence by the number of frames in the sequence. The result is further divided by the number of macroblocks in the frame to obtain the average number of search points per macroblock.

Since JMVM does not use inter-view prediction for the first and third views (except for the first frame of the third view), ΔT and ΔN are only calculated for the second view (V_1 in Fig. 3.1).

3.5.2 PDV-DE

Table 3.2 shows the results for PDV-DE. On average, PDV-DE achieves a time saving of over 35% compared to the TZ search mode of JMVM 6.0 while maintaining similar rate-distortion performance. The time saving does not vary much for different Quantization Parameter (QP) values. This is because PDV-DE exploits frames at the highest and second highest temporal levels and, for the same GOP size, the number of such frames is not affected by the QP value. Table 3.2 also shows that the Ballroom sequence has an average time saving of 34.09% with a standard deviation (SD) of 1.35 while the Exit sequence has an average time saving of 36.47% with a standard deviation of 0.72. The standard deviation values show that the Exit sequence achieves a more consistent time saving compared to the Ballroom sequence. The slightly larger time saving for the Exit sequence can be attributed to the fact that for frames at the same temporal level, PDV-DE reduces the search range primarily for 'simple' macroblocks, and the number of such macroblocks increases when there is less motion content. The decrease in bitrate for the Ballroom sequence (QP = 24, QP = 28, QP = 32, and QP = 36) suggests that at these QP values, more large modes were used which decreased the bitrate at the cost of a small decrease in PSNR.

3.5.3 SMCC-MDE

The results of SMCC-MDE are presented in Table 3.3. SMCC-MDE achieves a time saving of over 41% on average, with an average time saving of 43.97% (with a standard deviation of 3.02) and 39.60% (with a standard deviation of 2.19) for the Ballroom and Exit sequences respectively. The standard deviation values

Table 3.2: Performance of PDV-DE compared to JMVM 6.0.

	QP	ΔP	ΔB	ΔT (SD)	ΔN
Ballroom					
	20	-0.01	0.12	35.29 (0.01)	38.66
	24	-0.01	-0.19	35.52 (0.00)	39.15
	28	-0.00	-0.48	34.63 (0.03)	38.20
	32	-0.01	-0.62	32.95 (0.29)	37.69
	36	-0.04	-0.12	32.08 (0.29)	36.88
	Avg.	**0.01**	**-0.26**	**34.09**	**38.12**
	SD	0.01	0.26	1.35	0.79
Exit					
	20	-0.02	0.40	35.50 (0.10)	37.55
	24	-0.02	0.48	37.59 (0.20)	39.90
	28	-0.01	0.41	36.95 (0.15)	39.71
	32	-0.02	0.28	36.09 (0.48)	39.04
	36	-0.05	0.47	36.24 (0.25)	39.44
	Avg.	**0.03**	**0.41**	**36.47**	**39.13**
	SD	0.01	0.07	0.72	0.84

show that the Exit sequences achieves a more consistent time saving compared to the Ballroom sequence. The time saving increases with decreasing QP value. For example, for both sequences, for a QP value of 20, at least about 7% additional time saving is achieved compared to the time saving at a QP value of 36. One reason for this is that the algorithm uses estimated motion vectors to set the search range for ME. These estimated motion vectors depend on the difference between motion vectors of consecutive frames (see Eq. (3.1)). With fine quantization (low QP values), the difference between motion vectors of consecutive frames is small while with coarse quantization (high QP values), this difference is large. The rate-distortion performance of the algorithm is very similar to that of JMVM. The slight increase in bitrate is due to the more frequent use of small mode sizes, which increases the number of motion vectors. Overall, the number of search points saved per macroblock follows a similar trend to that of the time saving except for QP values of 28 and 32 for the Ballroom sequence, where the time saving increases while the saving in the number of search points decreases. One reason for this could be that all the search points are not of the same size. Smaller search points take less time to test while larger ones take longer. Hence,

Table 3.3: Performance of SMCC-MDE compared to JMVM 6.0.

	QP	ΔP	ΔB	ΔT (SD)	ΔN
Ballroom					
	20	0.00	0.30	46.42 (0.10)	48.19
	24	0.00	0.53	46.09 (0.18)	47.66
	28	0.01	0.81	45.28 (0.14)	48.48
	32	0.01	0.74	43.87 (0.16)	47.96
	36	0.00	0.77	38.20 (0.21)	45.57
	Avg.	**0.00**	**0.63**	**43.97**	**47.57**
	SD	0.00	0.19	3.02	1.04
Exit					
	20	0.01	0.04	42.25 (0.58)	46.62
	24	0.00	0.30	40.64 (0.21)	40.88
	28	0.01	0.57	40.39 (0.16)	40.67
	32	0.02	0.05	39.00 (0.11)	39.68
	36	-0.01	0.31	35.73 (0.85)	37.73
	Avg.	**0.00**	**0.25**	**39.60**	**41.12**
	SD	0.01	0.20	2.19	2.97

it is possible that in the above case, more smaller test points are searched which in turn increases the time saving but decreases the saving in the number of search points. The increase in bitrate, for different QP values, is different. This suggests that by reducing the original search range, SMCC-MDE slightly changes the overall mode distribution. Since this change is not predictable for different QP values, there is no clear trend of increasing or decreasing of the bitrate with respect to the QP value.

3.5.4 CLCMVC

Detailed results for the complete solution, called Complete Low-Complexity MVC (CLCMVC), are presented in this section. It is also shown, step by step, how the addition of each constituent method of CLCMVC reduces the number of search points and thus increases the overall time saving.

Table 3.4 shows the savings in time and number of search points for the four test sequences. The results for 'Exit' sequence show that JMVM searches on average 13,900 search points for different QP values, before a block is selected. By

Table 3.4: Comparison of fast MVC encoding techniques. S denotes the average number of search points per macroblock. ΔN and ΔT denote the percentage number of search points saving and the percentage time saving compared to JMVM 6.0.

Exit

Method/QP	36			32			28			24			20		
	S	ΔN	ΔT (SD)	S	ΔN	ΔT (SD)	S	ΔN	ΔT (SD)	S	ΔN	ΔT (SD)	S	ΔN	ΔT (SD)
JMVM	13906			13911			13901			13893			13882		
[60]	1101	92.08	76.92 (0.57)	1434	89.69	73.93 (0.40)	1751	87.40	71.48 (0.48)	2292	83.50	67.47 (0.43)	4284	69.14	53.73 (0.97)
[64]	555	96.01	87.96 (0.10)	647	95.35	85.67 (0.71)	735	94.71	85.20 (0.11)	931	93.30	83.00 (0.51)	1913	86.22	77.50 (0.36)
[34]	452	96.75	89.34 (0.11)	539	96.13	87.04 (0.77)	612	95.60	85.94 (0.36)	773	94.44	83.91 (0.89)	1458	89.5	79.5 (0.23)
[34]+[37]	314	97.74	90.31 (0.22)	466	96.65	88.41 (0.39)	547	96.07	86.44 (0.74)	731	94.74	84.45 (0.51)	1448	89.57	79.46 (0.21)
[34]+[37]+[63]	162	98.84	94.18 (0.18)	232	98.33	93.95 (0.05)	266	98.09	93.44 (0.11)	354	97.45	92.41 (0.24)	691	95.02	89.86 (0.52)
[34]+[37]+[63] + PDV-DE	113	99.19	95.38 (0.06)	165	98.81	95.11 (0.02)	205	98.53	94.6 (0.25)	273	98.03	93.69 (0.33)	514	96.30	91.46 (0.48)
CLCMVC	**109**	**99.22**	**95.42 (0.11)**	**162**	**98.84**	**95.37 (0.03)**	**189**	**98.64**	**95.05 (0.05)**	**246**	**98.23**	**94.33 (0.08)**	**439**	**96.84**	**92.27 (0.10)**

Ballroom

Method/QP	36			32			28			24			20		
	S	ΔN	ΔT	S	ΔN	ΔT	S	ΔN	ΔT	S	ΔN	ΔT	S	ΔN	ΔT
JMVM	13908			13900			13892			13882			13868		
[60]	2141	84.61	67.16 (0.17)	2734	80.33	64.58 (0.25)	3387	75.62	61.31 (0.38)	4108	70.41	56.33 (1.17)	5202	62.49	49.77 (1.06)
[64]	1113	92.00	78.98 (0.65)	1297	90.67	77.28 (0.20)	1513	89.11	76.27 (0.23)	1807	86.98	74.83 (0.44)	2384	82.81	72.93 (0.19)
[34]	1023	92.64	80.72 (0.63)	1208	91.31	78.73 (0.90)	1378	90.08	77.99 (0.90)	1626	88.29	76.25 (0.38)	2091	84.92	74.56 (0.53)
[34]+[37]	947	93.19	82.04 (0.62)	1132	91.86	79.38 (0.39)	1322	90.48	78.05 (0.06)	1585	88.58	76.9 (0.36)	2080	85.00	74.3 (0.71)
[34]+[37]+[63]	391	97.19	91.31 (0.27)	454	96.73	91.03 (0.47)	522	96.24	90.74 (0.11)	611	95.60	90.28 (0.42)	812	94.14	88.06 (0.73)
[34]+[37]+[63] + PDV-DE	305	97.81	93.15 (0.02)	359	97.42	92.90 (0.00)	473	96.60	92.22 (0.42)	562	95.95	90.92 (0.64)	651	95.31	90.01 (0.81)
CLCMVC	**287**	**97.94**	**93.61 (0.11)**	**341**	**97.55**	**93.45 (0.02)**	**400**	**97.12**	**93.11 (0.09)**	**448**	**96.77**	**92.63 (0.02)**	**546**	**96.06**	**91.62 (0.09)**

Vassar

Method/QP	36			32			28			24			20		
	S	ΔN	ΔT	S	ΔN	ΔT	S	ΔN	ΔT	S	ΔN	ΔT	S	ΔN	ΔT
JMVM	13917			13916			13912			13905			13893		
[60]	664	95.23	80.46 (0.20)	914	93.43	76.88 (0.64)	1164	91.63	76.36 (0.23)	1741	87.42	73.08 (0.16)	4208	69.71	58.59 (0.21)
[64]	395	97.16	90.41 (0.26)	502	96.39	89.57 (0.13)	577	95.85	88.94 (0.31)	823	94.08	87.73 (0.12)	2122	84.73	81.51 (0.25)
[34]	308	97.79	92.26 (0.21)	412	97.04	91.51 (0.11)	480	96.55	90.75 (0.14)	663	95.23	89.41 (0.21)	1699	87.77	83.79 (0.18)
[34]+[37]	143	98.97	93.55 (0.11)	227	98.37	92.62 (0.16)	231	98.34	91.64 (0.11)	621	95.53	89.94 (0.12)	1690	87.84	83.81 (0.09)
[34]+[37]+[63]	111	99.20	94.79 (0.04)	178	98.72	94.19 (0.06)	187	98.66	94.37 (0.13)	325	97.66	93.38 (0.02)	738	94.69	91.07 (0.03)
[34]+[37]+[63] + PDV-DE	59	99.58	95.70 (0.08)	123	99.12	94.78 (0.07)	118	99.15	94.91 (0.20)	234	98.32	93.96 (0.19)	501	96.39	91.44 (0.06)
CLCMVC	**36**	**99.74**	**95.84 (0.04)**	**57**	**99.59**	**95.58 (0.09)**	**47**	**99.66**	**95.26 (0.07)**	**201**	**98.55**	**94.22 (0.10)**	**376**	**97.29**	**92.06 (0.11)**

Race1

Method/QP	36			32			28			24			20		
	S	ΔN	ΔT	S	ΔN	ΔT	S	ΔN	ΔT	S	ΔN	ΔT	S	ΔN	ΔT
JMVM	13833			13806			13779			13772			13753		
[60]	1761	87.27	71.96 (0.12)	2177	84.23	69.35 (0.40)	2701	80.40	67.20 (0.12)	3365	75.57	62.75 (0.25)	4428	67.80	56.40 (0.04)
[64]	1064	92.31	81.36 (0.19)	1234	91.06	79.59 (0.24)	1448	89.49	78.42 (0.31)	1708	87.60	76.22 (0.29)	2120	84.59	73.61 (0.13)
[34]	927	93.30	81.54 (0.05)	1113	91.94	80.17 (0.11)	1351	90.20	78.57 (0.08)	1601	88.37	76.49 (0.10)	1973	85.65	73.62 (0.18)
[34]+[37]	762	94.49	86.52 (0.20)	987	92.85	84.79 (0.10)	1175	91.47	82.62 (0.20)	1537	88.84	79.49 (0.07)	1918	86.05	75.67 (0.06)
[34]+[37]+[63]	393	97.16	91.38 (0.22)	481	96.52	90.74 (0.15)	529	96.16	90.05 (0.19)	643	95.33	88.81 (0.33)	754	94.52	87.90 (0.14)
[34]+[37]+[63] + PDV-DE	271	98.04	93.55 (0.13)	334	97.58	93.24 (0.00)	354	97.43	92.58 (0.11)	422	96.94	91.60 (0.27)	526	96.18	90.39 (0.01)
CLCMVC	**265**	**98.08**	**93.67 (0.07)**	**314**	**97.73**	**93.41 (0.11)**	**326**	**97.63**	**92.92 (0.08)**	**397**	**97.12**	**92.07 (0.05)**	**476**	**96.54**	**91.17 (0.09)**

reducing the number of candidate modes as in [60], the number of search points can be reduced to about 1,100. This translates into an average time saving of around 68%. When the candidate modes reduction and selective disparity estimation methods are combined as in [64], the average time saving is further increased to about 83.8%. Combining candidate modes reduction and selective disparity estimation methods with the search range reduction method as in [34] takes the time saving to over 85%.

The state-of-the-art results are reported in [34]. The results in this section first show that the proposed novel framework, which combines state-of-the-art methods ([37], [63] and [34]) of different levels, achieves a significant reduction in encoding complexity compared to [34]. Table 3.4 shows that this combination can achieve, on average, a time saving of over 91.5%. Compared to [34], the complexity reduction is larger (13%) for sequences with high motion content (Ballroom, Race1) than for sequences with low motion content (6% for the Exit and Vassar sequences). There are two reasons for this. First, as inter-view redundancies are mainly found in still and low-motion regions, and all methods in [34] exploit inter-view redundancies, the room for improvement is small. Second, unlike the methods in the combination [34], which depend highly on inter-view redundancies, the method in [63] exploits the redundancies within a macroblock and the measure of such redundancies is not affected by the type of motion. Thus compared to [34], the novel framework proposed in this chapter achieves higher gains for high-motion content.

The addition of PDV-DE further increases the time saving by about 1.5%, compared to the combination [34]+[37]+[63]. More time saving is achieved for sequences with high motion content and at high bitrates. For example, the increase in time saving is, on average, over 2% for the Ballroom and Race1 sequences, which are representative of high motion sequences.

SMCC-MDE is the final constituent method in the CLCMVC framework. Its addition saves, on average, another 0.6% of the total time, the maximum being 1% for the Ballroom sequence. That takes the overall time saving to over 93.6%[1], which is an improvement of over 11% in encoding time saving compared to the

[1]Similar results were obtained for CLCMVC in [73] in which the test platform was a machine with Intel Celeron M 420 1.6 GHz processor with 2 GB RAM.

state-of-the-art [34]. Compared to the method in [34], CLCMVC saves more time at high bitrates. This is because the method in [34] relies heavily on the type of region to reduce the search range for ME and DE, while CLCMVC just uses it as one of the many indicators of the search range (others being the temporal levels and SMCC). At lower bitrates, the proportion of simple regions is bigger, so [34] is very successful. But as the bitrate increases, the proportion of simple regions decreases and so does the efficiency of [34]. Table 3.4 shows that among the four test sequences, the highest time saving is achieved for the Vassar sequence. This is understandable since, compared to the other sequences, it contains simpler texture and lesser motion. Compared to the method in [34], the proposed method saves on average around 32 s per GOP on our test platform. At high bitrates, the saving reaches about 40 s. The saving exceeds 43 s for video sequences with a high level of motion. The addition of SMCC-MDE and PDV-DE to the combination of [34], [37], and [63], results in saving an additional 4.8 s per GOP. The percentage saving in number of search points also corresponds to the behavior given by the time saving values.

Fig. 3.11 compares the rate-distortion performance of our method, JMVM, and state-of-the-art methods. The results show that our method does not penalize the rate-distortion performance.

Similar performance was observed for views $V3$, $V4$, and $V5$. The speedup and RD performance of CLCMVC compared to JMVM and [34] are summarized in Table 3.5.

