



School of Mechanical and Electrical Engineering  
Faculty of Health, Engineering and Sciences

# **ERROR-RESILIENT MULTI-VIEW VIDEO PLUS DEPTH BASED 3-D VIDEO CODING**

A thesis submitted by

**Pan Gao**, B Eng

for the award of

**Doctor of Philosophy**

2016



# Abstract

Three Dimensional (3-D) video, by definition, is a collection of signals that can provide depth perception of a 3-D scene. With the development of 3-D display technologies and interactive multimedia systems, 3-D video has attracted significant interest from both industries and academia with a variety of applications. In order to provide desired services in various 3-D video applications, the multi-view video plus depth (MVD) representation, which can facilitate the generation of virtual views, has been determined to be the best format for 3-D video data.

Similar to 2-D video, compressed 3-D video is highly sensitive to transmission errors due to errors propagated from the current frame to the future predicted frames. Moreover, since the virtual views required for auto-stereoscopic displays are rendered from the compressed texture videos and depth maps, transmission errors of the distorted texture videos and depth maps can be further propagated to the virtual views. Besides, the distortions in texture and depth show different effects on the rendering views. Therefore, compared to the reliability of the transmission of the 2-D video, error-resilient texture video and depth map coding are facing major new challenges.

This research concentrates on improving the error resilience performance of MVD-based 3-D video in packet loss scenarios. Based on the analysis of the propagating behaviour of transmission errors, a Wyner-Ziv (WZ)-based error-resilient algorithm is first designed for coding of the multi-view video data or depth data. In this scheme, an auxiliary redundant stream encoded according to WZ principle is employed to protect a primary stream encoded with standard multi-view video coding codec. Then, considering the fact that different combinations of texture and depth coding mode will exhibit varying robustness to transmission errors, a rate-distortion optimized mode switching scheme is proposed to strike

the optimal trade-off between robustness and compression efficiency. In this approach, the texture and depth modes are jointly optimized by minimizing the overall distortion of both the coded and synthesized views subject to a given bit rate. Finally, this study extends the research on the reliable transmission of view synthesis prediction (VSP)-based 3-D video. In order to mitigate the prediction position error caused by packet losses in the depth map, a novel disparity vector correction algorithm is developed, where the corrected disparity vector is calculated from the depth error. To facilitate decoder error concealment, the depth error is recursively estimated at the decoder.

The contributions of this dissertation are multifold. First, the proposed WZ-based error-resilient algorithm can accurately characterize the effect of transmission error on multi-view distortion at the transform domain in consideration of both temporal and inter-view error propagation, and based on the estimated distortion, this algorithm can perform optimal WZ bit allocation at the encoder through explicitly developing a sophisticated rate allocation strategy. This proposed algorithm is able to provide a finer granularity in performing rate adaptivity and unequal error protection for multi-view data, not only at the frame level, but also at the bit-plane level. Secondly, in the proposed mode switching scheme, a new analytic model is formulated to optimally estimate the view synthesis distortion due to packet losses, in which the compound impact of the transmission distortions of both the texture video and the depth map on the quality of the synthesized view is mathematically analysed. The accuracy of this view synthesis distortion model is demonstrated via simulation results and, further, the estimated distortion is integrated into a rate-distortion framework for optimal mode switching to achieve substantial performance gains over state-of-the-art algorithms. Last, but not least, this dissertation provides a preliminary investigation of VSP-based 3-D video over unreliable channel. In the proposed disparity vector correction algorithm, the pixel-level depth map error can be precisely estimated at the decoder without the deterministic knowledge of the error-free reconstructed depth. The approximation of the innovation term involved in depth error estimation is proved theoretically. This algorithm is very useful to conceal the position-erroneous pixels whose disparity vectors are correctly received.



# Certification of Dissertation

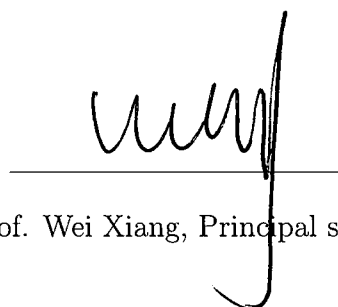
I certify that the ideas, designs and experimental work, results, analyses and conclusions set out in this dissertation are entirely my own effort, except where otherwise indicated and acknowledged.

I further certify that the work is original and has not been previously submitted for assessment in any other course or institution, except where specifically stated.



Pan Gao, Candidate

20/09/2016



Prof. Wei Xiang, Principal supervisor

20/09/2016



Prof. John Billingsley, Associate supervisor

/ /

# Acknowledgments

First, I would like to express my heartfelt gratitude to Prof. Wei Xiang, my principal supervisor for his endless commitment to directing the research and invaluable guidance. Without his continuous support and encouragement, this thesis would not have been completed.

I am also thankful to Prof. John Billingsley, my associate supervisor, for his instructive discussions and constructive suggestions to my thesis. My sincere thanks also go to Prof. Qiang Peng from Southwest Jiaotong University in China, my first supervisor, for introducing me to the challenging research area of video coding and transmission, and offering me the great opportunity to study abroad.

I would like to thank the Chinese Scholarship Council for providing financial support of my study at USQ during these three years. In addition, a special thanks goes to all my friends and research colleagues at USQ, for the stimulating discussions and the countless get-togethers throughout the years. I also thank all the administration staff at the School of Mechanical and Electrical Engineering. Their help made my life in Australia much easier. My thanks also go to Mark Butlin, who helped me to proofread the thesis.

Last, but not least, I would like to convey special thanks to my parents, without their support I would not have been able to reach this far in my studies. I am further greatly indebted to my wife Lijuan Zhang and my little precious son Yihan (George) Gao, who had to bear with me for the many times I had to work late into the night. Thanks for your patience and for your infinite love.

PAN GAO

*University of Southern Queensland, Australia*

*Mar. 2016*

# List of publications

The following publications were produced during the period of candidature:

## Journal Papers

[1] **P. Gao** and W. Xiang, “Disparity vector correction for view synthesis prediction-based 3-D video transmission,” *IEEE Trans. Multimedia*, vol. 17, no. 8, pp. 1153–1165, Aug. 2015. (USQ Publication Excellence Award 2015)

The work in the paper is presented in Chapter 5.

[2] **P. Gao** and W. Xiang, “Rate-distortion optimized mode switching for error-resilient multi-view video plus depth based 3-D video coding,” *IEEE Trans. Multimedia*, vol. 16, no. 7, pp. 1797–1808, Jun. 2014.

The work in the paper is presented in Chapter 4.

[3] **P. Gao**, Q. Peng, and W. Xiang, “Error-resilient multi-view video coding using Wyner-Ziv techniques,” *Multimedia Tools and Applications*, vol. 74, no. 17, pp. 7957–7982, Sept. 2015.

The work in the paper is presented in Chapter 3.

[4] W. Xiang, **P. Gao**, and Q. Peng, “Robust multiview three-dimensional video communications based on distributed video coding,” *IEEE Systems Journal*, Apr. 2015. (to appear, DOI:10.1109/JSYST.2015.2414662)

The work in the paper is presented in Chapter 3.

## Conference Papers

[1] **P. Gao** and W. Xiang “Modeling of packet-loss-induced distortion in 3-D synthesized views,” in *Proc. IEEE Int. Conf. Visual Communications and Image Processing (VCIP 2015)*, Dec. 2015, pp. 1–4.

The work in the paper is presented in Chapter 4.

[2] **P. Gao**, W. Xiang, and L. Zhang, “Transmission distortion modeling for view synthesis prediction based 3-D video streaming,” in *Proc. IEEE Int. Conf. Acoustics, Speech, and Signal Processing (ICASSP 2015)*, Apr. 2015, pp. 1448–1452.

The work in the paper is presented in Chapter 5.

[3] **P. Gao**, W. Xiang, J. Billingsley, and Y. Zhang, “Error-resilient multi-view video coding for next generation 3-D video broadcasting,” in *Proc. IEEE Int. Conf. Information and Communication Technology Convergence (ICTC)*, Jeju Island, South Korea, Oct. 2013, pp. 1022-1026.

The work in the paper is presented in Chapter 3.

## Other publication

[1] W. Xiang and **P. Gao**, “Distributed video coding-based robust mobile multi-view 3-D video communications,” *IEEE Communications Society (ComSoc) Multimedia Communications Technical Committee (MMTC) E-Letter*, vol. 9, no. 1, pp. 17-19, Jan. 2014.

# Contents

<b>Abstract</b>	<b>i</b>
<b>Acknowledgments</b>	<b>iv</b>
<b>List of Publications</b>	<b>v</b>
<b>List of Figures</b>	<b>x</b>
<b>List of Tables</b>	<b>xiii</b>
<b>Acronyms &amp; Abbreviations</b>	<b>xv</b>
<b>Chapter 1 Introduction</b>	<b>1</b>
1.1 Motivations . . . . .	2
1.2 Research Problems . . . . .	4
1.2.1 Forward Error Correction of Compressed Multi-View Video Signal . . . . .	5
1.2.2 Prevention of Error Propagation in Multi-View Video Plus Depth Map Coding . . . . .	6
1.2.3 Elimination of Prediction Position Errors in View Synthesis Prediction . . . . .	7
1.3 Contributions . . . . .	9
1.4 Research Methodology . . . . .	12
1.5 Organization . . . . .	13
<b>Chapter 2 Background</b>	<b>15</b>
2.1 2-D Video Coding Standards and Techniques . . . . .	15
2.2 Distributed Video Coding . . . . .	17
2.3 Multi-view Video Coding . . . . .	19
2.4 3-D Video Coding . . . . .	20
2.4.1 Overview of 3D-AVC . . . . .	21
2.4.2 Overview of 3D-HEVC . . . . .	23
2.5 Error-Resilient Coding Techniques . . . . .	27
2.5.1 Source-Level Error Resilient Video Coding . . . . .	27

2.5.2	Error Concealment by Post-Processing . . . . .	30
2.5.3	Interactive Error Control . . . . .	32
<b>Chapter 3 Error-Resilient Multi-view Video Coding Using Wyner-Ziv Techniques 35</b>		
3.1	Introduction . . . . .	35
3.2	Related Work . . . . .	37
3.3	Contributions of This Chapter . . . . .	39
3.4	Proposed Error-Resilient Scheme Using Embedded WZ Description	40
3.5	Transmission Distortion Model . . . . .	44
3.5.1	Transmission Distortion of DCT Coefficients in the Key Frames of the Odd Views . . . . .	46
3.5.2	Transmission Distortion of DCT Coefficients in the Key Frames of the Even Views . . . . .	47
3.6	Bit Rate Estimation for WZ Coding . . . . .	48
3.7	Complexity Analysis . . . . .	52
3.8	Simulation Results and Discussion . . . . .	54
3.9	Summary . . . . .	68
<b>Chapter 4 Rate-Distortion Optimized Mode Switching for Error-Resilient Multi-view Video Plus Depth Based 3-D Video Coding 71</b>		
4.1	Introduction . . . . .	71
4.2	Related Work . . . . .	72
4.3	Contribution of this Chapter . . . . .	74
4.4	MVD-based 3-D Video Coding Framework and Error Propagation	75
4.5	End-to-End Distortion Estimation for MVD-based 3-D Video Trans- mission . . . . .	76
4.5.1	Frequency Domain Analysis of the View Synthesis Distor- tion Caused by Depth Error . . . . .	79
4.5.2	Expected Texture and Depth Distortion Model . . . . .	81
4.6	Mode Switching Within a Rate-Distortion Framework . . . . .	85
4.6.1	Lagrange Multiplier Determination for Rate-Constrained Coder . . . . .	87
4.7	Experimental Results and Discussions . . . . .	88
4.7.1	Estimation Accuracy of the End-to-End Distortion Model	89
4.7.2	Error-Resilient MVD-based 3-D Video Coding . . . . .	90
4.7.3	Comparison with the Rate-distortion Optimization Model in 3D-ATM . . . . .	97
4.7.4	Statistical Results of MB Coding Mode Distribution . . . . .	99
4.7.5	Computational Complexity Analysis . . . . .	101
4.8	Summary . . . . .	104

<b>Chapter 5</b>	<b>Disparity Vector Correction for View Synthesis Prediction Based 3-D Video Transmission</b>	<b>105</b>
5.1	Introduction . . . . .	105
5.2	Related Work . . . . .	106
5.3	Contribution of This Chapter . . . . .	107
5.4	View Synthesis Prediction and Error Propagation . . . . .	108
5.5	Problem Formulation . . . . .	111
5.6	Proposed Disparity Vector Correction Algorithm . . . . .	113
5.6.1	Additional Remarks of the Proposed Scheme . . . . .	122
5.7	Experimental Results and Discussion . . . . .	123
5.7.1	Evaluation of the Effects of Slice Partitioning on Coding Efficiency and Error Resilience . . . . .	124
5.7.2	Verification of Depth Error Estimation . . . . .	127
5.7.3	Performance Comparison of the Right View . . . . .	127
5.7.4	Performance Comparison of the Decoder-Side Synthesized View . . . . .	133
5.7.5	Computational Complexity Analysis . . . . .	140
5.8	Summary . . . . .	141
<b>Chapter 6</b>	<b>Conclusion and Future Work</b>	<b>143</b>
6.1	Future Work . . . . .	145
	<b>References</b>	<b>149</b>

# List of Figures

1.1	Illustration of the effect of error propagation during texture transmission. . . . .	7
1.2	The impact of the depth reconstruction error on the quality of the synthesized view. . . . .	8
2.1	Typical HEVC video encoder. . . . .	16
2.2	Conventional pixel-domain and transform-domain DVC architecture.	18
2.3	Sample multi-view video compression structure for GOP length of eight in three view setup. . . . .	20
2.4	High-level flowchart of the texture encoder in 3D-AVC. . . . .	22
2.5	Basic encoder structure of 3D-HEVC with inter-view and inter-component prediction. . . . .	24
2.6	Synthesized view distortion change calculation with respect to a currently tested depth coding mode. . . . .	26
3.1	Encoding structure for WZ coding embedded multi-view video with the key frames of the odd views protected. . . . .	41
3.2	WZ-based error resilience multi-view video coding scheme against random packet losses. . . . .	43
3.3	Block diagram of the WZ encoder. . . . .	48
3.4	Estimation precision of conditional entropy. . . . .	50
3.5	Comparison between the measured and estimated transmission distortion for the key frames of “Ballroom” with a packet loss rate of 10%. . . . .	56
3.6	Total bit rate comparison for view 3 under different packet loss rates.	61



3.7	R-D performance of the odd views with a packet loss rate of 10%. Compared to the even views, the odd views directly benefits from various error protection algorithms. . . . .	62
3.8	Subjective image quality for the 37th frame of view 3 with 10% packet loss. . . . .	64
3.8	Subjective image quality for the 37th frame of view 3 with 10% packet loss. (con't) . . . . .	65
3.9	Distortion estimation performance with packet loss rate mismatch.	66
3.10	Performance for mismatch with an assumed packet loss rate of 10%.	67
3.11	Plots of PSNR versus bit error rate for the bursty errors. . . . .	68
4.1	A prediction structure for two-view based 3-D video coding. . . .	75
4.2	Comparison of percentage of three kinds of distortion in the chan- nel distortion with 10% packet loss rate. . . . .	84
4.3	Frame-by-frame comparison between the measured and estimated end-to-end distortions. . . . .	91
4.4	Comparison of the rate-distortion curves between the proposed method and the Bruno algorithm with 10% packet loss. . . . .	94
4.5	Subjective quality comparison for frame 39 of the BookArrival se- quence with 10% packet loss rate. . . . .	95
4.5	Subjective quality comparison for frame 39 of the BookArrival se- quence with 10% packet loss rate.(con't) . . . . .	96
4.6	PSNR comparison of each frame with a packet loss rate of 10%. .	98
4.7	PSNR versus the packet loss rate for the proposed scheme and the VSRDO model in 3D-ATM. . . . .	100
4.8	Comparison of the distribution of coding modes in the proposed mode switching approach under various packet loss rates. . . . .	102
5.1	A typical prediction structure for VSP based 3-D video coding (parallel camera setup). . . . .	110
5.2	Illustration of the effect of the prediction position error introduced by VSP during the reconstruction of the right view. . . . .	112

5.3	Illustration of the proposed disparity vector correction scheme when decoding the right view. . . . .	115
5.4	Impacts of slice number on the efficiency of the coding performance and error resilience. . . . .	125
5.5	Comparison between the measured and estimated depth map errors at the frame level. . . . .	128
5.6	PSNR comparison of each frame with a packet loss rate of 10%. . . . .	132
5.7	Subjective quality comparison for frame 45 of the BookArrival sequence with the packet loss rate 10%. . . . .	134
5.7	Subjective quality comparison for frame 45 of the BookArrival sequence with the packet loss rate 10% (con't). . . . .	135
5.7	Subjective quality comparison for frame 45 of the BookArrival sequence with the packet loss rate 10% (con't). . . . .	136
5.8	Decoder-side synthesized view quality versus packet loss rate. . . . .	139

# List of Tables

3.1	Average PSNR comparison with a variety of packet loss rates. . .	58
3.2	Bit rate (kbps@30Hz) comparison for WZ encoding with various packet loss rates. . . . .	61
4.1	Average PSNR comparison between the Bruno algorithm and the proposed method with a variety of packet loss rates. . . . .	92
4.2	Computational complexity comparison between the proposed method and the original JMVC 8.0. . . . .	103
5.1	Relative difference $e_\mu$ ( $e_\sigma$ ) between the $\hat{\mu}_t$ ( $\hat{\sigma}_t^2$ ) and $\tilde{\mu}_t$ ( $\tilde{\sigma}_t^2$ ). . . .	119
5.2	Description of MVD test sequences and simulation conditions. . .	123
5.3	PSNR comparison of the proposed algorithm with two different slice size settings (GOP = 30). . . . .	126
5.4	Correlation coefficient between the estimated and measured depth map errors at the MB level for each video sequence. . . . .	127
5.5	Average PSNR comparison for the right view video with a variety of packet loss rates. . . . .	130

5.6	Performance comparison for the texture at the packet loss rate of 10% when both VSP and TDCP are enabled. . . . .	131
5.7	Average PSNR comparison for the synthesized view video with a variety of packet loss rates. . . . .	137
5.8	Total bit rate comparison. . . . .	140
5.9	Computational complexity comparison between the proposed method and the VSP-enabled JMVC. . . . .	141

# Acronyms & Abbreviations

ADE	Actual depth error
ARQ	Automatic repeat request
AVC	Advanced Video Coding
BDPSNR	Bjontegaard Delta peak signal-to-noise ratio
B-VSP	Backward view synthesis prediction
CABAC	Context-adaptive binary arithmetic coding
CIVEP	Consideration of inter-view error propagation
DCT	Discrete cosine transform
DIBR	Depth image-based rendering
D-MVP	Depth-based motion vector prediction
DFT	Discrete Fourier Transform
DVC	Distributed video coding
EC	Error concealment
EDE	Estimated depth error
FEC	Forward error correction
FMO	Flexible macroblock ordering
FVV	Free Viewpoint Video
GOP	Group of Pictures
HEVC	High Efficiency Video Coding
JCT-3V	Joint Collaborative Team on 3D Video Coding Extension Development
JMVC	Joint multi-view video coding
JVDF	Joint view depth filtering
LDPCA	Low-density parity-check accumulated
MAD	Mean absolute difference
MB	Macro-block

MPEG	Moving Picture Experts Group
MSE	Mean square error
MTU	Maximum transmission unit
MVC	Multi-view video coding
MVD	Multi-view video plus depth
MVE	Motion vector extrapolation
MVP	Motion vector prediction
PRISM	Power-efficient, Robust, hIgh compression Syndrome-based Multimedia coding
PSNR	Peak Signal-to-Noise Ratio
QP	Quantization parameter
ROPE	Recursive optimal per-pixel estimate
R-D	Rate distortion
RPS	Reference picture selection
SI	Side information
SSD	Sum of squared difference
SVDC	Synthesized view distortion change
3-D	Three dimensional
3D-AVC	AVC-compatible 3-D video coding
3D-ATM	3D-AVC test model
3D-HEVC	HEVC-compatible 3-D video coding
3D-HTM	3D-HEVC test model
3DTV	3-D Television
TDCP	Translational disparity compensation prediction
VCEG	Video Coding Experts Group
VSP	View synthesis prediction
VSRDO	View synthesis-based rate-distortion optimization
VSRS	View synthesis reference software
WZ	Wyner-Ziv

# Chapter 1

## Introduction

With the success of three dimensional (3-D) blockbusters and the advancements in stereoscopic display and transmission technologies, 3-D video applications have been gathering momentum in recent years. Current 3-D video technology being introduced to homes is mostly based on stereo systems [1]. These systems use stereo video coding for pictures delivered by two input cameras. Typically, such stereoscopic systems only reproduce these two camera views and require wearing special 3-D glasses, such as anaglyph, polarized, and shutter glasses. In addition, stereoscopic 3-D video systems evoke 3-D perception by binocular parallax [2], in which scene is presented in a fixed perspective defined by two transmitted views, while further manipulations on depth perception require expensive computation with current technologies.

Free Viewpoint Video (FVV) and 3-D Television (3DTV) represent the next generation of video paradigms whose goal is the involvement of the observer or the depth perception without stereoscopic glasses [3]. There are several target fields for FVV and 3DTV covering a large number of areas like cinemas, home theatres, and video conferencing. In all these areas, the immersion is provided by representing the scene complying with the depth, and by displaying the 3-D world showing a high number of viewpoints to freely change the perspective and to give the spectator a real world experience. Generally speaking, immersion performance largely depends on the number of viewpoints that, when high, improves the sensation. However, when continuously increasing the number of cameras to capture the scene, a considerable amount of information has to be recorded or

transmitted [4]. To overcome this disadvantage, the number of viewpoints captured should be limited, bringing a data decrease, but also a worsening of the 3-D effect. Consequently, a strong need to render additional virtual views from the transmitted views arises, in order to support auto-stereoscopic displays, which emit different pictures depending on the position of the observer's eyes and do not require glasses for viewing.

Generally speaking, the virtual view generation needs to be set in a well-defined environment composed of one or more texture sequences and their corresponding depth sequences. This setting is usually called multi-view video plus depth (MVD) environment. In a common virtual view rendering setup, by knowing the camera parameters (extrinsic and intrinsic) and the captured view positions, a virtual intermediate view can be efficiently synthesized by a back projection of the nearest captured reference views to the 3-D scene coordinate, followed by a projection to the virtual view camera location. Since the MVD format enables the 3-D display to generate virtual images for arbitrary views by using the depth information, only a small number of the views needs to be encoded and transmitted [5]. With respect to the stereoscopic counterpart, the depth image-based representation 3-D system generally offers three obvious advantages. Firstly, the depth-based system can adjust the baseline distance of the presented stereo pair, which cannot be achieved easily in stereoscopic 3-D. Secondly, the depth-based system is suitable for glasses-free display, while stereoscopic 3-D systems require glasses to enable depth perception. Finally, the depth-based 3-D video framework exhibits higher coding and transmission efficiency. Therefore, in this study, we focus on the research of the coding and transmission techniques of the emerging multi-view video representation, including the pure multi-view video and MVD based video, and the stereo-paired video for 3-D viewing can be regarded as a special case of multi-view video.

## 1.1 Motivations

Although the MVD-based 3-D video systems facilitate the generation of intermediate views with high compression capabilities, they are not mature enough



at this stage. Given that the multi-view video signal is obtained from a set of cameras, there usually exists some illumination differences between different views. These illumination differences need to be compensated before inter-view prediction process. Depth maps, which play a key role in the synthesis of virtual views, need to be either captured with specialized apparatus or estimated from scene textures using stereo matching. Existing solutions for generating high quality depth maps are either costly or not sufficiently robust. Besides, there is a strong need to achieve efficient storage and robust transmission of this additional information. Typically, a depth map is composed of large homogeneous areas partitioned by sharp edges, with limited texture information. Knowing that the depth maps and textures are statistically different in nature, conventional video coding schemes being targeted for texture video cannot produce coding performance that is satisfactory for depth maps. Meanwhile, as the synthesized views are the ultimate information for viewing, depth compression may employ entirely different distortion measures for its rate-distortion optimization.

On the other hand, during transmission of MVD-based 3D video, one inherent problem of video and depth transmitted over unreliable channels is that information may be altered or lost during transmission due to channel noise. The effect of such information loss can be devastating for the transport of compressed video and depth because any damage to the compressed bit stream may lead to objectionable visual distortion at the decoder. These distortions on texture and depth will further result in annoying effects in rendered views. Moreover, as the depth map is used to render virtual views, the depth map error due to packet loss may introduce new types of distortion that are different from that of conventional 2-D video transmission. Thus, robustness to transmission errors is a crucial requirement. In addition, it is also critically important to define an appropriate evaluation procedure to measure 3-D video quality after transmission, as no well-defined process for evaluating the impact of depth coding and rendering results exists [6]. Therefore, in order to deploy a successful MVD-based 3-D system in the near future, there still are many unresolved issues.

While most previous works have been exclusively concerned with the compression performance by removing redundancies present in the MVD represen-

tation [7], in this thesis, we will focus primarily on addressing the problem of reliable or error-resilient transmission of MVD format based 3-D video over the error-prone channels, which has received much attention during the last few years because of the increasing bandwidth in the next generation network and rapidly growing demand for visual communication. Because the texture video and depth map are 2-D images, usual 2-D error-resilient algorithms are spontaneously thought of as appropriate for enhancing the robustness of 3-D video. However, since the new types of data format and distortion are introduced, conventional 2-D error-resilient approaches seem unable to efficiently improve the 3-D video streaming quality. The results of our study will also confirm the common idea that 3-D video is not just the extension of 2-D video and that commonly used 2-D video error-resilient algorithms are not sufficient to protect the 3-D video quality. Thus, due to the unique characteristics of MVD-based 3-D video coding and the subtle propagating behaviour of transmission errors, robustly delivering 3-D video is still an elusive open question to date. Considering the demand for high-quality visual content, this study is practically critical to ensure a comfortable, realistic, and immersive 3-D visual experience.

In the next sections, the specific research problems of this project, and the primary contributions in this dissertation are presented. Then, we give an explanation on how to carry out the research work, including data collection, experimental setup, and performance evaluation. Finally, the organization of this thesis is provided at the end of this chapter.

## 1.2 Research Problems

Due to the limited bandwidth of the transmission channels, video signals have to be compressed by efficient coding algorithms. To achieve high compression, most current video compression systems employ motion-compensated prediction between frames to exploit the temporal redundancy, followed by a spatial transformation to exploit the spatial redundancy, and the resulting parameters are entropy-coded followed by quantization to produce the compressed bit stream. The hybrid signal prediction/residual coding paradigm provides significant com-

pression efficiency, however the compressed signal is highly vulnerable to losses when transmitted over error-prone channels. A single packet loss may result in video quality degradation in an entire picture or an area of it. Moreover, the use of predictive coding will cause these errors to propagate to subsequent frames, thus significantly impacting on the received video quality. Usually, in multi-view video or MVD-based 3-D video system, both the 2-D video and depth are either independently encoded by common video compression techniques, or encoded by the improved coding framework that exploits the inter-component correlations between texture and depth. So the transmission of 3-D video also suffers from the same problem of transmission errors. Furthermore, as mentioned above, the varying characteristics of the input 3-D videos and the sophisticated data representation will make the problem much more difficult to solve. In view of this, in this project we will investigate three commonly encountered problems in 3-D video transmission, which are juxtaposed as follows.

### 1.2.1 Forward Error Correction of Compressed Multi-View Video Signal

In order to reduce the bandwidth requirement, an initial approach of coding MVD data was to compress multi-view texture and multi-view depth data independently with Multi-view video coding (MVC). This straightforward approach was accepted by the Moving Picture Experts Group (MPEG) 3DV standardization as an anchor technology [8]. However, when the compressed multi-view video bit stream coded with MVC is transmitted over an error-prone channel, it is extremely sensitive to transmission errors owing to the sophisticated inter-view prediction technique. If an error occurs in a frame of one view and cannot be effectively corrected, it can not only propagate to the following predictively coded frame in the same view, but also to the decoding frames in adjacent views. This error propagation problem would cause more substantial deterioration in video quality than that of single-view video transmission. Therefore, it is necessary to develop error resilient multi-view video coding algorithms to correct transmission errors and further achieve graceful quality degradation. Recently, distributed video coding, more specifically, Wyner-Ziv (WZ) coding [9], emerges as a promis-

ing scheme for error-control video coding, which has received increasing research attention. WZ encoding relies on a statistical coding framework rather than the closed-loop prediction used in conventional video coding, and thus successfully avoid the well-known drifting effect in the presence of transmission errors. Besides, WZ coding employs the channel coding to correct transmission errors in the side information, hence inheriting a joint source and channel coding framework. Inspired by this, how to use the built-in error resilience of distributed video coding to improve robustness for multi-view video transmission is one of the most important issues to be addressed in this thesis.

### 1.2.2 Prevention of Error Propagation in Multi-View Video Plus Depth Map Coding

As usual, during texture and depth map coding, the video frame is segmented into macroblocks (MBs) that are sequentially encoded. Each MB may be encoded in one of three coding modes: inter mode, inter-view mode, and intra mode. In inter/inter-view mode, the MB is first predicted from the previously decoded frame via motion/disparity compensation. Then the prediction error, or residue, is transform-coded. In intra mode, the original MB data are transform-coded directly without resource to prediction, which often requires much more bits than inter/inter-view mode. Although operation in inter/inter-view mode generally achieves higher compression efficiency, it is more sensitive to channel errors as it promotes error propagation. The effect of motion compensation on spatial and temporal error propagation is shown in Figure 1.1. As can be seen from Figure 1.1, it is clear that the transmission errors bring substantial degradation to the subjective quality of the reconstructed video sequence. Moreover, even in the predictive modes, the inter and inter-view modes will result in different levels of coding efficiency and robustness to packet loss. Since the virtual views are rendered from the compressed texture video and depth maps, transmission errors of the distorted texture video and depth map can be further propagated to the virtual views. From the error resilience perspective, by switching off the inter/inter-view prediction loop for certain MBs during texture and depth coding, the reconstructed blocks are no longer dependent on past frame and error

propagation caused by predictive mismatch is stopped. However, too many intra-coded MBs will significantly degrade the coding efficiency. Therefore, an optimal trade-off needs to be made when selecting the MB coding mode. In general, the trade-off problem can be formalized that, for a given total bit rate, how to determine the coding mode for each texture and depth MB such that the overall view synthesis distortion induced by texture error and depth error at the received side is minimized. Along this line, we will propose a rate-distortion optimized mode switching algorithm for 3-D video coding to strike the balance between the error resilience and compression efficiency.



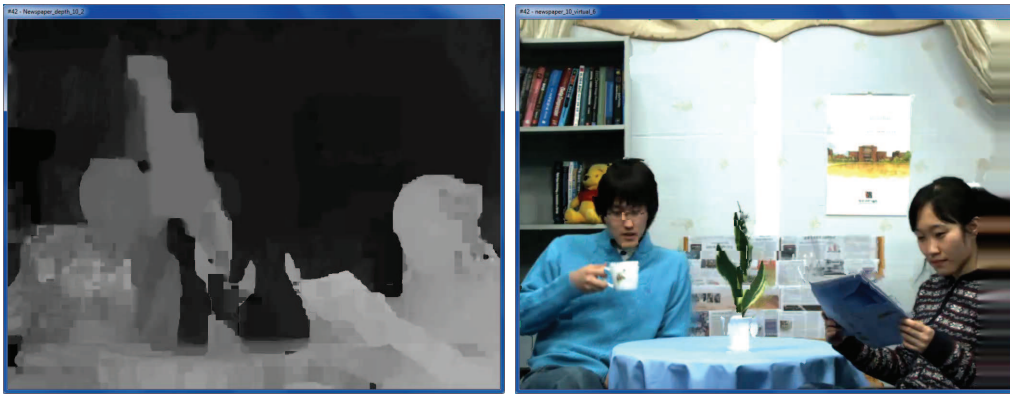
(a) *Decoded texture image without channel errors.* (b) *Decoded texture image with channel errors.*

Figure 1.1: Illustration of the effect of error propagation during texture transmission.

### 1.2.3 Elimination of Prediction Position Errors in View Synthesis Prediction

In order to exploit the essential features of multi-view video, view synthesis prediction (VSP) has been originally proposed for enhanced inter-view prediction in multi-view 3-D video coding [10]. Compared to traditional disparity compensated prediction, VSP can accurately represent the geometry transformation in a view-switch from one camera viewpoint to another. However, in VSP-based 3-D video transmission, while using the distorted texture video and depth map to synthesize the virtual reference view at the decoder, due to the depth map re-

constructed error caused by packet losses, the projection of the pixel in the base view will have a geometrical displacement in the virtual view. The impact of the depth reconstruction error on the picture quality of the synthesized virtual view is shown in Figure 1.2, in which the synthesized view is rendered from the erroneous depth images and error-free texture images. It can be clearly observed that the texture regions are projected to the wrong spatial locations in the synthesized image due to the reconstructed depth errors. If the virtual view is further used to predictively encode the dependent view, this geometry error may propagate to the dependent view along the disparity compensated path, and consequently cause the so-called prediction position error for the decoding of the dependent view. More precisely, when the packet of the current pixel in the dependent view is correctly received, the decoder can easily access the disparity vector and residue. However, due to the geometry error in the synthetic reference view, the disparity compensation reference pixel pointed to by the received disparity vector is distinctly from that used at the encoder. Continuing to use these fetched pixels for disparity compensation will result in a significant predictive mismatch and severely afflict the video quality of the dependent view. In order to effectively mitigate the effect of the prediction position error in the texture video of the dependent view, we will propose a novel disparity vector correction algorithm at the decoder.



(a) *Decoded depth image with channel error.* (b) *Rendered image of the virtual view.*

Figure 1.2: The impact of the depth reconstruction error on the quality of the synthesized view.

Motivated by the aforementioned issues, the thesis aims to solve the following detailed problems:

**Problem 1:** Given two dimensional error propagation in the temporal and inter-view directions in multi-view video streaming, how can one utilize the error correction capability of distributed video coding to achieve graceful quality degradation of multi-view video?

**Problem 2:** Given that different coding modes in texture and depth result in different levels of coding efficiency and robustness to packet losses, how can one optimize the selection of the coding modes to significantly improve the error resilience of 3-D video streaming while not sacrificing the coding efficiency too much?

**Problem 3:** Given that the depth errors caused by packet losses usually lead to geometry error and prediction position error in the VSP structure, how can one dramatically mitigate the adverse effect of the prediction position error at the receiver in a computationally efficient way?

## 1.3 Contributions

The thesis presents methods for reducing the quality degradation of both the coded views and synthesized views in MVD-based 3-D video transmission. These proposed methods operate in the application layer and involves 3-D video encoders and/or decoders. The goal of the research is to design a systematic error resilience framework for MVD-based 3-D video transmission over Internet or wireless channel. Specifically, the contributions in this project toward the above specific research problems can be summarized as follows:

- **Error-Resilient Multi-view Video Coding Using Embedded WZ**

**Description:** An efficient error-resilient scheme based on WZ coding for multi-view video transmission over error-prone channels is proposed in Chapter 3. At the encoder, the key frames of the odd views are protected by WZ encoding to generate the auxiliary bit-stream alongside the multi-view video coded bit-stream. At the decoder, error-concealed multi-view decoded frames are used as the side information (SI) for WZ decoding. Based

on the study on the characteristics of MVC and the complex propagating behaviour of channel errors, a recursive model to estimate the expected transmission distortion is developed in the transform domain, in which the channel-induced distortion takes into consideration both motion and disparity compensation induced inter-frame dependencies. With the proposed model, we propose a rate control strategy for WZ encoding to infer the minimum bit rate so as to correct the SI errors. The WZ bit rate estimation method exploits the correlation between the original bit-planes and the SI bit-planes as well as the bit-plane interdependency. Based on the rate allocation scheme, the proposed error-resilient scheme can transmit extra WZ bits minimally to protect the compressed multi-view video. Extensive experimental results show that the proposed WZ-embedded scheme outperforms Reed Solomon based forward error correction method by about 1.1 dB and outperforms the adaptive intra refresh algorithm by approximately 1.6 dB at the packet loss rate of 10%, demonstrating the potential of our proposed error correction algorithm.

- **Rate-Distortion-based Optimal Mode Switching for Error-Resilient**

**MVD Based 3-D Video Coding:** A rate-distortion optimized coding mode switching scheme is proposed in Chapter 4 to improve error resilience for MVD based 3-D video transmission over lossy networks. Firstly, we derive a new end-to-end distortion model for MVD-based 3-D video transmission. As compared to the previous MVD-based video distortion models in which distortion is measured by only investigating the expected texture video errors and depth errors on the synthesized virtual view, the proposed scheme characterizes both the end-to-end distortions in the rendered virtual view and the coded texture video due to packet losses. Moreover, inter-view error propagation for the texture video and depth map is also explicitly considered, and the compound impact of the texture error and depth error on the view synthesis distortion is mathematically analyzed in the frequency domain. Based on the proposed distortion model, an optimal mode decision algorithm is then performed in the texture video and depth map coding process. Taking into consideration the inherent correlation be-



tween the texture and depth, the optimization of texture coding mode with respect to the depth mode in the lossy environment is achieved through a local exhaustive search. Finally, to adapt the proposed algorithm to the rate or distortion constrained environment, we develop a convex scheme to determine the appropriate Lagrange Multiplier. Experimental results show that the proposed method provides significant improvements in terms of both objective and subjective evaluations, not only on the synthesized views, but the coded views as well.

- **Disparity Vector Correction for View Synthesis Prediction-Enhanced**

**3-D Video Coding and Transmission:** VSP is a crucial tool for enhancing the coding efficiency in the next-generation 3-D video systems. However, VSP will lead to catastrophic prediction position errors when the depth maps are corrupted by packet losses during transmission. In order to mitigate the prediction position errors, a novel disparity vector correction algorithm is proposed in Chapter 5. Firstly, we investigate the relationship between the rendering position errors and the depth errors according to the VSP procedure. The depth map errors due to packet losses and error propagation are then recursively estimated at the decoder without the use of the error-free reconstructed frames. In particular, by considering the piecewise smoothness characteristics of the depth map, the error introduced by packet loss in the current depth pixel is approximated by the difference between the co-located pixels from the two preceding depth frames. The accuracy of the depth error estimation is demonstrated via simulation results. Finally, based on the estimation of the reconstructed depth errors, the received disparity vectors can be corrected to find the matching synthesized pixels as those used at the encoder, and thereby the view synthesis based inter-view error propagation can be effectively stopped. Simulation results show that the proposed methods with the estimated and actual depth errors can provide significant improvements in both rate-distortion performance and subjective quality over known state-of-the-art error concealment schemes. This algorithm is useful as a complementary option with the encoder error-resilient scheme.

## 1.4 Research Methodology

In this section, we mainly describe how data would be collected; how theoretical results would be evaluated by experiments; and how the proposed techniques would be compared with known options in the literature.

The data we used in this thesis are original YUV multi-view video sequences. For example, the commonly used multi-view video sequences in the literature are: Ballroom, Exist, Race1, BookArrival, Lovebrid1, Newspaper, GT\_Fly, and Undo\_Dancer. These sequences are released from different research groups and institutes in the world, and adopted by the JVT/MPEG 3-D audio and visual (3DAV) group [7]. They are available to be downloaded from the public website. The first 3 sequences are pure multi-view video sequence, while the remaining sequences are multi-view texture sequences along with depth maps. These sequences are selected from different sources and have different spatiotemporal characteristics, which lead to different behaviour of compression algorithms. The detailed descriptions of the test sequences are provided in the corresponding main chapters.

These sequences are compressed using either Joint Multi-view Video Coding (JMVC) reference software or 3D-AVC Test Model (3D-ATM) [11]. For each sequence, we use four different quantization parameters to generate different quality video with different bit rate. For simulating the packet losses, the coded video frame is firstly packetized based on the real-time transfer (RTP) protocol specifications, and then the common conditions for low-delay IP/UDP/RTP packet loss resilient testing defined in [12] is used. We record the PSNR and bit rate to measure the compression performance for each sequence at each quantization parameter. To verify whether the proposed algorithm can work correctly, we always include the performance of the original JMVC or 3D-ATM at error-free setting as a benchmark. To compare the proposed techniques with the known option in the literature, we adopt two kinds of quantitative methods. On one hand, we do the performance comparison without bit rate control during coding. The purpose of this comparative design is to see how the proposed techniques affect both the bit rate and PSNR, which is useful for the applications where the bit rate may be not an priori. In this case, we use Bjontegaard Delta-PSNR (BD-PSNR) [13] to

measure the average difference between two rate-distortion curves. On the other hand, we compare the performances using joint rate control and error resilient algorithms. In this scenario, the permissible bit rate for each sequence is usually given, and the related rate control schemes are employed to ensure the output bit rate as close as possible to the target bit rate. For performance evaluation, we only need to do the PSNR comparison.

All the research methods in this thesis are made up of two components: end-to-end distortion estimation and rate-distortion optimization. For decoder-side distortion estimation, we use a divide-and-conquer method to quantify the effects of three individual terms on end-to-end distortion: 1) quantization distortion; 2) error propagation distortion; 3) error concealment distortion. While the quantization distortion is known to the encoder, the error propagation distortion and error concealment distortion depend on the particular channel realization and is unknown to the encoder. To estimate the error propagation distortion, we characterize the channel behaviour using the packet loss probability, which is modelled as an independent time-invariant packet erasure channel. To estimate the error concealment distortion, we simulate the error concealment at the encoder and then average the simulated distortions. The spatiotemporal correlation in the input video sequence is characterized by a recursion function introduced in [47]. To improve the performance of the video transmission system, the estimated distortion is incorporated into a rate-distortion framework based upon Lagrangian optimization. The basic idea of Lagrangian optimization is to introduce a Lagrange multiplier for the given constraint, which can thus be relaxed [148]. When the Lagrangian optimization is not able to reach points which are not on the convex hull of the rate-distortion curve, dynamic programming method may be used [149].

## 1.5 Organization

The dissertation begins with a review of 3-D video coding methods and some error-resilient techniques for video transport over lossy networks in Chapter 2.

The proposed error-resilient multi-view video coding with embedded WZ de-

scription is analyzed in Chapter 3. Section 3.4 presents the generic framework of the proposed WZ-based scheme. A mathematical transmission distortion model for multi-view video transmission and bit rate estimation for WZ encoding are studied and developed in Sections 3.5 and 3.6, respectively. In Section 3.7, an analysis of the added computational load of our proposed algorithm is presented. Experimental results are presented and discussed in Section 3.8, and Section 3.9 concludes this algorithm.

The rate-distortion-based optimal mode selection algorithm is analyzed in Chapter 4. Section 4.4 describes the typical encoding prediction structure for texture video and depth map coding. In Section 4.5, an end-to-end distortion estimation model for MVD-based 3-D video transmission is first proposed. Subsequently, in Section 4.6, the estimated overall distortion is incorporated into the rate-distortion based mode switching coders. The experimental results are presented and discussed in Section 4.7. Finally, concluding remarks are drawn in Section 4.8.

In Chapter 5, the research is moved to research into robust VSP-based 3-D video transmission. The rest of this chapter is organized as follows. Section 5.4 briefly describes the basic VSP-based encoding prediction structure for the texture video coding with enhanced inter-view prediction. In Section 5.5, we analyse the error propagation behaviour of the VSP-based 3-D video transmission, and define a new prediction position error induced by the depth map errors. Subsequently, in Section 5.6, a novel disparity vector correction method is proposed accordingly to eliminate the prediction position error. Experimental results are presented and discussed in Section 5.7. Finally, we provide the concluding remarks in Section 5.8.

Finally, Chapter 6 summarizes this thesis with some comments on future research directions.

# Chapter 2

## Background

This chapter reviews the recent emerging 3-D video coding methods and standards, as well as the error resilience techniques for video transport over unreliable networks. It starts in Section 2.1 with a brief overview of 2-D video coding standards, i.e., H.264/Advanced Video Coding (AVC) and High Efficiency Video Coding (HEVC), which are the basis for multi-view video and 3-D video coding. Then, distributed video coding is briefly reviewed in Section 2.2, for the benefit of the readers who are not familiar with this unique video coding paradigm. Section 2.3 reviews the multi-view video compression principle. Section 2.4 mainly discusses two most important 3-D video coding techniques and standards, i.e., the AVC-compatible 3-D video coding (3D-AVC) and HEVC-compatible 3-D video coding (3D-HEVC). 3D-AVC standard has been recently finalized, whereas 3D-HEVC is currently still under development. Section 2.5 provides a review of the general principles of error resilient coding technique in the video coding community. Since most key technologies of 3-D video coding are still in progress, there are very few works in the literature reported on error resilient 3-D video coding and transmission at this stage. Therefore, the very specific related works on reliably transmitting 3-D video will be discussed in the corresponding chapters.

### 2.1 2-D Video Coding Standards and Techniques

The video coding international standardization landscape includes two major players, the ISO/IEC MPEG and the ITU-T Video Coding Experts Group (VCEG).

The ITU-T produced H.261 [14] and H.263 [15], ISO/IEC produced MPEG-1 [16], and MPEG-4 Visual [17], and the two organizations jointly produced the H.262/MPEG-2 video [18], and H.264/MPEG-4 AVC [19] standards. The two standards that were jointly produced have had a particularly strong impact and have found their way into a wide variety of products that are increasingly prevalent in our daily lives. Following the same trends, MPEG and VCEG specified recently a new video coding standard, the HEVC standard [20], addressing ultra-high definition applications beside the usual H.264/AVC applications, and bringing an approximate 50% rate reduction for the same perceptual quality compared to the H.264/AVC most efficient high profile solution. The technological paradigm behind the major advances in all video compression standards is the so-called predictive video coding, which relies on redundancy and irrelevance removal as none contribute to the perceptual quality of the decoded video. While spatial, temporal and statistical redundancies are reduced using spatial transformation, motion compensated temporal prediction, and entropy coding tools, irrelevancy is mainly reduced through the quantization of the transform coefficients.

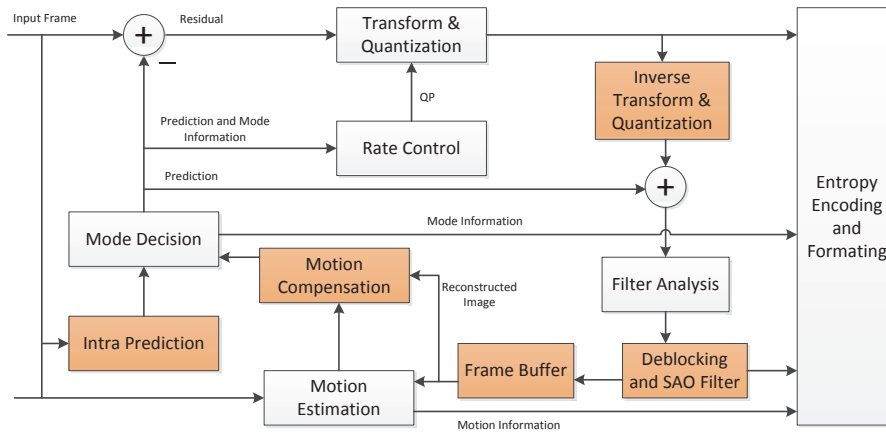


Figure 2.1: Typical HEVC video encoder.

In principle, the video coding layers of all video compression standards employ the same block-based hybrid video coding framework, which is briefly reviewed as follows. Figure 2.1 shows the key steps in the coding paradigm of HEVC. As illustrated, each picture is split into block-shaped regions, with the exact block partitioning being conveyed to the decoder. The first picture of a video sequence is coded using only intra prediction within the same picture. For all remaining

pictures of a sequence or between random access points, inter temporally predictive coding modes are typically used for most blocks. The encoding process for inter prediction consists of choosing motion data comprising the selected reference picture and motion vector to be applied for predicting the samples of each block. The encoder and decoder generate identical inter prediction signal by applying motion compensation using the motion vector and mode decision data, which are transmitted as side information. The residual signal of the intra or inter prediction, which is the difference between the original block and its prediction, is transformed by a linear spatial transform. The purpose of the transform is to reduce the spatial correlation between adjacent residual pixels, and to compact the energy of the residual pixels into a few coefficients. The transform coefficients are then scaled, quantized, entropy coded, and transmitted together with the prediction information. The encoder duplicates the decoder processing loop (see yellow-shaded boxes in Figure 2.1 ) such that both will generate identical predictions for subsequent data. Therefore, the quantized transformation coefficients are constructed by inverse scaling and are then inverse transformed to duplicate the decoder approximation of the residual signal. The residual is then added to the prediction, and the result of that addition may then be fed into the loop filters to smooth out artifacts induced by block-wise processing and quantization. The final picture representation is stored in a decoded picture buffer to be used for the prediction of subsequent pictures. Since the encoder dictates exactly how each code block is reconstructed, this hybrid video coding paradigm has a computationally expensive video encoder. However, due to the superior performance in compression, this kind of paradigm results in today's many practical video codec.

## 2.2 Distributed Video Coding

Distributed video coding (DVC) is the result of the information-theoretic bounds established for distributed source coding by Slepian and Wolf for lossless coding [92], and by Wyner and Ziv for lossy coding with decoder side information [93]. Lossless distributed source coding refers to two correlated memoryless sources independently encoded and jointly decoded by exploiting the statistical

dependencies. Slepian and Wolf proved that no compression efficiency is compromised in lossless coding by exploiting source correlation at the decoder only. Then Wyner and Ziv extended the Slepian-Wolf theorem to the lossy coding case, which shows that the same rate-distortion performance is also attainable if the source correlation is exploited at the encoder with Gaussian sources [93]. Based on DVC principles, it is well-known that there are two main DVC architectures proposed in the literature, namely, the PRISM (Power-efficient, Robust, hIgh compression Syndrome-based Multimedia coding) architecture [94] and the Stanford architecture [95], respectively. In the PRISM architecture, a WZ frame is transformed using a block-wise discrete cosine transform (DCT) and the transformed coefficients are quantized with a uniform scalar quantizer. Then a coding mode decision strategy will decide whether a block is coded as a WZ block or as an intra block. The block-based characteristic of PRISM allows for a better local adaption of the coding mode in order to cope with nonstationary statistical properties of video data. However, this block partitioning implies a short block-length which is a limiting factor for efficient channel coding.

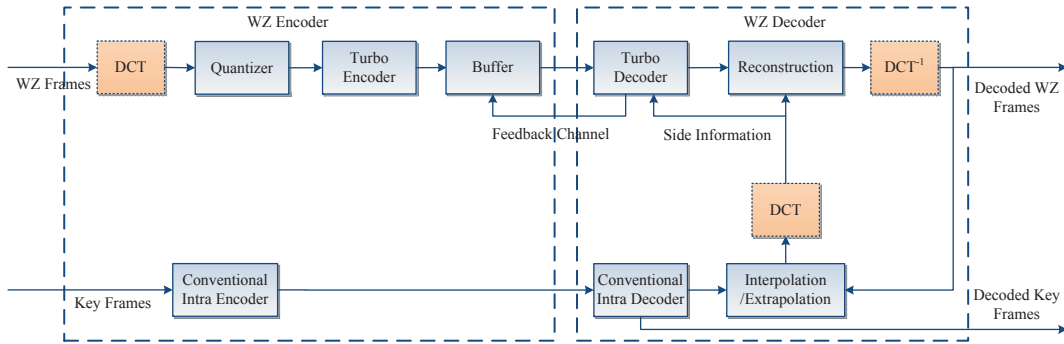


Figure 2.2: Conventional pixel-domain and transform-domain DVC architecture.

For ease of description, Figure 2.2 illustrates a block diagram of this architecture of the Stanford approach [95]. The video sequence is first divided into group of pictures (GOPs). The first frame of each GOP, referred to as key frame, is encoded using a conventional intra frame coding technique such as MPEG-4 or H.264/AVC intra mode coding. The remaining frames in a GOP are encoded using distributed coding principles and are referred to as WZ frames. In a pixel domain WZ version, the WZ frames first undergo quantization. Alternatively, in a



transform domain version, a DCT is applied prior to quantization. The quantized values are then fed into a punctured turbo coder. The systematic bits are discarded and only the parity bits of the turbo coder are properly stored in a buffer. The encoded bit-stream is thus composed of different parts: the H.264/AVC intra coded key frame stream, and the WZ stream. The information of the key frame is entirely sent to the decoder, while the parity bits are only partially sent, depending on the decoder requests to the encoder, provided iteratively through a feedback channel. At the decoder, the side information is generated by motion-compensated interpolation or extrapolation of previously decoded frames. And then the side information is used in the turbo decoder, along with the parity bits of the WZ frames, in order to reconstruct the bit planes, and subsequently the decoded video sequence. Since the motion compensation task is shift from the encoder to the decoder, this kind of video coding paradigm has a computationally inexpensive video encoder.

## 2.3 Multi-view Video Coding

At the early stage for multi-view video compression, a simulcast coding method was used as a straightforward solution. It individually encodes video information for each view with the existing single view video coding standard. Since the simulcast coding method does not consider an inter-view redundancy that exists between images at different views, high coding performance could not be achieved. In order to improve the coding performance, a multi-view video coding structure which allows temporal and inter-view prediction with hierarchical B pictures is proposed in [25]. This prediction structure gives high contribution to standardization of the multi-view video compression in MPEG [26]. Figure 2.3 illustrates a sample case for a GOP length of eight in three view configuration including video information captured from left, centre, and right cameras. A left view is encoded without an inter-view prediction and then a right view is encoded using the reconstructed left view as a forward reference. Finally, a centre view is encoded using the reconstructed left and right views as forward and backward references, respectively. Indices in I, P and B pictures represent

hierarchical levels of the prediction structure. B pictures of  $B_1$  to  $B_3$  are predicted from two nearest pictures of the next higher level, and the encoded pictures are employed as reference ones for pictures of the lower level. In the case of  $B_4$ , it has the lowest level, so it is not used as a reference frame for the other B pictures. This hierarchical prediction structure has been standardized into an extension of H.264/AVC, which is referred to MVC. Recently, HEVC has been extended to support encoding of multi-view video, namely MV-HEVC [27], similar to the MVC extension of H.264/AVC.

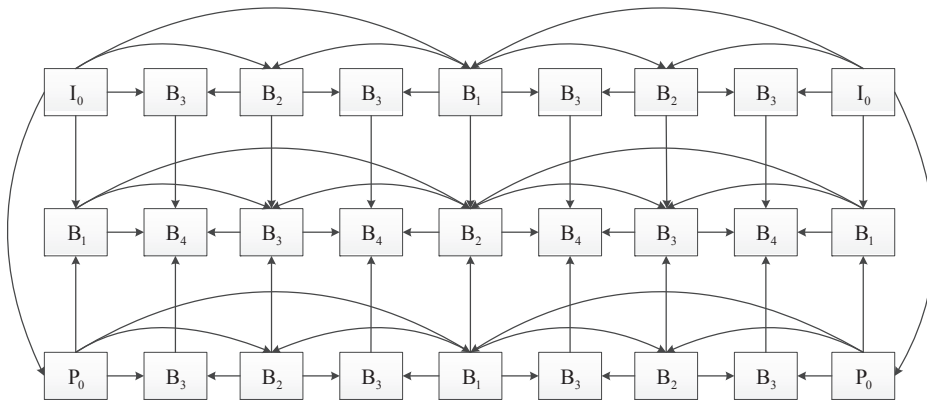


Figure 2.3: Sample multi-view video compression structure for GOP length of eight in three view setup.

## 2.4 3-D Video Coding

MPEG issued a Call for Proposals (CfP) on 3D video coding technology in March 2011 [28]. The aim of this CfP was to provide efficient compression of the MVD format and high quality intermediate view reconstruction. According to the different types of the base codes such as H.264/AVC and HEVC, two different standards, which are called AVC-compatible 3D video coding (3D-AVC) [29] and HEVC-compatible 3D video coding (3D-HEVC) [30], have been developed in the group. Some of the promising techniques from the response to CfP were included in the 3D-AVC test model (3D-ATM) and 3D-HEVC test model (3D-HTM) as the initial reference software, respectively. After the Joint Collaborative Team on 3D Video Coding Extension Development (JCT-3V) was established by VCEG

and MPEG in July 2012, more experts and companies have actively participated in the 3D video coding standardization. As a result, 3D-AVC has been recently finalized, and 3D-HEVC is currently still in progress while drafting this thesis.

### 2.4.1 Overview of 3D-AVC

A goal of the 3D-AVC development was an MVD coding system that is able to benefit from a wide deployment of H.264/AVC-based video services and from widely available hardware and software implementations of H.264/AVC. The intent was to allow only a limited number of changes to low-level processing and at the same time obtain a significant compression improvement compared to MVC-compatible coding. The encoder of 3D-AVC encodes the input MVD data into a bitstream, which consists of a sequence of access units. Each access unit consists of texture components and depth components representing one sampling of MVD data. Since the bitrate required for transmission of texture content is typically larger than the bitrate required for depth maps, a design concept is to utilize depth data for enhanced texture coding. In particular, a depth component can be coded prior to the texture component of the same view and hence used as inter-component prediction reference for the texture component. 3D-AVC also supports joint coding of texture and depth that have different spatial resolutions. Particularly, coding of depth data is supported at full, half and quarter spatial resolution compared to the resolution of the texture data. To enable some enhanced texture coding tools, the resolution of depth images is normalized to the resolution of luma texture images. Depth image normalization is implemented as in-loop up-sampling with bi-linear interpolation.

#### 2.4.1.1 Depth-based Enhanced Texture Coding Tools

3D-AVC includes two texture coding tools that utilize depth information: view synthesis prediction (VSP) and depth-based motion vector prediction (D-MVP). Figure 2.4 shows a high level flowchart of the texture coding in 3D-AVC with VSP and D-MVP modules marked in yellow colour. VSP is reviewed in the corresponding chapter, while the D-MVP is introduced below.

In H.264/AVC motion information associated with each prediction block of a

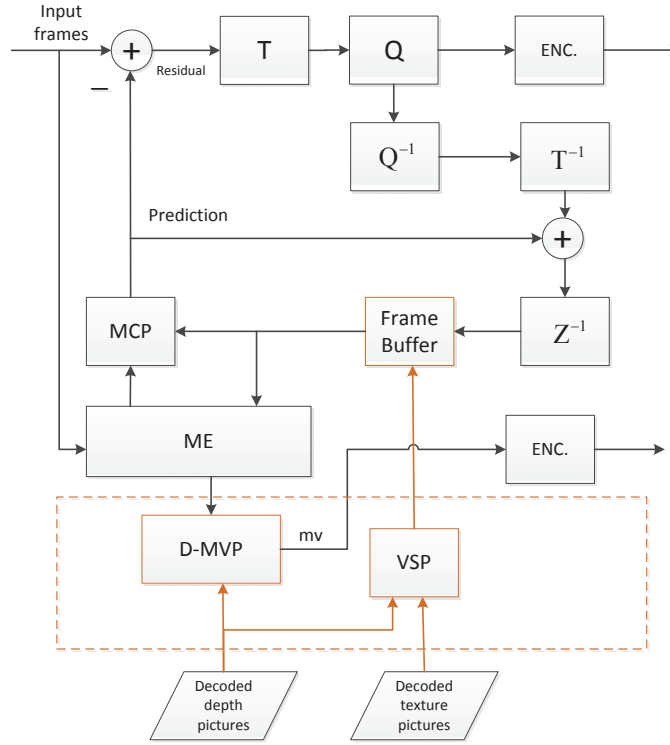


Figure 2.4: High-level flowchart of the texture encoder in 3D-AVC.

current block ( $C_b$ ) consists of three components, a reference index indicating the reference picture and two spatial components of motion vectors ( $mv_x, mv_y$ ). In order to reduce the required number of bits to encode the motion information, the blocks adjacent to  $C_b$  are used to produce a predicted motion vector ( $mvp_x, mvp_y$ ), and the difference between the actual motion information of  $C_b$  and  $mvp$  is transmitted. H.264/AVC specifies that components of the predicted motion vector are calculated by a median value of the corresponding motion vector components of the neighbouring blocks A, B and C. The layout of spatial neighbors (A, B, C) utilized in MVP is depicted in the top-left corner of  $C_b$ .

As described in more details in [31], the median MVP of H.264/AVC is not suitable for using more than one prediction (inter, inter-view, VSP), because it operates independently in the horizontal and vertical directions and also the magnitude of motion vector components can differ to a great extent in different prediction directions. Therefore, 3D-AVC restricted the conventional median MVP to identical prediction directions. All available neighbouring blocks are classified according to the direction of their prediction (inter, inter-view, VSP).

For example, if  $Cb$  uses an inter-view reference picture, all neighbouring blocks which do not utilize inter-view prediction are marked as not-available for MVP and are not considered in the median MVP. Furthermore, 3D-AVC introduced a new default candidate vector, when inter-view prediction is used in the original H.264/AVC design: if no motion vector candidates are available from the neighbouring blocks, the predicted motion vector is set to the average disparity which is computed by the depth value of  $Cb$ .

#### 2.4.1.2 Joint View Depth Filtering

The main idea of joint view depth filtering (JVDF) is that depth map filtering can utilize redundancy of multi-view depth map representation, and depth maps of all available viewpoints are filtered jointly. JVDF attempts to make depth maps of the same time instant consistent across views and hence removes depth estimation and coding errors. A detailed description and simulation results for JVDF can be found in [32], while a brief description of the JVDF algorithm is presented next.

All available depth maps are first warped to a single view  $m$ . Since warping results in multiple estimates of a noise-free depth map value in a spatial location  $(x_m, y_m)$ , a filtering is carried out for the samples of  $(x_m, y_m)$ . It is assumed that the depth value  $Z_m$  of view  $m$  is relatively accurate and therefore the correctly projected depth value  $Z_i$  from other views which describe the same object should be close in value to  $Z_m$ . A classification of similarity is defined through a confidence range and a threshold  $T$  on the absolute difference between  $Z_i$  and  $Z_m$ . The depth value  $Z_i$ , for which the absolute difference exceeds threshold  $T$  are excluded from joint filtering, whereas other depth values of location  $(x_m, y_m)$  are averaged in order to produce a “noise-free” estimate. After that, the produced “noise-free” estimate of the depth value is warped back to corresponding views that participated in joint filtering.

#### 2.4.2 Overview of 3D-HEVC

The presented 3D video coding extension of HEVC was developed for depth-enhanced 3D video formats, ranging from conventional stereo video to multi-view

video plus depth with two or more views and associated depth components. The basic structure of the 3D-HEVC encoder is shown in Figure 2.5.

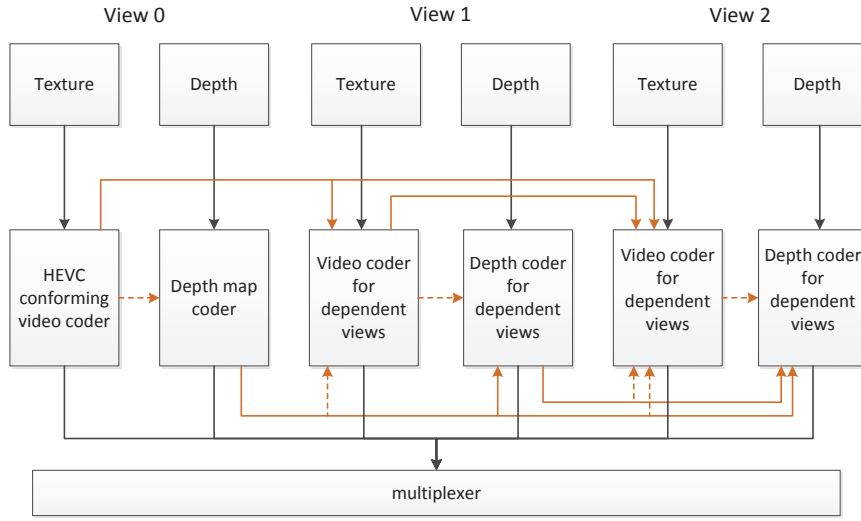


Figure 2.5: Basic encoder structure of 3D-HEVC with inter-view and inter-component prediction.

In order to provide backward compatibility with 2D video service, the base or independent view is coded using a fully HEVC compliant codec. For coding the dependent views and depth data, modified HEVC codecs are used, which are extended by including additional coding tools and an inter-component prediction technique that employ data from already coded components at the same time index, as indicated by the yellow arrows in Figure 2.5. In order to also support the decoding of video-only data, e.g., pure stereo video suitable for conventional stereo displays, the inter-component prediction can be configured in a way that video pictures can be decoded independently of the depth data. In summary, the HEVC design is extended by the following tools:

- Coding of dependent views using disparity-compensated prediction, inter-view motion prediction and inter-view residual prediction
- Depth map coding using new intra-coding modes, modified motion compensation and motion vector coding, and motion parameter inheritance
- Encoder control for depth-enhanced formats using view synthesis optimization with block-wise synthesized view distortion change and encoder-side

rendering model.

#### 2.4.2.1 Depth Map Coding

For the coding of depth maps, the same concepts of intra-prediction, motion-compensated prediction, disparity-compensated prediction, and transform coding as for the coding of the video pictures are used. However, in contrast to texture video, depth maps are characterized by sharp edges and large regions with nearly constant values. Therefore, different depth coding methods have been studied, including wavelet coding [33], mesh-based depth coding [34], as well as non-rectangular block partitioning for depth maps, such as wedgelet or platelet coding [35], and edge chain coding [36].

As the motion characteristics for the video and associated depth map in the MVD format is similar, inter-component motion vector prediction has been studied. In [37], a new inter coding mode for depth maps is added in which the partitioning of a block into sub-blocks and associated motion parameters are inferred from the co-located block in the associated texture video. Since the motion vectors of the texture signal are given in quarter-sample accuracy, whereas for the depth signal sample-accurate motion vectors are used, the inherited motion vectors are quantized to full pixel precision. For each block, based on a rate-distortion optimized policy, it can be adaptively decided, whether the partitioning and motion information is inherited from the co-located region of the texture video, or new motion data is transmitted.

#### 2.4.2.2 View Synthesis Optimization

To improve the rate-distortion optimization in depth map coding, a method is needed that relates the distortion of a depth map to the distortion of the synthesized view. Since encoding algorithms operate block-based, the mapping of depth distortion to the synthesized view distortion must be block-based as well. Moreover, the sum of partial distortions must be equal to the overall distortion of a block to enable an independent distortion calculation for all partitions of a sub-block, as hierarchical block structures are common practice in high efficiency encoders. However, disocclusion and occlusions prevent a bijective mapping of

the distorted depth map areas to distorted areas in the synthesized view. Hence, an exact mapping between the distortion of a block of the depth data and an associated distortion in the synthesized view is not possible regarding only the depth data within a currently processed block.

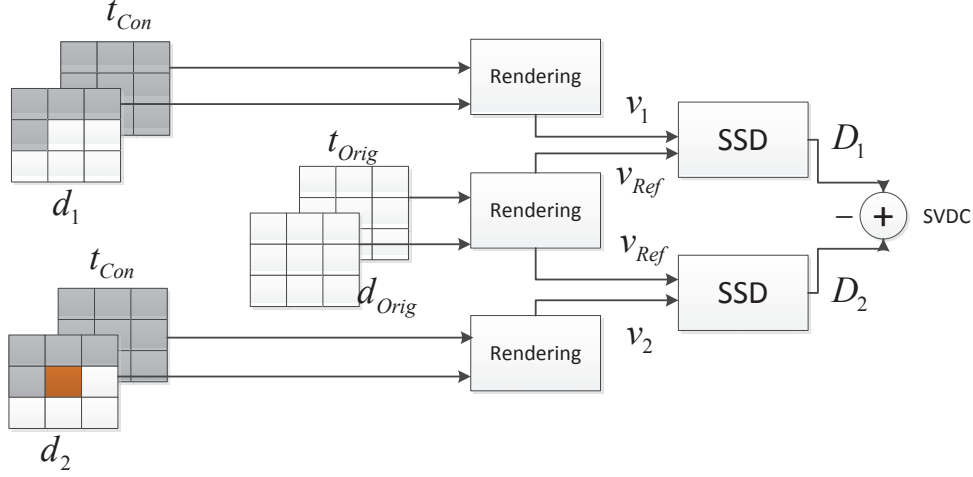


Figure 2.6: Synthesized view distortion change calculation with respect to a currently tested depth coding mode.