3.6 Conclusion

This chapter provided a framework for low-complexity MVC. The encoding process was first split into four levels (mode decision, prediction direction selection, reference frame selection, block matching). Then previous relevant techniques at each level were identified. Combining these techniques, made it possible to reduce the encoding complexity of JMVM 6.0 with the fast TZ search by about 91.5% on average. Two new techniques were also proposed: PDV-DE and SMCC-MDE. PDV-DE exploits the correlation between disparity vectors at high temporal levels in the same view to reduce the search range for disparity estimation. SMCC-MDE

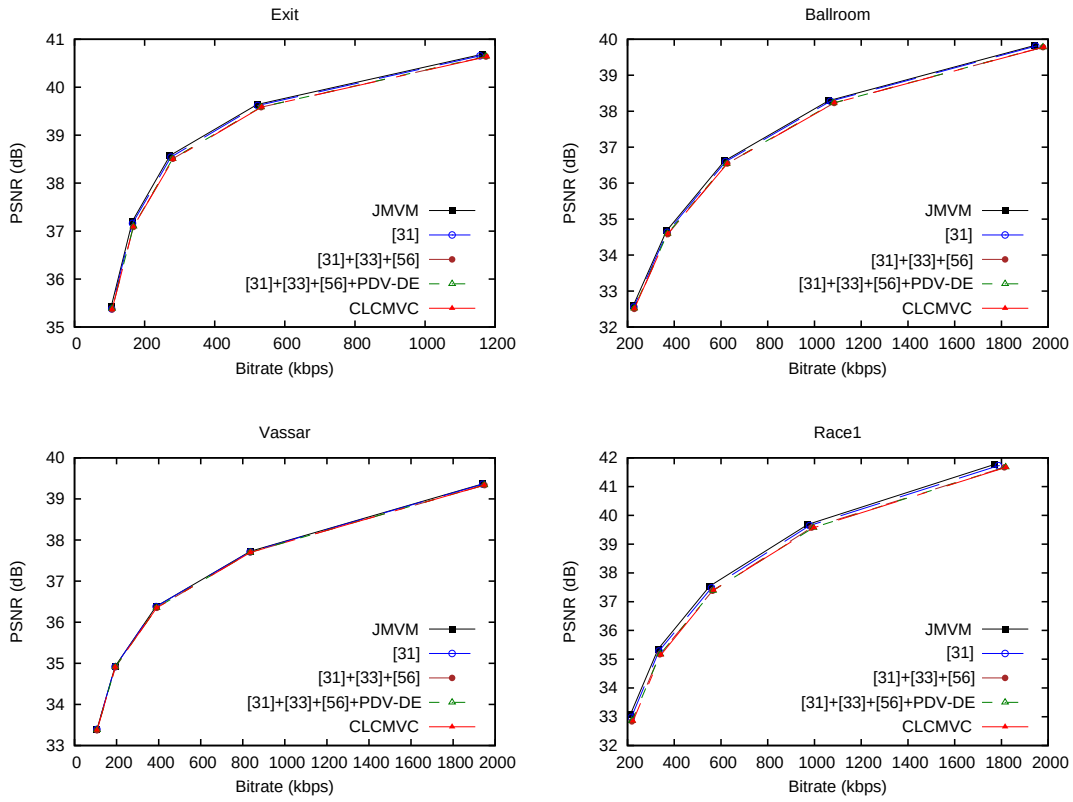


Figure 3.11: Rate-distortion performance of fast MVC encoding techniques.

Table 3.5: Comparison of fast MVC encoding techniques. ΔN, ΔT, ΔP, and ΔB denote the percentage number of search points saving, the percentage time saving, the difference in PSNR and the percentage increase in bitrate compared to JMVM 6.0. The results are shown for view $V4$ while views $V3$ and $V5$ were used as reference views.

Sequence	[34]				CLCMVC			
	ΔP	ΔB	ΔT (SD)	ΔN	ΔP	ΔB	ΔT (SD)	ΔN
Ballroom	-0.03	0.69	76.59 (0.67)	89.13	-0.07	3.06	91.11 (0.07)	96.31
Exit	-0.02	0.98	81.44 (0.47)	90.38	-0.06	3.60	91.74 (0.08)	97.24
Vassar	-0.01	0.04	86.64 (0.17)	96.04	-0.03	0.19	94.87 (0.08)	98.99
Race1	-0.05	0.97	78.27 (0.10)	90.57	-0.05	1.20	92.07 (0.08)	93.32
Average	**-0.02**	**0.67**	**80.73**	**91.53**	**-0.05**	**2.01**	**92.44**	**96.46**
SD	0.01	0.38	3.83	2.66	0.01	1.38	1.44	2.05

exploits the stereo motion consistency constraint to reduce the search range for both motion and disparity estimation. Integrating PDV-DE and SMCC-MDE in the proposed encoding framework reduced the encoding time and number of search points of JMVM 6.0 by about 93.7% and 96.9%, respectively. This was achieved at a negligible cost of 0.05 dB decrease in PSNR and 1.46% increase in bitrate.

It is expected that the proposed method will be particularly useful in applications characterized by high motion and high bitrates as in 3D TV sports since it is there that the improvement over the state-of-the-art [34] was the most significant in the performance study in this chapter. In the future, it is planned to extend the work on MVC to 3DVC [74] to jointly exploit texture and depth data in low complexity rate-distortion optimized encoders.

Chapter 4

Bayesian Early Mode Decision Technique for View Synthesis Prediction enhanced Multiview Video Coding

4.1 Introduction

The multiview plus depth representation is one of the most promising methods for providing multiview video services [75]. In a multiview plus depth representation, the information consists of multiple texture views together with their associated per-pixel depth maps. View synthesis uses the per-pixel depth maps and interpolation techniques to synthesize virtual views between camera views. Traditionally, it has been used to reduce the network and storage resource consumption of multiview video by providing N views (camera views plus synthesized views) at the decoder side while only K ($K < N$) camera views are captured, encoded, and transmitted. However, view synthesis can also improve the RD performance of MVC by providing new prediction modes for blocks to be encoded [29]. In particular, a VSP-based SKIP mode has been shown [76] to significantly improve the RD performance of MVC. Unlike the conventional SKIP mode, the View Synthesis Prediction (VSP) SKIP mode predicts a macroblock using a synthetic

reference frame. However, the RD optimized framework of MVC already uses a computationally complex motion and disparity estimation process. Adding the VSP SKIP mode in this framework further increases the computational complexity of the encoding.

In this chapter, an early mode selection technique is proposed to reduce the time complexity of a VSP SKIP-enhanced MVC coder. The proposed technique exploits the inter-view correlation between the RD costs of the VSP SKIP mode and uses Bayesian decision theory to restrict the number of candidate coding modes that are tested during the encoding. In this way, motion and disparity estimation can be skipped for a large proportion of macroblocks. Experimental results show that the encoding time can be reduced by up to 39.76% compared to the latest version of the MVC Joint Multiview Video Coding (JMVC) reference software with integrated VSP SKIP mode. This is achieved at the cost of 0.03 dB decrease in PSNR and 0.32% increase in the bitrate. Compared to a baseline technique, based on the inter-view correlations method in [37], this is a reduction of 12% in encoding time.

The remainder of the chapter is organized as follows. Section 4.2 briefly reviews the related work. Section 4.3 presents the VSP SKIP-enhanced MVC coder considered in this paper and studies optimal coding modes for this coder. The proposed Bayesian early mode decision technique is presented in Section 4.4. Section 4.5 evaluates the performance of the proposed method in terms of encoding time, bitrate, and peak signal-to-noise ratio (PSNR) and compares it to a baseline approach based on inter-view mode correlation. Conclusions are given in Section 4.6.

4.2 Related Work

No previous work has specifically addressed the problem of early mode selection for VSP SKIP-enhanced MVC. Most of the related work has focused on improving the quality of view synthesis [11], generating better depth maps [77], or pre- and post-processing of synthesized images for better prediction [78]. A number of fast algorithms [67], [68], [30], [61], [64], [34], [27] have been proposed to reduce the time complexity of motion estimation, disparity estimation, reference frame

selection, and mode decision processes in MVC. An early prediction technique for the Conventional SKIP mode in MVC was proposed in [37]. The method selects the Conventional SKIP mode for a macroblock if the corresponding macroblock identified by the Global Disparity Vector (GDV) [71] and its eight neighbouring macroblocks in a neighbouring view are encoded using this mode.

Bayesian decision theory was previously used for fast mode decision of H.264/AVC in [79] and Scalable Video Coding (SVC) in [80]. The techniques presented in these papers are not suitable for the proposed VSP-SKIP enhanced MVC coder as they do not exploit inter-view correlation. Moreover, the proposed technique is unique in its use of the RD cost of a mode as the observed feature in the Bayesian decision rule.

4.3 Preliminaries

In this chapter, an extended MVC coder is considered where a VSP SKIP mode is added to the existing eight Inter modes (Conventional SKIP, Inter16x16, Inter16x8, Inter8x16, Inter8x8, Inter8x4, Inter4x8, and Inter4x4) and two Intra modes (Intra16 and Intra4) [61]. An example of the encoding structure with three camera views, V_0, V_1, and V_2, and two synthesized views S_1 and S_2 is illustrated in Fig. 4.1.

Conventional SKIP is a special mode in which the macroblock is normally reconstructed by motion-compensated prediction using a motion vector that is derived as the median value of the motion vectors of the left, top and top-right macroblocks [81]. The VSP SKIP mode differs from the conventional SKIP mode in that the macroblock is reconstructed using the macroblock at the same position in a synthesized version of the current frame [29].

In order to synthesize the views required for VSP SKIP, the method proposed in [82] is used in this chapter, which is based on the image coordinate system, the camera coordinate system, and the world coordinate system. A pixel in the image coordinate system of the camera view is projected onto a pixel in the image coordinate system of the virtual view in two steps. First, using the intrinsic and extrinsic parameters of the reference camera and the depth information, the 3D point that corresponds to the pixel in the camera view is projected onto the world

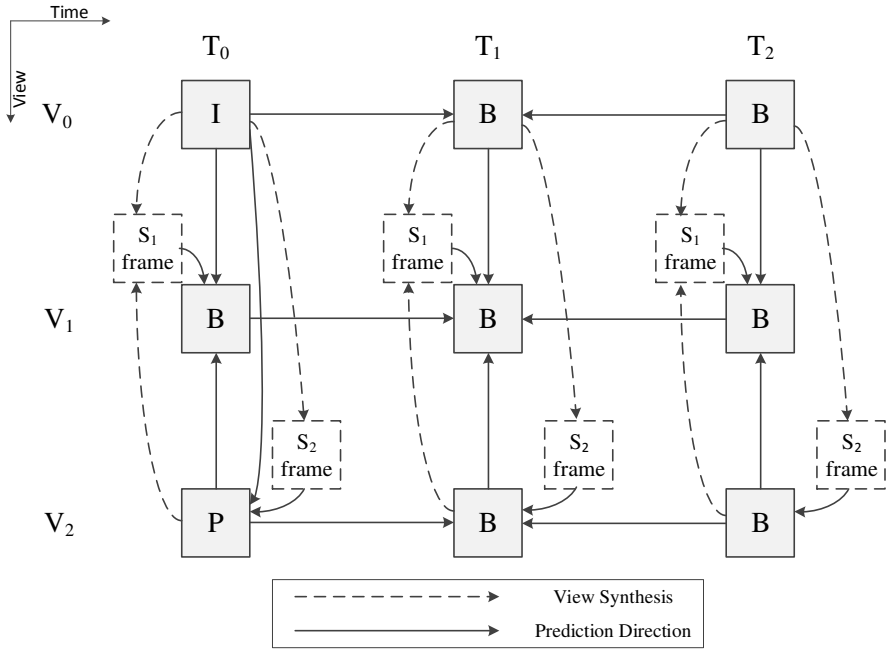


Figure 4.1: Prediction structure using view synthesis. V_0, V_1, and V_2 are camera views while S_1 and S_2 are synthesized views. Dotted lines represent the reference view(s) for view synthesis and solid lines refer to the prediction direction. The synthesized frames are used as reference frames for VSP prediction.

coordinate system. Then, from the world coordinate system it is projected onto the image coordinate system of the virtual view (using the camera parameters of the virtual view). When switching viewpoints, some background regions which are hidden behind foreground objects in the reference view, might appear in the virtual view and vice versa. This induces the hole problem. When a synthesized frame is created using only one reference frame, holes cannot be efficiently filled. This problem is solved by using two reference frames where the second reference frame is used to fill the holes.

Table 4.1 shows the proportion of modes (at the macroblock level) selected for view V_1 in the settings of Fig. 4.1 and for a VSP SKIP-enhanced JMVC 6.0 reference software [36]. The experiments were performed using four test sequences (Breakdancers and Ballet [51] of resolution 1024x768 and Poznan_Street and Poznan_Hall2 [83] of resolution 1920x1088 (See. Fig. 4.2)) and five quan-

tization parameter (QP) values (20, 24, 28, 32, 36). The number of frames in each sequence is 100, the Group of Pictures (GOP) size is 16, and the search range is $[\pm 16, \pm 16]$. In the table, Inter8x8 includes macroblocks encoded using sub modes Inter8x4, Inter4x8, and Inter4x4. It is observed that VSP SKIP, conventional SKIP and Inter16x16 are the dominant modes since their percentage proportions are very high compared to those of Inter16x8, Inter8x16, Inter8x8 and Intra modes. The dominance is more prominent in the presence of large homogeneous regions (as in the Ballet sequence). It is also evident from the table that Inter16x8, Inter8x16, Inter8x8 and the intra modes are rarely used. The standard deviation (SD) values in the table show that the proportion of each mode varies significantly from its mean value over the range of QP settings.

4.4 Bayesian Early Mode Decision Technique

This section first presents an overview of the Bayesian decision theory in Section 4.4.1. The proposed Bayesian decision rule based early mode termination method is then presented in Section 4.4.2.

4.4.1 Bayesian Decision Theory

Bayesian decision theory is a fundamental statistical approach that quantifies the trade-off between various decisions using probabilities and costs that accompany such decisions [84]. It assumes that the decision problem is posed in probabilistic terms and that the relevant probabilities are known. It has been commonly used for pattern classification problems in computer science. More recently, it has been widely used for identifying spam emails i.e., spam classification [85].

Consider that w denotes the state of nature, while $w = w_1$ and $w = w_2$ are its two possible states with prior probabilities $P(w_1)$ and $P(w_2)$ respectively. Let x be an observable variable and $p(x|w_1)$ and $p(x|w_2)$ be the respective probabilities of w_1 and w_2 given x. Then, Bayes decision rule can be used to decide between w_1 and w_2 according to the following:

Decide w_1 if $p(x|w_1)P(w_1) > p(x|w_2)P(w_2)$, otherwise decide w_2.

Table 4.1: Proportion (%) of coding modes for macroblocks.