To resolve this issue, a method that calculates the exact synthesized view distortion change (SVDC) for a particular rendering algorithm was proposed, which computes the change of the overall distortion of the synthesized view depending on the change of the depth data within a depth block while simultaneously considering depth data outside that block. Figure 2.6 illustrates the SVDC calculation. First, for each tested coding mode for the current depth block, two variants of depth data are used: Variant  $d_1$  consists of reconstructed depth values for already coded blocks and uncoded depth values for the remaining blocks, see grey and white blocks in the top of Figure 2.6, respectively. Variant  $d_2$  is similar, but for the current block, the reconstructed depth values from the actual mode under test are used, as shown by the yellow area in the bottom of Figure 2.6. Both depth variants  $d_1$  and  $d_2$  are further used to synthesize the intermediate views  $v_1$  and  $v_2$  with the coded and reconstructed texture data  $t_{Con}$ . For the sum of squared difference (SSD) calculation, also the reference portion  $v_{Ref}$  is available, which is synthesized in the initialization phase from uncoded texture and depth data  $t_{Orig}$  and  $d_{Orig}$ . Next, both distortion can be calculated:  $D_1 = SSD(v_1, v_{Ref})$



for depth variant 1 and  $D_2 = SSD(v_2, v_{Ref})$  for depth variant 2 with the current coding mode under test. Finally, the difference between these values is used as depth distortion measure:  $SVDC = D_2 - D_1$ .

## 2.5 Error-Resilient Coding Techniques

Error resilience techniques can be roughly classified into three categories as suggested in [38], [39]: source-level error-resilient video coding, error concealment by post-processing, and interactive error control. Source-level error-resilient video coding refers to those technique in which the transmitter injects a redundancy, often known as parity or repair symbols, to the transmitted data, enabling the receiver to recover the transmitted data when the bitstream is corrupted by transmission errors. Error concealment by post processing refers to the estimation of lost picture areas based on the correctly decoded samples as well as any other helpful information. Interactive error control refers to those techniques which require interactions between the source encoder and decoder, so that the encoder can adapt its operations based on the loss conditions detected at the decoder.

Besides, modern video compression formats also have syntax support for error resilience. For example, flexible macroblock ordering (FMO) is an error-resilient tool in H.264/AVC that can avoid loss of large contiguous regions to make error concealment more feasible and effective [40]. Another example is data partitioning [41], which allows decomposition of compressed single-view video into layers of different importance for preferential protection. Furthermore, it is possible to explicitly transmit additional data in various forms to aid error concealment [42].

### 2.5.1 Source-Level Error Resilient Video Coding

In this method, the encoder operates in such a way so that the transmission errors on the coded bitstream will not adversely affect the reconstructed video quality. Compared to codec that are optimized for coding efficiency, error-resilient coders typically are less efficient in that they use more bits to obtain the same video quality in the absence of any transmission errors. The objective of error-resilient coding is to design a scheme that can achieve the minimum end-to-end distortion

under a certain rate constraint. There are many ways to introduce redundancy into the bit stream to limit the distortion caused by packet losses. The commonly used error-resilient tools are intra macroblock (MB)/picture refresh [43], redundant picture coding [44].

### 2.5.1.1 Intra MB/Picture Refresh

One way to prevent the effect of error propagation is the insertion of intra-pictures [45]. However, it is costly to code an entire picture by intra-coding, because the size of intra-coded pictures is always much larger than that of inter-coded pictures. In addition to intra-picture coding, robust MB mode selection can be used. They aim at refreshing the most error prone areas as intra-coded macroblocks to avoid drastic channel error accumulation and can be classified into heuristic and rate-distortion optimized intra MB refresh algorithms.

The heuristic intra MB refresh algorithm typically uses a mapping between the packet loss rate and the refresh frequency to determine the number of intra MBs, and then apply intra-coding uniformly across the picture area. One example is to periodically code a certain number of intra MBs per picture in a pre-defined scan order. Another example of the heuristic algorithm is to code a certain number of MBs in intra mode at randomly selected MB locations [46].

Rate-distortion optimized intra refresh algorithms refer to the approach that incorporates the overall expected distortion within the rate-distortion framework in order to automatically choose the number and the placement of the intra-coded MBs. Typically, the objective of rate-distortion optimization is to find the best mode for encoding a block with minimum distortion given the bit rate. This constrained optimization problem can be converted to a unconstrained Lagrangian cost function that linearly combines bit rate budget and end-to-end distortion, and the mode selection of each MB is such that the cost is minimized. The rate-distortion optimized intra update algorithms can be categorized into two categories: optimal per-pixel estimation and model-based MB mode selection methods, which are reviewed in more details below.

The optimal per-pixel distortion estimation method aims at computing the expected distortion at the pixel level. One of the most well-known algorithms in

this category is the recursive optimal per-pixel estimate (ROPE) algorithm [47], which computes the mean squared error by recursively calculating the first and second moments of each pixel. The original ROPE algorithm operates at integer pixel precision, and therefore it has been extended in [48] and [49] to address cross-correlation terms between pixels for more accurate distortion estimation of fractional-pel motion estimation. Other extensions of the ROPE algorithm include refinement of the distortion estimation of DCT coefficients in the transform domain [50].

Model-based MB intra update algorithms generate and recursively update a block-level distortion map for each frame to approximate end-to-end distortion [51]. However, since inter-frame displacements involve sub-block motion vectors, a motion compensated block may inherit errors propagated from multiple blocks in previous frames. Hence, block-based techniques must involve a possibly rough approximation, whose errors may build up to seriously degrade estimation accuracy. Another model-based method for calculating the average expected distortion is to run several decoders, each for different packet loss patterns, at the encoder and to average the resulting distortions [52]. Although this decoder-simulation-based mode selection algorithm estimates the expected distortion reasonably accurately when the number of simulations is high enough, the disadvantage is that the computational complexity and storage requirements are impractical for many software and hardware platforms targeting low-delay applications.

### 2.5.1.2 Redundant Picture Coding

A redundant picture is a coded representation of a primary picture or a part of a primary picture. The decoder should not decode redundant pictures when the corresponding primary picture is correctly received and can be correctly decoded. However, when the primary picture is lost or cannot be correctly decoded, a redundant picture can be utilized to improve the decoded video quality. A redundant picture can be coded as an exact copy of the primary picture, or with different coding parameters. Redundant pictures do not even have to cover the entire region represented by the primary pictures.

Thanks to the flexibility of encoding redundant coded pictures, a number of encoding methods for redundant coded pictures have been proposed. A method for unequal error protection based on redundant coded pictures was proposed in [53]. In this method, the encoder creates a key picture periodically, which is either intra-coded or predicted from the previous key picture. Each picture is protected by coding a respective redundant coded picture as an exact copy of the key picture. Redundant pictures can also be encoded with some quality degradation by using larger quantization parameters than primary pictures, such that fewer bits will be used to represent redundant pictures. The method called systematic lossy error protection (SLEP) developed in [54], belongs to this category. Another method for coding redundant coded pictures using earlier reference pictures than those of the respective primary coded pictures was proposed in [55], in which a scheme for hierarchical placement of redundant coded pictures and their reference pictures is included. The allocation of redundant coded pictures was further developed in [56], which proposed an adaptive rate-distortion optimized algorithm for coding of redundant coded pictures.

## 2.5.2 Error Concealment by Post-Processing

Decoder error concealment refers to the recovery or estimation of lost information due to transmission errors. Error concealment algorithms can be generally categorized into spatial and temporal methods. In the spatial error concealment, only the information from the current coded picture or decoded picture is used. Temporal error concealment restores the corrupted blocks by exploiting temporal correlation between successive frames. A brief review of both spatial and temporal error concealment methods is provided below.

### 2.5.2.1 Spatial Error Concealment

Spatial error concealment can operate either in the frequency domain or in the sample domain. In frequency-domain concealment, the transform coefficients of missing blocks are reconstructed from the transform coefficients of the surrounding blocks under a smoothness constraint. For example, an average of the DC coefficients of the adjacent blocks can be used as a concealed coded block. In

another approach known as maximally smooth recovery [57], a limited number of DCT coefficients are estimated to provide the smoothest connection with the boundary pixels of the spatially adjacent blocks. In general, frequency-domain algorithms usually interpolate only the low-frequency transformation coefficients. In sample-domain concealment, the sample values of a missing block are derived from the sample values of the neighbouring blocks. For example, Salama *et al.* [58] proposed weighted pixel averaging, in which each pixel value in a MB to be concealed is formed as a weighted sum of the closest boundary pixels of the selected adjacent MBs. Wang *et al.* proposed a spatial error concealment method by minimizing the first-order derivative-based smoothness measure [59]. To suppress the induced blurring effect, the second-order derivatives were considered in [60]. Although such a smoothness constraint achieves good results for the flat regions, it may not be satisfied in the areas with high frequency edges. To resolve this issue, an edge-preserving algorithm was proposed to interpolate the missing pixels [61]. In [62], smooth and edge areas were effectively recovered based on selective directional interpolation. In [63], an orientation adaptive interpolation scheme derived from the pixel wise statistical model was proposed. In addition, a spatial error concealment method based on a Markov random field (MRF) was proposed in [64].

### 2.5.2.2 Temporal Error Concealment

The basic idea of temporal error concealment is to estimate the motion vector of a lost block. A simple strategy is to use a zero motion vector or a median of the motion vectors in the neighboring blocks. Chen *et al.* [65] proposed a side match criterion taking advantage of the spatial contiguity and inter-pixel correlation of image to select the optimal replacement among the motion vectors of spatially contiguous candidate blocks. The well-known boundary matching algorithm (BMA) proposed in [66] selected the motion vector that minimizes the total variation between the internal boundary and the external boundary of the reconstructed block as the best one to recover the corrupted block. There are also some sophisticated algorithms to obtain better replacements for the corrupted blocks. For example, a vector rational interpolation scheme [67], a bilinear motion

field interpolation algorithm [68], a Lagrange interpolation algorithm [69], and a dynamic programming algorithm [70] were proposed for error concealment.

In the aforementioned error concealment algorithms, to conceal the damaged MB, its neighboring MBs need to be correctly received. However, in a practical wireless network or Internet, consecutive packet losses are quite common due to the traffic congestion, and it is possible that all or some of the MBs surrounding a damaged MB are also lost. Thus, all the packets of one frame are very likely to be corrupted by channel errors. In such a scenario, the above mentioned error concealment methods will no longer perform well. An underlying solution to tackle this problem is to use the error concealment for the whole picture loss. A motion vector extrapolation (MVE) has been presented in [71] to combat the whole picture loss. It can extrapolate the motion vectors of the damaged MB from the last received picture and estimate the overlapped areas between the damaged MB and the motion extrapolation one. Based on the MVE method, Chen *et al.* proposed a pixel-based MVE (PMVE) method by extending the MVE method to the pixel level [72]. In order to solve the inaccurate problem of PMVE, Yan *et al.* [73] proposed a hybrid MVE (HMVE) method, which uses not only the extrapolated MVs of the pixels but also the extrapolated MVs of the block. To tackle the whole picture loss problem in H.264/SVC, Ji *et al.* [74] proposed a novel error concealment algorithm by utilizing the motion information of the co-located blocks in the temporally neighbouring previous and subsequent pictures.

### 2.5.3 Interactive Error Control

In the error-resilient techniques presented so far, the encoder and decoder operate independently as far as combating transmission errors is concerned. However, in certain practical applications, when a feedback channel can be set-up from the decoder to the encoder, the decoder can inform the encoder about which part of the transmitted information is corrupted by channel errors, and the encoder can avoid the use of the damaged area as a reference for coding of the future frames accordingly to suppress the effect of such errors. Usually, these techniques that automatically adjust the encoder operations based on the feedback information

from the decoder can reduce the coding gain loss, at the expense of increased complexity.

### 2.5.3.1 Reference Picture Selection

One way of taking advantage of an available feedback channel is to employ reference picture selection (RPS) [75]. If the encoder learns through a feedback channel about damaged parts of a previously coded frame, it can choose such a reference picture for inter prediction that is known to be correct and available based on the feedback. This requires that the encoder and decoder both store multiple past decoded frames. Information about the reference picture to be used is transmitted in the bitstream. Note that using RPS does not always necessarily bring extra delay in the encoder. The encoder does not have to wait for the arrival of the feedback information about the previous frame to code a current frame. Instead, it can choose to use the frame before the damaged frame as a reference. For example, when encoding the frame  $n + d$ , if the information about the damaged frame  $n$  does not arrive at the encoder, the decoder can select frame  $n - 1$  as the reference frame to code frame  $n + d$ . In this case, even if there are some errors between  $n + 1$  to  $n + d - 1$ , error propagation will be stopped from frame  $n + d$  onwards. Generally, the above discussed reference picture selection that reacts to receiver feedbacks by avoiding the use of notified loss-affected past frames as reference is called reactive RPS [76]. Another RPS alternative is to proactively use a reference picture with large prediction distance for error resilience based on some cost criterion [77], which is termed proactive RPS in this thesis. In proactive RPS, the encoder can first develop a method to model distortion in a candidate reference block or frame taking into account loss, and then choose the reference block or frame that leads to the smallest rate-distortion cost. Reactive RPS incurs a higher bit overhead only when needed, but suffers error propagation of up to one round-trip time at the decoder. Proactive RPS incurs constant overhead regardless of whether there are actually losses, and should only be applied preferentially to more important parts of the video. Further, the proactive RPS can be combined with the feedback further improving the performance.





# Chapter 3

## Error-Resilient Multi-view Video Coding Using Wyner-Ziv Techniques

### 3.1 Introduction

Multi-view video systems have become increasingly popular due to rapid uptake of interactive multimedia applications such as 3-D television, teleconference, surveillance and wireless sensor networks [78]. A multi-view video sequence can provide different perspective views of the same scene, offering interactivity as well as 3-D perception, which is the important input signal for high quality autostereoscopic display. Due to the huge increase in data volume with the number of the views, multi-view video technology has become an active research area focusing on both compression efficiency for storage and error resilience for transmission of multi-view video data. Several multi-view video compression techniques have been proposed in recent years [79]. The state-of-the-art standard for multi-view video coding is the MVC extension of H.264/AVC [80]. MVC employs motion and disparity estimation to fully explore both temporal and inter-view correlation to enable high compression efficiency. It also support backward compatibility with existing legacy system by structuring the MVC bit stream to include a base view. Due to the high quality encoding capability and support for backward compatibility, the MVC extension of H.264 was selected by the Blu-Ray Disc Association

as the coding format for 3-D video. As an amendment to H.264/AVC to support certain 3-D applications, MVC was originally designed for compressing multi-view video data. However, MVC can also be used directly in coding multi-view depth data by taking depth sequences as grayscale video.

Although inter-view prediction employed in MVC does improve the coding performance considerably compared to simulcast video coding, the compressed multi-view video signal is extremely sensitive to transmission error with regard to the delivery of 3-D video. In the case of packet-switched networks, packets may be discarded due to buffer overflow at intermediate nodes of the network, or may be considered lost due to long queuing delays. When this kind of packet loss happens to the predictive video streaming, a reconstruction error occurs. Further, such transmission errors from the current frame will propagate to the future frames along the motion compensation prediction path, and degrade video quality. As for transmission of multi-view video data, due to the joint design of motion and disparity compensation prediction, the error propagation problem will become more severe. If an error occurs in a frame of one view and cannot be effectively corrected, the transmission error will not only propagate to the subsequent frames in current view, but also spread to other dependent views through the disparity-compensated inter-view prediction, and thus an abrupt degradation of the total received multi-view video quality. This problem is exacerbated in wireless channels, where packet losses are far more frequent and bursty than in wire-line networks. Therefore, error-resilient multi-view video steaming has become a critically important topic of 3-D video research.

In this chapter, we will introduce an error-resilient framework for multi-view video communications based on the principle of distributed video coding theory, specifically solving the research problem 1 as defined in Chapter 1. The main component is a joint source/channel coding approach that selects the appropriate amount of WZ bits to insert into the source multi-view video bit stream, using knowledge of packet loss rate, the correction capability of WZ encoding, and the decoder error concealment. In the following, we first discuss a few key related work, and then outline our contributions.

## 3.2 Related Work

A lot of efforts have been made to protect the quality of video sequences with single view against channel errors. The widely used conventional error resilience mechanisms include automatic repeat request (ARQ) [81], forward error correction (FEC) [82], inserting more intra-coded macro-blocks [83], robust motion estimation [84], redundant slices [85], and so on. However, only a small number of algorithms have been proposed for robust multi-view video transmission.

In the area of stereoscopic video transmission, a rate distortion (R-D) optimization method for error-resilient stereoscopic video coding with inter-view refreshment was proposed in [86]. Tan *et al.* [87] introduced an end-to-end R-D model of stereo video that achieved optimal encoder bit rates and unequal error protection rates. In these works, error control was only considered for 3-D video with the left and right views, without regards to the characteristics of multi-view video signals. For transmitting generic multi-view video over packet lossy networks, Song *et al.* [88] proposed an efficient concealment algorithm based upon adaptive intra-view and inter-view correlation to provide reliable concealment results. Liu *et al.* [142] presented an error concealment method that incorporates the redundancy between the inter-views to generate the concealment frame. These two schemes mainly focus on recovering the lost block or frame at the decoder according to the received information, but they may not perform well if the neighbouring blocks or frames are also lost during transmission. Zhou *et al.* [90] developed a recursive mathematical model to estimate the expected channel-induced distortion considering both temporal and inter-view dependencies at both the frame and sequence levels, but do not include any error protection scheme. Dissanayake *et al.* [91] presented an error-resilient method by incorporating the disparity vectors into MVC so as to generate a redundant data stream.

More recently, motivated by emerging applications in low-complexity video coding and robust video transmission, distributed video coding has received increased research attention. It is an alternative paradigm for video compression based on the information theoretical results established by Slepian and Wolf for distributed lossless coding and Wyner and Ziv for lossy coding. The practical distributed video coding scheme, WZ coding, employs lossy video compression

with decoder side information where the temporal correlation of a video signal is exploited in the decoding phase rather than in the encoding one. In this way, the classic motion estimation process is no longer performed at the encoder, with a significant reduction in the computational complexity of the encoder. On the other hand, given that no prediction loop is used in the encoding phase, the distributed coding scheme has a good error resilience to transmission errors. Therefore, based on the intrinsic error resilience of WZ encoding, some robust single video coding schemes have been proposed in the past few years. Sehgal *et al.* [96] attempted to prevent temporal error propagation by periodically inserting WZ protected frames. Zhang *et al.* [97] utilized a unified WZ codec to accomplish joint source-channel coding through not only exploiting the correlation between the WZ frame and its side information (SI), but also protecting against channel errors. Zhang *et al.* [98] proposed a joint source-channel R-D optimized mode selection algorithm with WZ coding to optimize the overall R-D performance. Furthermore, the WZ coding technique has been applied in the protection of the region of interest area [99], multiple description coding [100], parity bits packetization strategy [101], etc. However, the above schemes are concerned with the using of WZ coding for single view video transmission, which are not directly applicable to multi-view video transmission because the inter-view correlation is not taken into consideration.

Generally, distributed video coding can be naturally extended to multi-view video scenario, as it can exploit both the redundancies already present in mono-view video and the inter-view correlation in multi-view video data. However, prior work on distributed compression of multi-view video has always been focused on compression performance by removing redundancies present in overlapping camera views, e.g., inter-view side information generation [102] and modelling correlation statistics [103], and using WZ coding to improve the robustness of multi-view video transmission is still relatively rare [104]. In 2010, Yeo and Ramchandran [105] presented a PRISM architecture based distributed multi-view video coding framework with disparity search for robustly delivering multi-view video data, which is referred to as PRISM-DS. For the PRISM-DS approach, the encoder at each camera did not have access to views observed from other cameras

and the temporal or inter-view redundancy was exploited to generate the SI at the WZ decoder.

However, due to the inherent difficulty in estimating the correlation model between the original frame to be coded and the SI, it suffered from much lower coding performance than the standard MVC. Moreover, this method is not compatible with the MVC standard.

### 3.3 Contributions of This Chapter

In this chapter, unlike the above mentioned PRISM-DS framework, we try to improve the robustness of standard multi-view encoded video by adding redundant information encoded according to WZ encoding principle. Specifically, at the multi-view encoder, the key frames of the odd views are protected with WZ coding in addition to being coded by the efficient MVC encoder. And the SI at the WZ decoder is the corresponding reconstructed key frames of the odd views after error concealment (EC). In this case, compared to the regular multi-view decoder reconstruction, the quality of the reconstructed key frames of the odd views can be improved by correcting channel errors after WZ decoding, and consequently the overall error resilience performance improvement could be achieved. The proposed error-resilient MVC scheme is backwards-compatible in the sense that a user with only the multi-view video decoder can ignore the extra WZ encoding data and decode the primary bit-stream. To the best of the authors' knowledge, this is the first one able to improve the multi-view 3-D video error resilience performance by using WZ encoding technique in a backwards-compatible fashion.

For WZ-based error-resilient multi-view video coding, the WZ encoded bits are the parity bits generated to correct channel errors occurred in the SI, i.e., the key frames of the odd views. In other words, the bit rate for the auxiliary bit-stream encoded by WZ principle should be determined by the possible transmission distortion in the key frames of the primary bit-stream. Therefore, one of our major contributions is the proposal of a recursive transmission distortion model at the transform domain to estimate the expected distortion of multi-view

video data. According to the propagating behavior of transmission errors, the proposed distortion model relates channel distortion in the current frame to that in the temporal reference frame or neighboring view frame, and allows for any motion-compensated and disparity-compensated EC method to be employed by the multi-view decoder. Moreover, the proposed model considers the error recovery for the key frames of the odd views with WZ compensation.

On the other hand, for the auxiliary bit-stream, existing WZ video coding solutions in the literature use a feedback channel based on decoder rate control strategy to adjust the bit rate. Typically, several iterative decoding operations may be needed to decode the video, especially when the quality of the SI is poor [106]. Aiming to considerably reducing the decoding complexity and delay, we propose a bit rate control strategy at the encoder to estimate the amount of parity information needed to achieve a target decoded quality based on the proposed transmission distortion model. The estimated bit rate of WZ encoding required to correct transmission errors can also be treated as a measure of the amount of the auxiliary bit-stream.

### 3.4 Proposed Error-Resilient Scheme Using Embedded WZ Description

Based on both temporal and inter-view redundancy of multi-view video data, researchers have proposed many different prediction structures. In the standard MVC structure adopted by the Joint Video Team (JVT) [107], the hierarchical B picture is not only used in the temporal prediction of each view, but also applied in inter-view prediction for key and non-key frames. This hybrid temporal and inter-view hierarchical B picture prediction can achieve the highest coding efficiency but cause a very high computational complexity to the encoder. So in some video conferencing applications with tight latency constraint, this prediction structure is unsuitable. In the proposed WZ-based error-resilient MVC scheme of this work, the multi-view KS\_IPP prediction structure developed in [108] is chosen. Compared to the MVC standard prediction structure, the KS\_IPP coding structure can lead to about 40% - 43% reduction of encoding complexity, while the

coding efficiency decreases just slightly [108]. The multi-view KS\_IPP structure is illustrated in Figure 3.1, for a multi-view video sequence with eight synchronized video cameras and a GOP length of 8, where  $V_n$  and  $T_n$  denote the captured views and time frames, respectively. The first picture of each view is the key picture and the so-called key frames are coded in regular intervals, which are depicted in grey in Figure 3.1. The KS\_IPP structure shows the extra correlation in addition to existing spatio-temporal correlation for multi-view video. For intra-view compression, temporal prediction with hierarchical B pictures is employed to exploit temporal dependency. In the inter-view direction, in order to reduce the encoding complexity, the traditional IPPP prediction structure is used for the key pictures to remove inter-view redundancy among different views, and this is motivated by the fact that the majority of gains in inter-view direction are obtained using prediction at the key frame positions. However, since the IPPP inter-view prediction belongs to the multi-view baseline profile and the hierarchical B inter-view prediction pertains to the multi-view high profile [109], the KS\_IPP prediction structure can be applied with a standard-compliant MVC coded bit-stream.

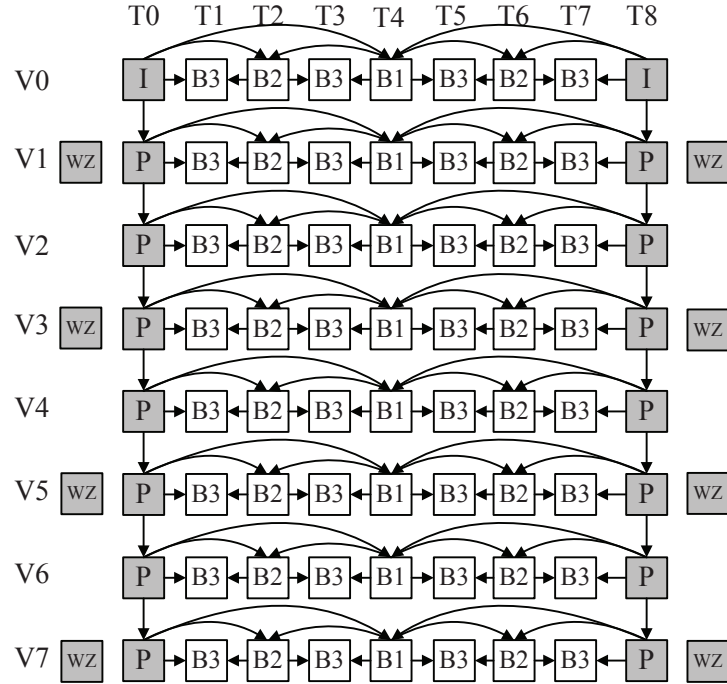


Figure 3.1: Encoding structure for WZ coding embedded multi-view video with the key frames of the odd views protected.

When the MVC bit-stream compressed by the KS\_IPP structure is transmitted over the error-prone channel, if an error occurs in non-key pictures during transmission, it propagates along the temporal direction only. By contrast, errors in key pictures will propagate temporally within the same view and also to frames in the adjacent view. Obviously, errors occurred in a key picture cause more significant degradation in the decoded picture quality than errors occurred in a non-key picture. Consequently, the key frames should be better protected. Intuitively, if every key frame is protected with embedded WZ encoding, transmission errors can be fully eliminated in the key frames, at the expense of the coding efficiency of multi-view video. In an effort to achieve a good balance between coding efficiency and error resilience, the key frames of the odd views are assumed to be WZ coded in the proposed encoding structure shown in Figure 3.1. Since the key frames of the even views or odd views have the same error propagation behavior facing the transmission errors, one can also assume that the key frames of the even views are WZ coded to the same effect. It should be noted that in other multi-view coding structures, error propagation may occur in non-key frames along inter-view direction. However, even in this case, the key frames protected by embedded WZ encoding can still provide better error robustness because the non-key frames are predicted by the key frames in each view.

Based on the propagating behavior of transmission errors caused by random packet losses, a WZ-based error-resilient MVC scheme shown in Figure 3.2 is proposed. At the transmitter, on one hand, the multi-view video bit-stream generated by motion-compensated and disparity-compensated predictive coding is transmitted as the primary stream. On the other hand, to prevent the prediction mismatch between the encoder and decoder, we use the reconstructed anchor frames of the odd views at multi-view encoder as the original information to feed into the WZ encoder. The WZ bit-stream is sent alongside the primary bit-stream as the auxiliary stream. As stated earlier, the bit rate of the auxiliary stream is dependent on the channel distortion of the anchor frames at the odd views. As a consequence, it is necessary to model the transmission distortion for multi-view video in the transform domain before estimating the bit rate of WZ encoding.



However, transmission distortion estimation requires the prior knowledge of the packet loss rate and the inter-view concealment strategy employed by the decoder [110], as showed in the “Transmission Distortion Analysis” module in Figure 3.2. Then based on the transmission distortion model, the WZ bit rate can be computed making use of previously decoded bits and the correlation noise model of the residual between the original information and its SI.

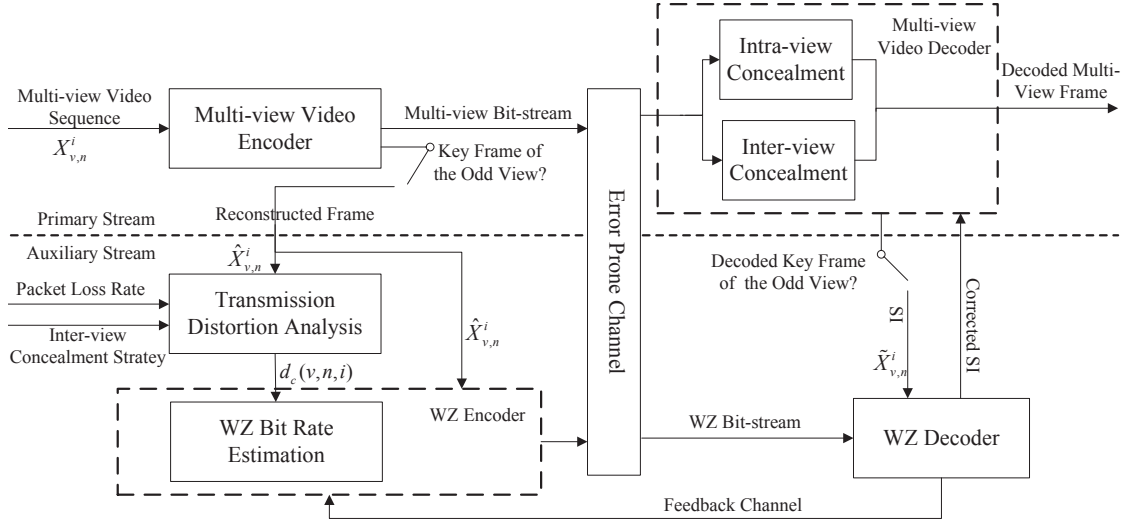


Figure 3.2: WZ-based error resilience multi-view video coding scheme against random packet losses.

At the receiver, although the multi-view decoded video frames are possibly erroneous due to packet drops, they are still highly correlated to the correct reconstruction. So the reconstructed key frames of the odd views after disparity-compensated concealment can be used as the SI by the WZ decoder. Then the auxiliary stream carrying parity bits is used to correct the SI, and the key frames of the odd views are able to be recovered in severe error conditions. Hence, the temporal and inter-view error propagation could be eliminated. After that, the decoded key frames of the odd views refined by WZ decoding are written back to the multi-view frame memory to serve as the reference frames for temporal and inter-view prediction. From the above analysis, it can be inferred that the multi-view video stream will exhibit superior transmission robustness improvement by embedded WZ encoding to correct channel errors in the multi-view decoded key frames of the odd views. With the proposed WZ compensation, a possible mismatch between the encoder and decoder side will mainly be the reconstructed

errors on the key frames of the even views and all the non-key frames. However, since the key frames of the odd views are fully recovered by embedded WZ encoding, a drastic reduction in picture quality can be avoided. In addition, since the proposed error-resilient algorithm just protects the frames which are more vulnerable to transmission errors, it can also be applied to other multi-view video coding structures such as KS\_PIP or KS\_IBP [108], provided that the transmission distortion estimation approach is properly adjusted to work with the desired prediction structure.

### 3.5 Transmission Distortion Model

To better describe the proposed transmission distortion model, some notations is first defined for the following derivations. Let  $X_{v,n}^i$  be the unquantized value of the DCT coefficient  $i$  in frame  $n$  at view  $v$ , and denote  $\hat{X}_{v,n}^i$  and  $\tilde{X}_{v,n}^i$  as the reconstructed values of  $X_{v,n}^i$  at the encoder and decoder, respectively. That is,  $\hat{X}_{v,n}^i$  and  $\tilde{X}_{v,n}^i$  are the quantized representation of  $X_{v,n}^i$  at the encoder and decoder, respectively. In the presence of transmission error,  $\hat{X}_{v,n}^i$  and  $\tilde{X}_{v,n}^i$  are normally different. Assume that the DCT coefficient  $j$  in frame  $r_{ref}$  at view  $v_{ref}$  is taken as the reference for DCT coefficient  $i$ . Let  $\hat{e}_{v,n}^i$  be the quantized transform coefficient of prediction residual signal, and thus we have  $\hat{X}_{v,n}^i = \hat{X}_{v_{ref},r_{ref}}^j + \hat{e}_{v,n}^i$ . If the current data packet is correctly received, the decoder reconstruction of  $X_{v,n}^i$  is  $\tilde{X}_{v,n}^i = \tilde{X}_{v_{ref},r_{ref}}^j + \hat{e}_{v,n}^i$ . When the DCT coefficient  $i$  is lost in transmission, the decoder conceals this error with  $\tilde{X}_{v_{ec},r_{ec}}^k$ , and  $k$  stands for the concealed DCT coefficient in frame  $r_{ec}$  at view  $v_{ec}$ . Moreover, as the anchor frames of the odd views are additionally protected by embedded WZ encoding, the concealed coefficient needs to be used as the SI for WZ decoding, if the lost coefficient belongs to an anchor frame of an odd view. The WZ-coded data will then be decoded by the auxiliary decoder for the correct reconstruction, preventing the prediction mismatch between the multi-view encoder and decoder. In this case, after WZ decoding,  $\hat{X}_{v,n}^i = \tilde{X}_{v,n}^i$  approximately holds.

Suppose a group of MBs in a frame forms a slice, and each slice has its own header, and is carried in a separate transport packet. We also assume that each

packet is independently lost with probability  $p$ . In this setting, the loss rate of a pixel equals the packet loss rate. Then the formula of the expectation of the channel-induced transmission distortion  $d_c(v, n, i)$  with respect to the probability of packet loss can be derived as follows

$$\begin{aligned}
d_c(v, n, i) &= E \left\{ (\hat{X}_{v,n}^i - \tilde{X}_{v,n}^i)^2 \right\} \\
&= (1-p) E \left\{ (\hat{X}_{v,n}^i - (\tilde{X}_{v_{ref}, r_{ref}}^j + \hat{e}_{v,n}^i))^2 \right\} + p E \left\{ (\hat{X}_{v,n}^i - \tilde{X}_{v_{ec}, r_{ec}}^k)^2 \right\} \\
&= (1-p) E \left\{ (\hat{X}_{v_{ref}, r_{ref}}^j - \tilde{X}_{v_{ref}, r_{ref}}^j)^2 \right\} \\
&\quad + p E \left\{ (\hat{X}_{v,n}^i - \hat{X}_{v_{ec}, r_{ec}}^k + \hat{X}_{v_{ec}, r_{ec}}^k - \tilde{X}_{v_{ec}, r_{ec}}^k)^2 \right\} \\
&= (1-p) E \left\{ (\hat{X}_{v_{ref}, r_{ref}}^j - \tilde{X}_{v_{ref}, r_{ref}}^j)^2 \right\} \\
&\quad + p E \left\{ (\hat{X}_{v,n}^i - \hat{X}_{v_{ec}, r_{ec}}^k)^2 \right\} + p E \left\{ (\hat{X}_{v_{ec}, r_{ec}}^k - \tilde{X}_{v_{ec}, r_{ec}}^k)^2 \right\} \\
&= (1-p) d_{c\_ref}(v_{ref}, r_{ref}, j) + p (d_{ec\_r}(v, n, i) + d_{c\_ec}(v_{ec}, r_{ec}, k))
\end{aligned} \tag{3.1}$$

where  $d_{ec\_r}(v, n, i)$  is the distortion between the reconstructed value and the error-concealed DCT coefficient at the encoder, which can be readily computed by simulating error concealment at the encoder;  $d_{c\_ref}(v_{ref}, r_{ref}, j)$  denotes the channel-induced distortion of the reference DCT coefficient; and  $d_{c\_ec}(v_{ec}, r_{ec}, k)$  represents the channel-induced distortion of the concealed DCT coefficient. It should be taken note that  $d_{ec\_r}(v, n, i)$  and  $d_{c\_ec}(v_{ec}, r_{ec}, k)$  are uncorrelated, and  $d_{c\_ref}(v_{ref}, r_{ref}, j)$  and  $d_{c\_ec}(v_{ec}, r_{ec}, k)$  can be recursively calculated from the reference and concealed frames, respectively.

As can be observed from (3.1), On one hand, even when the current data packet is received free of errors, the encoder and decoder may still be out of synchronization due to the past distortion caused by the reference frame. On the other hand, in the case of channel errors, the channel distortion is divided into two terms with the first term being the distortion caused solely by the EC algorithm, whereas the second term is attributed to the channel-induced distortion of the concealed frame. Note that the transmission distortion of the key frames of view 0 can be directly derived without considering error propagation because they are

typically coded as intra frames. Then, the transmission distortion of the following frames along the temporal direction can be computed as a weighted average of the error propagation distortion of the reference frames, based upon which the concerned frame is predicted in accordance with the prediction structure of hierarchical B pictures. The transmission distortion along the view direction can be computed as follows.

### 3.5.1 Transmission Distortion of DCT Coefficients in the Key Frames of the Odd Views

In the inter-view direction, the anchor frame of an odd view is predicted by its counterpart of the previous view through exploiting inter-view dependence based upon disparity compensation prediction. In this case, the propagated error mainly comes from the neighboring even view. Therefore, the expected transmission distortion of DCT coefficient  $i$  can be rewritten as

$$\begin{aligned} d_c(v, n, i) = & (1 - p)d_{c\_ref}(v - 1, n, j) \\ & + p(d_{ec\_r}(v, n, i) + d_{c\_ec}(v_{ec}, r_{ec}, k)). \end{aligned} \quad (3.2)$$

If each DCT coefficient value of the concealed picture is copied from the corresponding DCT coefficient of the previous view frame, the error-propagated distortion of the concealed DCT transform coefficient in (3.2) can be expressed as

$$d_{c\_ec}(v_{ec}, r_{ec}, k) = d_{c\_ec}(v - 1, n, i). \quad (3.3)$$

In practice, any sophisticated inter-view EC scheme can be used to replace this naive scheme and be accounted for in the algorithm.

For an intra-coded DCT coefficient, no error is propagated from the adjacent view frame, and only the transmission distortion is due to packet drops

$$d_c(v, n, i) = p(d_{ec\_r}(v, n, i) + d_{c\_ec}(v_{ec}, r_{ec}, k)). \quad (3.4)$$

As mentioned earlier, the amount of distortion introduced in (3.2) or (3.4) of the odd views can be compensated by WZ decoding.

### 3.5.2 Transmission Distortion of DCT Coefficients in the Key Frames of the Even Views

When the primary and auxiliary bit stream is transmitted to the receiver side, it is assumed that the auxiliary data are received without transmission errors. This assumption is based on the fact that the auxiliary data is only accounted for a small portion of the total bit stream, and an adequate amount of protection can be used to ensure that the auxiliary data is received correctly. And this assumption also has been adopted in the literature [97], [98], [111], [112]. Under this assumption, WZ decoding can always succeed with the aid of the SI, which is the error-concealed multi-view decoded DCT coefficients of the key frames of the odd views. With successful WZ decoding, the decoded DCT coefficients are identical to their reconstructed values at the multi-view encoder, thereby eliminating the accumulated effect of channel errors up to the key frame of an odd view. Therefore, the error-propagated distortion of the reference DCT coefficient and the error-concealed DCT coefficient from the odd views can be assumed to be zero, regardless of the current coefficient is inter-coded or otherwise. The transmission distortion of DCT coefficient  $i$  in the key frame of an even view can be estimated as

$$d_c(v, n, i) = p(d_{ec-r}(v, n, i)). \quad (3.5)$$

As can be seen from the above derivation of transmission distortion, the transmission distortion of a DCT coefficient is taken as the sum of several distortion items, which is different from the conventional recursive optimal per-pixel estimation (ROPE) method in [113] involved in keeping track of the first and second moments of the reconstructed pixel value. Since ROPE calculates the first and second moments of the decoded value for each pixel, it requires intense computation and is very sensitive to the approximation error caused by pixel averaging operations. In contrast, the proposed distortion model can suppress such approximation error. On the other hand, the proposed transmission distortion model in this chapter considers channel-induced distortion caused by disparity compensation prediction and disparity compensation error concealment at the transform domain. So it is more suitable to handle sophisticated error propagation for multi-view video transmission than the classic single view distortion

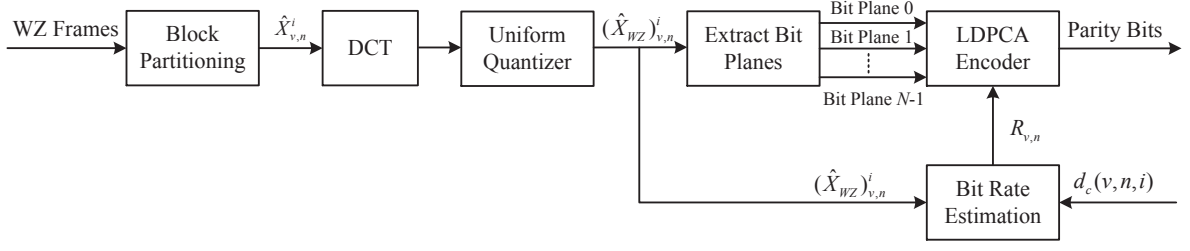


Figure 3.3: Block diagram of the WZ encoder.

models in [110], [113], [114].

### 3.6 Bit Rate Estimation for WZ Coding

In order to better represent the WZ bit rate estimation method, a more detailed block diagram of the WZ encoder is shown in Figure 3.3 to describe the data flow for the WZ encoder module in the dash box of Figure 3.2. In WZ encoding, a source frame is partitioned into blocks, and each block is transformed with DCT transform. DCT coefficients at the same position are grouped to form a coefficient band. Bit-planes are extracted from the quantized DCT coefficient bands and fed into the Slepian-Wolf coder which utilizes the low-density parity-check accumulated (LDPCA) [115] to generate the WZ bit-stream. The LDPCA codes which accumulate syndrome bits from conventional LDPC codes outperform the LDPC codes by a small margin with the advantage that different rates can be achieved without altering the generator matrix. Bit-planes are arranged in an increasing order with 0 corresponding to the least significant bit. At the WZ decoder, LDPCA decoding starts with incremental syndrome requests for each bit-plane. After all bit-planes are LDPCA decoded, the final reconstructed DCT coefficients are obtained by using minimum mean square error based reconstruction as in [116]. Following that, the reconstructed WZ frame is then obtained by applying the inverse DCT.

With an view to reduce decoding complexity and delay in more practical distributed video coding, we propose an encoder rate control strategy when encoding the key frames of the odd views. It is easy to see that the original DCT transform coefficient  $(\hat{X}_{WZ})_{v,n}^i$  is available at the WZ encoder side, while its SI  $(\tilde{X}_{WZ})_{v,n}^i$  is available at the WZ decoder side. Note that  $(\hat{X}_{WZ})_{v,n}^i$

and  $(\tilde{X}_{WZ})_{v,n}^i$  are the DCT transform representation of  $\hat{X}_{v,n}^i$  and  $\tilde{X}_{v,n}^i$  after WZ encoding, respectively. Assume that the previously decoded source bit-planes are  $\{(\hat{X}_{WZ})_{v,n}^{i,N-1}, \dots, (\hat{X}_{WZ})_{v,n}^{i,t+1}\}$ , the corresponding bit-planes of the SI are  $\{(\tilde{X}_{WZ})_{v,n}^{i,N-1}, \dots, (\tilde{X}_{WZ})_{v,n}^{i,t+1}\}$ , where  $N$  is the number of the bit-planes and  $t \in [0, N-1]$ . When estimating the bit rate of each bit-plane, it is important to estimate the crossover probability between the source and SI bit-planes. Suppose that  $l$  and  $u$  represent a lower bound and an upper bound of the decoded quantization DCT coefficient, the conditional probability that  $(\tilde{X}_{WZ})_{v,n}^{i,t}$  equals 0 or 1 given  $(\hat{X}_{WZ})_{v,n}^{i,t}$  can be obtained as follows.

If the currently decoded bit-plane of  $(\hat{X}_{WZ})_{v,n}^{i,t}$  equals 0, we can obtain  $l$  and  $u$  of the currently decoded quantization symbol from the previously decoded bit-planes as follows

$$\begin{cases} l = \sum_{z=t+1}^{N-1} 2^z \times (\hat{X}_{WZ})_{v,n}^{i,z} \\ u = 2^t - 1 + \sum_{z=t+1}^{N-1} 2^z \times (\hat{X}_{WZ})_{v,n}^{i,z} \end{cases} \quad (3.6)$$

where  $l$  and  $u$  correspond to the cases where all the subsequent un-decoded  $t$  bit-planes are 0 and 1, respectively.

In an ideal situation, the previously decoded bit-planes can be transmitted to the encoder through a feedback channel, and these decoded bit-planes could be further used for rate control for the following bit-planes. However, in this case, it will result in high decoding complexity and delay. Moreover, in practice, no feedback channel would have sufficient bandwidth to allow for the decoded bit-planes to be transmitted from the decoder to the encoder. In this algorithm, we assume that the previous bit-planes are decoded successfully, and are exactly the same as the corresponding source bit-planes at the WZ encoder. In practice, as some previous bit-planes might be decoded erroneously, this assumption will lead to mismatch between the WZ encoder and decoder. The mismatch will first affect the bounds  $l$  and  $u$  in (3.6) and (3.7), and then lead to errors in the estimation of the crossover probability through (3.11), and finally impact on the estimation accuracy of the conditional entropy in (3.13). The equations of (3.7), (3.11), and (3.13) will be shown later. In other words, the mismatch may lead to the parity rate underestimation or overestimation in the proposed system. However, since

the objective of the auxiliary WZ bit rate allocation is to ensure all the bit-planes are decoded correctly, i.e., the previously decoded bit-planes are the same as the source bit-planes, this mismatch is relatively small and often neglected in the literature [98], [117].

This claim can also be confirmed by our experimental results. The following Figure 3.4 shows the entropy estimation results of the test sequence. In our simulations, the first 150 frames of each sequence and the five most significant bit-planes of each coefficient band are considered. In Figure 3.4, the x-axis is the estimated entropy with the proposed method at the encoder side, and the y-axis is the actual entropy obtained at the decoder side with the standard WZ decoding. The results show that the estimation precision is fairly high.

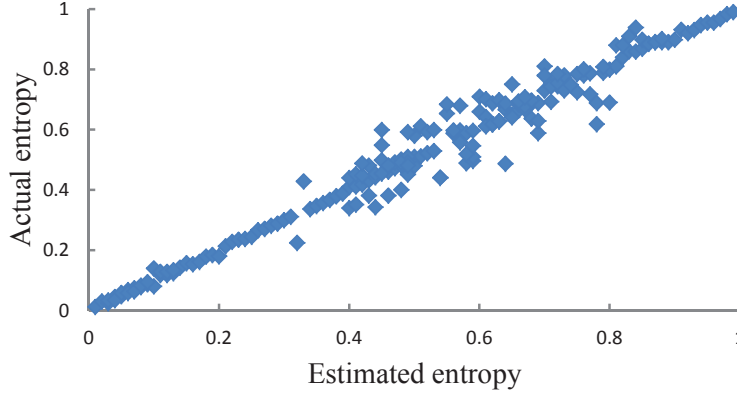


Figure 3.4: Estimation precision of conditional entropy.

Similarly, if the currently decoded bit-plane  $(\hat{X}_{WZ})_{v,n}^{i,t}$  equals 1,  $l$  and  $u$  are computed as follows

$$\begin{cases} l = 2^t + \sum_{z=t+1}^{N-1} 2^z \times (\hat{X}_{WZ})_{v,n}^{i,z} \\ u = 2^{t+1} - 1 + \sum_{z=t+1}^{N-1} 2^z \times (\hat{X}_{WZ})_{v,n}^{i,z}. \end{cases} \quad (3.7)$$

For WZ video coding in the transform domain, a Laplacian distribution is used to model the statistical correlation between the same DCT band in the original frame and the corresponding SI frame. Suppose that  $(\tilde{X}_{WZ})_{v,n}^y$  represents the DCT coefficient  $y$  in frame  $n$  at view  $v$ , which can be viewed as a random variable to the WZ encoder. Then, given  $(\hat{X}_{WZ})_{v,n}^i$  at the WZ encoder,  $(\tilde{X}_{WZ})_{v,n}^y$  follows



the following distribution

$$\begin{aligned} & f_{(\tilde{X}_{WZ})_{v,n}^y | (\hat{X}_{WZ})_{v,n}^i} ((\tilde{X}_{WZ})_{v,n}^y | (\hat{X}_{WZ})_{v,n}^i) \\ &= \frac{\alpha_{v,n}^i}{2} e^{-\alpha_{v,n}^i |(\hat{X}_{WZ})_{v,n}^i - (\tilde{X}_{WZ})_{v,n}^y|} \end{aligned} \quad (3.8)$$

where  $\alpha_{v,n}^i$  is the Laplacian parameter defined by

$$\alpha_{v,n}^i = \sqrt{\frac{2}{(\sigma_{v,n}^i)^2}} \quad (3.9)$$

where  $\sigma_{v,n}^i$  is the variance of the residual DCT coefficient between the WZ and SI frames that can be computed based upon the expected transmission distortion given in Section 3.5 as follows

$$\begin{aligned} (\sigma_{v,n}^i)^2 &= E \left\{ ((\hat{X}_{WZ})_{v,n}^i - (\tilde{X}_{WZ})_{v,n}^i)^2 \right\} \\ &= E \left\{ (\hat{X}_{v,n}^i - \tilde{X}_{v,n}^i)^2 \right\} \\ &= d_c(v, n, i). \end{aligned} \quad (3.10)$$

A binary symmetric channel (BSC) with a crossover probability of  $p_{v,n}^{i,t}$  is assumed to connect  $(\hat{X}_{WZ})_{v,n}^{i,t}$  and its SI  $(\tilde{X}_{WZ})_{v,n}^{i,t}$ .  $p_{v,n}^{i,t}$  can be derived according to the following formula

$$\begin{aligned} p_{v,n}^{i,t}((\tilde{X}_{WZ})_{v,n}^{i,t} = \theta | (\hat{X}_{WZ})_{v,n}^{i,t}) &= \int_{l \times \Delta}^{u \times \Delta} f_{(\tilde{X}_{WZ})_{v,n}^y | (\hat{X}_{WZ})_{v,n}^i} ((\tilde{X}_{WZ})_{v,n}^y | (\hat{X}_{WZ})_{v,n}^i) d((\tilde{X}_{WZ})_{v,n}^y) \\ &= \int_{l \times \Delta}^{u \times \Delta} \frac{\alpha_{v,n}^i}{2} \exp(-\alpha_{v,n}^i |(\hat{X}_{WZ})_{v,n}^i - (\tilde{X}_{WZ})_{v,n}^y|) d((\tilde{X}_{WZ})_{v,n}^y) \end{aligned} \quad (3.11)$$

where  $\theta \in \{0, 1\}$ ,  $\Delta$  corresponds to the quantization step size in WZ encoding, and  $(\tilde{X}_{WZ})_{v,n}^y$  is assumed to lie in the interval  $[l \times \Delta, u \times \Delta]$ .

More specifically,  $p_{v,n}^{i,t}$  can be calculated as

$$\begin{aligned} & p_{v,n}^{i,t}((\tilde{X}_{WZ})_{v,n}^{i,t} = \theta | (\hat{X}_{WZ})_{v,n}^{i,t}) \\ &= \begin{cases} p_{v,n}^{i,t}((\tilde{X}_{WZ})_{v,n}^{i,t} = 0 | (\hat{X}_{WZ})_{v,n}^{i,t}), & \text{if } (\hat{X}_{WZ})_{v,n}^{i,t} = 1 \\ p_{v,n}^{i,t}((\tilde{X}_{WZ})_{v,n}^{i,t} = 1 | (\hat{X}_{WZ})_{v,n}^{i,t}), & \text{if } (\hat{X}_{WZ})_{v,n}^{i,t} = 0. \end{cases} \end{aligned} \quad (3.12)$$

The minimum rate required to make  $(\hat{X}_{WZ})_{v,n}^{i,t}$  decodable at the decoder is the conditional entropy  $H((\tilde{X}_{WZ})_{v,n}^{i,t} | (\hat{X}_{WZ})_{v,n}^{i,t})$  which is a function of the conditional

probability  $p_{v,n}^{i,t}$

$$\begin{aligned} R(p_{v,n}^{i,t}) &= H((\tilde{X}_{WZ})_{v,n}^{i,t} | (\hat{X}_{WZ})_{v,n}^{i,t}) \\ &= -p_{v,n}^{i,t} \times \log(p_{v,n}^{i,t}) - (1 - p_{v,n}^{i,t}) \times \log(1 - p_{v,n}^{i,t}). \end{aligned} \quad (3.13)$$

Since WZ coding operates on a bit-plane basis, the encoding rate needs to be estimated for each bit-plane. As a result, each DCT band's bit rate  $R_{v,n}$  in frame  $n$  at view  $v$  can be estimated as follows

$$R_{v,n} = \sum_{t=0}^{N-1} \sum_{i=0}^{q-1} R(p_{v,n}^{i,t}) \quad (3.14)$$

where  $q$  is the number of transform coefficients contained in each DCT band. (3.14) is then used by the WZ encoder to determine the parity bit rate required to be transmitted to the WZ decoder. However, in the case of the encoder parity rate underestimation, a small amount of feedback for addition syndrome bits of the LDPCA code are allowed to ensure successful LDPCA decoding.

### 3.7 Complexity Analysis

The proposed error-resilient method introduces additional complexity for both the sender and receiver. However, the additional computational cost is well justified by the impressive error resilience performance improvements achieved. Extensive experimental evidence about the improved performance is reported in the next section. Since the WZ bit stream is transmitted as the redundant information for the multi-view video data bit stream, i.e., the WZ coders and the multi-view video coders work independently, the complexity for the WZ coders and multi-view video coders is analysed separately. As for the receiver side, because the SI for WZ decoder is the error concealed key frames of the odd views and error concealment is a preliminary post-processing technique employed at the multi-view video decoder, it can be regarded that no additional complexity is incurred at WZ and multi-view video decoder.

In contrast, the critical complexity increase originates in the transmission distortion and WZ bit rate estimation at the sender. In the distortion estimation module of the multi-view video encoder, for each DCT coefficient in the key

frames of the odd views, we need six additions and multiplications to compute the expected transmission distortion, which can be observed from (3.1) to (3.5). A DCT coefficient belonging to the key frames of the even views requires one addition and two multiplications. Furthermore, the error concealment algorithm also has to be implemented at the encoder for each DCT coefficient. However, the assumed approach which uses the coefficient at the corresponding positions in the previous frame to conceal the missing DCT coefficient requires negligible additional complexity. More elaborated error concealment algorithms could result in additional encoder complexity.

On the other hand, for the bit rate estimation at WZ encoder, it can be observed from (3.6) and (3.7) that the  $t$  th bit-plane requires  $(N - t - 1)$  additions and multiplications to calculate the lower bound and upper bound when the value of bit-plane  $t$  equals 0. When the value of bit-plane equals 1,  $(N - t)$  additions and  $(N - t - 1)$  multiplications are needed. Furthermore, from (3.8) to (3.13), the WZ encoder performs an integral operation to compute the crossover probability, and performs two additional additions, two additional multiplications, and two logarithm operations to estimate the minimum bit rate. However, the aforementioned bit rate estimation proposal represents a modest complexity increment in terms of arithmetic operations. In our implementation, the hardware platform is a laptop computer equipped with 2.40 GHz Intel (R) Core (M) 2 Duo CPU and 3G memory running Microsoft Windows 7 Professional. Based on the simulation results, an average increase of only 4.6% in execution time with respect to the original WZ encoder. On the other hand, according to the power-rate-distortion (P-R-D) model in [118], with the complexity increment, the energy cost of the proposed error-resilient encoding methods will also increase. We measure the power consumption data on the laptop computer running a WZ encoder software. A Tektronix current probe is used to measure the current (in amps) in the circuit, while the voltage is held constant at  $V_c$  volts. To eliminate the effect of the power consumption by programs running at the background by the operating system, we first measure the current consumption  $I_{\text{idle}}$  when no other tasks are running. Then, the current  $I_0(t)$  with the WZ encoder program running is measured, where  $t$  represents the encoding time. The difference  $I(t) = I_0(t) - I_{\text{idle}}$  is taken as the

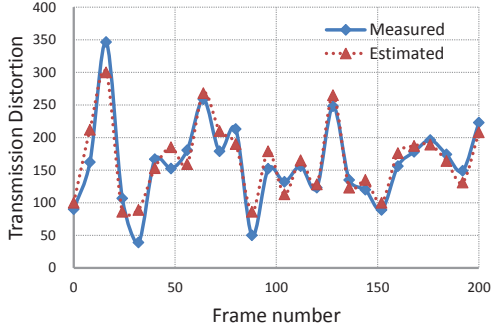
actual level for the WZ coder. For each video sequence,  $I_0(t)$  is recorded and the average energy is calculated as  $E = \int (I_0(t) - I_{\text{idle}}) V_c dt$  (joules). Through averaging all the test sequences, the average encoder energy cost of the proposed error-resilient algorithm is shown to increase only 10 mJ per frame compared with the original WZ coder. This demonstrates that the proposed WZ encoder still satisfies the primary requirement of low-encoding complexity and limited energy supply.

### 3.8 Simulation Results and Discussion

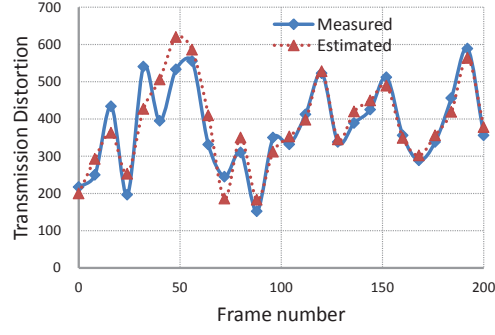
In this section, we report experimental results that demonstrate the performance of the proposed formulation. The simulation is based on the the JMVC (Joint Multi-view Video Coding) version 8.0 of the MVC reference software [119], which is mainly used to encode multi-view video sequences. The standard sequences of Ballroom, Exit and Race1 released by the JVT/MPEG 3-D audio and visual (3DAV) group are chosen for our simulations. Among these sequences, Ballroom contains complex scenes, and Exit is a smooth sequence, and Race1 is a sequence shot by moving set but fixed relative position cameras. The spatial and temporal resolutions for the sequences are 640x480 and 30Hz, respectively. A total of 200 frames in the test sequences is used at the encoder. Context-adaptive binary arithmetic coding (CABAC) is used as the entropy coding scheme and the functions of the variable prediction size and the loop filter are turned on. The size of the GOP is 8, and the search range for disparity estimation and motion estimation is 64. In WZ encoding, the reconstructed key frames of the odd views out of the multi-view video encoder are fed as input into the WZ encoder, after these key frames being partitioned into  $4 \times 4$  blocks that are transformed by a  $4 \times 4$  DCT transform. The DCT coefficients are quantized by scalar quantizer. The quantization matrixes  $\bar{M}^1$ ,  $\bar{M}^3$ ,  $\bar{M}^5$ , and  $\bar{M}^7$  in [120] are used in the quantizer, corresponding to the quantization parameters (QPs) 22, 27, 32, and 37 for the primary stream. Then we use the LDPCA codes to generate parity check bits as the redundant bit-stream. At the WZ decoder, the SI is obtained through error-concealed multi-view decoded key frames of the odd views.

There is only one I-frame in a GOP for each multi-view video sequence, which is assumed to be received error free. This assumption is to ensure that the transmitted GOP is decodable. Each row of the macro-blocks in a frame constitutes a single slice, which is carried in a separate transport packet. It should be noted that the packet length in our simulations is within the limit of the maximum transmission unit for Ethernet. The packet size in this setting is usually around  $700 \sim 1000$  bytes. The reason for this selected packetization strategy is based on the consideration of the trade-off between the efficiency of the error concealment and the compression performance. More specifically, if we employ one packet for each coded frame, the performance degrades significantly as the packet loss rate increases. This is because that the loss of a packet results in the loss of an entire coded frame. Moreover, this approach usually produces packets which exceed the desired maximum packet size of 1500 bytes. On the other hand, if we use one packet for less than one row of the MBs, this packetization scheme can facilitate error concealment by the decoder. However, it will induce larger packetization overhead, which severely reduces the available video bit rate. Therefore, the use of one packet for each row of the MBs can maintain reasonable performance levels over a wide range of packet loss rates. To simulate packet loss, the common condition for wire-line, low delay IP/UDP/RTP packet loss resilient testing defined in [12] is used. In our experiments, the EC method proposed in [88] is employed at the multi-view decoder side, which conceals erroneous blocks by using inter-view and intra-view correlation. The experimental results reported in this section consider 5%, 10%, and 20% average packet loss rates. These packet loss rates are simulated using the respective error pattern files defined in [121]. For each packet loss rate, 300 simulation runs are performed, each one using a different packet loss pattern. For the objective video quality assessment, the luminance peak signal-to-noise ratio (Y-PSNR) is averaged over all decoded frames and all the channel realizations.

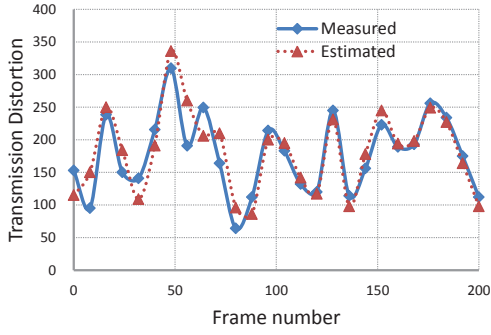
Figure 3.5 compares the measured transmission distortion and that estimated with the proposed method for the key frames of the Ballroom sequence. These plots show that the estimated distortion at the encoder is very close to its actual measured counterpart at the decoder. The measured transmission distortion is



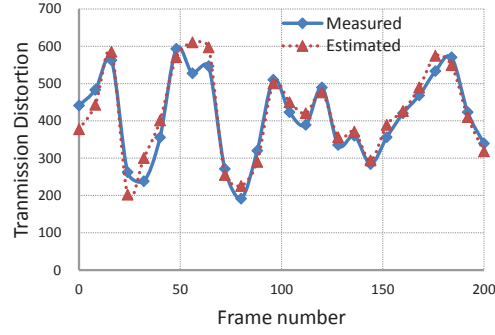
(a) View 2



(b) View 3



(c) View 4



(d) View 5

Figure 3.5: Comparison between the measured and estimated transmission distortion for the key frames of “Ballroom” with a packet loss rate of 10%.

obtained by computing the mean squared error between the actually decoded frames and the error-free reconstructed frames, and the estimated distortion is obtained by our proposed transmission distortion model. Note that the estimated transmission distortion of the key frames of the odd views is larger than that of the previous key frames of the even views at the same time index because of the error propagation from the even views. It is also observed that the estimated transmission distortion is greater than the measured distortion in some case. This is mostly due to the fact the concealment algorithm adopted in our simulation [88] is much more sophisticated than the one assumed at the encoder.

In order to evaluate the performance of the proposed WZ-based error-resilient algorithm, the proposed method, the Dissanayake algorithm [91], and the “JMVC” scheme are compared. “JMVC” represents the basic scheme that only the aforementioned error concealment method is employed at the JMVC decoder to recover

the erroneous region. The Dissanayake algorithm, to the best knowledge of the authors, is so far the latest research work in this area. In the Dissanayake algorithm, different views of a multi-view video are encoded using scalable layers, in which each layer represents a different camera view. The disparity vector is incorporated into the MVC encoder to generate a redundant data stream for the enhancement layer. Then the decoder provides error resilience in two ways. First, if the primary packet of the enhancement layer is lost, then the redundant data stream is decoded in place of the lost data. Secondly, if both primary and redundant packets are lost, the frame copy error concealment method is employed to recover the errors. As shown in Table 3.1, the proposed error-resilient algorithm for MVC can improve on the reconstructed quality of each view with a variety of packet loss rates. Compared to the “JMVC” codec, the PSNR of view 0 remains the same, and the PSNR gain of the odd views is better than that of the even views. This is because the key frames of view 0 are intra-coded without transmission errors, and the key frames of the odd views are additionally protected by WZ coding. Since a higher packet loss rate will result in larger transmission distortion, the SI can be recovered from channel errors to a greater extent and thus the PSNR at the decoder can be better improved. The Exit sequence containing background scenes achieves the maximum decoder PSNR gain among the test sequences, because transmission errors in the error-concealed key frames of the odd views in “Exit” are the largest. When comparing the proposed algorithm with the Dissanayake scheme, it is also clear that the proposed algorithm yields consistent and significant performance gains over the Dissanayake scheme with different packet loss rates. The reason for that is, the Dissanayake scheme aims to protect the coded bit-stream of the enhancement layer view and cannot eliminate error propagation from the previous frames of the base layer view, whereas the proposed WZ-based error-resilient approach attempts to protect the enhancement layer frames and remove transmission errors either occurring on the current frame or propagated from the previous frame of the base layer view.

Table 3.1: Average PSNR comparison with a variety of packet loss rates.

Sequence	View	Y-PSNR (dB) at different loss rates								
		5%			10%			20%		
		JMVC	Dissanayake	Proposed	JMVC	Dissanayake	Proposed	JMVC	Dissanayake	Proposed
Ballroom	0	33.83	33.83	33.83	32.56	32.56	32.56	29.56	29.56	29.56
	1	30.11	30.13	31.93	26.76	27.78	28.81	23.21	24.58	25.48
	2	29.31	29.40	29.49	25.47	25.61	25.76	21.47	21.56	21.83
	3	28.41	30.23	31.64	24.89	25.52	26.37	20.70	23.42	25.97
	4	27.13	28.43	29.13	23.17	24.57	25.46	19.76	21.13	22.08
	5	27.17	29.11	30.08	22.60	24.53	26.73	19.02	23.27	26.18
	6	25.97	26.42	27.07	22.03	23.41	24.38	18.66	21.12	22.56
	7	25.55	27.33	28.28	21.52	24.23	26.59	18.07	22.19	24.76
	Average	28.43	29.36	30.18	24.87	26.03	27.08	21.30	23.35	24.80
Race1	0	31.21	31.21	31.21	28.08	28.08	28.08	25.58	25.58	25.58
	1	28.73	28.82	28.93	25.09	25.13	25.32	21.79	22.18	23.02
	2	27.45	27.69	27.92	24.31	24.56	24.71	20.19	20.41	20.57
	3	26.19	27.11	28.04	22.88	23.98	26.20	19.39	20.12	22.42
	4	26.41	26.89	27.15	22.16	23.12	23.75	19.20	19.98	20.92
	5	25.11	26.13	27.83	21.31	23.41	25.59	18.59	20.34	22.23
	6	24.05	25.02	25.94	21.16	21.68	22.43	17.90	19.21	20.68
	7	24.14	26.48	28.51	21.22	23.42	25.82	17.86	20.23	22.53
	Average	26.66	27.42	28.16	23.27	24.18	25.24	20.06	21.01	22.24



continued from previous page

Sequence	View	Y-PSNR (dB) at different loss rates								
		5%			10%			20%		
		JMVC	Dissanayake	Proposed	JMVC	Dissanayake	Proposed	JMVC	Dissanayake	Proposed
Exit	0	36.79	36.79	36.79	36.03	36.03	36.03	34.73	34.73	34.73
	1	33.23	33.89	34.04	29.98	30.98	32.80	26.02	27.88	31.96
	2	31.79	31.81	31.97	28.13	28.50	28.64	23.88	24.61	25.03
	3	31.04	32.33	34.04	27.46	30.12	33.05	23.24	25.36	31.43
	4	30.29	30.38	30.41	25.54	26.13	26.30	21.83	22.13	22.62
	5	30.27	32.12	33.16	24.77	27.22	30.79	21.23	24.68	29.38
	6	27.26	27.98	28.67	22.59	23.48	24.36	20.02	20.99	21.56
	7	26.42	28.43	30.97	21.63	24.52	28.74	19.17	24.73	27.83
Average		<b>30.89</b>	<b>31.59</b>	<b>32.51</b>	<b>27.02</b>	<b>28.37</b>	<b>30.09</b>	<b>23.76</b>	<b>25.64</b>	<b>28.06</b>

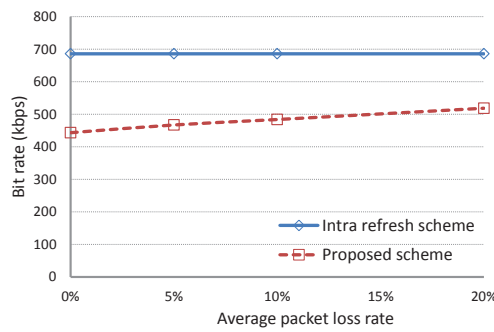
Table 3.2 presents bit rate required for the key frames of the odd views to recover from error propagation. As can be observed from the table, the multi-view streams require fewer additional coded streams to achieve better robustness against channel errors. A larger packet loss rate results in a higher WZ bit rate due to a greater number of transmission errors. The view 1 needs the minimum WZ bit rate in each test sequence because the error propagated from view 0 is the smallest. Moreover, in terms of WZ bit rate for odd views 3, 5, and 7 of each sequence, it can be seen that the proposed scheme will not cause error accumulation of the transmission distortion, and thus can achieve a better error resilience performance. The Exit sequence requires the maximum WZ bit rate of 8.35%, 14.73%, and 26.55% at the packet loss rates of 5%, 10%, and 20%, respectively. Because the WZ encoding protects the quantized symbols of the key frames at the odd views, transmission errors of which in the Exit sequence are the largest. In conclusion, the proposed scheme is very effective in improving error resilience of MVC coded video sequences with a less number of auxiliary bits. When there are no errors, the PSNRs of the proposed scheme and the JMVC method are identical. As for the bit rate, due to the inclusion of the additional WZ bit stream for protecting the key frames of the odd views, the total bit rate of the proposed scheme will be slightly higher than that of the JMVC method. Therefore, in the error-free environment, the rate-distortion performance of the proposed scheme will be slightly worse than that of the JMVC method.