	Breakdancers sequence					
Mode\QP (Texture)	36	32	28	24	20	Avg. (SD)
VSP SKIP	**40.36**	**34.50**	**28.82**	**24.69**	**20.16**	**29.70 (7.12)**
Conventional SKIP	45.48	46.82	46.09	41.98	32.83	42.64 (5.18)
Inter16x16	11.60	14.37	17.73	20.62	23.64	17.59 (4.29)
Inter16x8	0.96	1.63	2.49	4.23	6.44	3.15 (1.98)
Inter8x16	0.97	1.46	2.75	4.33	6.19	3.14 (1.92)
Inter8x8	0.59	1.14	1.96	3.58	7.7	2.99 (2.56)
Intra modes	0.04	0.08	0.16	0.58	3.25	0.82 (1.23)
	Ballet sequence					
Mode\QP (Texture)	36	32	28	24	20	Avg. (SD)
VSP SKIP	**29.91**	**26.54**	**23.46**	**20.76**	**18.28**	**23.79 (4.12)**
Conventional SKIP	64.30	65.66	66.41	65.97	61.86	64.84 (1.65)
Inter16x16	4.80	6.16	7.61	9.38	13.68	8.33 (3.08)
Inter16x8	0.45	0.72	1.10	1.51	2.17	1.19 (0.61)
Inter8x16	0.36	0.60	0.91	1.48	2.25	1.12 (0.68)
Inter8x8	0.19	0.29	0.47	0.83	1.56	0.66 (0.50)
Intra modes	0.00	0.03	0.03	0.08	0.20	0.06 (0.07)
	Poznan_Street sequence					
Mode\QP (Texture)	36	32	28	24	20	Avg. (SD)
VSP SKIP	**18.29**	**14.30**	**12.23**	**10.24**	**6.52**	**12.32 (3.94)**
Conventional SKIP	76.99	79.02	78.26	73.92	62.44	74.13 (6.10)
Inter16x16	3.94	5.25	7.08	11.81	21.50	9.91 (6.38)
Inter16x8	0.47	0.79	1.20	1.96	4.34	1.76 (1.39)
Inter8x16	0.25	0.53	0.92	1.45	3.57	1.34 (1.18)
Inter8x8	0.02	0.07	0.24	0.51	1.27	0.42 (0.46)
Intra modes	0.04	0.05	0.08	0.10	0.35	0.12 (0.11)
	Poznan_Hall2 sequence					
Mode\QP (Texture)	36	32	28	24	20	Avg. (SD)
VSP SKIP	**14.64**	**11.25**	**10.00**	**9.36**	**9.03**	**10.86 (2.04)**
Conventional SKIP	77.03	78.70	78.16	73.48	58.99	73.27 (7.37)
Inter16x16	7.24	8.93	10.24	14.16	22.53	12.62 (5.46)
Inter16x8	0.42	0.46	0.65	1.13	3.24	1.18 (1.06)
Inter8x16	0.52	0.53	0.76	1.37	3.38	1.31 (1.08)
Inter8x8	0.00	0.01	0.04	0.12	0.39	0.11 (0.15)
Intra modes	0.14	0.12	0.16	0.38	2.45	0.65 (0.90)

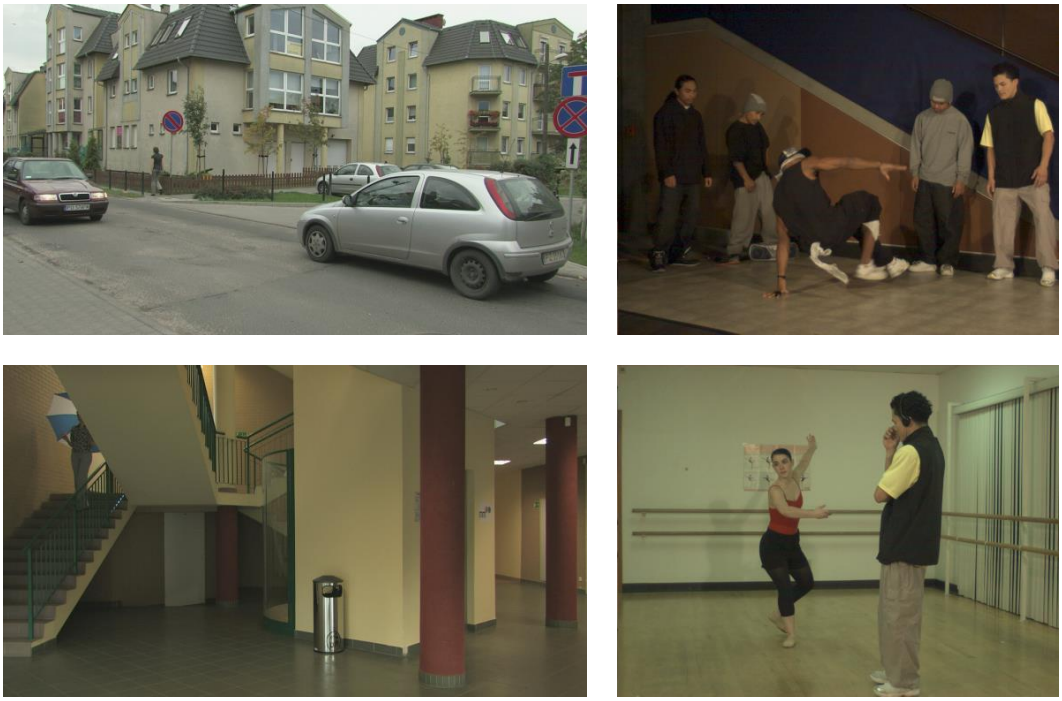


Figure 4.2: Multiview plus depth test dataset: Poznan_Street (top-left) and Poznan_Hall2 (bottom-left) from Poznan University, and Breakdancers (top-right) and Ballet (bottom-right) from Microsoft Research

4.4.2 Proposed Method

In the standard encoding scheme, the encoder considers all coding modes and selects one with minimum RD cost. The reference views (V_0 and V_2 in Fig. 4.1) are encoded first, followed by the bidirectionally predicted view (V_1 in Fig. 4.1). However, as observed in Section 4.3, VSP SKIP or Conventional SKIP may be selected much more often than the other modes. In this section, it is proposed to exploit the correlation between RD costs across views and Bayesian decision theory to avoid testing unlikely coding modes during the encoding of V_1.

Let $m_1, m_2, \ldots, m_8$ denote VSP SKIP, Conventional SKIP, Inter16x16, Inter16x8, Inter8x16, Inter8x8, Intra16, and Intra4 modes, respectively, where, as before, Inter8x8 includes sub modes Inter8x4, Inter4x8, and Inter4x4. For a given macroblock in the predicted view (V_1), let $P(m_i|x)$ denote the a posteriori probability of selecting mode m_i given an observation x of the VSP SKIP RD cost of this macroblock in V_1. From Bayes theorem, we have $P(m_i|x) = \frac{P(m_i)p(x|m_i)}{p(x)}$ where $P(m_i)$ is the a priori probability of mode m_i, $p(x|m_i)$ is the conditional Probability Density Function (PDF), and $p(x)$ is the mixture density function. Since $p(x) > 0$, Bayes decision rule implies that mode m_i should be selected if $P(m_i)p(x|m_i) > P(m_j)p(x|m_j)$. While the values of $P(m_i)$ and $p(x|m_i)$ are unknown in V_1, we can estimate them from their respective values in V_2. Fig. 4.3 shows that this approach is reasonable since the probability density functions $p(x|m_i)$ in V_1 and V_2 are similar. Here a lognormal distribution was used to model the probability density function of random variable x. Figure 4.4 shows the normalized histograms of VSP SKIP RD cost for different modes and lognormal distribution fit for the Breakdancers sequence. Lognormal distribution was also found to be an appropriate model for the other sequences (Ballroom, with inter-camera distance of 5cm; Poznan_Street and Poznan_Hall2 sequences, with inter-camera distances of 13.5cm each). Several models were considered and the lognormal one was selected based on the Bayesian information criterion [86].

Bayes decision rule does not lead to perfect mode selection as its optimality holds in a probabilistic sense only. In order to account for the Bayes error and the fact that $P(m_i)$ and $p(x|m_i)$ are estimates from a different view, the proposed method selects not only the optimal mode m_i^* in Bayesian sense but also any

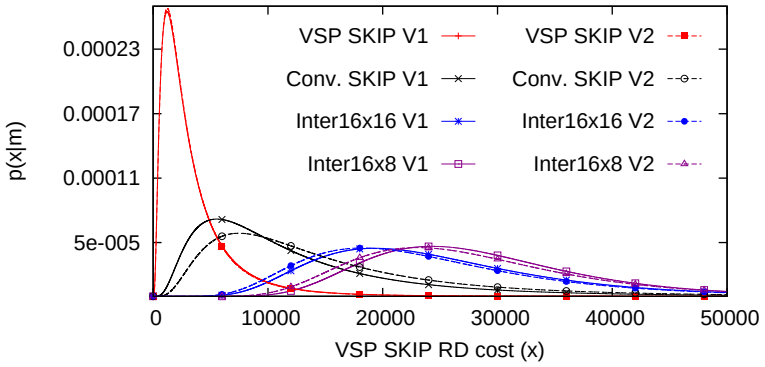


Figure 4.3: Comparison of conditional probability density functions (PDFs) in V_1 and V_2 for the Breakdancers sequence (QP = 36).

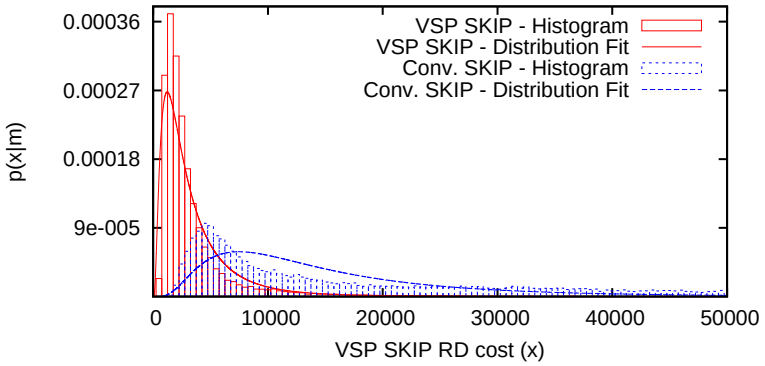


Figure 4.4: Normalized histograms of VSP SKIP RD cost for different modes and lognormal distribution fit for the Breakdancers sequence (QP = 36).

other mode m_i such that

$$\frac{P(m_i^*)p(x|m_i^*) - P(m_i)p(x|m_i)}{\sum_{j=1}^{8} P(m_j)p(x|m_j)} \leq e \tag{4.1}$$

where $e \in [0, 1]$ is a tolerance threshold. Its value can be set to control the number of candidate modes that are tested. More candidate modes are tested at higher values of e. Testing more modes improves the compression efficiency but slows down the encoding process. Hence, e can be set to provide a trade-off between compression efficiency and encoding speed.

The proposed algorithm is summarized in Algorithm 1.

Algorithm 1 Bayesian Early Mode Decision Technique

Pre-Processing:

Encode V_0.
Encode V_2.

Input: Threshold e.

Output: Modes to be checked for all macroblocks in V_1.

Calculate the a priori probabilities $P(m_i),\ i = 1, \ldots, 8$ in V_2.
Estimate the conditional probability density functions $p(x|m_i),\ i = 1, \ldots, 8$ in V_2.
while not all macroblocks are encoded **do**
    Determine the RD cost x of VSP SKIP for the current macroblock in V_1.
    Determine $i^* = \arg\max_i P(m_i)p(x|m_i),\ i = 1, \ldots, 8$ from V_2.
    for $i = 1$ to 8 **do**
        if $\frac{P(m_i^*)p(x|m_i^*) - P(m_i)p(x|m_i)}{\sum_{j=1}^{8} P(m_j)p(x|m_j)} \leq e$ **then**
            check mode i.
        end if
    end for
end while

4.5 Experimental Results and Discussion

The proposed fast mode decision technique was implemented in the VSP SKIP-enhanced MVC coder described in Section 4.3. As a baseline approach, the experiments used a method that extends the interview correlation technique proposed in [37] by selecting a SKIP mode (Conventional SKIP or VSP SKIP) for a macroblock if the macroblock at the same position and its eight neighbouring macroblocks in V_2 are encoded using the same SKIP mode. The simulations were run on a machine with Intel Core i5 dual core 2.67 GHz CPU and 4 GB RAM.[1]

Table 4.2 compares the performance of the proposed technique and the baseline approach to that of the standard coder. Parameters ΔPSNR, ΔR, and ΔT denote the increase in PSNR, increase in bitrate, and saving in encoding time, respectively, compared to the standard coder.

[1]The simulations were also tested on another machine with Intel Core i5 1.60 GHz CPU and 4 GB RAM and the results were consistent.

Table 4.2: Comparison of the baseline approach and the proposed method with respect to the standard coder.

Breakdancer sequence									
	Baseline			Proposed ($e = 0$)			Proposed ($e = 0.995$)		
QP	ΔPSNR	ΔR(%)	ΔT(%) (SD)	ΔPSNR	ΔR(%)	ΔT(%)(SD)	ΔPSNR	ΔR(%)	ΔT(%)(SD)
20	-0.02	0.02	5.86 (0.02)	-0.77	12.10	44.81 (0.01)	-0.06	0.89	26.74 (0.01)
24	-0.01	0.21	8.59 (0.07)	-0.63	14.22	46.89 (0.02)	-0.06	0.91	28.84 (0.04)
28	-0.02	0.68	12.23 (0.12)	-0.41	14.56	46.41 (0.08)	-0.07	0.29	29.64 (0.08)
32	-0.03	1.36	13.39 (0.23)	-0.21	11.07	50.51 (0.13)	-0.07	0.14	34.01 (0.16)
36	-0.07	1.48	15.36 (0.59)	-0.15	4.94	55.05 (0.35)	-0.05	0.06	36.20 (0.43)
Average	**-0.03**	**0.75**	**11.07**	**-0.43**	**11.38**	**48.73**	**-0.06**	**0.46**	**31.09**
SD	0.02	0.59	3.42	0.24	3.47	3.67	0.01	0.37	3.48
Ballet sequence									
	Baseline			Proposed ($e = 0$)			Proposed ($e = 0.995$)		
20	-0.01	0.01	10.83 (0.04)	-0.38	12.86	48.24 (0.14)	-0.04	0.43	22.32 (0.06)
24	-0.03	0.28	23.76 (0.01)	-0.32	13.70	49.65 (0.04)	-0.05	0.81	25.15 (0.03)
28	-0.07	1.16	24.46 (0.01)	-0.22	12.16	50.31 (0.02)	-0.07	0.77	27.27 (0.09)
32	-0.08	0.70	29.15 (0.17)	-0.15	9.37	50.44 (0.76)	-0.07	0.66	33.11 (0.19)
36	-0.07	0.43	30.42 (0.01)	-0.10	4.87	47.08 (0.04)	-0.03	0.26	35.82 (0.07)
Average	**-0.05**	**0.51**	**23.72**	**-0.23**	**10.59**	**49.14**	**-0.05**	**0.59**	**28.73**
SD	0.03	0.39	6.94	0.10	3.21	1.29	0.01	0.02	5.01
Poznan_Street sequence									
	Baseline			Proposed ($e = 0$)			Proposed ($e = 0.995$)		
20	-0.01	0.01	3.83 (0.11)	-0.07	1.83	46.79 (0.13)	-0.01	0.06	18.95 (0.17)
24	-0.02	0.08	5.98 (0.15)	-0.08	3.68	53.98 (0.14)	0.00	0.11	25.54 (0.23)
28	-0.02	0.40	23.54 (0.23)	-0.12	6.59	49.91 (0.20)	-0.01	0.06	27.35 (0.29)
32	-0.02	1.14	24.23 (0.42)	-0.14	9.72	46.85 (0.43)	-0.01	0.08	39.76 (0.62)
36	-0.03	1.28	30.98 (0.06)	-0.21	10.48	31.70 (0.44)	-0.01	0.04	33.55 (0.48)
Average	**-0.02**	**0.58**	**17.71**	**-0.12**	**6.46**	**45.85**	**-0.01**	**0.07**	**27.03**
SD	0.01	0.53	10.80	0.05	3.35	7.55	0.01	0.02	5.01
Poznan_Hall2 sequence									
	Baseline			Proposed ($e = 0$)			Proposed ($e = 0.995$)		
20	-0.02	-0.09	2.64 (0.12)	-0.21	1.56	50.90 (0.07)	-0.05	0.91	21.24 (0.11)
24	-0.05	1.01	15.97 (0.03)	-0.15	6.90	47.65 (0.02)	-0.01	0.25	26.72 (0.03)
28	-0.10	3.03	26.71 (0.02)	-0.21	11.64	54.29 (0.01)	0.00	-0.44	31.92 (0.01)
32	-0.18	2.61	27.42 (0.40)	-0.41	17.81	47.36 (0.40)	-0.01	0.33	36.22 (0.49)
36	-0.22	2.78	29.48 (0.03)	-0.63	9.76	44.86 (0.03)	-0.02	-0.14	39.63 (0.03)
Average	**-0.11**	**1.87**	**20.44**	**-0.28**	**9.49**	**48.18**	**-0.02**	**0.18**	**31.15**
SD	0.00	1.21	10.07	0.18	5.36	3.26	0.02	0.46	6.57
Overall Average	**-0.04**	**0.93**	**18.23**	**-0.33**	**10.98**	**47.97**	**-0.03**	**0.32**	**29.50**
Overall SD	0.03	0.55	4.65	0.11	1.87	1.27	0.02	0.21	1.73

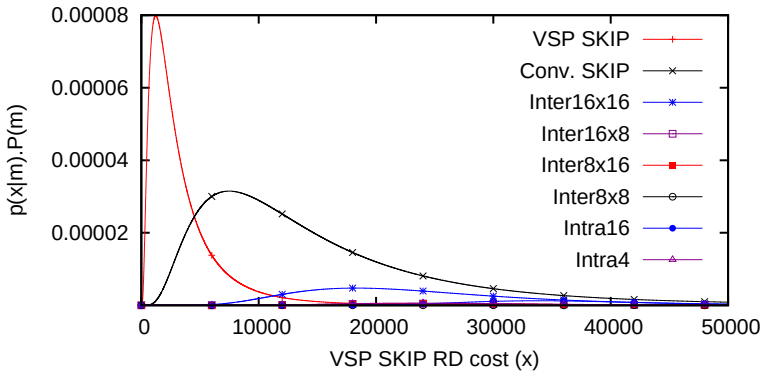


Figure 4.5: Product of conditional PDFs and *a priori* probabilities of different modes for the Breakdancers sequence (QP = 36).