With various packet loss rates, Figure 3.6 demonstrates that the proposed algorithm is more efficient in terms of total bit rate than the conventional intra refresh method in mitigating error propagation. For a fair comparison with the proposed algorithm, the key frames of the odd views are intra-coded. The QPs of the primary bit-stream are set to 32. In this case, the intra-coded key frames would have the same effect in preventing channel error propagation as embedded WZ encoding. As can be seen from the Figure 3.6, with the increase of the packet loss rate, the total bit rate of the proposed algorithm become greater. When the packet loss rate is increased to a certain level, the proposed algorithm may perform worse than the intra refresh method due to the overly increase of the WZ bit rate. However, in practical packet-switched networks, the decoded

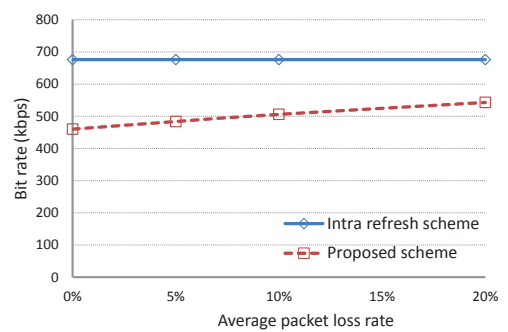
Table 3.2: Bit rate (kbps@30Hz) comparison for WZ encoding with various packet loss rates.

Sequence	View	MVC bit rate	WZ coding bit rate		
			5%	10%	20%
Ballroom	1	484.25	16.81	26.42	47.35
	3	443.26	23.82	40.56	75.27
	5	437.75	23.12	40.90	74.84
	7	495.54	23.73	40.49	74.13
	<b>Average</b>	<b>471.17</b>	<b>21.87</b>	<b>37.09</b>	<b>67.90</b>
Race1	1	465.77	14.61	24.24	44.92
	3	459.94	23.80	46.08	83.04
	5	411.61	24.73	44.50	81.17
	7	480.24	23.03	47.48	81.91
	<b>Average</b>	<b>497.98</b>	<b>21.54</b>	<b>40.58</b>	<b>72.76</b>
Exit	1	160.97	14.27	22.43	39.92
	3	176.1	18.17	34.57	60.51
	5	212.37	19.48	33.90	60.62
	7	268.33	18.42	33.16	62.59
	<b>Average</b>	<b>210.57</b>	<b>17.59</b>	<b>31.02</b>	<b>55.91</b>

video quality is perceptually unacceptable for the human visual system when the channel packet loss rate exceeds 20%.



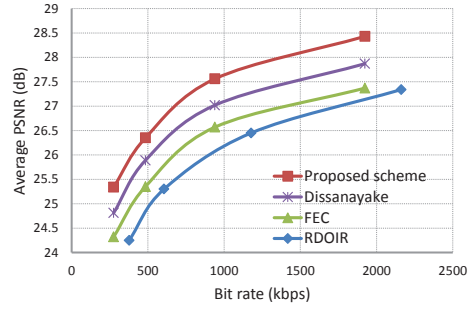
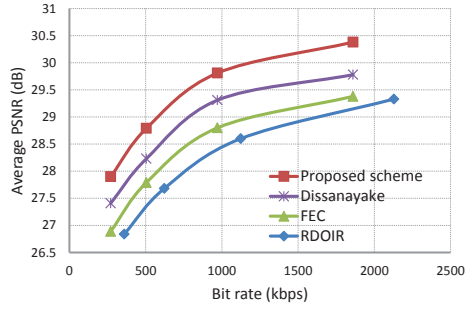
(a) View 3 in "Ballroom"



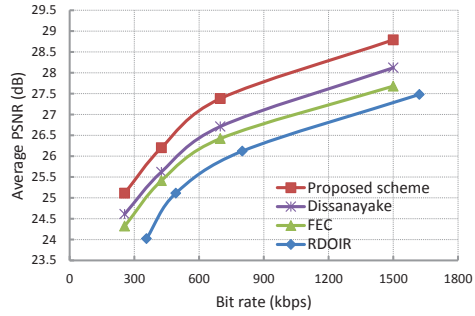
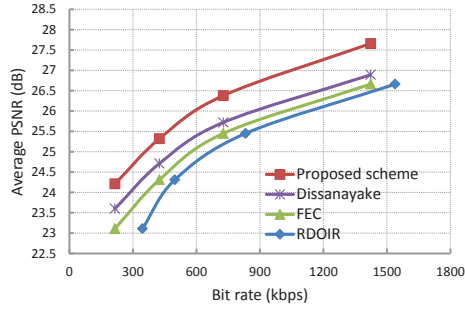
(b) View 3 in "Race1"

Figure 3.6: Total bit rate comparison for view 3 under different packet loss rates.

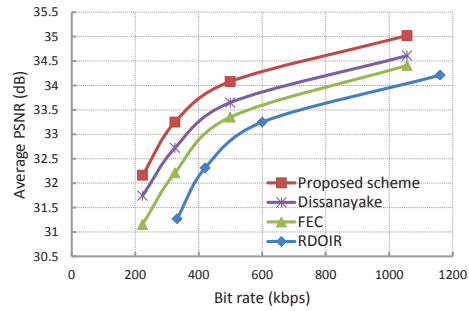
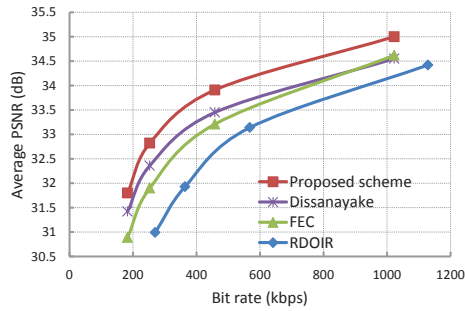
Since the R-D optimized macro-block intra refresh technique achieves significant gains over the traditional robust intra update methods (random, regular), the option is also selected here as the basic algorithm for comparison, which is



(a) View 1 of the *Ballroom* sequence (b) View 3 of the *Ballroom* sequence



(c) View 1 of the *Race1* sequence (d) View 3 of the *Race1* sequence



(e) View 1 of the *Exit* sequence (f) View 3 of the *Exit* sequence

Figure 3.7: R-D performance of the odd views with a packet loss rate of 10%. Compared to the even views, the odd views directly benefits from various error protection algorithms.

denoted by “RDOIR”. In the experiment of RDOIR, we firstly estimate the end-to-end distortion for the key frames of the odd views in a lossy transmission environment. Then based on the availability of the estimated distortion, the R-D optimized coding mode selection is employed to implement intra refresh at macro-block level for the key frames of the odd views. Another widely adopted error resiliency tool FEC is also utilized for comparative evaluation. In the same manner, the FEC by means of  $(N, K)$  Reed-Solomon codes are employed to protect the key frames of the odd views.  $K$  is fixed and set equal to the number of data slices in a frame, while the  $N - K$  is chosen to have the same total bit rate of the proposed WZ-based error resilient MVC scheme. Figure 3.7 compares the rate-distortion performances between the proposed scheme and the other three approaches. To generate the rate-distortion curves, the QPs of the primary bit-stream are set to 22, 27, 32, and 37, which correspond to four points in the rate-distortion curves. The results presented in Figure 3.7 clearly demonstrate that the proposed scheme also achieves higher rate-distortion performance gains over the Dissanayake approach, the FEC scheme, and the RDOIR algorithm. The average comparative PSNR gains are about 0.6 dB, 1.1 dB, and 1.6 dB at the packet loss rate of 10%, respectively. It should be specially mentioned that, in order to make a fair comparison, we also assumed that the FEC bit stream and intra-coded MBs (frames) are received correctly without transmission errors.

To subjectively evaluate the simulation results, Figure 3.8 shows a comparison of the 37th reconstructed frame of the Ballroom sequence from RDOIR, FEC, the Dissanayake scheme and our proposed scheme. To make the visual differences between these error-resilient algorithms more clear, we have zoomed the areas impaired by packet losses in some images, which are correspondingly shown in Figures 3.8 (b), (d), (f). It can be seen that the proposed scheme can significantly reduce the effect of drift, preserving the details in the picture, especially in the moving region. It is important for the subjective visual quality since the motion information in an image takes a dominating role in human perception.

In all the previous experiments, the channel packet loss rate is assumed to be available at the encoder. However, in practical situation, feedback packet loss rate information may be delayed from the decoder. Therefore, the packet

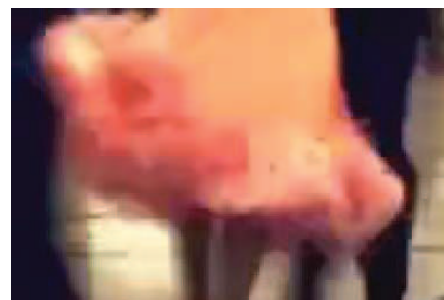
(a) *RDOIR scheme*(b) *Zoom in of the area impaired by packet losses in RDOIR scheme*(c) *FEC scheme*(d) *Zoom in of the area impaired by packet losses in FEC scheme*(e) *Dissanayake scheme*(f) *Zoom in of the area impaired by packet losses in Dissanayake scheme*

Figure 3.8: Subjective image quality for the 37th frame of view 3 with 10% packet loss.

(g) *Proposed scheme*

Figure 3.8: Subjective image quality for the 37th frame of view 3 with 10% packet loss. (con't)

loss rate used by the encoder in transmission distortion estimation process may not be exactly identical to the actual packet loss rate. Clearly, this packet loss rate mismatch will adversely compromise the accuracy of the expected multi-view video distortion model, and then degrade the overall error resilience performance of the proposed WZ-based error resilient algorithm. Firstly, the impact of this packet loss rate mismatch on the overall distortion estimation performance is investigated. In this test, the distortion estimation performance is measured by the “Distortion Difference Ratio (DDR)”, which is defined in [131]. The results for all the test sequences are shown in Figure 3.9 . As can be observed, the DDR shows an upward trend as the packet loss rate mismatch increases. However, the increase of the DDR with the mismatch is insignificant.

To further evaluate the performance of the proposed error-resilient approach when the estimated packet loss rate does not match the actual one, we use 10% packet loss rate in the distortion estimation, whereas, the actual packet loss rate is varied from 5% to 20%. Some selected results are given in Figure 3.10 for a given total bit rate. The proposed WZ based error resilient algorithm outperforms RDOIR and FEC at all selected packet loss rates. Compared to FEC, at 5% packet loss rate, the average PSNR gain is about 0.6 dB for the view 3 of the Exit



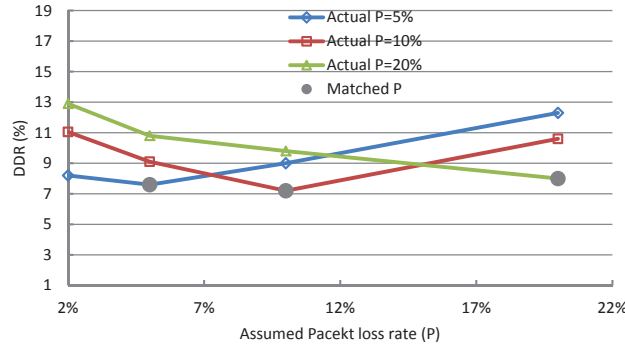


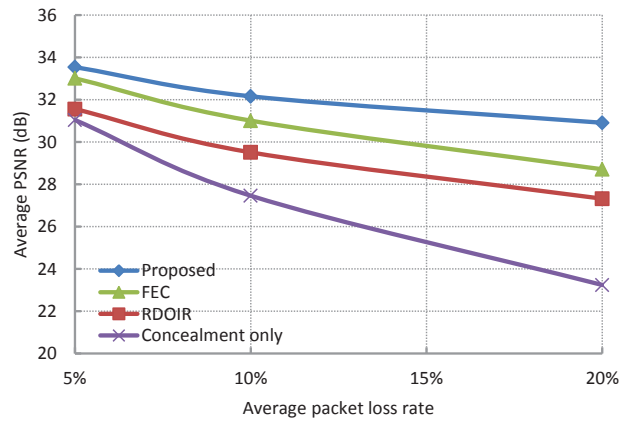
Figure 3.9: Distortion estimation performance with packet loss rate mismatch.

sequence and increases to up to 2.2 dB at 20% packet loss rate. We also notice that when the packet loss rate increases to about 15%, the RDOIR method gives better performance than FEC for the Race1 sequence. This is mainly because the error correction capability of FEC is exceeded due to the lost data packets.

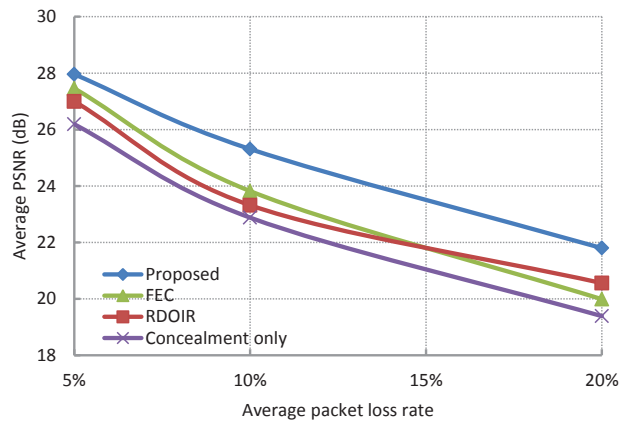
It should be noted that, in practice, due to the unreliability and heavy traffic of the feedback channel, it is better to consider statistical estimation of packet loss rate instead of using a fixed packet loss rate. For the online measurement and estimation of the packet loss probability, we need to know the main source of packet loss in IP-based video service. Generally, the major reason for packet loss in IP networks is the network congestion and long queuing delay [113]. The network congestion is actually the buffer overflow at the outgoing interface in network nodes. Therefore, the estimation of packet loss probability is usually calculated from the buffer overflow probability in an infinite buffer system based on the stochastic characteristics of input traffic [122]. Without loss of generality, the Gaussian model is considered to represent the stochastic input process. On the other hand, the packet loss probability caused by network delay may also be randomly varying and usually follow a shifted Gamma distribution [191]. It should be noted that, the proposed framework in this chapter is general and not limited to any specific input traffic or network delay model. All that is needed is a stochastic model of the input traffic and delays.

Many researchers have studied the actual network loss behavior, and most of these studies agree that internet packet loss often exhibits finite temporal dependency, which means if the current packet is lost, then the next packet is





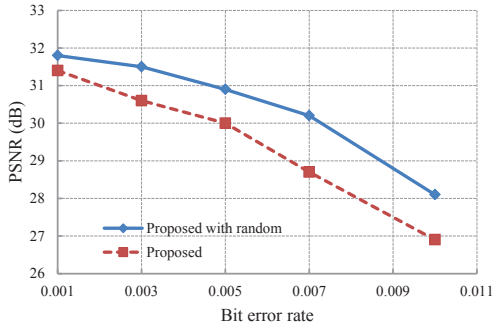
(a) View 3 in “Exit”



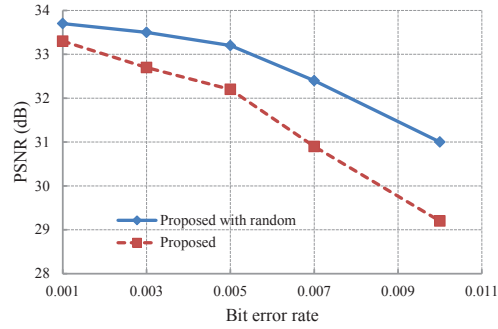
(b) View 3 in “Race1”

Figure 3.10: Performance for mismatch with an assumed packet loss rate of 10%.

also likely to be lost [123]. This leads to bursty packet losses. In order to cope with bursty errors, we combine the proposed WZ based error-resilient approach with a random permutation scheme. More specifically, for the multi-view video sequences, in addition to the key frames of the odd views protected by the WZ encoding technique, a random permutation of the ordering of the macroblocks is generated for all the views. For simplicity, the proposed method with the random permutation scheme is denoted by “Proposed with random”. To validate the performance of the “Proposed with random” scheme, the average burst length is set to 240 bits with the average bit error rates of  $1 \times 10^{-3}$ ,  $3 \times 10^{-3}$ ,  $5 \times 10^{-3}$ ,  $7 \times 10^{-3}$ , and  $1 \times 10^{-2}$ . The plots of the PSNR versus bit error rates averaged over 100 runs for the “Proposed with random” and “Proposed” schemes are shown in Figure 3.11. The results demonstrate that the “Proposed with random” scheme significantly outperforms the “Proposed” scheme in the bursty error case. This is expected because each bursty error can be decomposed into several “individual” errors after random permutation, and consequently the channel error propagation can be effectively mitigated.



(a) “Ballroom” sequence



(b) “Race1” sequence

Figure 3.11: Plots of PSNR versus bit error rate for the bursty errors.

### 3.9 Summary

Through utilizing inter-view correlation and the WZ video coding technique, an error-resilient coding algorithm is proposed for MVC. The key frames of the odd views are protected by WZ encoding to mitigate error propagation. Their

corresponding concealed reconstructed counterparts are employed as the SI for WZ decoding. Then WZ decoding is used to correct channel errors of the SI. In the proposed WZ-based MVC structure, a transmission distortion model is presented first, in which any motion and disparity EC method is allowed to be used at the decoder. Then the parity bits can be estimated using the proposed distortion model, where the previous bit-planes are assumed to be decoded successfully. Although this assumption may cause slight mismatch on the transmitted parity bits, the effect of mismatch is always negligible verified by our simulation results. Extensive experimental results are presented to demonstrate that the proposed algorithm can render the multi-view coded bit-stream more resilient to channel errors.

The proposed WZ-based error-resilient scheme is independent from the specific MVC codec used to encode the multi-view video bit stream. Only the transmission distortion prediction algorithm needs to be adjusted to the specific codec adopted. Although there exist some reconstruction mismatch errors on the key frames of the even views and the non-key frames, a drastic reduction in the overall picture quality can be efficiently avoided. The proposed method introduces additional encoder complexity to the MVC and WZ coders, which is mainly caused by the transmission distortion and WZ bit rate estimation module. However, the additional complexity is well justified with performance improvement (of about 1.1 dB in PSNR over FEC-based solutions and 1.6 dB over intra refresh solutions).



# Chapter 4

## Rate-Distortion Optimized Mode Switching for Error-Resilient Multi-view Video Plus Depth Based 3-D Video Coding

### 4.1 Introduction

3-D video is the visual content of the well-known 3-D television (3DTV) and free viewpoint television (FVT) [124]. The main challenge of the 3-D video system lies in storage and transmission of tremendous amounts of multi-view data. As discussed in the previous chapter, one possibility would be to transmit this high number of views using the MVC profile of H.264/AVC [125]. However, when increasing the number of cameras to capture the scene, the bit rate required for coding multi-view video with MVC increases approximately linearly with the number of coded views. So MVC is inappropriate for delivering 3-D content with a large number of views. Depth image-based rendering (DIBR) presents a promising solution for efficient delivery of 3-D video, in which any desired viewpoint can be rendered from a limited number of texture videos, e.g. 2-3 views, and their corresponding depth maps [126]. Due to reduction in the amount of data being transmitted, MVD format has emerged as an efficient data representation for 3-D video system.

With the development of electronic and communication technology, streaming

3-D video and videoconferencing are rapidly increasing in popularity. However, due to the lack of end-to-end quality of service (QoS) guarantee in today's network, 3-D video coder design is facing major new challenges. In unreliable underlying networks, transmission of compressed video is highly susceptible to channel errors. The use of motion compensation prediction causes these errors to propagate to subsequent frames, thus significantly impacting on the received video quality. In MVD-based 3-D video systems, the 2-D video and depth information are either independently encoded by common video compression techniques, or jointly encoded through exploiting the correlations between the texture video and depth map [127], [128]. So the transmission of 3-D video certainly suffers from the same problem of transmission errors. Moreover, since the virtual views are rendered from the compressed texture videos and depth maps, transmission errors of the distorted texture videos and depth maps can be ultimately propagated to the virtual views. However, it is well-known that depth maps are used to aid in the view rendering process. Therefore, the distortion of the depth map due to packet losses will cause incorrect projection of texture video pixels, which may lead to unexpected holes or overlaps in the synthesized virtual view. Thus, compared to the reliability of transmitting 2-D video, robust transmission of texture videos and depth maps over error-prone networks is a more challenging problem.

## 4.2 Related Work

Extensive efforts have been dedicated to improve on the quality of 2-D video against channel errors. Given that the background of error-resilient techniques is already reviewed in Chapter 2, we know that the error control methods are classified into two categories. The first category focuses on link-layer reliability, typically, forward error correction (FEC) and automatic repeat request; the second category considers the intrinsic source dependence, attempting to minimize the quality deterioration using error-resilient video coding methods. Among various error-resilient video coding techniques, the mode switching technique is very popular and widely adopted. The mode switching approach, which is standard-compatible, is useful to combat the adverse effect of packet loss. By switching

off the inter and inter-view prediction loop for certain macroblocks (MBs), the reconstructed blocks no longer depend on past frames and error propagation is stopped. Early related work is mainly based on heuristic intra refresh techniques without rate-distortion consideration, such as randomly or periodically inserting intra MBs. Recently, a number of rate-distortion optimized techniques have been proposed for coding mode switching in error-prone environments. An early proposal of mode selection based on rate-distortion framework to combat packet loss appeared in [129]. A significant improvement to rate-distortion based mode selection was proposed in [130]- [132], where the expected end-to-end distortion is estimated first at the pixel level by recursively calculating the first and second moments of the reconstructed pixel value, and then the estimated distortion is incorporated into a rate-distortion based mode switching process. Compared to the early heuristic mode switching strategies, rate-distortion based mode switching methods have contributed to a significant improvement on the error resilience performance. On the other hand, in order to reduce the computational complexity and memory costs, several block-level end-to-end rate-distortion optimization schemes [133]- [136] have been developed for video coding in packet loss environments, in which a block-level distortion map is recursively updated for each frame. Further, joint optimization of mode switching, error concealment, and channel coding has been considered in [137].

However, to the best knowledge of the authors, only a limited number of publications have reported on robust multi-view 3-D video coding. Macchiavello *et al.* proposed a reference frame selection algorithm at the block level for loss-resilient depth map coding to minimize expected synthesized view distortion [139], where the encoder has the flexibility to choose the reference frame with long prediction distance for motion compensation. Thereafter, this idea was extended to encoding of both textures and depth maps [140]. However, in these two algorithms, inter-view error propagation is not considered in the transmission distortion modeling and only the distortion in synthesized views is characterized. In MVD-based 3-D video coding, since both the rendered virtual view and the coded view would be presented for viewing at the receiver side, in order to achieve the optimal rate-distortion performance, it is reasonable to consider both qualities of the coded

texture videos and the virtual views.

### 4.3 Contribution of this Chapter

In this chapter, we mainly target the research problem 2 as defined in Chapter 1 so as to improve the overall performance of MVD-based 3-D video in packet loss scenarios. The major contributions of this chapter are two folds.

1. The first one is the proposal of a recursive pixel-level end-to-end distortion model for MVD-based video transmission over lossy packet-switched networks. In the overall distortion estimation, we take both the expected texture video distortion and virtual view distortion into consideration. Compared to the 2-D video distortion which affects only the pixel intensity, the depth distortion in 3-D video causes position errors in the rendered virtual view. Therefore, a new expected distortion metric is proposed to capture the effect of the texture video and depth reconstructed errors on the synthesized virtual views, which also considers the time-temporal and inter-view error propagation of texture video and depth map encoding. Especially, the effect of the depth error on the view warping is analyzed in the frequency domain.
2. The second contribution is the introduction of a rate-distortion optimized mode switching algorithm, which, to the best of the authors' knowledge, is the first one able to improve the MVD-based 3-D video error resilience performance by exploiting the mode decision strategy. The main novelty of this lies in that the original source coding distortion of each texture and depth MB is replaced by the expected overall distortion of decoder MB reconstruction, which accounts for the impact of packet losses. Then, based on the end-to-end estimated distortion, the encoder can optimally select the intra, inter or inter-view mode for each MB during joint texture video and depth map encoding. During the optimization process, we explicitly consider the inherent dependency between the texture mode and depth mode.



## 4.4 MVD-based 3-D Video Coding Framework and Error Propagation

For ease of exposition, we use two-view based 3-D video to elaborate on the idea presented in this chapter, which can be easily extended to the case of multiple views. In a two-view based 3-D video, suppose that one view is regarded as left view and the other view is regarded as right view, each view is composed of a texture video and the corresponding depth map. Texture videos are captured with multiple cameras, and the corresponding depth maps are generated by the depth estimation methods. A depth map is usually represented as a gray scale video sequence that describes the positions of objects in the scene. As shown in Figure 4.1, the left view is encoded by the conventional video coding scheme with the temporal motion-compensated prediction. For the right view, the video is encoded with both the temporal motion-compensated prediction and the disparity-compensated prediction that is employed to exploit the inter-view correlation. Since the depth maps can be treated as the monochromatic videos, they are also encoded using the same prediction structure. At the decoder, the audience desired viewpoint videos are synthesized with the decoded texture videos and depth maps by the DIBR technique, and ultimately provide 3-D video experiences for the end users.

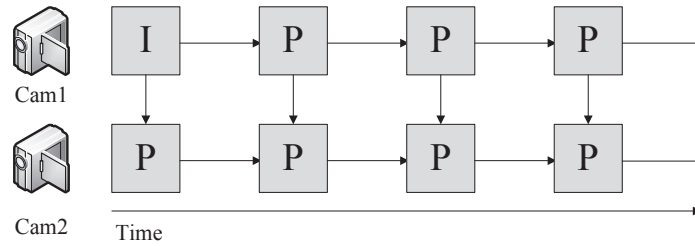


Figure 4.1: A prediction structure for two-view based 3-D video coding.

According to the MVD-based 3-D video coding framework, when transmitting the texture video and associated depth streams over an error-prone channel, the bit streams are extremely sensitive to transmission errors. Compared to single-view video transmission, the mismatch error will not only propagate to the subsequent frames of the current view, but also spread to other dependent views through the disparity-compensated based inter-view prediction. Furthermore, in

a typical DIBR-based 3-D video system, the color images of the virtual view are commonly synthesized by the decoded texture videos and depth maps of the neighboring reference views. Therefore, errors in the texture video and depth map will still propagate to the synthesized view. On one hand, the texture video transmission errors result in distortion to both the luma and chroma values of synthesized pixels. On the other hand, the depth map transmission errors lead to geometry distortion in the virtual view images, which is fundamentally different from the distortion affecting luma and chroma data in standard 2-D video. More specifically, errors in depth values at a given pixel position, affect the position in the synthesized view where this pixel will be used for interpolation, which in turn will translate into errors in the luma and chroma of the rendered view. To mitigate the effects of error propagation, an error-resilient MVD-based 3-D video coding technique is much desired to enable higher packet loss resilience.

In the next, an end-to-end distortion estimation algorithm is first developed with respect to the transmission distortion of the texture video and depth map. Then, based on the estimated distortion, a new rate-distortion optimized mode switching scheme is derived to improve the error resilience performance of the texture video stream and depth stream.

## 4.5 End-to-End Distortion Estimation for MVD-based 3-D Video Transmission

As mentioned before, since the depth map is not directly used for viewing, minimizing the depth map distortion does not guarantee the optimal quality in the virtual views. So when encoding the depth map, it is more appropriate to consider its effect to the rendered view quality instead of the distortion in compressed depth map itself. Moreover, since both the coded views and virtual views will be presented for human viewing, their decoded qualities are equally important for visual experience. Therefore, in this work, we will model and characterize the total expected decoder distortion of both the coded view video and synthesized view video at the encoder.

Denote by  $T_{x,y}$  the original value at pixel position  $(x, y)$  in the texture video.

Let  $\tilde{T}_{x,y}$  be the reconstructed value of the texture video at the decoder. In a DIBR system, a virtual view can be rendered using a set of reference video frames and their corresponding depth maps. So in a similar manner, let  $V_{x',y'}$  be the original value at pixel position  $(x', y')$  in the synthesized view rendered by the original texture video and depth map, and  $\tilde{V}_{x'',y''}$  be the reconstructed value at pixel position  $(x'', y'')$  in the synthesized view rendered by the decoded texture video and depth map. It should be noted that  $(x', y')$  is a warped pixel position for the rendered view corresponding to  $(x, y)$  in the texture video by the pre-defined warping function. Due to the depth map reconstructed error, it causes the projection of pixel  $(x, y)$  to move from  $(x', y')$  to  $(x'', y'')$  in the rendered view, and this effect is known as geometry distortion. So using the mean square error (MSE) as the distortion metric, the total end-to-end distortion of MVD-based 3-D video can be approximately decomposed into the following two components

$$\begin{aligned} \text{MSE}_{\text{Total}}(T_{x,y}, V_{x',y'}) &= \text{MSE}_T(T_{x,y}) + \text{MSE}_V(V_{x',y'}) \\ &= E \left\{ (T_{x,y} - \tilde{T}_{x,y})^2 \right\} + E \left\{ (V_{x',y'} - \tilde{V}_{x'',y''})^2 \right\} \end{aligned} \quad (4.1)$$

where  $\text{MSE}_T(T_{x,y})$  and  $\text{MSE}_V(V_{x',y'})$  are the expected distortion of pixel  $(x, y)$  of the texture video and the expected distortion of pixel  $(x', y')$  of the synthesized view at the decoder, respectively.  $E \{ \cdot \}$  represents expectation.

Since error-resilient distortion model of texture video has been widely studied in the literature, we will mainly derive a recursive distortion model to capture the effect of packet losses in depth map and texture video on the synthesized view distortion. In general, if there is no information loss in either the texture or depth map stream, the exact view synthesis distortion can be measured between the rendered view and the ground truth at the encoder. However, because the rendered view can be generated for any arbitrary viewpoint, the ground truth is always not available. In addition, to estimate the distortion of the rendered view, the computational complexity is prohibitively high if the actual view rendering processing is performed during the depth coding process. So instead, the view synthesis distortion is always approximated using the reference video to be compatible with block processing [141]. In the case of the error-prone environment, as the local video characteristics of the reference video would also be very similar to those of the synthesized view video, reconstruction errors in the rendered

virtual view can still reflect on the source reference views [142]. Since the depth maps of different views have large amounts of uniform contents, the impact of different views on the same virtual view may be rather similar. For simplicity, we only consider the impact from one adjacent view in the following distortion derivations. When the cameras are in parallel positions, the rendered virtual view distortion can be represented as

$$\begin{aligned} \text{MSE}_V(V_{x',y'}) &= E \left\{ (V_{x',y'} - \tilde{V}_{x'',y''})^2 \right\} \\ &= w_r^2 E \left\{ (T_{x,y} - \tilde{T}_{x-\Delta p(x,y),y})^2 \right\} \end{aligned} \quad (4.2)$$

where  $w_r$  is the weighting factor of the rendered virtual image from a particular view. If a pixel in the rendered view picture is only occluded in one reference view, the corresponding weighting factor is set to 0, while the other one is set to 1. And  $\Delta p(x, y)$  indicates the translational rendering position error, which is already proven that it is proportional to depth map error as in [141], [143], i.e.,

$$\Delta p(x, y) = \alpha(D_{x,y} - \tilde{D}_{x,y}) \quad (4.3)$$

where  $D_{x,y}$  and  $\tilde{D}_{x,y}$  indicate the original and reconstructed values of  $(x, y)$  in the depth image, respectively, and  $\alpha$  is the proportional coefficient determined by the following equation

$$\alpha = \frac{fL}{255} \left( \frac{1}{Z_{\text{near}}} - \frac{1}{Z_{\text{far}}} \right) \quad (4.4)$$

where  $f$  is the focal length,  $L$  is the baseline between the reference view and rendered view,  $Z_{\text{near}}$  and  $Z_{\text{far}}$  are the values of the nearest and farthest depth of the scene, respectively.

Then, (4.2) can be further derived as follows:

$$\begin{aligned} \text{MSE}_V(V_{x',y'}) &= w_r^2 E \left\{ (T_{x,y} - \tilde{T}_{x-\Delta p(x,y),y})^2 \right\} \\ &= w_r^2 E \left\{ (T_{x,y} - \tilde{T}_{x,y} + \tilde{T}_{x,y} - \tilde{T}_{x-\Delta p(x,y),y})^2 \right\} \\ &= w_r^2 E \left\{ (T_{x,y} - \tilde{T}_{x,y})^2 \right\} + w_r^2 E \left\{ (\tilde{T}_{x,y} - \tilde{T}_{x-\Delta p(x,y),y})^2 \right\} \\ &\quad + 2w_r^2 E \left\{ (T_{x,y} - \tilde{T}_{x,y})(\tilde{T}_{x,y} - \tilde{T}_{x-\Delta p(x,y),y}) \right\} \end{aligned} \quad (4.5)$$

where  $E \left\{ (\tilde{T}_{x,y} - \tilde{T}_{x-\Delta p(x,y),y})^2 \right\}$  represents the average view rendering distortion induced by depth map errors, and  $E \left\{ (T_{x,y} - \tilde{T}_{x,y})^2 \right\}$  represents the average view rendering distortion induced by texture errors, and  $E \left\{ (T_{x,y} - \tilde{T}_{x,y})(\tilde{T}_{x,y} - \tilde{T}_{x-\Delta p(x,y),y}) \right\}$

approximates to zero [144]. As suggested in (4.5), the view rendering distortion due to depth errors can be represented by the squared difference between pixel  $(x, y)$  in the reconstructed texture image and its position shifted counterpart. In order to further analyse  $E \left\{ (\tilde{T}_{x,y} - \tilde{T}_{x-\Delta p(x,y),y})^2 \right\}$ , the Discrete Fourier Transform (DFT) is employed.