For the proposed technique, all encoding steps are taken into consideration in the calculation of the encoding time. This includes the estimation of $P(m_i)$ and the fitting of the lognormal distribution model to the samples in V_2 (Steps 1 and 2 in the main part of Algorithm 1).

The table shows results that correspond to the Bayes decision rule ($e = 0$ in Algorithm 1) and to $e = 0.995$. For $e = 0$, the proposed approach reduced the encoding time by 47.97% on average. However, the loss in rate-distortion performance was significant. For $e = 0.995$, the average saving in encoding time was smaller but still significant (29.50%) while the loss in rate-distortion performance was negligible. The overall standard deviation (SD) of 1.73 shows that the algorithm performs consistently well for all the test sequences. It is to be noted that, after testing many different values of e experimentally, the value $e = 0.995$ was found to provide a good trade off between compression efficiency and encoding time. Further measurements showed that for this value of e, our algorithm found the optimal mode for 95.03% of the macroblocks on average (see Table 4.3) for the four test sequences. Table 4.3 shows that the termination accuracy for different sequences become more consistent as the QP value increases. It also shows that among the tested sequences, the termination accuracy of Poznan_Street is the most consistent with a standard deviation of 1.67.

The results show that the reduction in encoding time, generally, increases

Table 4.3: Percentage of macroblocks for which the proposed algorithm finds the optimal mode. The results are shown for $e = 0.995$.

QP\Sequence	Poznan_Hall2	Poznan_Street	Breakdancer	Ballet	Avg. (SD)
20	78.91	94.86	93.25	86.67	88.43 (5.63)
24	92.99	97.68	95.08	90.56	94.08 (2.35)
28	97.67	98.20	95.95	93.41	96.31 (1.67)
32	98.29	99.31	97.48	95.79	97.72 (1.15)
36	99.08	99.50	98.95	96.95	98.62 (0.88)
Average	**93.39**	**97.91**	**96.14**	**92.68**	**95.03**
SD	7.54	1.67	1.96	3.72	3.64

as QP increases. This is because when QP increases, the quantization becomes coarser, fewer details are preserved, and more large modes are selected. This benefits the proposed method, which mostly predicts large modes such as VSP SKIP, Conventional SKIP, and Inter16x16 (Fig. 4.5).

The baseline approach reduced the encoding time by only 18.23% on average. Since the synthesized view S_2 is constructed using only one reference frame, its quality is lower than that of S_1, which is constructed from two reference frames (Fig. 4.1). Consequently, the contribution of VSP SKIP in V_2 is smaller than in V_1, and the VSP SKIP mode decisions in V_1 cannot be efficiently predicted using the VSP SKIP mode decisions in V_2.

4.6 Conclusion

An early mode decision technique for View Synthesis Prediction-enhanced Multiview Video Coding was proposed in this chapter. The proposed method uses Bayesian decision theory to speed up the encoding by reducing the number of candidate coding modes. It reduced the encoding time of the VSP SKIP-enhanced JMVC 6.0 by up to 39.76% while preserving the RD performance. This is a reduction of around 12% compared to a baseline technique based on the inter-view correlations method in [37]. It is expected that more time savings can be achieved if the proposed method is combined with techniques ([87], [34]) that can efficiently predict non-VSP coding modes. The experimental results obtained in this chapter also show that the methods based on inter-view mode correlation might not

be suitable for predicting the VSP SKIP mode because of the difference in the quality of the synthesized reference frames in neighbouring views.

Chapter 5

Temporal and Inter-view Consistent Error Concealment Technique for Multiview plus Depth Video Broadcasting

5.1 Introduction

In video broadcasting, video data is compressed and transmitted to the home over satellite, cable, or terrestrial delivery channels. Because modern video compression schemes use entropy coding and inter-frame coding, a single transmission error can lead to error propagation that may affect several frames. In order to protect the transmitted data against transmission errors, video broadcasting systems generally use forward error correction (FEC). However, FEC cannot guarantee perfect recovery of the transmitted data. For this reason, FEC is often used in conjunction with error concealment at decoder, which aims at masking the effect of residual transmission errors.

In this chapter, error concealment is studied in the context of MVD [9] video. It is assumed that the multiview texture and depth videos are compressed independently of each other using MVC [24] (Fig. 5.1) standard (commonly used ([88, 89, 49]) for the compression of MVD videos).The texture data and depth

maps are encapsulated into separate packets and are broadcasted over a channel with transmission errors. Due to the high compression efficiency of MVC, it is possible to compress and encapsulate a complete frame of a low resolution sequence into a single packet. Thus, the loss of a single packet can result in the loss of a whole frame. In case frames are encapsulated into multiple packets, whole frame loss is still highly probable for two reasons: (i) multiple packet burst losses are very common in video broadcast [90, 91] and are likely to corrupt all the packets of one frame, (ii) many decoders discard the full video frame even if a single packet containing parts of the video frame data is lost [92]. Hence, it is considered that transmission errors lead to the loss of complete frames, such that efficient error concealment techniques are required to reduce their effect on the decoded video quality.

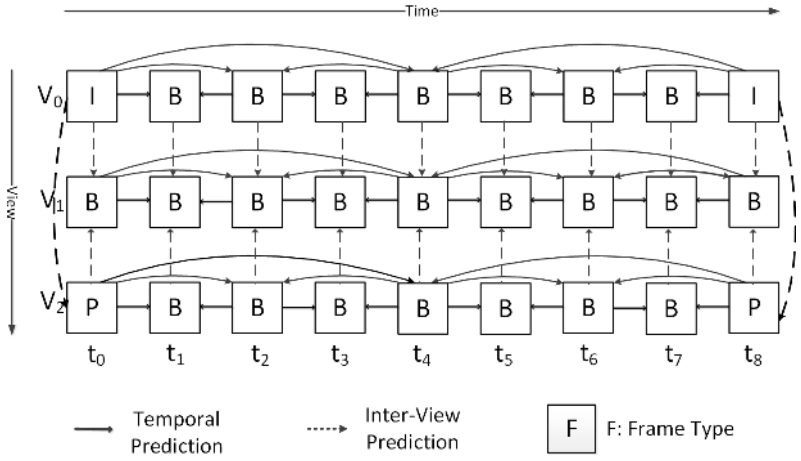


Figure 5.1: A typical MVC prediction structure

For multiview video, it is not only important for the concealment technique to reconstruct the individual frames with high fidelity but also to preserve the consistency between frames i.e., the corresponding pixels in the neighbouring frames (of the same view as well as the neighbouring views) should have consistent color information. The consistency requirement is ignored in existing error concealment methods [93]-[94]. In many 3D applications, however, frames are not viewed independently. Consequently, inconsistent frames can lead to an inconsistent reconstruction of 3D scenes which may negatively affect the viewing

experience.

This chapter addresses this fundamental problem by proposing a scene-consistent error concealment method for MVD videos. A novel metric for consistent reconstruction is introduced. The proposed method exploits inter-view and temporal correlations, as well as the geometry of MVD frames to build a set of candidate blocks for error concealment. The proposed new consistency-based cost function is then used to select the best candidate blocks for concealment. Experimental results show that, compared to the baseline BMA and two standard methods, the proposed method can reconstruct the lost frames with high fidelity while maintaining at the same time a high level of consistency between frames of the same view (temporal consistency) and those of the neighbouring views (inter-view consistency). Normally, a user is expected to be watching the video from a particular viewpoint for some time before switching to another view. In other words, compared to watching a frame from the same view, the probability of view switching is low. Nevertheless, it is still important to consider inter-view consistency, since it is not only affected if a frame in a view is lost while switching to it but also by the loss of a past frame in that view. Hence, both temporal and inter-view consistencies are considered in this chapter. The proposed method could be useful in reducing flickering artefacts in 3D videos which are often caused by inconsistencies in video frames.

The remainder of the chapter is organized as follows. In Section 5.2, existing error concealment techniques for 2D and 3D video are reviewed. In Section 5.3, the proposed new metric and the proposed scene-consistent error concealment method are introduced. Simulation results are presented in Section 5.4. Finally, conclusions are given in Section 5.5.

5.2 Related Work

An overview of existing error concealment methods for 3D video is presented in this section. Methods proposed for 2D video are also briefly overviewed as many error concealment methods for 3D video are extensions of the 2D ones. Recently, many error concealment methods have been proposed for 2D and 3D video applications. For example, Yan and Gharavi [95], Ji, Zhao, and Gao [96]

and Guo et al. [97] proposed error concealment methods for H.264/AVC [20] and H.264/SVC [22] based 2-D video and [93, 38, 98, 99, 39, 94] target error concealment for 3D video transmission.

For 2D video, a Motion Vector Extrapolation (MVE) [100] based hybrid motion vector extrapolation method is proposed in [95]. This method considers extrapolated motion vectors (MVs) at both the pixel and the block levels and discards inaccurate MVs on the basis of their euclidean distances from other MVs in the selected set of candidate MVs. Ji, Zhao, and Gao [96] and Guo et al. [97] propose error concealment methods for scalable video coding (SVC). The method proposed in [96] is based on the temporal direct mode which is usually used in regions with slow or no motion. That is why for content with fast motion or complex texture, it might not be as efficient. The authors in [97], propose the Intra-layer and Inter-layer concealment methods. The Intra-layer methods use the information of the same spatial or quality layer to conceal a lost frame while the Inter-layer methods use the information of the base layer to conceal a lost frame from one of the enhancement layers. While it might be possible to extend these methods to recover lost MVD frames, they do not address the issue of inconsistencies in the recovered frames.

For 3D video, Song et al. [93] proposes three error concealment methods, temporal bilateral error concealment, inter-view bilateral error concealment, and multihypothesis bilateral error concealment for MVC. The first uses spatio-temporal correlations in each view, the second uses inter-view correlation, while the third recovers the motion and disparity vectors of the lost block using the block matching principle [101]. For block losses in video plus depth (V+D) format, Liu, Wang and Zhang [38] jointly consider the depth and neighbouring spatial and temporal information to recover the lost MVs for the corrupted blocks. The application of these methods is limited to the scenario of block losses since they depend on the availability of correctly decoded neighbouring MBs from the same frame as that of the lost MBs.

Among the methods proposed for whole-frame loss concealment in 3D video, inter-view motion vector correlation of MVC is exploited in [98]. This method first estimates the overall disparity between corresponding frames from neighbouring views. In case a frame in one view is lost, its corresponding MBs are identified in

a neighbouring view using the overall average disparity. The MVs of the corresponding MBs are then used to reconstruct the lost frame. This method assumes that global disparity is a good approximation of local disparities. This might not always be the case and hence its efficiency decreases as the difference between global and local disparities increase. For frame losses in stereo plus depth format, Chung, Sull and Kim [99] use 3D image warping technique to determine matching pixels between neighbouring views and perform the reconstruction based on the similarities of the motion vectors and the intensity differences of matching pixels. Hewage et al. [39] proposes to share motion vectors between the texture and depth videos in case a frame from either of them is lost. This method might not be very efficient when a frame contains objects with different textures placed at the same depth. Similarly, for frame losses in V+D format, Yan and Zhou [94] propose to use depth differences as a measure of the reliability of the MVs in a set of candidate MVs.

Generally, all the above methods involve the following two steps: 1. Extract several candidates for error concealment. 2. Use an evaluation criteria to discard less likely candidates and select the final candidate. The first step is non-trivial in both block and frame loss methods. The second step is even more complicated. Block based methods usually use some extension of Boundary Matching Algorithm (BMA) [40] which finds the difference between the outer boundary pixels of the available neighbouring blocks and the inner boundary pixels of the concealed block, while frame loss methods are usually based on simple intuitions such as the maximum overlap method [95] in case of MVE. In such methods, the MVs extrapolated from the pixels in the previous frame may not be accurate, i.e., some MVs are likely to be wrongly extrapolated, especially in large motion scenes. Another problem with these methods is that they only aim to recover the contents of the lost frame without taking into consideration the effect on the consistency between the frames. In this way, the consistency between the frames that represent the 3D scene might be disturbed. Scene consistency in 3D video has been studied in the context of seam carving [102], image segmentation [103], feature points detection [104] and view synthesis [105, 106]. It has not been applied to the error concealment problem. For whole frame losses in 3D video, it is desirable to have a cost function for selecting candidate data that can efficiently

conceal the lost frames by recovering the contents of the lost frames with high consistency between their inter-view and temporal neighbours and hence provide a consistent viewing experience to the users.

5.3 Scene consistent error concealment

A new scene consistent error concealment technique is proposed. It uses the inter-view, temporal and geometric information of the neighbouring texture as well as depth frames to recover the lost frames with high consistency in MVD sequences.

Section 5.3.1 gives an overview of the MVD video format and the view synthesis process. The proposed scene consistency model is presented in Section 5.3.2 while Section 5.3.3 describes the methods used to create the four candidate MBs for reconstruction which are used by the proposed consistency model in Section 5.3.2.

5.3.1 Preliminaries

A typical MVD setup is illustrated in Fig. 5.2. Frame $F_{v,t}$ has two temporal neighbours F_{v,t^-} and F_{v,t^+}, two view neighbours $F_{v^-,t}$ and $F_{v^+,t}$ and a depth frame $D_{v,t}$.

Let (i,j) denote the position of a pixel in frame $F_{v,t}$. Its corresponding positions (i_1,j_1) and (i_2,j_2) in frames $F_{v^-,t}$ and $F_{v^+,t}$ can be found using 3D warping (e.g., as in [82]). This technique uses the depth value $D_{v,t}(i,j)$ corresponding to (i,j), the intrinsic matrices $A(v)$, $A(v^+)$ and $A(v^-)$ and the translation vectors $T(v)$, $T(v^+)$ and $T(v^-)$ of views v, v^+ and v^- respectively and the rotation matrix $R(v)$ of view v. The intrinsic matrix $A(u)$ for view u represents the transformation from the camera coordinate system of view u to its image coordinate system while a translation vector $T(u)$ and a rotation matrix $R(u)$ describe the displacement of the camera from the origin and the direction of the camera, respectively [107]. Using these quantities, pixel (i,j) in $F_{v,t}$ is first projected into world coordinates $[u,v,w]$ via

$$[u,v,w] = R(v).A^{-1}(v).[i,j,1].D_{v,t}(i,j) + T(v) \tag{5.1}$$

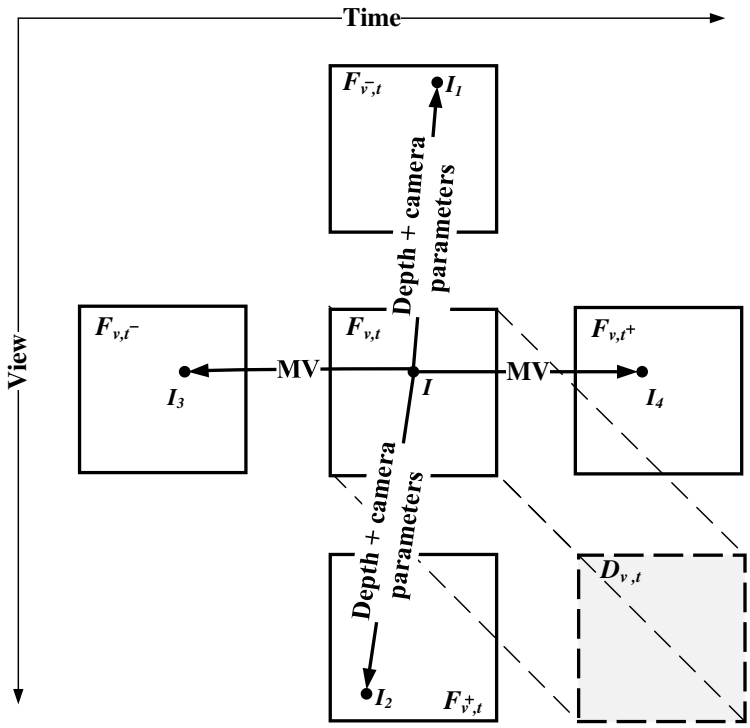


Figure 5.2: Proposed scene consistency model. The pixel values I, I_1, I_2, I_3, and I_4 represent $F_{v,t}(i,j)$, $F_{v^-,t}(i_1,j_1)$, $F_{v^+,t}(i_2,j_2)$, $F_{v,t^-}(i_3,j_3)$, and $F_{v,t^+}(i_4,j_4)$ respectively.

Next, the world coordinates are mapped onto the target coordinates $[i',j',k']$ of the frame in a target view, v', via

$$[i',j',k'] = A(v').R^{-1}(v).[u,v,w] - T(v') \tag{5.2}$$

Finally, to obtain the pixel location (i',j') (where (i',j') represents (i_1,j_1) and (i_2,j_2) when $v' = v^-$ and $v' = v^+$, respectively), the target coordinates are converted to homogeneous form, i.e., $(i',j') = (i'/k', j'/k')$. Collectively we refer to the intrinsic, translation and rotation matrices as camera parameters. The warping method introduced in this section is commonly used for view synthesis. In this work, they are used to find the pixel positions (i_1,j_1) and (i_2,j_2) in Section 5.3.2.