#### 4.5.1 Frequency Domain Analysis of the View Synthesis Distortion Caused by Depth Error

Generally, the DFT of  $\tilde{T}_{x,y}$  is given below

$$\begin{aligned} \tilde{\Phi}_{x,y}(\omega_j, \omega_k) &= \text{DFT}(\tilde{T}_{x,y}) \\ &= \frac{1}{WH} \sum_{x=0}^{W-1} \sum_{y=0}^{H-1} \tilde{T}_{x,y} \exp \{ -j(\omega_j x + \omega_k y) \} \end{aligned} \quad (4.6)$$

where  $W$  and  $H$  represent the width and height of the texture images, respectively, and the discrete frequencies  $\omega_j, \omega_k$  are equal to,  $\frac{2\pi}{W}j$  and  $\frac{2\pi}{H}k$ , with  $j = 0, \dots, W-1$ , and  $k = 0, \dots, H-1$ , respectively.

By applying the shift theorem of the DFT to (4.6), we can obtain the DFT version of  $\tilde{T}_{x-\Delta p(x,y),y}$  as follows

$$\tilde{\Phi}_{x-\Delta p(x,y),y}(\omega_j, \omega_k) = \tilde{\Phi}_{x,y}(\omega_j, \omega_k) \exp(-j\omega_j \Delta p(x, y)) \quad (4.7)$$

As can be seen from (4.7), the shift in warping location is translated into a complex exponent in the frequency domain via DFT.

According to the Parseval's theorem in signal processing, the estimated view rendering distortion induced by depth errors is given by

$$\begin{aligned} &E \left\{ (\tilde{T}_{x,y} - \tilde{T}_{x-\Delta p(x,y),y})^2 \right\} \\ &= \frac{1}{W^2 H^2} \sum_{j=0}^{W-1} \sum_{k=0}^{H-1} \left| \tilde{\Phi}_{x,y}(\omega_j, \omega_k) \right|^2 |1 - \exp(-j\omega_j \Delta p(x, y))|^2 \end{aligned} \quad (4.8)$$

The Taylor series expansion of  $|1 - \exp(-j\omega_j \Delta p(x, y))|^2$  yields the following polynomial

$$\begin{aligned} |1 - \exp(-j\omega_j \Delta p(x, y))|^2 &= 2 - 2 \cos(\omega_j \Delta p(x, y)) \\ &= \frac{2(\omega_j \Delta p(x, y))^2}{2!} - \frac{2(\omega_j \Delta p(x, y))^4}{4!} + \frac{6(\omega_j \Delta p(x, y))^6}{6!} - \dots \end{aligned} \quad (4.9)$$

Since the higher order terms in (4.9) are insignificant for small  $\omega_j \Delta p(x, y)$ , they could be approximated to zero. As a result, the following approximation

holds

$$|1 - \exp(-j\omega_j \Delta p(x, y))|^2 = \frac{2(\omega_j \Delta p(x, y))^2}{2!} \quad (4.10)$$

Therefore, (4.8) can be rewritten as

$$\begin{aligned} E \left\{ (\tilde{T}_{x,y} - \tilde{T}_{x-\Delta p(x,y),y})^2 \right\} &= \frac{1}{W^2 H^2} \sum_{j=0}^{W-1} \sum_{k=0}^{H-1} |\tilde{\Phi}_{x,y}(\omega_j, \omega_k)|^2 (\omega_j \Delta p(x, y))^2 \\ &= \frac{1}{W^2 H^2} \sum_{j=0}^{W-1} \sum_{k=0}^{H-1} \left[ |\tilde{\Phi}_{x,y}(\omega_j, \omega_k)|^2 \omega_j^2 \right] \cdot (\Delta p(x, y))^2 \quad (4.11) \\ &= \psi_r \cdot (\Delta p(x, y))^2 \end{aligned}$$

where

$$\psi_r = \frac{1}{W^2 H^2} \sum_{j=0}^{W-1} \sum_{k=0}^{H-1} \left[ |\tilde{\Phi}_{x,y}(\omega_j, \omega_k)|^2 \omega_j^2 \right]$$

For a particular rendered virtual view, (4.11) implies that  $E \left\{ (\tilde{T}_{x,y} - \tilde{T}_{x-\Delta p(x,y),y})^2 \right\}$  can be characterized by a linear model and expressed as follows [142], [145]

$$E \left\{ (\tilde{T}_{x,y} - \tilde{T}_{x-\Delta p(x,y),y})^2 \right\} = \|\Delta p(x, y)\|^2 \times \psi_r \quad (4.12)$$

where  $\psi_r$  is the linear parameter associated with image contents, which can be readily computed from the energy density of the input texture video of the adjacent view.

Based on (4.3) and (4.12), the relationship between  $E \left\{ (\tilde{T}_{x,y} - \tilde{T}_{x-\Delta p(x,y),y})^2 \right\}$  and the expected depth distortion  $\text{MSE}_D(D_{x,y})$  of pixel  $(x, y)$  can be approximately defined by

$$E \left\{ (\tilde{T}_{x,y} - \tilde{T}_{x-\Delta p(x,y),y})^2 \right\} = \alpha^2 \psi_r \times E \left\{ (D_{x,y} - \tilde{D}_{x,y})^2 \right\}. \quad (4.13)$$

Thus, based on (4.1), (4.5), and (4.13), the total expected distortion of MVD-based 3-D video can be rewritten as

$$\text{MSE}_{\text{Total}}(V_{x',y'}, T_{x,y}) = (w_r^2 + 1) E \left\{ (T_{x,y} - \tilde{T}_{x,y})^2 \right\} + w_r^2 \alpha^2 \psi_r \times E \left\{ (D_{x,y} - \tilde{D}_{x,y})^2 \right\}. \quad (4.14)$$

As can be observed from (4.14), the total distortion of MVD-based 3-D video can be modeled as a linear combination of the distortions of the transmitted texture video and depth map. Since  $\tilde{D}_{x,y}$  and  $\tilde{T}_{x,y}$  cannot be accessed at the encoder due to possible packet losses in the channel, accurate and robust modeling of the expected texture video and depth map distortion remains a challenging problem.

In this work, based on the characteristics of MVD-based 3-D video coding and the propagating behavior of transmission errors, we develop a general recursive function to estimate the expected decoder distortion through characterizing the packet loss probability, which explicitly takes into account the channel-induced distortion caused by both motion and disparity compensation prediction. Since the textures and depth maps of two neighboring captured views are encoded separately by H.264/MVC using the same prediction structure, the derivation for the expected texture and depth error will be exactly the same. Thus in the following, we will only derive the expected end-to-end distortion in the coded depth map due to channel losses.

#### 4.5.2 Expected Texture and Depth Distortion Model

Before modeling the expected texture and depth distortion, we make some assumptions. Without loss of generality and for simplicity, it is assumed that the underlying depth bit stream is packetized at the slice level. That is, data for coding an integer number of MBs are transmitted in a separate transport packet. So suppose the packet loss rate is known as  $p$ , which is equivalent to the slice loss rate<sup>1</sup>. Furthermore, it is assumed that the losses of two different packets occur independently. The overall expected distortion of pixel  $(x, y)$  in the depth frame  $t$  of view  $s$  can be decomposed as

$$\begin{aligned}
 \text{MSE}_D(D_{x,y}(s, t)) &= E \left\{ (D_{x,y}(s, t) - \tilde{D}_{x,y}(s, t))^2 \right\} \\
 &= (1 - p)(d_s(D_{x,y}(s, t)) + d_{ep}(D_{\theta(x,y)}(s_{ref}, t_{ref}))) \\
 &\quad + p d_{ec}(D_{x,y}(s, t))
 \end{aligned} \tag{4.15}$$

where  $d_s(D_{x,y}(s, t))$  denotes the familiar source coding distortion, which is the distortion between the original and error-free reconstructed signals.  $d_s(D_{x,y}(s, t))$  can be either exactly calculated by actual coding or estimated by a model-based approach for fast approximation.  $d_{ep}(D_{\theta(x,y)}(s_{ref}, t_{ref}))$  is the error propagation distortion introduced by the reference pixel  $\theta(x, y)$  in the depth frame  $t_{ref}$  of view

---

<sup>1</sup>We assume that the packet loss rate is available at the encoder. This can be either specified as part of the initial negotiations, or adaptively calculated from information provided by the transmission protocol.

$s_{ref}$ , and  $\theta(\cdot)$  is the operator to calculate the spatial position of the reference pixel.  $d_{ec}(D_{x,y}(s, t))$  is the error concealment distortion in case pixel  $(x, y)$  is lost.

Since much early research in the rate-distortion theory has been undertaken to investigate the source coding distortion [146], in order to optimally estimate the end-to-end depth distortion at the encoder, we mainly focus on the computation of  $d_{ep}(D_{\theta(x,y)}(s_{ref}, t_{ref}))$  and  $d_{ec}(D_{x,y}(s, t))$  due to potential packet losses. Usually, the sum of  $d_{ep}(D_{\theta(x,y)}(s_{ref}, t_{ref}))$  and  $d_{ec}(D_{x,y}(s, t))$  is called channel distortion or transmission distortion.

For future error computation, the error propagation distortion of the current coding pixel can be calculated as

$$\begin{aligned} d_{ep}(D_{x,y}(s, t)) = & (1 - p)d_{ep}(D_{\theta(x,y)}(s_{ref}, t_{ref})) \\ & + p(d_{ec,r}(D_{x,y}(s, t)) + d_{ep}(D_{\rho(x,y)}(s_{ec}, t_{ec}))) \end{aligned} \quad (4.16)$$

where  $d_{ec,r}(D_{x,y}(s, t))$  denotes the distortion between the error-free reconstructed and error concealed values at the depth map encoder. As can be observed from (4.16), the term  $d_{ep}(D_{\theta(x,y)}(s_{ref}, t_{ref}))$  is the error propagation distortion of the reference pixel when the current data packet is received free of errors. On the other hand, in the case of channel errors, the distortion is divided into two terms with the first term being the distortion caused solely by the error concealment algorithm, whereas the second term is attributed to the error propagation distortion of the concealed pixel  $\rho(x, y)$  of view  $s_{ec}$ . And  $\rho(\cdot)$  refers to the operator to calculate the spatial position of the concealed pixel.  $d_{ec,r}(D_{x,y}(s, t))$  can be readily measured by simulating packet losses at the encoder with the prior knowledge of the packet loss rate, while  $d_{ep}(D_{\theta(x,y)}(s_{ref}, t_{ref}))$  and  $d_{ep}(D_{\rho(x,y)}(s_{ec}, t_{ec}))$  can be recursively calculated under the given inter dependencies established during motion compensation prediction and error concealment processes.

Note that the error propagation distortion of the first depth map frame of the left view can be directly derived without considering error propagation because they are typically coded as intra frames. Then, the error propagation distortion of the pixel in the following depth frames can be recursively calculated in accordance with the prediction structure.



#### 4.5.2.1 Error Propagation Distortion of Pixels in the Depth Frames of the Left View

Because the depth frames of the left view is temporally predicted only by motion-compensated prediction, the expected error propagation distortion of pixel  $(x, y)$  with the inter coding mode can be rewritten as that of single-view video transmission

$$d_{ep}(D_{x,y}(s, t)) = (1-p)d_{ep}(D_{\theta(x,y)}(s, t-1)) + p(d_{ec.r}(D_{x,y}(s, t)) + d_{ep}(D_{\rho(x,y)}(s_{ec}, t_{ec}))). \quad (4.17)$$

If each pixel in the concealed picture is directly copied from the co-located pixel of the previous time depth frame, the error propagated distortion of the concealed pixel in (4.17) can be expressed as

$$d_{ep}(D_{\rho(x,y)}(s_{ec}, t_{ec})) = d_{ep}(D_{x,y}(s, t-1)). \quad (4.18)$$

For an intra-coded pixel, no transmission errors will be propagated from a temporal frame of the depth map, and the spatial error propagation within the same frame caused by intra-prediction is rather limited and often considered negligible in the literature [132], [147].

The reason for that lies in two-fold. The first one is, compared with inter/inter-view coding modes, very few MBs in a P-frame will be coded in an intra-mode (The percentage of intra-coded MBs is usually around 1%-3%). In the second, when a MB is intra-coded using intra-prediction, it is always predicted by a weighted sum of several previously coded neighboring MBs in the same frame. In this case, the error propagation will be attenuated by the intra-prediction. Moreover, if constrained intra-prediction is used, the error propagation distortion in received intra-coded MBs is zero.

This claim is also confirmed by our experimental results. In the following, we provide some statistical information about the error propagation distortion induced by intra-prediction. Figure 4.2 shows the percentages of the three different kinds of error propagation distortion contained in the channel distortion, namely, distortion induced by intra-prediction, distortion induced by inter/inter-view prediction, distortion induced by error concealment. (The channel distortion refers to the sum of the error propagation distortion and error concealment distortion

in (4.15), which is the mean squared error between the reconstructed frame at the encoder and the decoded video frame at the receiver.) As can be observed from Figure 4.2, the percentage of error propagation distortion induced by intra-prediction within the same frame is relatively small compared with the other error propagation distortion in the channel distortion.

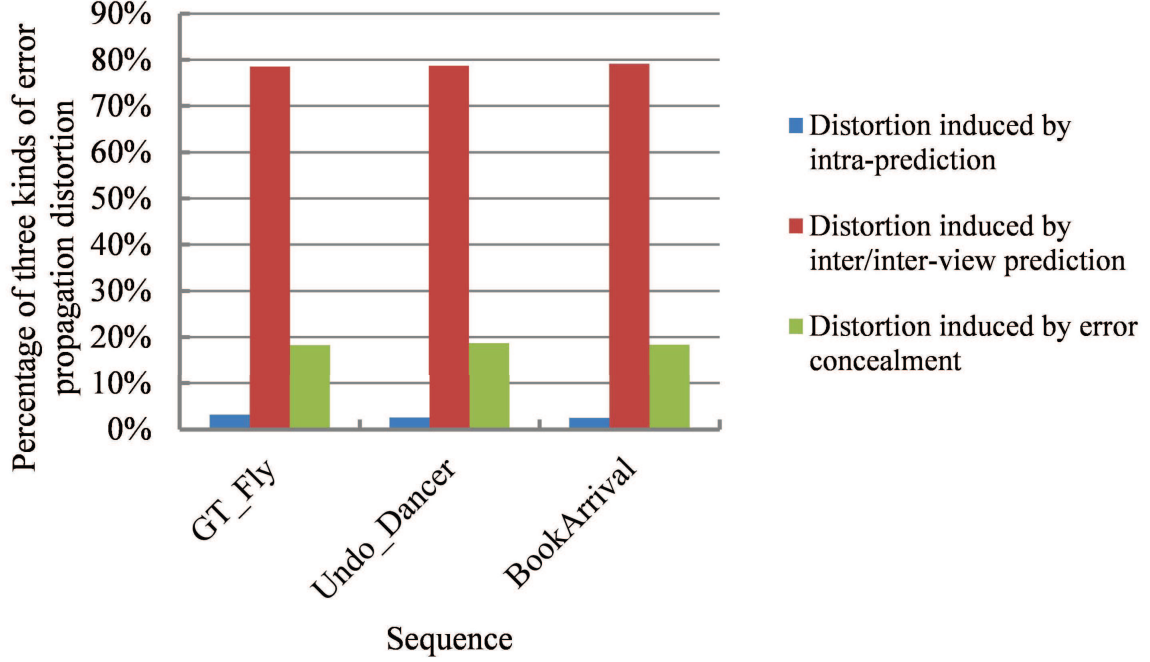


Figure 4.2: Comparison of percentage of three kinds of distortion in the channel distortion with 10% packet loss rate.

Therefore, we consider that error-propagated distortion for intra-coded pixels is due only to video packet drops. So the error propagated distortion of (4.16) for the intra mode needs to be modified as

$$d_{ep}(D_{x,y}(s, t)) = p(d_{ec,r}(D_{x,y}(s, t)) + d_{ep}(D_{\rho(x,y)}(s_{ec}, t_{ec}))). \quad (4.19)$$

#### 4.5.2.2 Error Propagation Distortion of Pixels in the Depth Frames of the Right View

If the current pixel is in the depth frames of the right view, the error propagation distortion not only comes from itself, but also comes from the depth frame of the left view. For the inter and intra coding modes, the derivation of the error propagation distortion of the depth frame of the right view is identical to that of

the depth map of the left view. However, for the inter-view coding mode, because the depth frame of the right view is additionally predicted by its counterpart depth frame of the left view through exploiting inter-view dependence based upon disparity compensation, the expected error propagation distortion for inter-view mode can be rewritten as

$$d_{ep}(D_{x,y}(s, t)) = (1-p)d_{ep}(D_{\theta(x,y)}(s-1, t)) + p(d_{ec.r}(D_{x,y}(s, t)) + d_{ep}(D_{\rho(x,y)}(s_{ec}, t_{ec}))). \quad (4.20)$$

As  $d_{ep}(D_{x,y}(s, t))$  can be calculated using the above mentioned derivations, the only thing left in the overall distortion formula (4.15) is how to compute  $d_{ec}(D_{x,y}(s, t))$ , which is given by the following

$$d_{ec}(D_{x,y}(s, t)) = d_{ec.o}(D_{x,y}(s, t)) + d_{ep}(D_{\rho(x,y)}(s_{ec}, t_{ec})). \quad (4.21)$$

Observe that  $d_{ec}(D_{x,y}(s, t))$  consists of the error propagated distortion of the concealed pixel  $\rho(x, y)$  and the distortion  $d_{ec.o}(D_{x,y}(s, t))$  between the original and concealed pixels.  $d_{ec.o}(D_{x,y}(s, t))$  can also be readily calculated by running the decoder error concealment method.

## 4.6 Mode Switching Within a Rate-Distortion Framework

Upon the availability of the end-to-end distortion, the rate-distortion optimized mode decision for MVC encoder in packet loss environments can be readily derived. Conventional multi-view video coders perform mode decision by minimizing the source distortion, without consideration of packet losses or channel errors. In our approach, we incorporate the overall expected distortion within the rate-distortion framework in order to optimally select the encoding mode for each texture and depth MB. This enables minimization of the overall distortion of both the decoded texture video and the rendered view video for the given packet loss rate and bit rate. Three kinds of modes are considered, i.e., inter, inter-view, and intra modes. The rate-distortion optimized mode selection problem is thus equivalent to minimizing the following Lagrangian cost function [148]

$$\min_{\text{mode}}(J_{\text{MB}}) = \min_{\text{mode}}(D_{\text{MB}} + \lambda R_{\text{MB}}) \quad (4.22)$$

where  $D_{\text{MB}}$  represents the MB-level end-to-end distortion, which is the sum of the distortion contributions of the individual pixels

$$D_{\text{MB}} = \sum_{(x,y) \in \text{MB}} \text{MSE}_{\text{Total}}(T_{x,y}, V_{x',y'}). \quad (4.23)$$

And in (4.22),  $R_{\text{MB}}$  is the bit budget for coding the texture or depth MB associated with the selected mode, which can be readily calculated with the transform coefficients and residual error. For the intra mode, the rate term  $R_{\text{MB}}$  includes the bits needed for the coding of the MB header and the transformed-coefficients for the given texture or depth MB. As for the inter or inter-view coding mode, the rate term  $R_{\text{MB}}$  is the number of bits for coding the MB header, the motion/disparity vector and reference picture, and the transform coefficients of the residual MB. The bit rate for all the coding modes is obtained after entropy coding, as used in traditional video coding in error-free environments.  $\lambda$  is the Lagrange multiplier to control the rate-distortion tradeoff. For the error-prone environment, extensive experimental evidence suggests that there is no significant performance difference between using the Lagrange multiplier tailored to the error-free or error-prone environment, which has also been confirmed in [151]. So the  $\lambda$  is set to the following value tailored to error-free environment.

$$\lambda = 0.85 \times 2^{(\text{QP}-12)/3} \quad (4.24)$$

where QP is the quantization parameter. It should be noted that, another reason for setting the Lagrange multiplier in the error-prone transmission environment to equal that of error-free channel, is based on the consideration of the related rate control strategy adopted. Since the output bit rate is not a priori information in this section, we do not need to vary the Lagrange multiplier. By setting the derivative of the end-to-end rate-distortion cost function to zero, we can easily obtain the Lagrange multiplier in the error-prone environment that is equal to the error-free Lagrange multiplier. This consideration of multiplier has low complexity compared to the known iterative approach introduced in [149], and was demonstrated to work well for error-prone packet-switched network [151].

Note that in (4.22), each MB is independently optimized. This is based on the assumption that the rate and distortion for a given block are impacted only by

the current block and its respective operational coding modes. This assumption is also employed in almost all the video coding standards. In the following, we summarize the main optimization procedure.

While encoding a texture MB, we try all the possible modes for both the texture MB and the corresponding depth MB. For each possible coding mode combination, the total expected distortion and the bit rate are then calculated. With the inter coding mode for the texture or depth MB, the expected distortion of each pixel in the coded texture or depth map is obtained using (4.15), (4.17) and (4.21), whereas for the intra coding mode, the expected decoder distortion is computed with (4.15), (4.19) and (4.21). As for the inter-view coding mode, the expected reconstructed distortion is computed using (4.15), (4.20) and (4.21). Based on the expected distortion of the texture and depth map, the total expected end-to-end distortion of MVD-based 3-D video can be estimated using (4.14). After the rate-distortion costs of all possible coding mode combinations are computed, the optimal mode for the texture MB can be decided via (4.22). When encoding a depth map MB, since the coding mode of the co-located texture MB is already determined, we need to try the possible modes only for the depth map. In a similar way, the optimal mode for depth map coding can be determined using (4.22).

#### 4.6.1 Lagrange Multiplier Determination for Rate-Constrained Coder

In the above approach, the lagrange multiplier is fixed to a constant, and the solution to the unconstrained Lagrangian cost for the given  $\lambda$  results in minimum end-to-end distortion for several possible bit rate point. However, in some scenarios, the bit rate must be controlled to maintain a constant local-average bit rate over time, in which case it is desirable to find a particular value for  $\lambda$  so that upon optimization of (4.22), the resulting bit rate closely matches a given rate constraint  $R_{\text{budget}}$ . Because of the monotonic relationship between  $\lambda$  and bit rate [150], we propose a fast convex search algorithm to find the optimal multiplier for the given  $R_{\text{budget}}$  as follows.

*Step 1:* Find initial  $\lambda_1$  and  $\lambda_2$  such that the resulting  $\sum_{\text{MB}} R(\lambda_1)$  and  $\sum_{\text{MB}} R(\lambda_2)$

satisfy:  $\sum_{\text{MB}} R(\lambda_1) \leq R_{\text{budget}} \leq \sum_{\text{MB}} R(\lambda_2)$ . If either of the equalities hold, the problem is solved. As with most iterative solutions, the choice of a good initial operating point is the key to a fast convergence. In this work, it is assumed the two values of  $\lambda$ ,  $\lambda_1$  and  $\lambda_2$  can be judiciously chosen. A conservative choice for a solvable problem would be  $\lambda_1 = 0$  and  $\lambda_2 = \infty$ .

*Step 2:* Otherwise, let  $\lambda_3 = \sqrt{\lambda_1 \lambda_2}$ , and obtain  $\sum_{\text{MB}} R(\lambda_3)$ . If  $(1 - \delta_1)R_{\text{budget}} \leq \sum_{\text{MB}} R(\lambda_3) \leq (1 + \delta_1)R_{\text{budget}}$ , then the problem is solved. The  $\delta_1$  is a vanishingly small positive number picked to ensure that the lower rate point is picked. Else if  $\sum_{\text{MB}} R(\lambda_3) > (1 + \delta_1)R_{\text{budget}}$ , let  $\lambda_2 = \lambda_3$ . Otherwise let  $\lambda_1 = \lambda_3$ . Repeat step 2.

It is important to note that the fine-tuning of rate is accomplished via a single parameter,  $\lambda$ , with the desirable outcome that no matter what bit rate results, the distortion of the frame will be minimum for that rate. The same technique can also be used to solve the dual optimization problem, i.e., minimize the rate for given distortion constrained.

## 4.7 Experimental Results and Discussions

In this section, we evaluate the performance of the proposed scheme. The JMVC version 8.0 of the MVC reference software is adopted to encode multi-view video sequences and depth maps, and the view synthesis reference software (VSRS) 3.5 [182] is used to render the virtual view. The standard video plus depth sequences “BookArrival”, “Lovebird1”, “Newspaper”, “GT\_Fly”, and “Undo\_Dancer” are chosen for our simulations. Among these sequences, for “BookArrival” containing 16 views with 6.5 cm spacing between adjacent views, the views 8 and 10 are used as the left and right reference views, respectively. For “Lovebird1” containing 12 views with 3.5 cm spacing between adjacent views, the views 6 and 8 are adopted as the left and right views, respectively. For “Newspaper” containing 9 views with 5 cm spacing between adjacent views, views 4 and 6 are served as the left and right reference views, respectively. For the GT\_Fly sequence, the views 5 and 9 are used as the left and right views to render the virtual view “6”. For the Undo\_Dancer sequence, the views 2 and 5 are employed as the left and right views to synthesize

the virtual view “3”. The first three sequences have a resolution of  $1024 \times 768$  samples, while the remaining ones have a resolution of  $1920 \times 1088$  samples. For both texture video and depth map, context-adaptive binary arithmetic coding (CABAC) is used as the entropy coding scheme, and the functions of the variable prediction size and the loop filter are turned on. The search range for disparity estimation and motion estimation is 64. For each multi-view video sequence, each view is encoded with the GOP size of 100 frames, where the first frame in the left view is coded as an I-frame, and the remaining frames are coded as P-frames.

There is only one I-frame in the texture and depth map of the left view, which is assumed to be received error-free. This assumption is to ensure that the transmitted GOP is decodable. Each row of the MBs in a frame constitutes a single slice, which is carried in a separate transport packet. It should be noted that the packet length for all the P frames in our simulations are within the limit of the maximum transmission unit (MTU) for Ethernet. The random packet loss pattern is employed to simulate packet losses. Different packet loss rates 5%, 10% and 20% are tested on both the texture video and depth streams. To simulate the channel, at each packet loss rate, 300 packet loss patterns are randomly generated. For the objective video quality assessment, the luminance peak signal-to-noise ratio (Y-PSNR) is averaged over all the decoded frames and all the implemented channel conditions. In our experiments, the simple yet efficient error concealment method that each damaged block is directly replaced by the co-located one in the previous time picture is employed at the multi-view decoder [152]. The distortion of virtual view synthesis is calculated between the virtual view images synthesized by the original texture and depth images and the decoded texture and depth images.

#### 4.7.1 Estimation Accuracy of the End-to-End Distortion Model

In order to verify the estimation accuracy of the proposed end-to-end distortion model, we compare the estimated and the measured distortions for the BookArrival and Newspaper sequences under packet loss rate of 10%. In the experiments of this subsection, the QP is set to 32 for texture video and depth map coding.

In order to conduct a fair comparison, the measured distortion of MVD video also contains the actually decoded texture video distortion and the actual view rendering distortion. As can be observed from Figure 4.3, the estimated distortion curve sometimes takes on a little jitter due to the impacts of detailed view blending and hole filling algorithms in DIBR. However, it is still clear that the estimated curve shows the similar trend as the measured one. Therefore, the proposed end-to-end distortion model can be utilized to substitute source distortion model for error-resilient MVD-based 3-D video coding. It should be noted that, since the QP value only affects the source coding distortion that comprises only a very small portion of the overall end-to-end distortion, the selection of QP values has no significant impact on the accuracy of the proposed end-to-end distortion model.

#### 4.7.2 Error-Resilient MVD-based 3-D Video Coding

To conduct an objective quality evaluation, the proposed robust MVD-based 3-D video coding algorithm is compared with the Bruno algorithm [140], [153], which is the latest research work in this area to the best knowledge of the authors while drafting this chapter. Table 4.1 summarizes simulation results with a constant QP of 32 for the test sequences at various packet loss rates. As can be observed from Table I, the proposed algorithm yields consistent and significant gains over the Bruno algorithm for the texture video and synthesized virtual view. It is also clear that the Newspaper sequence achieves the maximum average PSNR gains among the test sequences of 2.19 dB, 2.90 dB, and 1.84 dB at the packet loss rates of 5%, 10%, and 20%, respectively. The reason for that is, the Newspaper sequence captures nearer scene than the other sequences, which results in larger geometry error in (4.3), and consequently larger rendered view distortion. Through our proposed rate-distortion optimized mode switching algorithm, more MBs will be selected to be intra-coded so as to suppress error propagation.

It should be noted that, although, the proposed algorithm outperforms the Bruno algorithm in terms of PSNR, the complexity of the proposed algorithm will be higher than the Bruno algorithm due to the joint texture and depth map coding. In our implementation, an average increase of 1.9% with respect



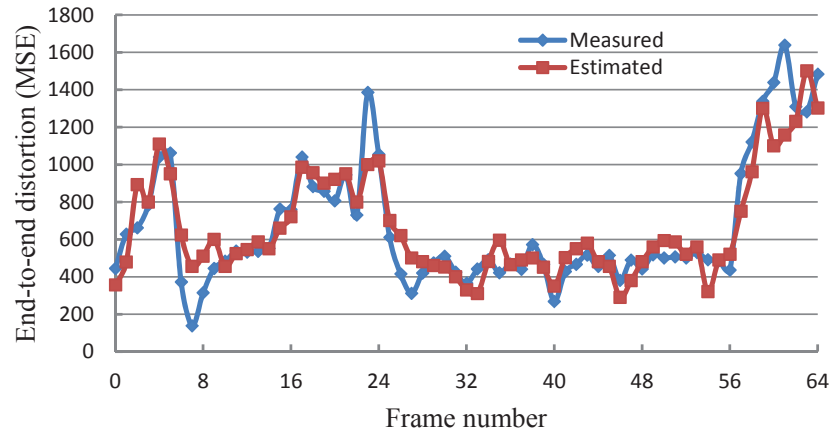
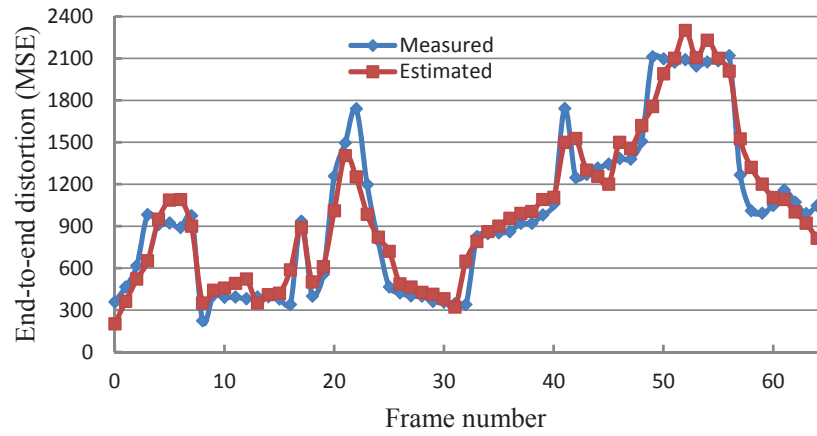
(a) *BookArrival*.(b) *Newspaper*.

Figure 4.3: Frame-by-frame comparison between the measured and estimated end-to-end distortions.

to the Bruno algorithm was registered in terms of the percentage of encoding time. This complexity increase is not significant. Therefore, we can conclude that, compared to the Bruno algorithm, the proposed algorithm offers significant performance benefits at complexity cost that can be neglected.

Table 4.1:

Average PSNR comparison between the Bruno algorithm and the proposed method with a variety of packet loss rates.

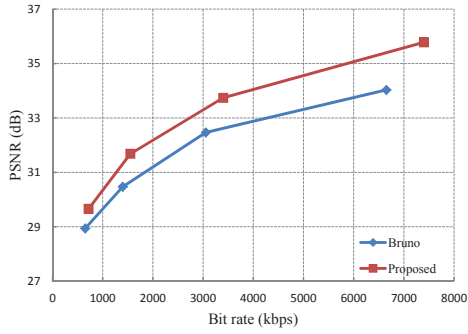
Sequence	View	Y-PSNR (dB) at different loss rates					
		5%		10%		20%	
		Bruno	Proposed	Bruno	Proposed	Bruno	Proposed
Lovebird1	6	31.31	32.84	30.60	30.77	28.33	29.82
	8	32.21	32.72	30.47	31.48	28.91	29.43
	“7”	32.70	34.14	31.52	32.23	29.43	31.11
	<b>Average</b>	<b>32.07</b>	<b>33.23</b>	<b>30.86</b>	<b>31.49</b>	<b>28.89</b>	<b>30.12</b>
BookArrival	8	31.61	32.35	28.52	30.45	26.21	27.12
	10	30.12	32.24	27.10	29.82	24.19	26.16
	“9”	30.22	31.30	27.51	29.54	24.53	25.85
	<b>Average</b>	<b>30.65</b>	<b>31.96</b>	<b>27.71</b>	<b>29.94</b>	<b>24.98</b>	<b>26.38</b>
Newspaper	4	28.49	30.72	26.13	28.56	22.78	25.1
	6	27.69	29.83	24.16	27.58	21.58	23.44
	“5”	28.41	30.62	25.15	27.99	22.66	23.99
	<b>Average</b>	<b>28.20</b>	<b>30.39</b>	<b>25.14</b>	<b>28.04</b>	<b>22.34</b>	<b>24.18</b>
GT_Fly	5	34.78	35.55	33.46	34.92	31.74	33.15
	9	33.83	34.62	32.46	33.91	30.58	32.92
	“6”	34.22	34.94	32.68	34.24	31.17	32.67
	<b>Average</b>	<b>34.27</b>	<b>35.04</b>	<b>32.87</b>	<b>34.36</b>	<b>31.16</b>	<b>32.91</b>
Undo_Dancer	2	31.79	32.89	29.67	30.28	27.89	28.70
	5	29.57	30.40	27.88	28.81	26.71	27.23
	“3”	30.59	31.67	28.75	29.91	27.29	28.41
	<b>Average</b>	<b>30.65</b>	<b>31.65</b>	<b>28.76</b>	<b>29.66</b>	<b>27.30</b>	<b>28.11</b>

Figure 4.4 compares the rate-distortion performance between the proposed scheme and the Bruno approach using the Bjontegaard Delta PSNR (BDPSNR) [13] method. The QPs of this experiment are set to 22, 27, 32, and 37. Both the input texture video and corresponding depth map are coded and decoded using the same QP. Since the virtual views are generated with DIBR technology, the bit

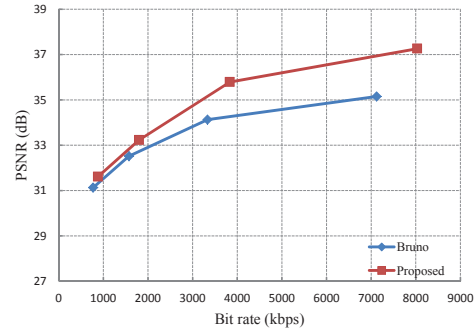
rate for views “7”, “5”, and “6” is the average bit rate for both the textures and depth maps of the left and right coded views. In addition, since the effect of the left and right views on the rendered view is actually similar, only the comparative results for the right captured views and synthesized views are given. A positive BDPSNR means that the proposed method outperforms the conventional Bruno method. From the curves in Figure 4.4, it can be seen that the proposed scheme significantly improves the rate-distortion performance by a gain of 1.01 dB in PSNR compared with the Bruno scheme on average. Besides, the performance gain tends to increase with the bit rates due to the higher quality of the texture video and depth map at low QP case.

Since PSNR may not be meaningful for error resilience and concealment, subjective performance is also evaluated. To subjectively evaluate the simulation results, Figure 4.5 shows a comparison of the 39th frame in the coded view and synthesized virtual view from the Bruno method and our proposed algorithm. For the coded texture video, it can be seen that errors occurred in regions with complex motion are faithfully recovered with our proposed method. This is important for the subjective visual quality since the motion information in an image takes a dominating role in human perception. Meanwhile, the subjective view rendering results also clearly confirm the effectiveness of our proposed scheme. As shown in Figure 4.5(c), the contour deformation has occurred around the head of the man. This is because the depth map errors lead to geometric errors in the captured scene, and the subsequent texture pixel is copied to the wrong spatial location in the synthesized image. Instead, our proposed algorithm can produce better visual rendering results as shown in Figure 4.5(d).

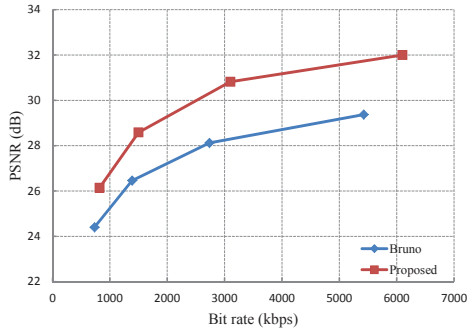
Since disparity-compensated prediction is an important tool for exploiting the redundancy between different views, transmission errors of the distorted texture video or depth map of the left view will propagate to the right view and consequently affect the quality of the right view video. Therefore, as shown in Section 4.5, the proposed distortion model explicitly takes into consideration inter-view error propagation. In order to demonstrate that the received video quality is improved by considering inter-view error propagation, we compare the PSNR results of each frame which are achieved with and without consideration of inter-



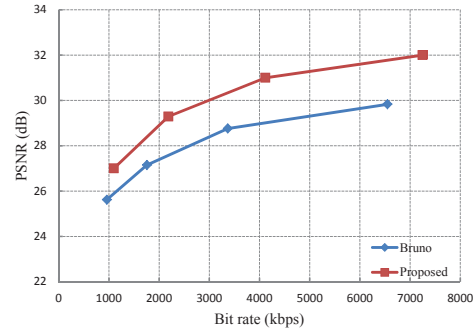
(a) Coded view 8 of the Lovebird1 sequence (BDPSNR gain of 0.74 dB).



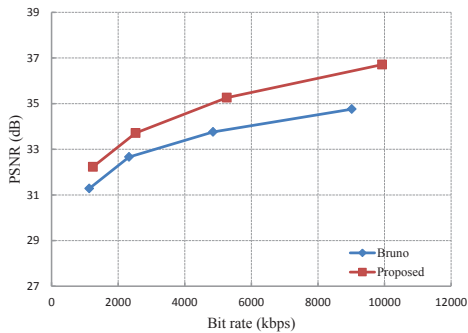
(b) Rendered view "7" of the Lovebird1 sequence (BDPSNR gain of 0.62 dB).



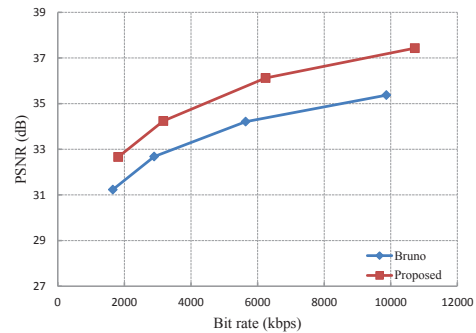
(c) Coded view 6 of the Newspaper sequence (BDPSNR gain of 1.47 dB).



(d) Rendered view "5" of the Newspaper sequence (BDPSNR gain of 1.23 dB).



(e) Coded view 9 of the GT\_Fly sequence (BDPSNR gain of 0.88 dB).



(f) Rendered view "6" of the GT\_Fly sequence (BDPSNR gain of 1.04 dB).

Figure 4.4: Comparison of the rate-distortion curves between the proposed method and the Bruno algorithm with 10% packet loss.



(a) *Decoded texture image of view 10 with the Bruno method.*



(b) *Decoded texture image of view 10 with the proposed method.*

Figure 4.5: Subjective quality comparison for frame 39 of the BookArrival sequence with 10% packet loss rate.



(c) *Rendered texture image of view “9” with the Bruno method.*



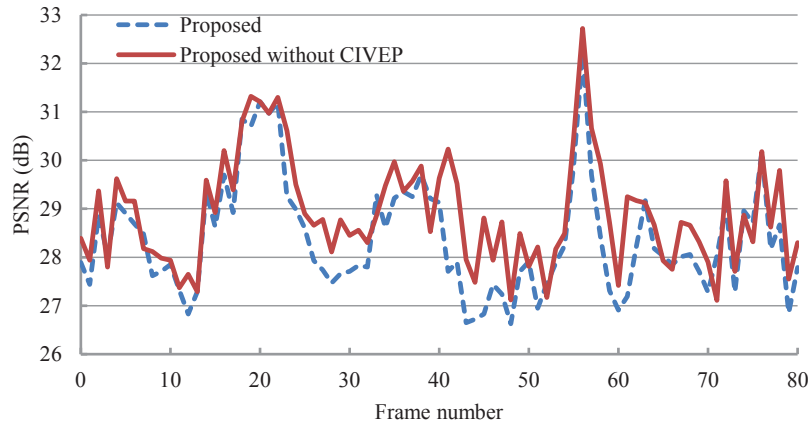
(d) *Rendered texture image of view “9” with the proposed method.*

Figure 4.5: Subjective quality comparison for frame 39 of the BookArrival sequence with 10% packet loss rate.(con’t)

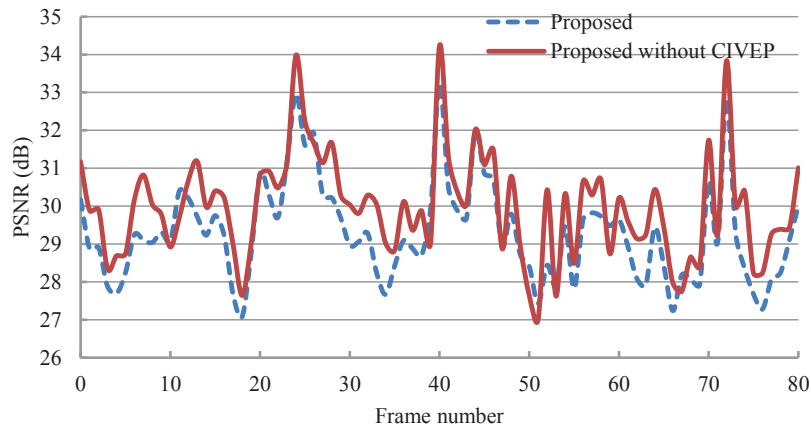
view error propagation. Figure 4.6 shows the PSNR comparison of each frame for the coded view and synthesized view with a packet loss rate of 10%, where the proposed algorithms with and without consideration of inter-view error propagation (CIVEP) are denoted by “Proposed” and “Proposed without CIVEP”, respectively. From the performance comparison, it can be seen that the received quality achieved by the “Proposed” algorithm is always better than that of the “Proposed without CIVEP” algorithm. For the coded view, the average PSNR of the “Proposed” approach is 28.81 dB, which is higher than that of the “Proposed without CIVEP”, i.e., 28.30 dB. For the rendered view, the average PSNR for the “Proposed” approach is 29.91 dB, while for the “Proposed without CIVEP” method it is 29.27 dB. As for some frames of the coded and synthesized views, the PSNR of the “Proposed” approach can be up to 1.3 dB higher than that of the “Proposed without CIVEP” method. From the above comparison, we can conclude that it is of great importance to consider the inter-view dependence during multi-view video distortion analysis and modeling.

### 4.7.3 Comparison with the Rate-distortion Optimization Model in 3D-ATM

In order to further evaluate the comparative performance of the proposed rate-distortion optimized mode switching algorithm, we compare the proposed method with the view synthesis-based rate-distortion optimization (VSRDO) algorithm introduced in [154], which is employed in the 3D-AVC Test Model (3D-ATM) [185]. Since the VSRDO model focuses only on efficient depth map compression without consideration of packet losses, for fair comparison, we incorporate our proposed end-to-end texture and depth distortion model (i.e., (4.15), (4.16), and (4.21)) into the VSRDO model. In this test, the target bit rate is fixed for each test sequence (texture video plus depth) under various packet loss rates, i.e., 4.9 Mbps for the GT\_Fly sequence and 3.3 Mbps for the BookArrival sequence. In texture video and depth map coding, the advanced adaptive rate control mechanism described in [155] is employed in the proposed method and the VSRDO algorithm to ensure the resulting total bit rate as close as possible to the target bit rate. Figure 4.7 plots the average PSNRs of the coded and rendered views



(a) Coded view 5 of the *Undo\_Dancer* sequence.



(b) Rendered view "3" of the *Undo\_Dancer* sequence.

Figure 4.6: PSNR comparison of each frame with a packet loss rate of 10%.



versus the packet loss rate for the proposed error-resilient scheme and the VSRDO algorithm. As can be seen from Figure 4.7, when the packet loss rate is set to 0 (i.e., error-free environment), the performances of the proposed method and the VSRDO scheme are almost identical. However, in the event of packet losses, our proposed scheme significantly outperforms the VSRDO scheme by around 1.39~1.68 dB.

The reason why the proposed method performs superiorly to the VSRDO model can be explained as follows. In the VSRDO model, the view rendering distortion induced by depth errors is assumed to be a distance between the interpolated curves by the uncompressed and decoded group of pixels. In other words, the view rendering distortion due to depth errors can be approximated by the square of the product of the position shift by depth errors and the difference between adjacent reconstructed texture video pixels. In the error-free environment, the reconstructed texture information used to compute the view synthesis distortion can be directly obtained at the encoder. However, in the error-prone environment, due to the possible packet loss in the channel, the corresponding reconstructed texture pixel values are not available and cannot be precisely estimated. This will thus lead to inaccuracy of estimation of the view rendering distortion, and consequently misguide the rate-distortion optimized mode switching. Moreover, the VSRDO algorithm optimizes the depth map coding and texture coding independently, where the view synthesis distortion is employed for depth map coding and the conventional distortion metric is used for texture video coding. This optimization method ignores the inter-dependency between the expected texture video and depth MB distortions. By contrast, our proposed method jointly optimizes the encoding modes of the texture MBs and depth map MBs by minimizing the total expected distortion, in which the expected distortion of both the coded video and synthesized view video is modeled.

#### 4.7.4 Statistical Results of MB Coding Mode Distribution

In order to validate that the proposed mode switching scheme works correctly, Figure 4.8 shows the average percentages of texture and depth map MBs using each coding mode from the proposed coding mode switching method under

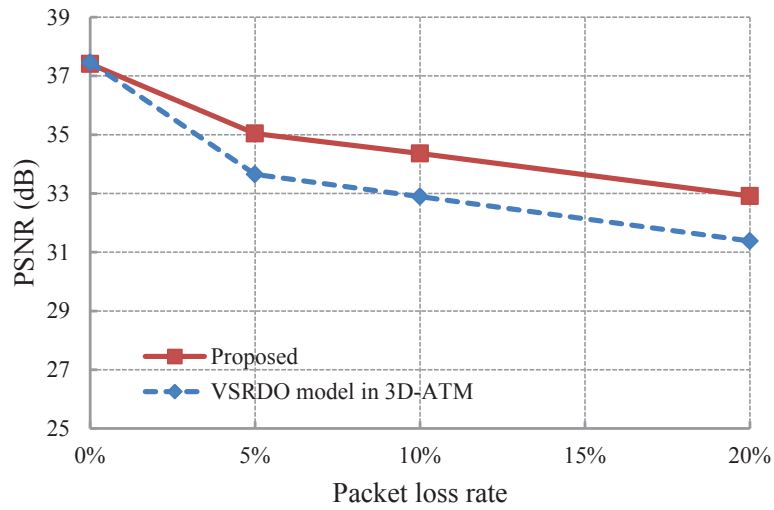
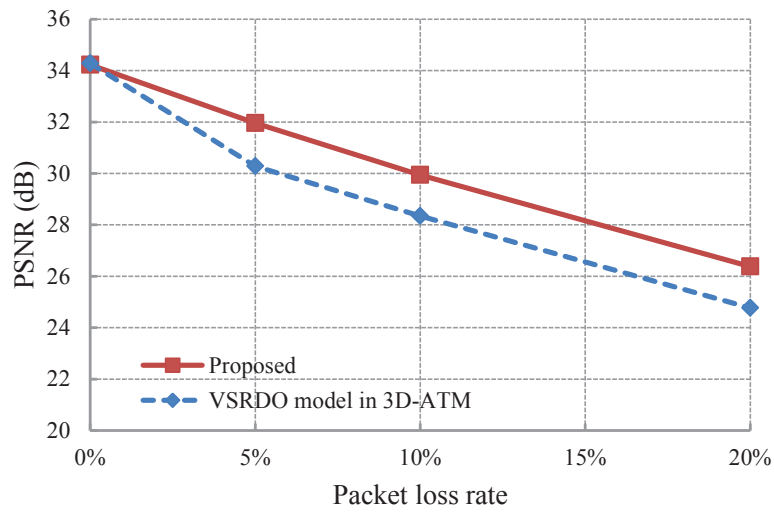
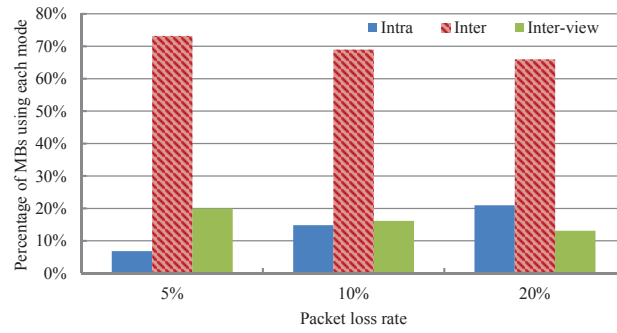
(a) “*GT\_Fly*” sequence.(b) “*BookArrival*” sequence.

Figure 4.7: PSNR versus the packet loss rate for the proposed scheme and the VSRDO model in 3D-ATM.

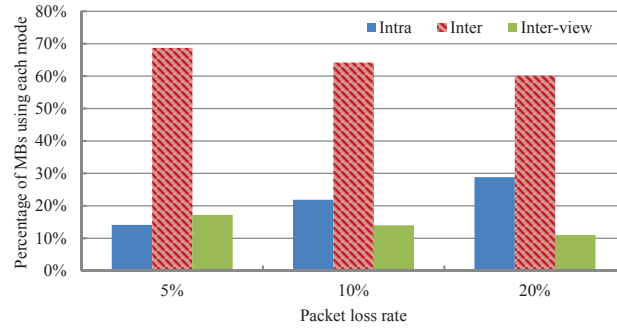
different error rates, where QPs are set to 32 and 37 for Figure 4.8(a)–4.8(b) and Figure 4.8(c)–4.8(d), respectively. As can be observed from Figure 4.8, a large number of MBs select the inter mode during texture and depth map encoding. This is expected because the temporal correlation between two consecutive frames is usually higher than the spatial correlation in a frame or between two neighbouring views. Compared to the inter coding mode, the average inter-view coding mode selection percentage is only about 11.02% – 20.28% under different loss rates. As the packet loss rate increases, the percentage of intra coded texture or depth map MBs also increases. This is due to the fact, with high packet loss rates, the probability of propagated mismatch errors is high, and then more intra MBs are required to mitigate the mismatch error propagation and limit the distortion caused by packet losses. Meanwhile, in the case of two different QP settings, it is observed that the increments of intra refresh percentage for error-resilient video coding are almost the same. Based on the achieved number of intra coded MBs, it can be concluded that the proposed rate-distortion optimized mode switching scheme can adaptively intra refresh MBs to efficiently stop the channel error propagation. In addition, it should be noted that, since the depth map typically consists of many smooth regions, the percentage of intra coded MBs of the depth map is higher than that of the texture video with the same QP setting.

#### 4.7.5 Computational Complexity Analysis

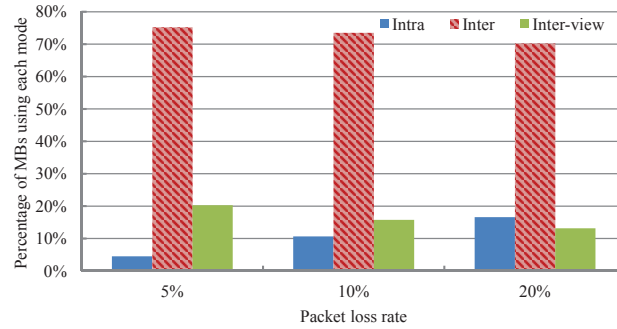
The proposed error-resilient algorithm introduces additional complexity for the MVC encoder. However, the additional computational costs are very modest and well justified by the considerable error resilience performance improvements achieved. Since the proposed rate-distortion optimized mode switching method is used in substitution of the standard rate-distortion optimization employed in coding of texture and depth map, it imposes nearly no additional computational complexity costs. Therefore, the critical complexity increase originates in the end-to-end distortion estimation for MVD-based 3-D video streaming. As can be observed from (4.15), the derivation for the expected texture or depth map distortion involves determining the error propagation distortion and the error conceal-



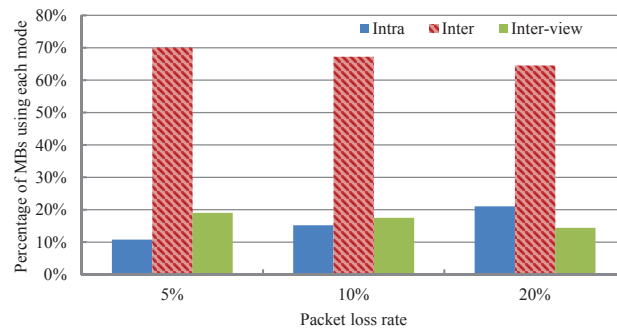
(a) *Texture video of the BookArrival sequence at  $QP = 32$ .*



(b) *Depth map of the BookArrival sequence at  $QP = 32$ .*



(c) *Texture video of the BookArrival sequence at  $QP = 37$ .*



(d) *Depth map of the BookArrival sequence at  $QP = 37$ .*

ment distortion defined by (4.16) and (4.21), respectively. As for the computation of the error propagation distortion, since the second term in (4.16) represents the distortion between the reconstructed and error concealed values, this term contributes one addition and one multiplication to the complexity count. So it needs four additions and three multiplications to calculate the error propagation distortion for each pixel. In a similar way, calculating the error concealment distortion in (4.21) requires two additions and one multiplication. Thus, based on (4.15), (4.16) and (4.21), it requires eight additions and six multiplications to derive the expected reconstructed distortion for each pixel in coded texture video and depth map. Finally, considering the expected distortion in the synthesized view, the estimation of the overall end-to-end distortion for MVD-based 3-D video in (4.14) requires a total of fifteen additions and twelve multiplications. Furthermore, the error concealment algorithm also has to be implemented at the encoder for each block. However, the assumed approach which uses the block at the corresponding position in the previous frame to conceal the missing block requires negligible additional complexity. Table 4.2 shows the encoder complexity comparison between the proposed method and the original JMVC 8.0 without error resiliency consideration, from which the complexity of the proposed method increases only by 4.3% as opposed to the original JMVC 8.0. This small increase in computational complexity is acceptable with the significant video quality improvement.

Table 4.2:

Computational complexity comparison between the proposed method and the original JMVC 8.0.

Sequence	Encoding time comparison		Time increment
	JMVC 8.0 (s)	Proposed (s)	
Lovebird1	5312.36	5478.38	3.1%
BookArrival	5824.65	6096.76	4.5%
Newspaper	5489.17	5672.72	3.3%
GT_Fly	9235.45	9693.91	4.8%
Undo_Dancer	10546.21	11128.37	5.3%
<b>Average</b>	<b>7281.56</b>	<b>7614.03</b>	<b>4.3%</b>

## 4.8 Summary

In this chapter, we have proposed an efficient mode switching algorithm to improve the coded and synthesized view video reconstruction qualities. An overall expected distortion model for MVD-based 3-D video is first derived, in which we focus mostly on the theoretical analysis of the impact of the texture video reconstruction errors and depth map reconstruction errors on the quality of the virtual view. It is shown that the synthesis distortion due to depth errors is the product of the magnitude of squared position error and the energy density of the reconstructed texture. The accuracy of the proposed distortion model is demonstrated via simulation results. The benefit of considering inter-view dependence during multi-view video distortion modeling is also verified.

Then, based on the derived distortion model, a new rate-distortion optimized error resilience algorithm is developed to adaptively select the inter-view, inter or intra coding mode for encoding the texture and depth map. In particular, the proposed optimization approaches can fine tune the total rate or distortion and thus follow any bit rate or distortion profile. Compared to existing error-resilient schemes for MVD-based 3-D video transmission, our experimental results show significant performance gains on both objective and subjective visual quality, at the cost of modest additional complexity.

We also show the percentages of different texture and depth modes chosen during encoding, which clearly illustrates that the proposed mode switching scheme can truly strike the optimal trade-off between coding efficiency and error resilience performance.

# Chapter 5

## Disparity Vector Correction for View Synthesis Prediction Based 3-D Video Transmission

### 5.1 Introduction

As discussed before, to enable 3DTV and FVV functionalities, the MVD representation, which facilitates DIBR [156], has been generally determined to be the best format for representing 3-D video because of its low complexity and compatibility with the current legacy devices [157]. This MVD format consists of texture video and depth sequences for a limited number of nearby camera views of the same natural scene. By using the MVD representation, user-chosen virtual views can be efficiently rendered with high compression capabilities.

Although the MVD representation could greatly reduce the data volume of 3-D video being transmitted, the presence of multiple cameras as well as additional depth information brings new challenges for compression. Since the depth maps can be treated as the monochromatic videos, they can also be compressed by existing compression techniques, such as H.264/AVC and MVC [158]. In a straightforward way, the multi-view video and depth map are independently encoded using existing compression standards, just as what is done in the previous chapter, which is called the conventional MVD-based 3-D video coding framework. However, in this framework, the 3-D video coding efficiency is not fully

optimized because the redundancy between the multi-view video and depth map is not fully exploited. Therefore, some other MVD-based coding schemes have been proposed to exploit this inter-component correlation to further increase the overall coding efficiency. View synthesis prediction (VSP) was first introduced into MVC in [159], which synthesized additional virtual frames as complementary reference frames for non-translational disparity-compensated prediction. Based on the principle of VSP, Yea *et al.* [160] devised a rate-distortion optimized MVD framework to improve the coding performance. Besides, Shimizu *et al.* [161] also designed a related VSP scheme, in which the original video of base views and the residue of enhancement views are encoded by a traditional video coding process. To improve the performance of VSP, an adaptive depth quantization scheme was developed in [162]. In [163], in order to reduce the additional decoder complexity, Tian *et al.* proposed a backward VSP design using the depth map of the current view to perform a pixel-based warping, where the disparity vector of the current depth block is derived from the neighboring blocks in the texture-first coding order scenario. The VSP based 3-D video coding framework is termed the improved MVD-based coding framework in this chapter. Due to the potential coding efficiency improvement on the conventional 3-D coding framework, VSP has been adopted into both the upcoming H.264/AVC-based [164] and HEVC-based [165] 3-D video coding standards.

## 5.2 Related Work

Similar to 2-D video, 3-D video is also very sensitive to transmission errors due to its hybrid predictive coding structure, where the errors from the current frame may propagate to the future frames and often bring significant degradation to the video quality at the receiver end. To achieve higher packet loss resilience, a number of techniques have been proposed to enhance the robustness of 3-D video transmission against packet losses. They could be broadly classified into two categories, namely decoder error concealment and source-level error-resilient coding. Given that a texture sequence and its associated depth represent the same scene from the same point of view, they should have similar motion characteristics.



Therefore, some methods have been developed recently to conceal the corrupted 3-D videos by making use of the depth information [152], [166]. These works exploit the statistical correlation between the texture video and corresponding depth map to better select the missing motion vectors based on the matching criterion of depth similarity. However, these methods may not work well if depth map boundaries are fuzzy and not well aligned with the texture video. Furthermore, they only focus on a single view plus depth and do not consider the inter-view geometry correlation. To better exploit the inter-view geometry information provided by the depth signal, an enhanced temporal error concealment method has been proposed by estimating the missing motion vectors of a corrupted macroblock (MB) with the help of the synthesized information [167]. The experimental results of this algorithm have shown superior performance compared with previously proposed methods, while still ignoring the effect of the common geometry error due to the loss of depth information. On the other hand, in the aspect of error-resilient 3-D video coding, Machiavello *et al.* introduced a reference frame optimization scheme at the block level for loss-resilient depth map coding to minimize the expected synthesized view distortion [139]. Thereafter, this idea was extended to encoding of both texture and depth map [168]. In these two algorithms, inter-view error propagation on transmission distortion is not considered and only the distortion in the synthesized views is modeled. In [169], in order to fully improve the overall quality of reconstructed 3-D video, a rate-distortion optimized coding mode switching scheme was presented for robust MVD-based 3-D video coding, in which the summative end-to-end distortions of both the rendered view and coded texture video are characterized. Although the above error control methods can contribute to some reasonable improvements on the error resilience performance of 3-D video systems, they are all built upon the conventional MVD-based 3-D video coding framework.

### 5.3 Contribution of This Chapter

In this chapter, we will concentrate our efforts on developing a post-processing error correction scheme for the new VSP based 3-D video coding framework,

which mainly solves the research problem 3 defined in Chapter 1. Compared to the conventional 3-D video coding framework, the transmission error of the coded texture video and depth map will first propagate to the synthesized reference view along the VSP path, and then particularly lead to prediction position error for the dependent view. Such error propagation may cause substantial deterioration to the video quality of both coded and rendered views. To mitigate the effects of the newly presented inter-view error propagation, we first analyse the monotonic relationship between the rendering position error and the depth error, in which the depth error caused by lost packets is estimated based on the received depth map bit stream and the deterministic knowledge of the actual loss pattern. Then based on the derived reconstruction depth errors, the disparity vectors can be corrected to find the matching pixels in the synthetic reference picture. Our simulations show that the estimated depth error model is highly accurate, and the performance of the proposed algorithm with the estimated depth error is approximately close to that of the proposed algorithm with the real depth error, which can give the optimum rate-distortion performance. To the best of the authors' knowledge, this work is the first and important attempt to improve the error resilience performance of the compressed VSP based 3-D video streaming.

In the following, we first analyze the error propagation problem in the VSP, and define a new kind of transmission error, i.e., prediction position error. Then, we provide a detailed description of the algorithm used to solve the prediction position shift problem.

## 5.4 View Synthesis Prediction and Error Propagation

Disparity-compensated prediction is a well-known technique for exploiting the redundancy among simultaneously captured views of the same scene, which can provide compression gains when the temporal correlation is lower than the spatial correlation. e.g., objects entering or leaving the scene, or fast motion. However, it does not utilize some essential features of multi-view video. While block translation is good for predicting temporally adjacent frames, it is less accurate

for predicting spatially adjacent ones because the disparity of an object in one frame relative to another depends on the distance of the object to the camera, as well as the camera parameters. To exploit these new features of multi-view video, view synthesis has been proposed for enhanced prediction in multi-view 3-D video coding. As illustrated in Figure 5.1, following the 3-D video coding standard specification in [157], we emphasize on a two-view coding configuration with 1D parallel camera setting, in which view synthesis is employed as an alternative means of prediction. The figure shows a left view and a right view, with each view composed of a texture video and a depth map. Specifically, the left view is firstly encoded by the traditional motion-compensated prediction, which can be compatible with the H.264/AVC or HEVC standard bit stream. Then, a virtual version of the right target viewpoint is synthesized from the already encoded left view according to the reconstructed depth information and camera parameters. This virtual view will exhibit a object structure more similar to the original right view. However, after the projection, there will be some dis-occluded area in the synthesized view. For the sake of simplicity, we assume the encoder performs the linear interpolation to fill these dis-occlusion holes [170]. More sophisticated inpainting methods can be found in [171]. It should be noted that we use the reconstructed depth map instead of the original depth to synthesize the virtual reference view at the encoder side <sup>1</sup>. Finally, based on the synthesized reference view, disparity-compensated prediction is employed to encode the texture video of the right view in addition to the existing temporal prediction <sup>2</sup>. Note that in this study, the VSP is applied to texture coding only. Nevertheless, similar procedures of the VSP can also be applied to the depth component [172].

According to the improved 3-D video coding framework, when the texture

---

<sup>1</sup>Nevertheless, our proposed disparity vector correction algorithm in the following can also be applied to the case when the original depth map is used to generate the synthetic reference view.

<sup>2</sup>Since the error propagation behavior under traditional disparity-compensated prediction has been extensively studied, the transmission errors of that can be effectively mitigated by the previous approaches developed in the conventional MVD-based 3D video coding framework. In order to focus on the analysis of the error propagation behaviour under view synthesis based inter-view prediction, we disable the translational disparity compensation prediction directly from the left view.

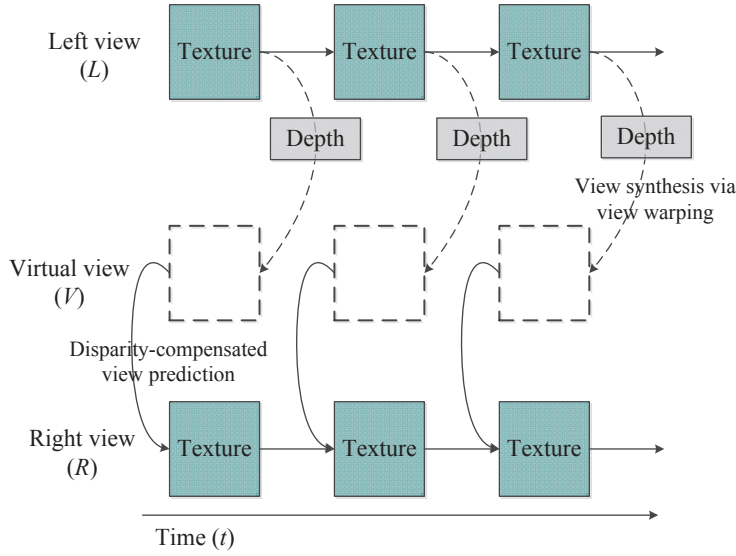


Figure 5.1: A typical prediction structure for VSP based 3-D video coding (parallel camera setup).

video stream and its associated depth stream are transmitted over an error-prone channel, they are highly susceptible to channel errors due to the complex prediction loops. Therefore, it normally requires development of error control techniques to guarantee the coded and synthesized view quality at the decoder. Since the depth map is encoded by traditional joint motion/disparity-estimation-based MVC structure, its error propagation behaviour is actually the same as that of the MVC-based 3-D video transmission. Therefore, the error control methods proposed for the conventional 3-D video coding framework can be used to suppress depth error propagation. As for the texture video transmission, the left view and right view exhibit different characteristics facing transmission errors. For the left view, the texture video is encoded by the temporal motion-compensated prediction without the use of depth map. So the error propagation behaviour of the texture video of the left view is also similar to that of single view video transmission. On the other hand, due to the view synthesis based inter-view prediction, the transmission errors of the texture frame in the right view not only come from itself, but also come from the synthetic reference frame. As for the latter case, since the synthesized frame is rendered from the coded texture and depth map of the left view, obviously, the transmission errors of the distorted texture and depth map will propagate to the synthesized virtual view frame along the warping path.

However, because the depth maps are only used to aid in view rendering but not themselves directly viewed by the end users, the distortion of depth map caused by packet losses will lead to geometry displacement in the virtual view images, which is fundamentally different from the channel distortion afflicting luma and chroma data in standard 2-D video. Note that the synthesized view images will still be used as the disparity compensation reference frames for coding the right view. In this way, the resulting geometry errors will further propagate to the right view, which often cause more severe degradation in video presentation quality than the ordinary transmission errors. Therefore, unlike other experimental approaches for error control reported in the literature, this work will mainly focus on how to eliminate the new type of error propagation introduced by VSP.

## 5.5 Problem Formulation

To better understand the proposed algorithm, we analyse the formulation of the above-mentioned view synthesis based inter-view error propagation problem in this section. The whole illustration of the effect of the prediction position error induced by VSP is presented in Figure 5.2. At the encoder, for a pixel  $(x_l, y_l)$  in the encoded texture image of the left view, it can be projected to the pixel  $(x', y')$  in the virtual image using the camera parameters and compressed depth information. While encoding the right view, pixel  $(x', y')$  is used as a reference pixel to predict the pixel  $(x_r, y_r)$  in the right texture image<sup>3</sup>. After encoding the right view, the residual signal of prediction and disparity information of pixel  $(x_r, y_r)$  are transmitted over the error-prone networks.

At the receiver, the texture video and depth map of left view are first decoded separately using the conventional temporal motion-compensation-based decoder. While using the distorted texture video and depth map to synthesize the virtual reference view at the decoder, due to the reconstructed depth map errors that have occurred during transmission, the projection of the pixel  $(x_l, y_l)$  moves from  $(x', y')$  to  $(x'', y'')$ , and this effect is known as geometry error. This geometry error may continuously propagate to the right view along the disparity-compensated

---

<sup>3</sup>It is assumed that the inter-view correlation between the right texture image and the virtual view image for this pixel is larger than its temporal correlation.