5.3.2 Scene consistency model

The proposed scene consistency model is presented in this section. It consists of two parts: (i) inter-view consistency and (ii) temporal consistency. For mathematical simplicity, we define the respective inconsistencies and minimize them. Let $F_{v,t}(i,j)$ denote the intensity of the pixel (i,j) in frame $F_{v,t}$. Then, the Inter-view Inconsistency (IVI) at position (i,j) of $F_{v,t}$ is defined as

$$IVI(i,j) = |F_{v,t}(i,j) - F_{v^-,t}(i_1,j_1)| + |F_{v,t}(i,j) - F_{v^+,t}(i_2,j_2)|. \tag{5.3}$$

where positions (i_1,j_1) and (i_2,j_2) are obtained using the 3D warping method explained in the previous section. In order to obtain high inter-view consistency, the intensity values $F_{v,t}(i,j)$, $F_{v^-,t}(i_1,j_1)$ and $F_{v^+,t}(i_2,j_2)$ should be similar.

Similarly, the Temporal Inconsistency (TI) at position (i,j) in $F_{v,t}$ is defined as

$$TI(i,j) = |F_{v,t}(i,j) - F_{v,t^-}(i_3,j_3)| + |F_{v,t}(i,j) - F_{v,t^+}(i_4,j_4)|. \tag{5.4}$$

where the positions (i_3,j_3) and (i_4,j_4) in frames F_{v,t^-} and F_{v,t^+} respectively are obtained by using the motion vector MV associated with $F_{v,t}(i,j)$ (Fig. 5.2), i.e., $(i_3,j_3) = (i+MVx, j+MVy)$ and $(i_4,j_4) = (i-MVx, j-MVy)$. Objects usually move between frames with quite regular motion. So if a motion vector can be used to trace an object in a past frame, the same motion vector can be used to trace the object in a future frame as well ([98]). In order to obtain high temporal consistency, the intensity values $F_{v,t}(i,j)$, $F_{v,t^-}(i_3,j_3)$ and $F_{v,t^+}(i_4,j_4)$ should be similar.

Finally, IVI and TI are combined to form the Inconsistency Cost Function (ICF).

$$ICF(i,j) = \alpha.IVI(i,j) + (1-\alpha).TI(i,j) \tag{5.5}$$

where $\alpha \in [0,1]$ is a weight factor. This metric is used to select the best macroblock to use in the concealment method in order to maximize consistency.

5.3.3 Candidate MBs for reconstruction

The proposed concealment method, a set $\mathcal{C}$ of four candidate macroblocks is defined as follows. The first candidate, MB_{r_1}, is built by using the motion vectors (MVs) of the collocated MB in the corresponding depth frame $D_{v,t}$ as in [39]. This method is named as Depth Motion Vector Sharing (DMS) (See Fig. 5.3).

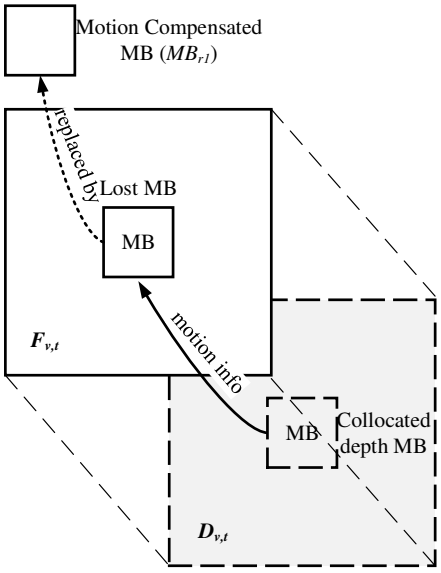


Figure 5.3: Depth Motion Vector Sharing (DMS) used to create candidate MB_{r1}

The next two candidates MB_{r_2} and MB_{r_3} are obtained as in [98] by using the MVs of the MBs in the frames $F_{v^-,t}$ and $F_{v^+,t}$ identified by using the global disparity between the current view and the respective left and right views. In this paper, we represent the global disparity between two views by the associated Global Disparity Vector (GDV) [71]. This method is named as Inter-view Motion Vector Sharing (IMVS) (See Fig.5.4).

The last candidate, MB_{r_4}, is constructed with view-synthesis [82]. First synthesized version of the lost MB is created using the left reference frame $F_{v^-,t}$ and its corresponding depth frame $D_{v^-,t}$. Then its second synthesized version is created using the right reference frame $F_{v^+,t}$ and its corresponding depth frame $D_{v^+,t}$. Finally, the two synthesized versions are merged such that the holes in one version are filled using the texture from the other. This fills up most of the

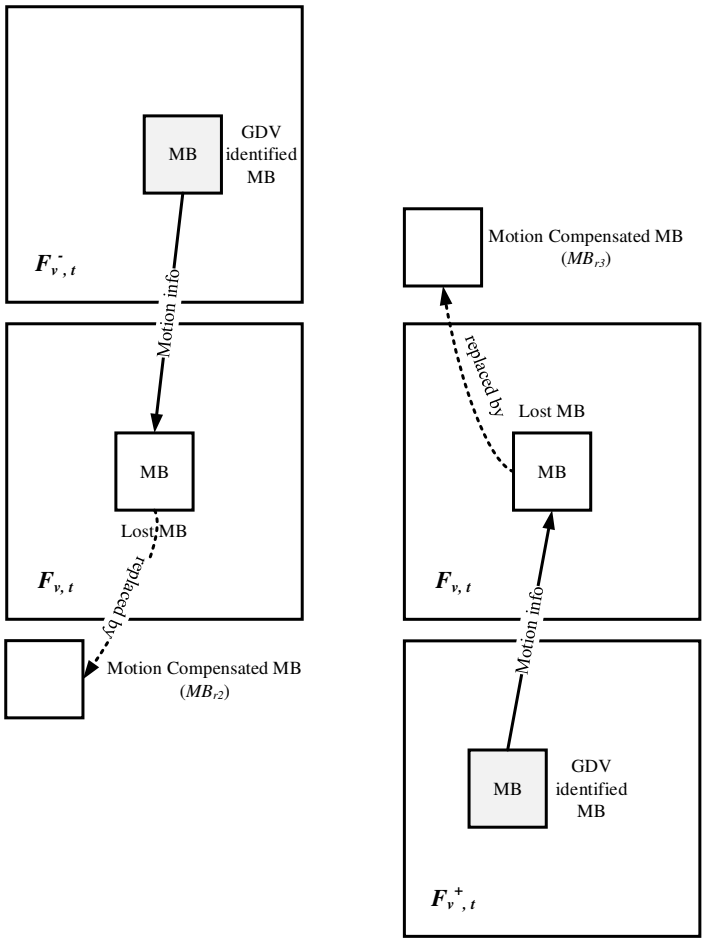


Figure 5.4: Inter-view Motion Vector Sharing (IMVS) used to create two candidate MBs, (a) MB_{r2} and (b) MB_{r3}.

large holes. In order to fill the remaining small holes, the morphological close operation is used. This method is named as View Synthesis Concealment (VSC) (See Fig. 5.5).

In order to reconstruct an MB in a lost frame, the receiver first defines a set $\mathcal{C}$ of MBs from available frames (see next Section 5.3.3 for an example). Then each 4×4 block in a MB of the lost frame is reconstructed as the 4×4 block in the same location in $\mathcal{C}$ that minimizes $\sum_{i,j} ICF(i,j)$.

A flowchart of the complete algorithm is shown in Fig. 5.6.

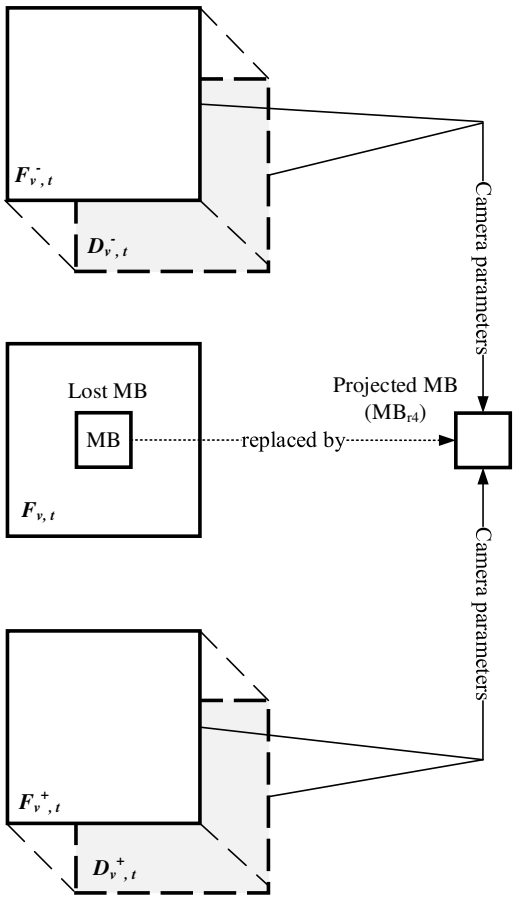


Figure 5.5: View Synthesis Concealment (VSC) used to create candidate MB_{r4}

5.4 Simulation Results

In this section, simulation results are presented to compare the performance of the proposed method to that of a baseline technique. The baseline method uses the same three methods DMS, IVMS and VSC but selectively combines them using a slightly modified version of the BMA technique [40]. BMA is commonly used for recovering a lost block for which spatially neighbouring left, right, top and bottom MBs are available. In the frame loss scenario considered in this chapter, these MBs are not available so the first row and the first column of MBs of the lost frame are recreated using DMS. Each of the remaining MBs is recreated in BMA by finding the difference between the outer boundary pixels of its left and

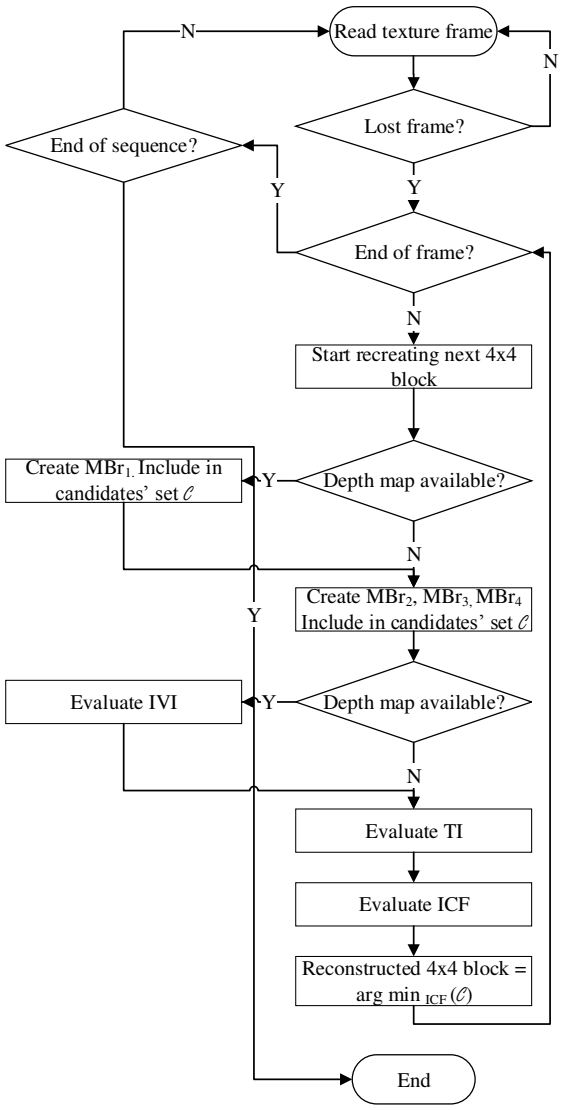


Figure 5.6: Flowchart of the proposed scene-consistent error concealment algorithm which uses the new consistency metric (ICF) to choose between candidate blocks to reconstruct each block of the lost frame. In each frame, the macroblocks and the 4x4 blocks are scanned in raster order.

top MBs and the inner boundary pixels of each of the candidate reconstructed MBs. The candidate reconstructed MB for which such a difference is the smallest is chosen for concealment of the current MB.

JMVC 8.5 reference software [41] was used to encode three texture views and their associated depth maps of four standard 3D video sequences (1024 x 768 Ballet, 1024 x 768 Breakdancers [51] and 1920 x 1088 Poznan_Street and 1920 x 1088 Poznan_Hall2 [83]). The value of α was set to 0.5 to give equal weights to temporal and inter-view inconsistencies. 100 frames of each texture and depth view of the test sequences were used. Random loss model for 5%, 10% and 20% Packet Loss Rates (PLRs) was considered and the simulations were repeated 50 times. Similar PLR values have been observed in [108]. For all sequences, each frame consisted of one slice, the frame rate was 25 frames per second. Considering the broadcast scenario, GOP size was set to 12. Each frame of the texture and depth sequences was encapsulated in a separate packet. Hence packet loss rate corresponds to frame loss rate. The transmission order was $V0$, $V2$, $V1$ and for each view, texture frames were transmitted before the depth frames. The quantization parameter (QP) was set to 28 for the texture. Although the depth map requires about 15 to 20% of the bit rate required to encode the original video [109], a higher bit rate (with a QP value of 20) was used to obtain high quality depth maps. This is important as they are also used in view synthesis. The texture and depth data were packetized separately. If a depth frame needed in the calculation of $IVI(i,j)$ was lost during transmission, $IVI(i,j)$ was skipped. An alternative would be to first recover the lost depth frames using conventional concealment algorithms (see [95] and [100]) and then to use the proposed complete error concealment strategy including IVI computations.

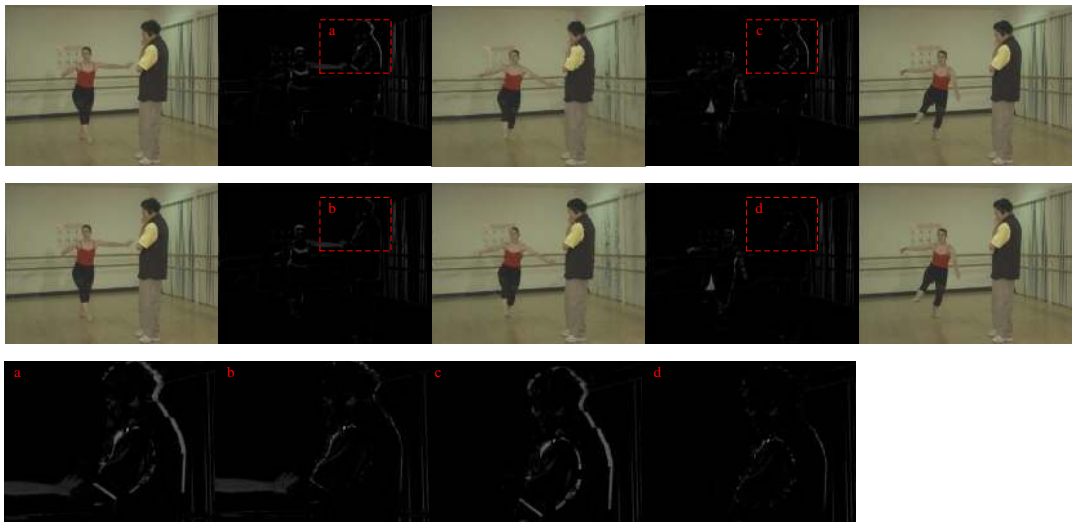
In the first experiment, the proposed consistency model is validated, i.e., the temporal and inter-view consistency of the proposed approach are visually compared with the baseline BMA method. In order to evaluate the temporal consistency, the difference between the reconstructed frame and its temporal left and right neighbours is studied. Fig. 5.7 shows that the proposed method achieves better temporal consistency compared to the baseline approach. In fact, the proposed method overcomes the limitation of the baseline approach. For example, the baseline approach is not efficient when it faces a complex texture or object boundaries (e.g., the outline of the man in Fig. 5.7). In contrast, the proposed method reduces such inconsistencies as it is evident from the visual comparisons.

In order to evaluate the inter-view consistency, we study the difference between

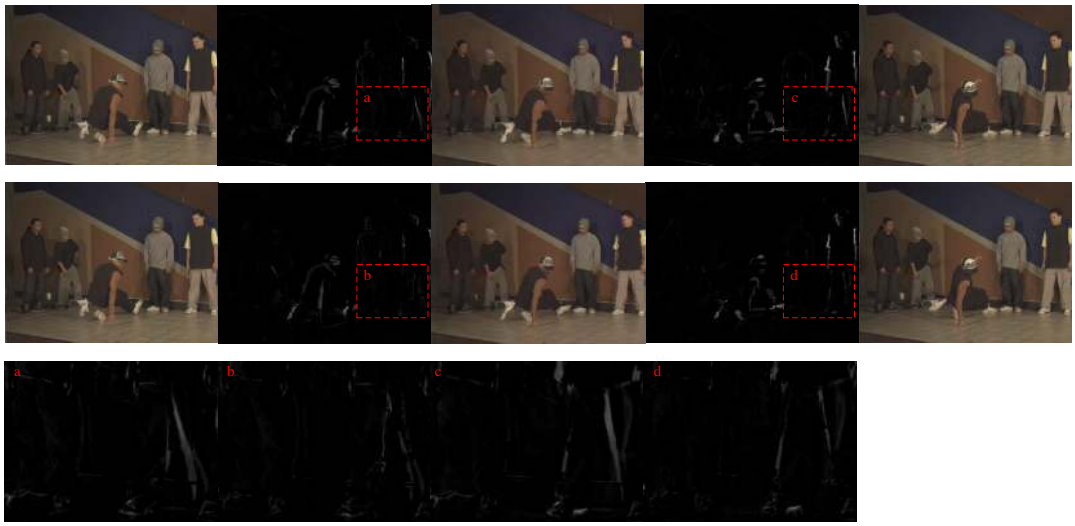
the projection of the reconstructed frame in its view-neighbouring left and right frames and the respective view-neighbouring left and right frames is studied. Fig. 5.8 and Fig. 5.9 show that the proposed method achieves better view consistency compared to the baseline approach.

In the second experiment, the two approaches are compared based on their PSNR performance. The following is evaluated: (i) the average PSNR of the reconstructed frames and (ii) the average PSNR of all the frames (including reconstructed, affected, and unaffected frames).

Table 5.1 presents a comparison of the average PSNR of the concealed frames for all PLRs for DMS, IMVS, the baseline BMA and the proposed approaches. It shows that the proposed method can reconstruct lost frames with higher fidelity than DMS, IMVS and the baseline approach. Table 5.2 shows the average PSNR over the whole sequence for the two approaches at different PLRs. Four trends are observed: (i) combining methods generally improve the PSNR performance (ii) the proposed method has higher average PSNR over the whole sequence compared to DMS, IMVS and the baseline approach, (iii) as the PLR increases, so does the gain of the proposed approach, (iv) as the baseline distance between cameras decrease (from 20 cm for Breakdancer and Ballet sequences to 13.5 cm for Poznan_Street and Poznan_Hall2 sequences), the gain of the proposed approach increases. The improved PSNR performance of the proposed approach compared to the baseline BMA approach can be mainly attributed to the fact that the baseline approach relies on the decoded left and top blocks. Due to high spatial correlation between neighbouring blocks in regions with consistent texture, these blocks are sufficient to recover the lost blocks but at object boundaries, this correlation decreases and the spatially neighbouring top and left blocks are not as useful. This limitation is largely overcome by the proposed approach which can accurately track blocks at object boundaries in a neighbouring view. The increase in gain for camera arrangements with short baseline distances can be attributed to the better inter-view correlations in such setups. This does not only show that the proposed method efficiently recovers the lost frames but it also limits error propagation to other frames. The visual results obtained in Fig. 5.10 show the 7th and 4th frame of $V1$ of the Breakdancer and Poznan_Hall2 sequences respectively. The cropped and zoomed parts of the frames confirm the



(a) Ballet



(b) Breakdancer

Figure 5.7: Comparison of Temporal Consistency. Top: baseline BMA method, Centre: proposed method, Bottom: Zoomed difference images. The top two rows show (in order from left to right): the frame F_{v,t^-}, the difference of the reconstructed frame $F_{v,t}$ and the frame F_{v,t^-}, the reconstructed frame $F_{v,t}$, the difference of the reconstructed frame $F_{v,t}$ and the frame F_{v,t^+}, and the frame F_{v,t^+} respectively. The bottom row shows zoomed portions of the difference images in the top two rows. For these results, a frame in view 1 was dropped and then concealed using the baseline BMA and the proposed methods. The zoomed parts of the difference images show higher temporal consistency (smaller magnitude of the white color) of the proposed method compared to the baseline BMA method.

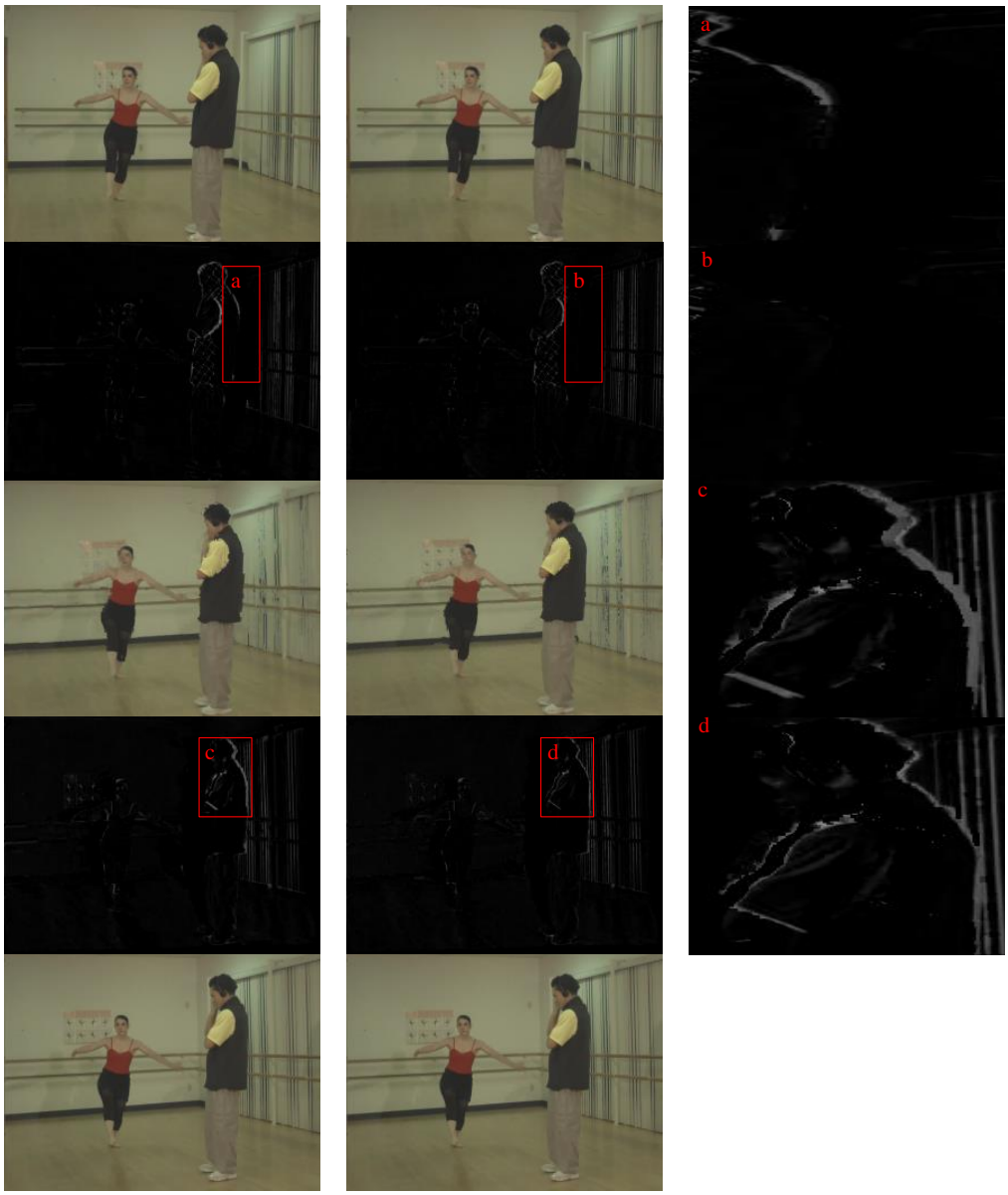


Figure 5.8: Comparison of Inter-view Consistency. Left: baseline method, Centre: proposed method, Right: Zoomed difference images. The first two columns show (from top to bottom): the frame $F_{v^-,t}$, the difference of the warped frame from $F_{v,t}$ to $F_{v^-,t}$ and the frame $F_{v^-,t}$, the reconstructed frame $F_{v,t}$, the difference of the warped frame from $F_{v,t}$ to $F_{v^+,t}$ and the frame $F_{v^+,t}$, and the frame $F_{v^+,t}$ respectively. The third column shows the zoomed difference images from the first two rows. For these results, a frame in view 1 of the Ballet sequence was dropped and then concealed using the baseline BMA and the proposed method. The zoomed parts of the difference images show higher inter-view consistency (smaller magnitude of the white color) of the proposed method compared to the baseline BMA method.

Figure 5.9: Comparison of Inter-view Consistency. Left: baseline method, Centre: proposed method, Right: Zoomed difference images. The first two columns show (from top to bottom): the frame $F_{v^-,t}$, the difference of the warped frame from $F_{v,t}$ to $F_{v^-,t}$ and the frame $F_{v^-,t}$, the reconstructed frame $F_{v,t}$, the difference of the warped frame from $F_{v,t}$ to $F_{v^+,t}$ and the frame $F_{v^+,t}$, and the frame $F_{v^+,t}$ respectively. The third column shows the zoomed difference images from the first two rows. For these results, a frame in view 1 of the Breakdancer sequence was dropped and then concealed using the baseline BMA and the proposed method. The zoomed parts of the difference images show higher inter-view consistency (smaller magnitude of the white color) of the proposed method compared to the baseline BMA method.

gains of the proposed method over the baseline technique. The gains are particularly visible in high texture regions such as the face in the Breakdancer sequence and the railing in the Poznan_Hall2 sequence.

5.5 Conclusion

In this chapter, a consistent error concealment method was proposed to recover lost frames when MVD 3D video is broadcast over an error-prone delivery channel. The proposed method uses a cost model that combines temporal and view consistency criteria to reconstruct lost blocks from a set of candidate blocks. Simulation results show that the proposed method does not only outperform a baseline method based on conventional error concealment technique and two standard methods in reconstruction fidelity but also gives more consistent frames. In conclusion, the proposed consistent error concealment method is capable of significantly improving the quality of MVD based 3D video that has been corrupted by transmission errors.

The proposed cost model is generic and flexible in the choice of the underlying error concealment methods that are used to generate candidate blocks. The choice of the methods to create candidate blocks for reconstruction in Section 5.3.3 is motivated by the idea that MBs reconstructed using view-synthesis are expected to have better inter-view consistency while those obtained using motion compensation are expected to have better temporal consistency. So a selective combination of these methods based on an overall inconsistency evaluation criteria would result in frames consistent in both the inter-view and temporal directions. Another motivation is to make available a diverse set of candidate blocks such that the concealment process is not dependent on the availability of a particular frame. Moreover, the current value of α in Eq. (5.5) assigns equal weight to temporal and inter-view inconsistencies. Adapting the value of α according to the scene or requirements might be even more useful. Finally, the application of the proposed method is not limited to the whole-frame loss case. In case, a frame is encapsulated into multiple packets, some of which are lost during transmission, it is possible that a decoder will not drop all the packets. In this case, applying the proposed consistency model in combination with conventional methods (such

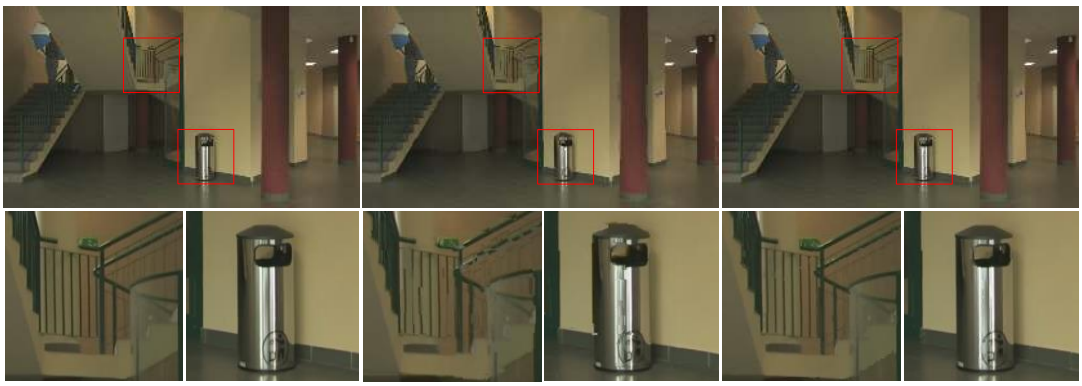
Table 5.1: Average PSNR (dB) Over concealed frames for different Packet Loss Rates (PLRs). The following notations are used in the table: P = Proposed, B = Baseline BMA.

Sequence	PLR	No Error	DMS [98]	IMVS [39]	B	P	P - DMS	P - IMVS	P - B
Ballet	5%	37.05	25.80	25.87	26.50	27.08	1.28	1.21	0.58
	10%	36.97	24.73	25.15	25.34	26.67	1.94	1.52	1.33
	20%	37.02	23.97	24.75	24.73	26.36	2.39	1.61	1.63
Breakdancer	5%	35.90	25.44	27.46	26.79	27.47	2.03	0.01	0.68
	10%	35.80	24.71	26.76	26.39	26.77	2.06	0.01	0.38
	20%	35.79	24.23	26.31	26.04	26.34	2.21	0.03	0.40
Poznan_Street	5%	41.24	29.30	28.90	29.67	31.86	3.06	3.46	2.19
	10%	41.09	27.87	27.59	28.62	31.19	3.32	3.60	2.57
	20%	41.05	26.86	26.59	27.58	30.64	3.78	3.95	3.06
Poznan_Hall2	5%	44.26	33.48	34.72	34.82	36.45	2.97	1.73	1.63
	10%	44.07	32.37	33.84	33.94	35.76	3.39	1.92	1.82
	20%	44.04	31.50	33.10	33.29	35.40	3.90	2.30	2.11

Table 5.2: Average PSNR (dB) Over all frames for different Packet Loss Rates (PLRs). The following notations are used in the table: P = Proposed, B = Baseline BMA.

Sequence	PLR	No Error	DMS [98]	IMVS [39]	B	P	P - DMS	P - IMVS	P - B
Ballet	5%	36.98	33.03	33.04	33.11	33.28	0.25	0.24	0.16
	10%	36.98	31.93	32.03	32.00	32.39	0.46	0.36	0.39
	20%	36.98	29.65	30.00	30.12	30.71	1.06	0.71	0.59
Breakdancer	5%	35.82	28.41	28.61	28.56	28.63	0.21	0.02	0.08
	10%	35.82	27.47	28.21	28.27	28.36	0.89	0.15	0.09
	20%	35.82	26.93	27.92	27.76	27.93	1.00	0.01	0.17
Poznan_Street	5%	41.17	36.62	36.59	37.16	37.41	0.79	0.82	0.35
	10%	41.17	34.90	34.84	35.95	36.64	1.74	1.80	0.69
	20%	41.17	31.42	31.35	33.98	35.10	3.68	3.75	1.12
Poznan_Hall2	5%	44.10	39.70	39.90	40.29	40.49	0.79	0.59	0.19
	10%	44.10	38.17	38.61	39.43	39.79	1.62	1.18	0.36
	20%	44.10	35.00	35.93	37.35	38.31	3.31	2.38	0.96

(a) 7th frame of Breakdancer sequence



(b) 4th frame of Poznan_Hall2 sequence

Figure 5.10: Comparison of visual results for a frame in view 1 with no error (left), reconstructed using the baseline method, BMA (centre), and reconstructed using the proposed method (right). For each sequence, the top row contains the full frame while the bottom row contains zoomed part of the frames. For these results, a frame in view 1 was dropped and then concealed using the baseline BMA and the proposed method. The difference in visual quality for the baseline BMA and the proposed methods can be seen in the zoomed parts.

as [38] and [101]) for recovering block losses may prove to be very effective.

Chapter 6

Conclusions and Future Work

This chapter provides a summary of the thesis and discussed its limitations and proposes ideas for future extension of the work done in this thesis. The summary is provided in Section 6.1 while the limitations and suggestions for future extension are discussed in Section 6.2.

6.1 Thesis Summary

This thesis addressed two problems:

- The MVC encoder uses a complex variable block size motion and disparity estimation process to efficiently compress multiview videos. This makes the MVC encoder very slow and makes it undesirable for use in power constrained devices.
- Compressed 3D videos can suffer from packet loss during transmission, which can degrade the viewing quality of the 3D video at the decoder.

The thesis provided solutions for these problems by proposing:

- fast encoding techniques for multiview videos based on MVC,
- a fast encoding technique for MVD videos based on a view synthesis prediction-enhanced MVC coder, and

- a consistent error concealment technique for multiview plus depth video broadcasting.

Chapter 3 describes the first three contributions of the thesis. These include a previous disparity vector based fast disparity estimation method (PDV-DE), a stereo-motion consistency constraint based fast motion and disparity estimation method (SMCC-MDE) and a complete low-complexity encoding solution for MVC (CLCMVC).

PDV-DE was motivated by an analysis of the distribution of optimal disparity vectors at different temporal levels of a GOP for different types of macroblocks under two different conditions: (i) when the search centre is determined using median prediction, and (ii) when the search centre is determined by the previous disparity vector. It was found that at higher temporal levels and for 'Simple' macroblocks, using previous disparity vector as the search centre instead of the conventional median prediction allows to reduce the search range considerably without a compromise on the RD performance. Hence, an adaptive search range strategy was proposed which reduced the encoding time of the JMVM 6.0 encoder by around 40% with negligible difference in the RD performance. The results of this contribution were published in [30] and [31].

SMCC-MDE exploited the stereo motion consistency constraint which is a geometrical constraint between the motion and disparity vectors of stereo videos. Using this constraint, it first estimates the final motion and disparity vectors and then uses these estimates to find the final motion and disparity vectors in a limited search range. The proposed technique reduced the encoding time of the JMVM 6.0 encoder by around 50% without compromising much on the RD performance. The results of this contribution were published in [31].

CLCMVC is a complete low-complexity encoding framework for multiview video coding. It combines PDV-DE and SMCC-MDE with other state-of-the-art methods in a unique framework that allows the gains of these methods to add up. It has been shown that CLCMVC can reduce the encoding time of the JMVM 6.0 encoder by over 93% without compromising significantly on its RD performance. Compared to the state-of-the-art [34], the reduction in encoding time is over 11%. The results of this contribution were published in [31].

Chapter 4 describes the fourth contribution of the thesis. In this chapter, an enhanced JMVC encoder was considered which included an additional View-synthesis prediction (VSP) based SKIP mode. VSP SKIP mode has been shown to improve the compression efficiency of MVC and is a candidate tool for modern 3D video codecs. The idea of this contribution was to speed up the encoding process of VSP SKIP mode enhanced encoders by predicting the optimal modes early and thus testing a limited number of candidate prediction modes during the motion/disparity estimation process. The motivation behind the proposed method was an analysis of the relationship between the VSP SKIP mode RD cost and the optimal prediction mode. It was found that this relationship is very similar across views. Hence bayesian decision rule was used to first observe the VSP SKIP mode RD cost while encoding a macroblock and then based on this observation, test only the most probable modes in the bayesian sense. It was found that using the proposed technique, over 33% of the encoding time can be reduced without a noticeable degradation in the RD performance. The results of this contribution were published in [35]. VSP modes are expected to be included in upcoming video coding standards such as 3D-AVC [49]. The proposed method can be useful in reducing the encoding time of such future video codecs.

Chapter 5 describes the last contribution of the thesis. The motivation for this contribution was the lack of emphasis of the existing frame-loss error concealment methods on the important aspect of consistency in 3D videos during the error concealment process. Unlike 2D videos, in 3D videos, frames are not viewed independently. That is why while in 2D videos, independently recovering frames might be sufficient, in 3D videos, this can lead to inconsistent reconstruction of the 3D scene and hence in poor viewing experience. So a scene consistent frame-loss error concealment method is proposed which takes into consideration temporal and inter-view consistency while recovering lost frames. Experimental results have shown that compared to the baseline BMA method and two standard error concealment methods, the proposed method can recover lost frames with high consistency while maintaining a reasonable PSNR performance. The results suggest that compared to the standard methods, the proposed method might be more helpful in reducing the flickering artefacts in 3D video which are caused by inconsistencies between frames.

6.2 Limitations and Future Work

This section presents limitations of the work and suggests some future directions in which this work can be extended:

- In Chapter 3, experiments were performed using four standard multiview test sequences recommended by the Joint Video Team [72]. The test dataset is representative of different video features such as outdoor/indoor, fast/slow motion, human/vehicles, rectangular/circular structures, translational/rotational motion. However, all the four sequences were generated using a 1D camera array. The performance is not guaranteed for sequences generated using a 2D camera array and will depend on the interview correlation between views. Moreoever, the tests were performed using five different QP values ranging from 20 to 36. A wide range of bitrates are covered by this range that is why QP values outside of this range were not tested. However, as the performance of the proposed algorithm did not vary dramatically for different QP values, it is expected that the results will not be significantly different if experiments are performed using other QP values.

- In Chapter 3, the compression performance of the proposed fast encoding techniques was evaluated by comparing its Rate-Distortion plots with those of the standard JMVM 6.0 implementation. The maximum PSNR loss compared to the standard JMVM 6.0 was of 0.07 dB. Visual evaluation of the decoded frames did not show a significant difference between the different methods. That is why they were not included in the results.

- In the Bayesian early mode decision method in Chapter 4, the value of epsilon was set experimentally. It was noticed that this value is not significantly effected by the type of motion in a video. Further experiments can be performed to see how this value changes for different baseline distances, multi-camera setups, natural/indoor sequences. If a relationship is found, the algorithm can be modified by introducing an epsilon value that adapts to the type of content.

- The Bayesian early mode decision method in Chapter 4 was applied to only one view i.e. $V1$ (the middle view) because the middle view requires most

of the computational effort since it is the only view in the Hierarchical B pictures prediction structure in MVC where inter-view prediction is enabled. The algorithm can be extended and applied to view $V2$ by using the RD Cost of Conventional SKIP mode in $V2$ as the observation variable (x)

- The Bayesian early mode decision method in Chapter 4 can be combined with other fast encoding methods for MVC such as the proposed CLCMVC method proposed in Chapter 3. It has been shown in [110] that the time saving can be increased by another 30% without a significant compromise in the RD performance when combined with the selective disparity estimation method in [64].

- The consistency model in Chapter 5 is generic and can also be applied in combination with other methods apart from the ones in Section 5.3.3 used to obtain the candidate reconstructed MBs. The choice of methods in Section 5.3.3 was motivated by the idea that MBs reconstructed using view-synthesis have better inter-view consistency while those obtained using motion-compensation have better temporal consistency. So a selective combination of these methods based on an overall inconsistency evaluation criteria would result in frames consistent in both the inter-view and temporal directions. A further study on the application of the proposed consistency model on other error concealment methods (for example, as discussed in Section 5.2) might help improve the consistency performance. Moreover, the value of α in Eq. 5.5 gives equal weights to temporal and inter-view inconsistencies. Adapting α according to the scene or requirements may lead to better results. The results may also be further validated by using different loss models such as the Gilbert-Elliot model in [111].

- The HEVC encoder is very slow. Encoding a 10 second High Definition (HD) resolution video takes around 20 hrs. Even for the fastest 'intra-only' configuration, the encoding time may exceed 1000 times real time [112]. Its multiview extension [113] is expected to include several new tools to efficiently compress multiview videos. For example two new tools, inter-view motion vector prediction and inter-view residual prediction have been

included in the current phase of the collaborative work on the standardization of the multiview extension of HEVC. This will further slow down the encoder. Considering that the fundamental compression techniques such as motion/disparity estimation, transform coding have not be entirely changed but enhanced in HEVC, the fast encoding techniques introduced in Chapter 3 and 4 can be modified for use in HEVC encoder as well.

The fast encoding techniques proposed in Chapters 3 and 4 can be used as a starting point for any fast encoding algorithm for the multiview extension of HEVC.

- The thesis proposed software based solutions for low-complexity encoding of multiview videos. An important aspect of video encoders is the hardware platform on which they are deployed. Different hardware platforms have different memory, power and processing resources available which can have a profound effect on the performance of the MVC encoder/decoder. A review of state-of-the-art in hardware architectures for MVC encoder/decoder e.g. ARM and Intel based processors is required to further investigate the possibility of speeding up the encoding process by efficient use of hardware resources. For example harnessing the parallel processing capabilities of modern on-chip GPUs for video encoding/decoding is an active area of research. The idea is that expensive encoder functions such as motion estimation and disparity estimation can be offloaded to a multi-core GPU where they are performed in parallel.

  A study of the hardware architectures will also provide an opportunity to optimize the decoder for power and memory utilization. This will be particularly useful for low-power, low-memory devices such as mobile phones and tablet devices.

- VGA resolution (640x480), XGA resolution (1024x768), and HD resolution (1920x1080) videos were considered in this thesis. Beyond-HD resolutions (such as 4K UHD) have gained a rapid popularity and are expected to become widely available soon. The inclusion of Ultra High Definition (UHD) content in the test dataset would have further validated the results. They

were not included in the test dataset because this resolution was beyond the scope of the MVC common test conditions set by the standard and were not publicly available until recently.

- The experimental results only contain objective test results. No subjective tests were performed. For the methods in Chapter 3 and Chapter 4, subjective evaluation is not expected to lead conclusions significantly different than those obtained using PSNR as the decoded frames for different methods are hardly different. For the consistent error concealment method in Chapter 5, the results can be extended by including a subjective evaluation criteria based on Mean Opinion Scores (MOSs).

References

[1] "Cisco visual networking index: Forecast and methodology, 2012-2017," Cisco, May 2013.

[2] *3D display technology and market forecast report*, DisplaySearch.

[3] H. Urey, K. V. Chellappan, E. Erden, and P. Surman, "State of the art in stereoscopic and autostereoscopic displays," *Proceedings of the IEEE*, vol. 99, no. 4, pp. 540–555, 2011.

[4] M. Lambooij, M. Fortuin, I. Heynderickx, and W. IJsselsteijn, "Visual discomfort and visual fatigue of stereoscopic displays: a review," *Journal of Imaging Science and Technology*, vol. 53, no. 3, pp. 1–14, 2009.

[5] N. A. Dodgson, "Autostereoscopic 3d displays," *Computer*, vol. 38, no. 8, pp. 31–36, 2005.

[6] "Dimenco display." [Online]. Available: http://www.dimenco.eu/displays/

[7] A. Vetro, S. Yea, and A. Smolic, "Towards a 3D video format for autostereoscopic displays," in *Proc. SPIE Conference on Applications of Digital Image Processing XXXI*, vol. 7073, San Diego, CA, Sept. 2008.

[8] A. Smolic, K. Mueller, P. Merkle, C. Fehn, P. Kauff, P. Eisert, and T. Wiegand, "3D video and free viewpoint video - technologies, applications and MPEG standards," in *Proc. IEEE ICME*, Atlanta, GA, Oct. 2006.

[9] P. Merkle, A. Smolic, K. Muller, and T. Wiegand, "Multi-view video plus depth representation and coding," in *Proc. IEEE ICIP*, San Antonio, TX, USA, Sept. 2007.

[10] C. Fehn, "Depth-image-based rendering (DIBR), compression, and transmission for a new approach on 3d-tv," in *Proc. SPIE Conference on Stereoscopic Displays and Virtual Reality Systems XI*, CA, USA, Jan. 2004, pp. 93–104.

[11] D. Tian, P.-L. Lai, P. Lopez, and C. Gomila, "View synthesis techniques for 3D video," in *Proc. SPIE Conference on Applications of Digital Image Processing XXXII*, vol. 7443, 2009.

[12] P. Frojdh, A. Norkin, and R. Sjoberg, "Next generation video compression," Ericsson, Tech. Rep., Apr. 2013.

[13] M. Liou, "Overview of the p× 64 kbit/s video coding standard," *Communications of the ACM*, vol. 34, no. 4, pp. 59–63, 1991.

[14] K. Rijkse, "H. 263: Video coding for low-bit-rate communication," *IEEE Communications Magazine*, vol. 34, no. 12, pp. 42–45, 1996.

[15] G. Cote, B. Erol, M. Gallant, and F. Kossentini, "H.263+: video coding at low bit rates," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 8, no. 7, pp. 849–866, 1998.

[16] "H.263," ITU-T Recommendation, 2005.

[17] K. Brandenburg and G. Stoll, "ISO/MPEG-1 audio: A generic standard for coding of high-quality digital audio," *Journal of the Audio Engineering Society*, vol. 42, no. 10, pp. 780–792, 1994.

[18] I. E. Richardson, *H. 264 and MPEG-4 video compression: video coding for next-generation multimedia*. John Wiley & Sons, 2004.

[19] "Generic coding of moving pictures and associated audio information - part 2: Video," ITU-T and ISO/IEC JTC 1, 1994.

[20] T. Wiegand, G. J. Sullivan, G. Bjntegaard, and A. Luthra, "Overview of the H.264/AVC video coding standard," *IEEE Transactions on Circuits and Systems for Video Technology*, no. 13, pp. 560–576, July 2003.

[21] "Advanced video coding for generic audiovisial services," ITU-T Rec. H.264 and ISO/IEC 14496-10 (AVC), 2012.

[22] H. Schwarz, D. Marpe, and T. Wiegand, "Overview of the scalable video coding extension of the H.264/AVC standard," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 17, no. 9, pp. 1103–1120, 2007.

[23] Y.-S. Ho and K.-J. Oh, "Overview of multi-view video coding," in *Proc. 14th International Workshop on Systems, Signals and Image Processing (IWSSIP 2007)*, 2007, pp. 5–12.

[24] A. Vetro, T. Wiegand, and G. Sullivan, "Overview of the Stereo and Multiview Video Coding extensions of the H.264/MPEG-4 AVC standard," *Proceedings of the IEEE*, vol. 99, no. 4, pp. 626–642, 2011.

[25] G. J. Sullivan, J. Ohm, W.-J. Han, and T. Wiegand, "Overview of the high efficiency video coding (HEVC) standard," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 22, no. 12, pp. 1649–1668, 2012.

[26] A. Vetro, A. Tourapis, K. Muller, and T. Chen, "3d-tv content storage and transmission," *IEEE Transactions on Broadcasting*, vol. 57, no. 2, pp. 384–394, 2011.

[27] B. Zatt, M. Shafique, S. Bampi, and J. Henkel, "A multi-level dynamic complexity reduction scheme for multiview video coding," in *Proc. IEEE ICIP*, Brussels, Belgium, Sept. 2011, pp. 749–752.

[28] P. Merkle, A. Smolic, K. Muller, and T. Wiegand, "Efficient prediction structures for multiview video coding," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 17, no. 11, pp. 1461–1473, 2007.

[29] C. Lee and Y.-S. Ho, "A framework of 3d video coding using view synthesis prediction," in *Proc. Picture Coding Symposium (PCS 2012)*, Krakow, Poland, May 2012.

[30] S. Khattak, R. Hamzaoui, S. Ahmad, and P. Frossard, "Low-complexity multiview video coding," in *Proc. Picture Coding Symposium (PCS 2012)*, Krakow, Poland, May 2012, pp. 97–100.

[31] S. Khattak, R. Hamzaoui, S. Ahmad, and P. Frossard, "Fast encoding techniques for multiview video coding," *Signal Processing: Image Communication*, vol. 28, no. 6, pp. 569–580, 2013.

[32] Y. Chen, P. Pandit, S. Yea, and C. Lim, "JMVM 6.0 software," Joint Video Team (JVT), Oct. 2007.

[33] S. Khattak, "Framework for low-complexity multiview video coding," in *Proc. 14th annual post graduate symposium on the convergence of Telecommunications, Networking, and Broadcasting (PGNet2013)*, Liverpool, UK, June 2013.

[34] L. Shen, Z. Liu, T. Yan, Z. Zhang, and P. An, "View-adaptive motion estimation and disparity estimation for low complexity multiview video coding," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 20, no. 6, pp. 925–930, 2010.

[35] S. Khattak, R. Hamzaoui, T. Maugey, S. Ahmad, and P. Frossard, "Bayesian early mode decision technique for view synthesis prediction-enhanced multiview video coding," *IEEE Signal Processing Letters*, vol. 20, no. 11, pp. 1126–1129, Nov. 2013.

[36] Y. Chen, P. Pandit, S. Yea, and C. Lim, "Draft reference software for MVC (JMVC 6.0)," Joint Video Team (JVT), July 2009.

[37] L. Shen, Z. Liu, T. Yan, Z. Zhang, and P. An, "Early skip mode decision for MVC using inter-view correlation," *Image Commun.*, vol. 25, no. 2, pp. 88–93, Feb. 2010.

[38] Y. Liu, J. Wang, and H. Zhang, "Depth image-based temporal error concealment for 3-d video transmission," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 20, no. 4, pp. 600–604, 2010.

[39] C. T. E. R. Hewage, S. Worrall, S. Dogan, and A. Kondoz, "Frame concealment algorithm for stereoscopic video using motion vector sharing," in *Proc. IEEE ICME*, Hannover, Germany, June 2008.

[40] W.-M. Lam, A. Reibman, and B. Liu, "Recovery of lost or erroneously received motion vectors," in *Proc. IEEE ICASSSP*, Minneapolis, MN, USA, Apr. 1993.

[41] "JMVC 8.5, garcon.ient.rwthaachen.de, 2011."

[42] "MPEG requirements sub-group, requirements on multi-view video coding," July 2009.

[43] A. Smolic, K. Mueller, P. Merkle, P. Kauff, and T. Wiegand, "An overview of available and emerging 3d video formats and depth enhanced stereo as efficient generic solution," in *Proc. Picture Coding Symposium (PCS 2009)*, Chicago, IL, USA, May 2009, pp. 1–4.

[44] H. Brust, A. Smolic, K. Mueller, G. Tech, and T. Wiegand, "Mixed resolution coding of stereoscopic video for mobile devices," in *3DTV Conference: The True Vision-Capture, Transmission and Display of 3D Video, 2009*, Potsdam, Germany, May 2009, pp. 1–4.

[45] C. Fehn, P. Kauff, M. O. De Beeck, F. Ernst, W. Ijsselsteijn, M. Pollefeys, L. Van Gool, E. Ofek, and I. Sexton, "An evolutionary and optimised approach on 3d-tv," in *Proc. IBC*, vol. 2, 2002, pp. 357–365.

[46] P. Merkle, Y. Morvan, A. Smolic, D. Farin, K. Mueller, P. de With, and T. Wiegand, "The effects of multiview depth video compression on multiview rendering," *Signal Processing: Image Communication*, vol. 24, no. 1, pp. 73–88, 2009.

[47] A. Bourge, J. Gobert, and F. Bruls, "Mpeg-c part 3: Enabling the introduction of video plus depth contents," in *Proc. IEEE Workshop on Content Generation and Coding for 3D-television*, Eindhoven, The Netherlands, June 2006.

[48] Y. Chen, M. M. Hannuksela, T. Suzuki, and S. Hattori, "Overview of the MVC+D 3D video coding standard," *Journal of Visual Communication and Image Representation*, vol. 25, no. 4, pp. 679 – 688, 2014.

[49] M. M. Hannuksela, D. Rusanovskyy, W. Su, L. Chen, R. Li, P. Aflaki, D. Lan, M. Joachimiak, H. Li, and M. Gabbouj, "Multiview-video-plus-depth coding based on the advanced video coding standard," *IEEE Transactions on Image Processing*, vol. 22, no. 9, pp. 3449–3458, 2013.

[50] S. Ivekovic, A. Fusiello, and E. Trucco, "Fundamentals of multiple-view geometry," *3D Videocommunication: Algorithms, Concepts and Real-Time Systems in Human Centred Communication*, pp. 91–113.

[51] C. L. Zitnick, S. B. Kang, M. Uyttendaele, S. Winder, and R. Szeliski, "High-quality video view interpolation using a layered representation," *ACM Trans. Graph.*, vol. 23, no. 3, pp. 600–608, 2004.

[52] D. Scharstein, *View synthesis using stereo vision*. Springer-Verlag, 1999.

[53] M. Bertalmio, L. Vese, G. Sapiro, and S. Osher, "Simultaneous structure and texture image inpainting," *IEEE Transactions on Image Processing*, vol. 12, no. 8, pp. 882–889, 2003.

[54] M. Tanimoto, T. Fujii, and K. Suzuki, "Reference softwares for depth estimation and view synthesis," 2008.

[55] M. Tanimoto, T. Fujii, and K. Suzuki, "View synthesis algorithm in view synthesis reference software 2.0 (VSRS2.0)," 2009.

[56] K.-J. Oh, S. Yea, and Y.-S. Ho, "Hole filling method using depth based in-painting for view synthesis in free viewpoint television and 3-d video," in *Proc. Picture Coding Symposium (PCS 2009)*, Chicago, IL, May 2009, pp. 1–4.

[57] J.-I. Jung and Y.-S. Ho, "Virtual view synthesis using temporal hole filling with bilateral coefficients," in *IEEE RIVF International Conference on Computing and Communication Technologies, Research, Innovation, and*

Vision for the Future (RIVF), Ho Chi Min City, Vietnam, Feb. 2012, pp. 1–4.

[58] Y. Zhang, S. Kwong, L. Xu, and G. Jiang, "Direct mode early decision optimization based on rate distortion cost property and inter-view correlation," *IEEE Transactions on Broadcasting*, vol. 59, no. 2, pp. 390–398, 2013.

[59] J. Reichel, H. Schwarz, and M. Wien, "Joint scalable video model 8," ISO/IEC JTC1/SC29/WG11 and ITU-T Q6/SG16, July 2006.

[60] L. Shen, T. Yan, Z. Liu, Z. Zhang, P. An, and L. Yang, "Fast mode decision for multiview video coding," in *Proc. IEEE ICIP*, Cairo, Egypt, Nov. 2009.

[61] H. Zeng, K.-K. Ma, and C. Cai, "Fast mode decision for multiview video coding using mode correlation," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 21, no. 11, pp. 1659–1666, 2011.

[62] C.-C. Chan and C.-W. Tang, "Coding statistics based fast mode decision for multi-view video coding," *Journal of Visual Communication and Image Representation*, vol. 24, no. 6, pp. 686 – 699, 2013.

[63] Y. Zhang, S. Kwong, G. Jiang, and H. Wang, "Efficient multi-reference frame selection algorithm for hierarchical b pictures in multiview video coding," *IEEE Transactions on Broadcasting*, vol. 57, no. 1, pp. 15–23, 2011.

[64] L. Shen, Z. Liu, S. Liu, Z. Zhang, and P. An, "Selective disparity estimation and variable size motion estimation based on motion homogeneity for multi-view coding," *IEEE Transactions on Broadcasting*, vol. 55, no. 4, pp. 761–766, 2009.

[65] W. Zhu, X. Tian, F. Zhou, and Y. Chen, "Fast disparity estimation using spatio-temporal correlation of disparity field for multiview video coding," *IEEE Transactions on Consumer Electronics*, vol. 56, no. 2, pp. 957–964, 2010.

[66] Z.-P. Deng, Y.-L. Chan, K.-B. Jia, C.-H. Fu, and W.-C. Siu, "Fast motion and disparity estimation with adaptive search range adjustment in stereoscopic video coding," *IEEE Transactions on Broadcasting*, vol. 58, no. 1, pp. 24–33, 2012.

[67] Z.-P. Deng, Y.-L. Chan, K.-B. Jia, C.-H. Fu, and W.-C. Siu, "Iterative search strategy with selective bi-directional prediction for low complexity multiview video coding," *Journal of Visual Communication and Image Representation*, vol. 23, no. 3, pp. 522 – 534, 2012.

[68] Y. Zhang, S. Kwong, G. Jiang, X. Wang, and M. Yu, "Statistical early termination model for fast mode decision and reference frame selection in multiview video coding," *IEEE Transactions on Broadcasting*, vol. 58, no. 1, pp. 10–23, 2012.

[69] M. Shafique, B. Zatt, and J. Henkel, "A complexity reduction scheme with adaptive search direction and mode elimination for multiview video coding," in *Proc. Picture Coding Symposium (PCS 2012)*, Krakow, Poland, May 2012.

[70] I. Patras, N. Alvertos, and G. Tziritas, "Joint disparity and motion field estimation in stereoscopic image sequences," in *Proc. 13th International Conference on Pattern Recognition*, Vienna, Austria, Aug., 1996.

[71] H. S. Koo, Y. J. Jeon, and B. M. Jeon, "MVC motion skip mode," ISO/IEC JTC1/SC29/WG11 and ITU-T Q6/SG16, Apr. 2007.

[72] "Common test conditions for multiview video coding," ISO/IEC JTC1/SC29/WG11 and ITU-T Q6/SG16, July 2006.

[73] W. Zhu, Y. Zheng, P. Chen, and J. Feng, "Fast macroblock encoding algorithm based on rate-distortion activity for multiview video coding," *Signal Processing: Image Communication*, 2014.

[74] "Overview of 3d video coding," ISO/IEC JTC1/SC29/WG11, May 2008.

[75] P. Aflaki, M. Hannuksela, D. Rusanovskyy, and M. Gabbouj, "Nonlinear depth map resampling for depth-enhanced 3-d video coding," *IEEE Signal Processing Letters*, vol. 20, no. 1, pp. 87–90, 2013.

[76] S. Yea and A. Vetro, "View synthesis prediction for multiview video coding," *Signal Processing: Image Communication*, vol. 24, no. 12, pp. 89 – 100, 2009.

[77] Z. Ni, D. Tian, S. Bhagavathy, J. Llach, and B. Manjunath, "Improving the quality of depth image based rendering for 3d video systems," in *Proc. IEEE ICIP*, Cairo, Egypt, Nov. 2009.

[78] M. Solh and G. AlRegib, "Hierarchical hole-filling for depth-based view synthesis in ftv and 3d video," *IEEE Journal of Selected Topics in Signal Processing*, vol. 6, no. 5, pp. 495–504, 2012.

[79] I. Richardson, M. Bystrom, and Y. Zhao, "Fast H.264 skip mode selection using an estimation framework," in *Proc. Picture Coding Symposium (PCS 2006)*, Beijing, China, Apr. 2006.

[80] C.-H. Yeh, K.-J. Fan, M.-J. Chen, and G.-L. Li, "Fast mode decision algorithm for scalable video coding using bayesian theorem detection and markov process," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 20, no. 4, pp. 563–574, 2010.

[81] A. Saha, K. Mallick, J. Mukherjee, and S. Sural, "SKIP prediction for fast rate distortion optimization in H.264," *IEEE Transactions on Consumer Electronics*, vol. 53, no. 3, pp. 1153–1160, 2007.

[82] E. Martinian, A. Behrens, J. Xin, and A. Vetro, "View synthesis for multiview video compression," in *Proc. Picture Coding Symposium (PCS 2006)*, Beijing, China, Apr. 2006.

[83] M. Domanski, T. Grajek, K. Klimaszewski, M. Kurc, O. Stankiewicz, J. Stankowski, and K. Wegner, "Poznan multiview test sequences and camera parameters," ISO/IEC JTC1/SC29/WG11, Oct. 2009.

[84] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern classification.* John Wiley & Sons, 2012.

[85] I. Androutsopoulos, J. Koutsias, K. V. Chandrinos, G. Paliouras, and C. D. Spyropoulos, "An evaluation of naive bayesian anti-spam filtering," in *Proc. Workshop on Machine Learning in the New Information Age, 11th European Conference on Machine Learning (ECML 2000)*, Barcelona, Spain, May 2000.

[86] G. Schwarz, "Estimating the dimension of a model," *The Annals of Statistics*, vol. 6, no. 2, pp. 461–464, 1978.

[87] B. Micallef, C. Debono, and R. Farrugia, "Fast inter-mode decision in multi-view video plus depth coding," in *Proc. Picture Coding Symposium (PCS 2012)*, Krakow, Poland, May 2012.

[88] F. Shao, G. Jiang, M. Yu, K. Chen, and Y.-S. Ho, "Asymmetric coding of multi-view video plus depth based 3-d video for view rendering," *IEEE Transactions on Multimedia*, vol. 14, no. 1, pp. 157–167, 2012.

[89] J. Y. Lee, H.-C. Wey, and D.-S. Park, "A fast and efficient multi-view depth image coding method based on temporal and inter-view correlations of texture images," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 21, no. 12, pp. 1859–1868, 2011.

[90] C. Hellge, T. Schierl, and T. Wiegand, "Mobile tv using scalable video coding and layer-aware forward error correction," in *Proc. IEEE ICME*, Hannover, Germany, June 2008, pp. 1177–1180.

[91] W. Fischer, *Digital video and audio broadcasting technology: a practical engineering guide.* Springer, 2010.

[92] K. Singh and G. Rubino, "Quality of experience estimation using frame loss pattern and video encoding characteristics in DVB-H networks," in *Proc. International Packet Video Workshop (PV2010)*, Hong Kong, China, Dec. 2010, pp. 150–157.

[93] K. Song, T. Chung, Y. Oh, and C.-S. Kim, "Error concealment of multi-view video sequences using inter-view and intra-view correlations," *Journal of Visual Communication and Image Representation*, vol. 20, no. 4, pp. 281 – 292, 2009.

[94] B. Yan and J. Zhou, "Efficient frame concealment for depth image-based 3-d video transmission," *IEEE Transactions on Multimedia*, vol. 14, no. 3, pp. 936–941, 2012.

[95] B. Yan and H. Gharavi, "A hybrid frame concealment algorithm for H.264/AVC," *IEEE Transactions on Image Processing*, vol. 19, no. 1, pp. 98–107, 2010.

[96] X. Ji, D. Zhao, and W. Gao, "Concealment of whole-picture loss in hierarchical b-picture scalable video coding," *IEEE Transactions on Multimedia*, vol. 11, no. 1, pp. 11–22, 2009.

[97] Y. Guo, Y. Chen, Y.-K. Wang, H. Li, M. Hannuksela, and M. Gabbouj, "Error resilient coding and error concealment in scalable video coding," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 19, no. 6, pp. 781–795, 2009.

[98] S. Liu, Y. Chen, Y.-K. Wang, M. Gabbouj, M. Hannuksela, and H. Li, "Frame loss error concealment for multiview video coding," in *Proc. ISCAS*, Seattle, WA, USA May 2008.

[99] T.-Y. Chung, S. Sull, and C.-S. Kim, "Frame loss concealment for stereoscopic video plus depth sequences," *IEEE Transactions on Consumer Electronics*, vol. 57, no. 3, pp. 1336–1344, 2011.

[100] Y. Chen, K. Yu, J. Li, and S. Li, "An error concealment algorithm for entire frame loss in video transmission," in *Proc. Picture Coding Symposium (PCS 2004)*, San Francisco, CA, USA, Dec. 2004, pp. 15–17.

[101] W.-Y. Kung, C.-S. Kim, and C.-C. Kuo, "Spatial and temporal error concealment techniques for video transmission over noisy channels," *IEEE*

Transactions on Circuits and Systems for Video Technology, vol. 16, no. 7, pp. 789–803, 2006.

[102] T. Basha, Y. Moses, and S. Avidan, "Geometrically consistent stereo seam carving," in *Proc. IEEE ICCV*, Barcelona, Spain, Nov. 2011.

[103] G. Floros and B. Leibe, "Joint 2d-3d temporally consistent semantic segmentation of street scenes," in *Proc. IEEE CVPR*, Providence, RI, USA, June 2012.

[104] A. Tankus and Y. Yeshurun, "Scene-consistent detection of feature points in video sequences," in *Proc. IEEE CVPR*, Kauai, HI, USA, Dec. 2001.

[105] T. Maugey, P. Frossard, and G. Cheung, "Consistent view synthesis in interactive multiview imaging," in *Proc. IEEE ICIP*, Orlando, FL, USA, Sept. 2012, pp. 2717–2720.

[106] I. Ahn and C. Kim, "Depth-based disocclusion filling for virtual view synthesis," in *Proc. IEEE ICME*, Melbourne, Australia, July 2012.

[107] R. Tsai, "A versatile camera calibration technique for high accuracy 3d machine vision metrology using off-the-shelf tv cameras and lenses," *IEEE Journal of Robotics and Automation*, vol. 3, no. 4, pp. 323–344, 1987.

[108] R. Di Bari, M. Bard, A. Arrinda, P. Ditto, G. Araniti, J. Cosmas, K. K. Loo, and R. Nilavalan, "Measurement campaign on transmit delay diversity for mobile dvb-t/h systems," *IEEE Transactions on Broadcasting*, vol. 56, no. 3, pp. 369–378, 2010.

[109] A. Smolic, K. Mueller, N. Stefanoski, J. Ostermann, A. Gotchev, G. Akar, G. Triantafyllidis, and A. Koz, "Coding algorithms for 3dtv a survey," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 17, no. 11, pp. 1606–1621, 2007.

[110] A. Gul, "Bayesian early mode decision technique for multiview video coding," Master's thesis, De Montfort University, 2013.

[111] J.-P. Ebert and A. Willig, "A gilbert-elliot bit error model and the efficient use in packet level simulation," Technical University of Berlin, Tech. Rep., Mar. 1999.

[112] F. Bossen, B. Bross, K. Suhring, and D. Flynn, "HEVC complexity and implementation analysis," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 22, no. 12, pp. 1685–1696, Dec. 2012.

[113] K. Muller, H. Schwarz, D. Marpe, C. Bartnik, S. Bosse, H. Brust, T. Hinz, H. Lakshman, P. Merkle, F. H. Rhee *et al.*, "3d high-efficiency video coding for multi-view video and depth data," *IEEE Transactions on Image Processing*, vol. 22, no. 9, pp. 3366–3378, 2013.