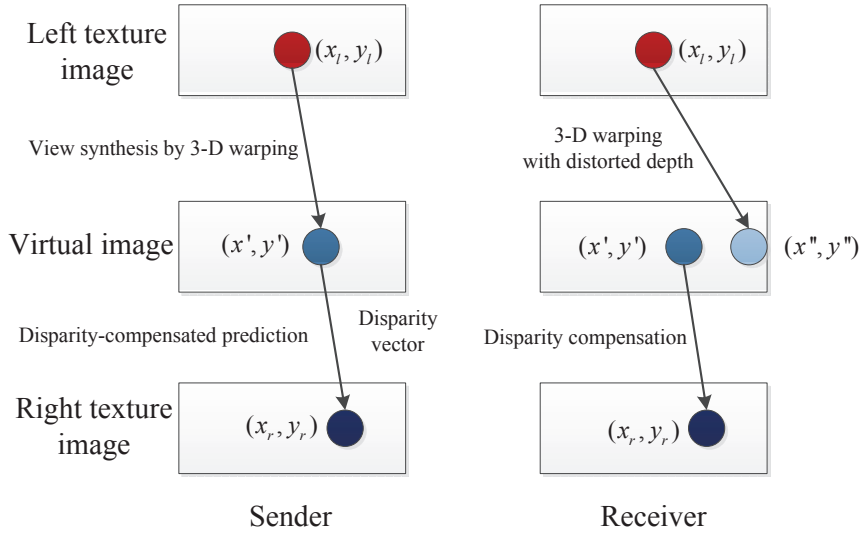


Figure 5.2: Illustration of the effect of the prediction position error introduced by VSP during the reconstruction of the right view.

prediction path. During the reconstruction of the pixel  $(x_r, y_r)$  of the right view, if the packet containing the disparity information and residue is lost, the decoder performs error concealment in a manner identical to that of conventional 3-D video transmission. For example, the disparity vector of a missing MB can be estimated as the median of the disparity vectors of the neighboring MBs in the current texture frame or the corresponding depth map frame. The pixels in the texture frame of the left view, which are pointed to by the estimated disparity vector, are then used to replace the missing pixels in the current frame. Contrarily, if the current packet is received correctly, the decoder can access to both the disparity vector and the prediction residue. However, due to the rendering position error of 3-D warping, the disparity compensation reference pixel at position  $(x', y')$  in the synthetic image pointed to by the received disparity vector is potentially different from the synthesized pixel used by the encoder. Continuing to use pixel  $(x', y')$  for disparity-compensated reconstruction will lead to a significant prediction mismatch between the encoder and decoder. As a matter of fact, the correct disparity compensation reference pixel here should be pixel  $(x'', y'')$ . Therefore, in this case, the incorrect projection of the pixel in the left reference image will cause the coordinate difference between the disparity compensation reference pixels, i.e., the so-called “prediction position error”, for the decoding of the right view. This prediction position error in turn will translate into errors

in the luminance or chrominance of the right view. At the same time, it should be noted that the reconstruction errors of the texture frame in the left view will also spread to the right view through the 3-D warping operation.

To be more precise, given that the data packet of pixel  $(x_r, y_r)$  is received error-free, the reconstructed pixel  $(x_r, y_r)$  will be affected by two major types of transmission errors. The first one is the above defined “prediction position error”, which is mainly induced by the reconstructed error of the left depth map. The second one is the already existing errors in the inexact prediction pixel  $(x', y')$ . This type of errors represents the pixel intensity change, which is caused by directly copying the erroneous texture pixel from the left view to the synthesized pixel  $(x', y')$ . Since the rendering view quality of an arbitrary viewpoint at the decoder highly depends on the quality of the texture videos and depth maps of both the coded views, in order to minimize the reconstructed synthesized distortion of an interpolated view, it is imperative to contain the adverse effects of network packet losses that may arise during texture video and depth map transmission. Toward this goal, on one hand, to mitigate the effect of the prediction position error, it is much desired to let  $(x_r, y_r)$  find the actual prediction pixel  $(x'', y'')$  in the virtual image during decoding. This can be regarded as disparity vector correction, which we will elaborate in the next section. On the other hand, since the texture errors of the left view affect the rendered view quality by simply adding noise to the luminance or chrominance level of each pixel, the synthesized view error inherited from the texture error of the left view can be compensated by the aforementioned methods in the conventional 3-D video coding framework.

## 5.6 Proposed Disparity Vector Correction Algorithm

In the emerging 3-D video communication system, in order to produce the large number of high-quality views required for an auto-stereoscopic display with the currently prevailing network, an effective error control mechanism against possible transmission errors is indispensable. In light of this, we develop a novel disparity vector correction algorithm for the VSP based MVD data transmission in this

section.

The major contributions of this proposed scheme distinguished from the previous work are two-fold.

Firstly, we propose to use the depth map error to correct the geometry error for the pixels whose disparity vectors are correctly received. This idea is potentially different from the previous work which attempt to provide an estimate of the lost disparity vector based on the correctly decoded samples as well as any other helpful information.

Secondly, the pixel-level depth map error can be precisely estimated at the decoder with the deterministic knowledge of the actual loss pattern, in which the approximation of the innovation term is analytically demonstrated through theoretical derivation and experimental observations. The proposed depth error model is also distinctly different from the distortion estimation approaches in the previous work performed at the encoder. Therefore, based on these two contributions, the prediction position error can be effectively eliminated using some very simple and intuitive formulas, with negligible extra computational complexity. In the following, we describe the details of the proposed disparity vector correction scheme.

As analysed before, even if the bit streams of the right view are received correctly, the decoded texture pixels still suffer from the prediction position errors. To prevent that kind of error propagation, it is necessary to correct the disparity vectors to find the corresponding reference pixels as those used in the encoder. As clearly illustrated in Figure 5.3, when decoding pixel  $(x_r, y_r)$ , it is appropriate to use pixel  $(x'', y'')$  synthesized from pixel  $(x_l, y_l)$  for disparity compensation. However, according to the received disparity vector, the decoder can only find the pixel at the position  $(x', y')$ . Therefore, the key technology is to correct the disparity vector to locate the matching prediction pixel  $(x'', y'')$  from the synthetic image. In other words, the disparity error  $\Delta d_t^{x_l, y_l}$  between positions  $(x', y')$  and  $(x'', y'')$  in the synthetic reference view needs to be determined when decoding pixel  $(x_r, y_r)$ .

At the encoder, when the virtual view is synthesized using the compressed texture video and depth map of the left view, according to the 3-D warping



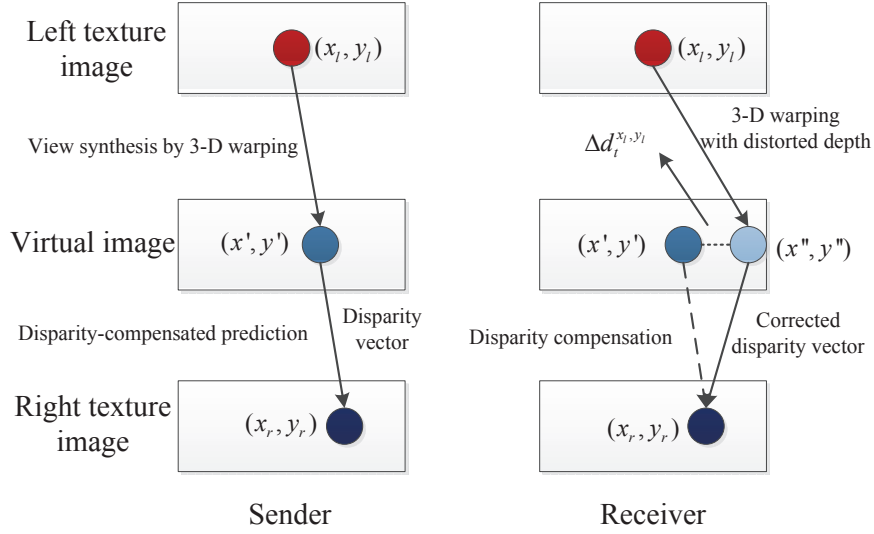


Figure 5.3: Illustration of the proposed disparity vector correction scheme when decoding the right view.

process in the virtual view generation, the horizontal disparity between pixel  $(x_l, y_l)$  in the left reference view image and the corresponding pixel  $(x', y')$  in the rendered view image can be calculated by the following equation as [173], [174]<sup>4</sup>:

$$d_t^{x_l, y_l} = fL / \hat{Z}_t^{x_l, y_l} \quad (5.1)$$

where  $f$  represents the common focal length of the left view camera, and  $L$  denotes the baseline distance between the virtual viewpoint and the left view camera.  $\hat{Z}_t^{x_l, y_l}$  is the physical depth value of pixel  $(x_l, y_l)$  in compressed depth frame  $t$ , which has a relationship with the pixel value  $\hat{D}_t^{x_l, y_l}$  of pixel  $(x_l, y_l)$  in the compressed depth map image; i.e.,

$$\hat{Z}_t^{x_l, y_l} = \left[ \frac{\hat{D}_t^{x_l, y_l}}{255} \times \left( \frac{1}{Z_{\text{near}}} - \frac{1}{Z_{\text{far}}} \right) + \frac{1}{Z_{\text{far}}} \right]^{-1} \quad (5.2)$$

where  $Z_{\text{near}}$  and  $Z_{\text{far}}$  are the values of the nearest and farthest depth of the scene, corresponding to depth pixel values 0 and 255, respectively.

While using the decoded texture video and depth map to synthesize the virtual view, due to the depth map reconstructed error, the projection of pixel  $(x_l, y_l)$  moves from  $(x', y')$  to  $(x'', y'')$ , as shown in Figure 5.3. Based on (5.1) and (5.2), the corresponding disparity reconstruction error  $\Delta d_t^{x_l, y_l}$  between these two warped

<sup>4</sup>Note that the vertical disparity is zero due to the parallel camera setup.

pixels can be estimated from the depth error, which is given by

$$\begin{aligned}
\Delta d_t^{x_l, y_l} &= |fL/\hat{Z}_t^{x_l, y_l} - fL/\tilde{Z}_t^{x_l, y_l}| \\
&= \frac{fL}{255} \left( \frac{1}{Z_{\text{near}}} - \frac{1}{Z_{\text{far}}} \right) |\hat{D}_t^{x_l, y_l} - \tilde{D}_t^{x_l, y_l}| \\
&= \frac{fL}{255} \left( \frac{1}{Z_{\text{near}}} - \frac{1}{Z_{\text{far}}} \right) |e_t^{x_l, y_l}|
\end{aligned} \tag{5.3}$$

where  $\tilde{D}_t^{x_l, y_l}$  and  $\tilde{Z}_t^{x_l, y_l}$  denote the pixel value and the depth value of pixel  $(x_l, y_l)$  in the decoded depth map, respectively.  $e_t^{x_l, y_l}$  is used to represent the reconstructed error of pixel  $(x_l, y_l)$  in depth frame  $t$  in the left view at the decoder due to packet losses. It can be concluded from (5.3) that the rendering position error has a linear relationship with the depth error, focal length and the camera baseline distance with the given texture video. Note that the expression in (5.3) can also be derived in the lossy compression of MVD in a similar way [175]. In order to calculate the disparity error between pixels  $(x', y')$  and  $(x'', y'')$ , the depth error  $e_t^{x_l, y_l}$  of pixel  $(x_l, y_l)$  must be first computed.

Recently, a number of methods of estimating the channel-induced distortion at the encoder have been presented, in which the actual error pattern is unknown and a statistical characterization of the channel is given [176], [177], [178]. The goal of these models is to optimally choose encoding parameters to obtain the best average video quality across the range of possible packet losses. However, since the depth error is used to determine the disparity error in decoding the right view, the depth distortion of the left view should be estimated at the decoder side. In comparison with the distortion estimation methods at the encoder, distortion estimation at the decoder is somewhat simplified by the deterministic knowledge of the actual error pattern. However, the unavailability of the error-free reconstructed depth frames, which is typical in practical applications, complicates the estimation of the distortion. In this work, based on the received depth map bit stream and the deterministic knowledge of actual loss pattern, we try to recursively estimate the reconstructed error induced by packet losses in the depth map sequence at the decoder. In order to estimate  $e_t^{x_l, y_l}$  in (5.3), the proposed method considers two separate cases, depending on whether pixel  $(x_l, y_l)$  of the depth map frame has been lost or correctly received.

If pixel  $(x_l, y_l)$  is an intra-coded pixel in the depth map of the left view and correctly received, then the decoder can reconstruct the pixel exactly, and thus

$e_t^{x_l, y_l} = 0$ . If pixel  $(x_l, y_l)$  is an inter-coded pixel, it is assumed that pixel  $(x_l, y_l)$  in depth frame  $t$  is predicted from pixel  $\theta(x_l, y_l)$  in frame  $t-1$ , where  $\theta(\cdot)$  refers to the operator to calculate the spatial position of the reference pixel. When pixel  $(x_l, y_l)$  is received correctly, the decoder adds the received difference signal  $\hat{r}_t^{x_l, y_l}$  to the motion-compensated prediction signal

$$\tilde{D}_t^{x_l, y_l} = \tilde{D}_{t-1}^{\theta(x_l, y_l)} + \hat{r}_t^{x_l, y_l}. \quad (5.4)$$

Then the depth error of pixel  $(x_l, y_l)$  at the decoder can be written as

$$\begin{aligned} e_t^{x_l, y_l} &= \hat{D}_t^{x_l, y_l} - \tilde{D}_t^{x_l, y_l} \\ &= (\hat{D}_{t-1}^{\theta(x_l, y_l)} + \hat{r}_t^{x_l, y_l}) - (\tilde{D}_{t-1}^{\theta(x_l, y_l)} + \hat{r}_t^{x_l, y_l}) \\ &= \hat{D}_{t-1}^{\theta(x_l, y_l)} - \tilde{D}_{t-1}^{\theta(x_l, y_l)} \\ &= e_{t-1}^{\theta(x_l, y_l)}. \end{aligned} \quad (5.5)$$

It can be observed from (5.5) that the decoder error is purely caused by the mismatch between the prediction references when pixel  $(x_l, y_l)$  is correctly received.

If pixel  $(x_l, y_l)$  in the current depth frame (i.e., intra-coded or inter-coded pixel) is lost during transmission, then the decoder performs a pixel copy concealment from the same position of the previous depth frame  $t-1$ , i.e., the concealed motion vector is set to zero. The decoder depth error of  $(x_l, y_l)$  can thus be represented as

$$\begin{aligned} e_t^{x_l, y_l} &= \hat{D}_t^{x_l, y_l} - \tilde{D}_{t-1}^{x_l, y_l} \\ &= (\hat{D}_t^{x_l, y_l} - \hat{D}_{t-1}^{x_l, y_l}) + (\hat{D}_{t-1}^{x_l, y_l} - \tilde{D}_{t-1}^{x_l, y_l}) \\ &= (\hat{D}_t^{x_l, y_l} - \hat{D}_{t-1}^{x_l, y_l}) + e_{t-1}^{x_l, y_l}. \end{aligned} \quad (5.6)$$

Defining  $\hat{\delta} = \hat{D}_t^{x_l, y_l} - \hat{D}_{t-1}^{x_l, y_l}$ , the above equation can be rewritten as

$$e_t^{x_l, y_l} = \hat{\delta} + e_{t-1}^{x_l, y_l}. \quad (5.7)$$

The second term in (5.7) is the propagation of errors from previous packet losses occurred in the depth map frame. The first term is an innovation term, characterizing the new error introduced by the lost packets in the current depth map frame. Since a depth map explicitly captures the 3-D structure of a scene, it contains large areas of smoothly changing grey levels and only jumps at the object

boundary.  $\hat{\delta}$  can be estimated as the difference among the co-located pixels of the two previously decoded depth map frames, i.e.,  $\tilde{D}_{t-1}^{x_l, y_l} - \tilde{D}_{t-2}^{x_l, y_l}$ . For the initial frame in which one or two previous frames are not available,  $\hat{\delta}$  is estimated from spatially neighbouring pixels with similar depth.

In order to justify that the innovation term in (5.7) can be estimated using the difference between the co-located pixels from the two preceding depth frames, we develop a statistical model. If we define  $\hat{\mu}_t = E[\hat{D}_t^{x_l, y_l}]$  as the mean value, where  $E[\cdot]$  denotes the expectation operation, then we can write  $\hat{\delta}^2$  due to this pixel being lost as

$$\begin{aligned}\hat{\delta}^2 &= E[(\hat{D}_t^{x_l, y_l} - \hat{\mu}_t + \hat{\mu}_t - \hat{D}_{t-1}^{x_l, y_l} + \hat{\mu}_{t-1} - \hat{\mu}_{t-1})^2] \\ &= E[(\hat{D}_t^{x_l, y_l} - \hat{\mu}_t)^2] + E[(\hat{D}_{t-1}^{x_l, y_l} - \hat{\mu}_{t-1})^2] \\ &\quad - 2E[(\hat{D}_t^{x_l, y_l} - \hat{\mu}_t)(\hat{D}_{t-1}^{x_l, y_l} - \hat{\mu}_{t-1})] \\ &\quad + (\hat{\mu}_t - \hat{\mu}_{t-1})^2.\end{aligned}\tag{5.8}$$

Note that, while deriving (5.8), it is assumed that  $(\hat{D}_t^{x_l, y_l} - \hat{\mu}_t - \hat{D}_{t-1}^{x_l, y_l} + \hat{\mu}_{t-1})$  is uncorrelated with  $(\hat{\mu}_t - \hat{\mu}_{t-1})$  [179]. If we let  $\hat{\sigma}_t^2 = E[(\hat{D}_t^{x_l, y_l} - \hat{\mu}_t)^2]$  represent the variance of  $\hat{D}_t^{x_l, y_l}$ ,  $\hat{\beta}_{t,t-1} = E[(\hat{D}_t^{x_l, y_l} - \hat{\mu}_t)(\hat{D}_{t-1}^{x_l, y_l} - \hat{\mu}_{t-1})]$ , and assume that  $\hat{\sigma}_t^2 = \hat{\sigma}_{t-1}^2$ , then we can write this as

$$\hat{\delta}^2 = 2(\hat{\sigma}_t^2 - \hat{\beta}_{t,t-1}) + (\hat{\mu}_t - \hat{\mu}_{t-1})^2.\tag{5.9}$$

In a similar manner, denoted by  $\tilde{\delta} = \tilde{D}_t^{x_l, y_l} - \tilde{D}_{t-1}^{x_l, y_l}$  the corresponding difference between the decoder reconstructed depth pixels, the squared magnitude of  $\tilde{\delta}$  can also be derived as follows

$$\tilde{\delta}^2 = 2(\tilde{\sigma}_t^2 - \tilde{\beta}_{t,t-1}) + (\tilde{\mu}_t - \tilde{\mu}_{t-1})^2.\tag{5.10}$$

To better denote the energy of the difference signal after error concealment at the decoder, we add a “tilde” to the relevant quantities.

In order to validate  $\hat{\delta}^2 = \tilde{\delta}^2$ , we first compare the terms  $\hat{\mu}_t$  and  $\hat{\sigma}_t^2$  with the terms  $\tilde{\mu}_t$  and  $\tilde{\sigma}_t^2$  over the test sequences and decoding setting, respectively. The average relative difference  $e_\mu$  ( $e_\sigma$ ) between the  $\hat{\mu}_t$  ( $\hat{\sigma}_t^2$ ) and  $\tilde{\mu}_t$  ( $\tilde{\sigma}_t^2$ ), defined by

$$e_\mu = \frac{1}{T} \sum_{t=1}^T \frac{|\hat{\mu}_t - \tilde{\mu}_t|}{\tilde{\mu}_t} \times 100\% \tag{5.11}$$

for each test is listed in Table 5.1. Here,  $T$  is the total number of depth frames. It can be seen that  $e_\mu$  and  $e_\sigma$  are very small, which implies that  $\hat{\mu}_t$  and  $\hat{\sigma}_t^2$  are approximately equal to  $\tilde{\mu}_t$  and  $\tilde{\sigma}_t^2$ , respectively.

Table 5.1:  
Relative difference  $e_\mu$  ( $e_\sigma$ ) between the  $\hat{\mu}_t$  ( $\hat{\sigma}_t^2$ ) and  $\tilde{\mu}_t$  ( $\tilde{\sigma}_t^2$ ).

Depth sequence	Packet loss ratio	Relative difference $e_\mu$	Relative difference $e_\sigma$
Lovebird1	5%	0.8%	0.6%
Lovebird1	10%	1.1%	0.8%
BookArrival	5%	1.8%	1.2%
BookArrival	10%	2.1%	1.5%
GT_Fly	5%	1.3%	0.9%
GT_Fly	10%	1.5%	1.3%
Undo_Dancer	5%	1.6%	1.3%
Undo_Dancer	10%	1.7%	1.2%

Based on the statistical results of  $\hat{\mu}_t = \tilde{\mu}_t$  and  $\hat{\sigma}_t^2 = \tilde{\sigma}_t^2$ ,  $\hat{\beta}_{t,t-1}$  can then be proved to approximate to  $\tilde{\beta}_{t,t-1}$  as follows. Since the reconstructed depth images at the encoder and decoder are highly correlated, and the value of the reconstructed pixel at the decoder can be approximated by its value at the encoder plus a zero-mean white noise variable, both the pixel values  $(\hat{\mu}_t, \hat{\mu}_{t-1})$  and  $(\tilde{\mu}_t, \tilde{\mu}_{t-1})$  can be modeled by a 2-D Gaussian distribution with the same Gaussian parameter  $\rho$  [180], [181], i.e.,  $(\hat{\mu}_t, \hat{\mu}_{t-1}) \sim N(\hat{\mu}_t, \hat{\mu}_{t-1}; \hat{\sigma}_t^2, \hat{\sigma}_{t-1}^2; \rho)$  and  $(\tilde{\mu}_t, \tilde{\mu}_{t-1}) \sim N(\tilde{\mu}_t, \tilde{\mu}_{t-1}; \tilde{\sigma}_t^2, \tilde{\sigma}_{t-1}^2; \rho)$ . Therefore, the covariances  $\text{Cov}(\hat{\mu}_t, \hat{\mu}_{t-1})$  and  $\text{Cov}(\tilde{\mu}_t, \tilde{\mu}_{t-1})$  are  $\rho \hat{\sigma}_t \hat{\sigma}_{t-1}$  and  $\rho \tilde{\sigma}_t \tilde{\sigma}_{t-1}$ , respectively, and thus  $\text{Cov}(\hat{\mu}_t, \hat{\mu}_{t-1}) = \text{Cov}(\tilde{\mu}_t, \tilde{\mu}_{t-1})$ . According to these definitions, we can have the following equality

$$\begin{aligned}
\hat{\beta}_{t,t-1} &= E[(\hat{D}_t^{x_l, y_l} - \hat{\mu}_t)(\hat{D}_{t-1}^{x_l, y_l} - \hat{\mu}_{t-1})] \\
&= E(\hat{D}_t^{x_l, y_l} \hat{D}_{t-1}^{x_l, y_l}) - \hat{\mu}_t \hat{\mu}_{t-1} \\
&= \text{Cov}(\hat{\mu}_t, \hat{\mu}_{t-1}) \\
&= \text{Cov}(\tilde{\mu}_t, \tilde{\mu}_{t-1}) = \tilde{\beta}_{t,t-1}.
\end{aligned} \tag{5.12}$$

In the last, based on all the above results of  $\hat{\mu}_t = \tilde{\mu}_t$  ( $\hat{\mu}_{t-1} = \tilde{\mu}_{t-1}$ ),  $\hat{\sigma}_t^2 = \tilde{\sigma}_t^2$ , and  $\hat{\beta}_{t,t-1} = \tilde{\beta}_{t,t-1}$ , it is easy to obtain  $\hat{\delta} = \tilde{\delta}$  from (5.9) and (5.10), i.e.,  $\hat{D}_t^{x_l, y_l} - \hat{D}_{t-1}^{x_l, y_l} = \tilde{D}_t^{x_l, y_l} - \tilde{D}_{t-1}^{x_l, y_l}$ . In addition, recall that in the event of packet losses, the decoder uses zero-motion concealment to recover the missing pixel in the corrupted depth image, so that  $\tilde{D}_t^{x_l, y_l} - \tilde{D}_{t-1}^{x_l, y_l}$  can also be approximated simply

by  $\tilde{D}_{t-1}^{x_l, y_l} - \tilde{D}_{t-2}^{x_l, y_l}$ , which tends to be zero<sup>5</sup>. Finally,  $\hat{D}_t^{x_l, y_l} - \hat{D}_{t-1}^{x_l, y_l} = \tilde{D}_{t-1}^{x_l, y_l} - \tilde{D}_{t-2}^{x_l, y_l}$  can be proved. This indicates it is quite reasonable to estimate the innovation term using the difference between the co-located pixels from the two preceding already decoded depth frames. Moreover, our experimental results in Section 5.7 will also confirm the depth error estimation model derived based on this assumption is very accurate.

After the reconstructed error  $e_t^{x_l, y_l}$  of the depth map is estimated, the disparity error  $\Delta d_t^{x_l, y_l}$  between the warped pixels  $(x', y')$  and  $(x'', y'')$  in the synthesized reference view can be determined, and finally the initially received disparity vector for pixel  $(x_r, y_r)$  can be corrected. However, since disparity vectors are obtained on a block basis for disparity-compensated process, they may not accurately represent the disparity field at the pixel level. Thus, based on the corrected disparity vectors of all the pixels within one block, we can derive the corrected disparity of each block in the texture frame of the right view. Denote the set of corrected disparities of all the pixels in block  $B_k$  by  $\Psi(B_k) = \{d_t(B_k^j) | j = 1, 2, \dots, N\}$ , where  $N$  denotes the number of elements in  $\Psi(B_k)$ , and the subscript  $k$  indicates the block spatial index. We use the vector median filter to obtain the most likely representative corrected disparity vector  $\mu_{VM}$  of block  $B_k$ , and the generation process is formulated as

$$\mu_{VM} = \arg \min_{d_t(B_k^j) \in \Psi(B_k)} \sum_{i=1}^N \|d_t(B_k^j) - d_t(B_k^i)\|. \quad (5.13)$$

After  $\mu_{VM}$  of one block is obtained, the decoder can perform the general disparity compensation operation for the block in the texture frame of the right view.

In order to clearly describe how the proposed disparity vector correction algorithm works, we summarize the proposed novel approach in Algorithm 1.

---

<sup>5</sup>It should be noted that, when the pixel  $\tilde{D}_{t-1}^{x_l, y_l}$  is correctly received, there will be some transmission errors between  $\tilde{D}_{t-1}^{x_l, y_l}$  and  $\tilde{D}_{t-2}^{x_l, y_l}$ . However, due to the smooth characteristic of the depth map, the value of  $\tilde{D}_{t-1}^{x_l, y_l} - \tilde{D}_{t-2}^{x_l, y_l}$  is still approximately close to zero.

---

**Algorithm 1** Proposed disparity vector correction algorithm
 

---

**Require:** texture and depth compressed bit stream

- 1: Initialization of  $f$ ,  $L$ ,  $Z_{\text{near}}$  and  $Z_{\text{far}}$  in (5.3)
- 2: **loop** for all available pixels for each texture block
- 3:   **if** Texture pixel  $(x_r, y_r)$  is lost **then**
- 4:       Do the error concealment for  $(x_r, y_r)$
- 5:   **else**
- 6:       Perform the correction of the disparity vector of  $(x_r, y_r)$
- 7:       **if** Depth pixel  $(x_l, y_l)$  is correctly received **then**
- 8:           **if** Pixel  $(x_l, y_l)$  is intra-coded **then**
- 9:               Set the depth error of  $(x_l, y_l)$  to zero
- 10:          **else**
- 11:               Estimate the depth error  $e_t^{x_l, y_l}$  using (5.5)
- 12:          **end if**
- 13:       **else**
- 14:           Estimate the depth error  $e_t^{x_l, y_l}$  using (5.7)
- 15:       **end if**
- 16:       Estimate the rendering position error  $\Delta d_t^{x_l, y_l}$  using (5.3)
- 17:   **end if**
- 18: **end loop**
- 19: Estimate the block-level disparity vector using (5.13)
- 20: Disparity compensation and reconstruction

**Ensure:** The concealed texture MB of the right view

---

### 5.6.1 Additional Remarks of the Proposed Scheme

In the above, we propose to correct the rendering position error in the synthesized view (or dependent view ) through using the estimated depth error to get the displacement vector. One may argue that, as an alternative, the estimated depth error can be directly applied to conceal the corrupted depth map, and then the corrected depth map is employed to generate the synthetic reference view without any geometry errors, and thus there is no further error occurred in the dependent texture. In this case, the operation of getting the displacement vector is no longer required, which makes this alternative approach slightly simpler. However, in comparison to this approach (referred to as the second approach in the following), the significant advantage of our proposed approach lies in that it is more versatile and robust. The main reasons are explained as follows.

Firstly, our proposed algorithm can be additionally applied to the case, where the reconstructed depth errors in (5.3) contain both source coding distortion and channel distortion. This case typically arises when the original depth map is used to generate the synthetic reference view at the encoder. More specifically, if one uses the original depth map instead of the reconstructed depth map to synthesize a virtual reference view during the VSP procedure at the encoder, the reconstructed depth errors  $e_t^{x_l, y_l}$  in (5.3) at the decoder will contain both quantization errors (source coding distortion) and channel errors, i.e.,  $e_t^{x_l, y_l} = \left| D_t^{x_l, y_l} - \tilde{D}_t^{x_l, y_l} \right|$ , where  $D_t^{x_l, y_l}$  denotes the original pixel value of depth pixel  $(x_l, y_l)$ . As such, the resulting displacement vector between two warped pixels depends on the end-to-end depth errors. Consequently, using the estimated depth error to derive the displacement vector and then to correct the received disparity vector can truly and accurately account for the effects of both source coding distortion and channel distortion. On the contrary, the second approach that directly applies the estimated depth error to conceal depth can only account for the channel distortion. That is, this approach will neglect the impact of the quantization errors on the geometry error between these two warped pixels. As a matter of fact, the second approach can be regarded as a special case of our proposed algorithm, when the distance between the two warped pixels is 0.

Secondly, the second approach is not applicable to Backward VSP (B-VSP)



based upon the coding order of  $T_0D_0T_1D_1$ . The reason for this is that, in the texture-first coding order B-VSP, the depth component is coded after its corresponding texture component and the depth map of the dependent view is used to perform 3-D warping. In other words, the depth of the dependent view is not available when decoding the texture of the dependent view with backward warping, which prohibits the estimated depth error being directly applied to the depth. By contrast, our proposed algorithm does not require that the depth map always physically exists. So it can be easily used in BVSP based upon both the depth-first coding order and texture-first coding order.

## 5.7 Experimental Results and Discussion

In this section, the performance of the proposed scheme is extensively evaluated. The Joint Multi-view Video Coding (JMVC) version 8.0 [119] of the H.264/MVC reference software is appropriately modified to encode both the multi-view video sequences and depth maps, and View Synthesis Reference Software (VSRS) 3.5 [182] is used to render the synthetic reference view at the encoder and the virtual intermediate views at the decoder. The standard multi-view video plus depth sequences “BookArrival”, “Lovebird1”, “Newspaper”, “GT\_Fly” and “Undo\_Dancer” are chosen for our simulations. For each multi-view video sequence, each view is encoded with the GOP size of 30 or 100 frames, where the first frame in the left view is coded as an I-frame, and the remaining frames are coded as P-frames. The descriptions of the used MVD test sequences and the other coding parameters are listed in Table 5.2. Note that on the basis of the prediction structure illustrated in Figure 5.1, both the VSP and regular temporal prediction are used during texture video coding.

Table 5.2:  
Description of MVD test sequences and simulation conditions.

Test sequence	Input views	Synthesized view	Frame rate	Resolution	Encoder parameters
Lovebird1	6-8	“7”	16.7	1024 × 768	Symbol mode: CABAC Variable prediction size: Enabled Loop filter: Enabled Search range: 64 QP for texture and depth: 32
BookArrival	8-10	“9”	30	1024 × 768	
Newspaper	4-6	“5”	30	1024 × 768	
GT_Fly	5-9	“6”	25	1920 × 1088	
Undo_Dancer	2-5	“3”	25	1920 × 1088	

Each coded frame is partitioned into slices, where each depth slice contains four horizontal rows of MBs, and each texture slice contains a horizontal row of MBs due to higher associated bit rates. This slice size selection will be confirmed in Section 5.7.1. Each coded slice is then carried in a separate packet. It should be noted that the packet length of all the frames in our simulations is within the limit of the maximum transmission unit (MTU) for the Ethernet. The random packet loss pattern is employed to simulate packet losses [183]. Since the proposed method aims to mitigate the prediction position error introduced by the reconstructed depth error, in order to clearly demonstrate the effectiveness of the proposed scheme, packet loss is only simulated for the depth map stream. In this case, all the transmission errors of the texture video of the right view come from the synthetic reference frames <sup>6</sup>. To simulate the channel, at each packet loss rate, different packet loss patterns are randomly generated. For objective video quality assessment, the peak signal-to-noise ratio (PSNR) is averaged over all decoded frames under all the implemented channel conditions. In our experiments, the error concealment method where each damaged block in the depth map is directly replaced by the co-located one in the previous frame is employed at the multi-view video decoder.

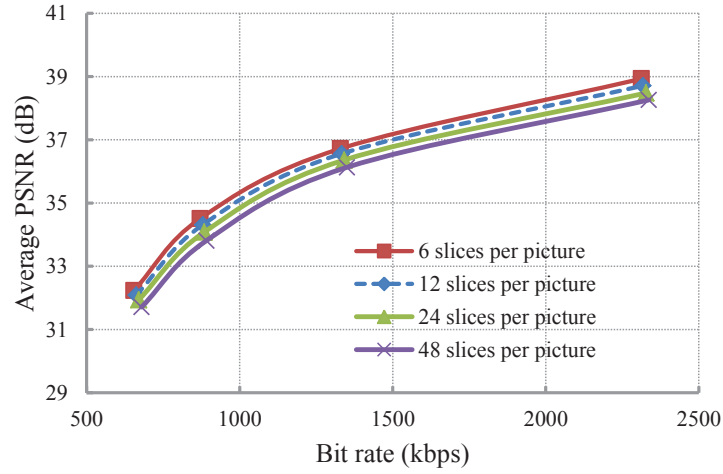
### 5.7.1 Evaluation of the Effects of Slice Partitioning on Coding Efficiency and Error Resilience

It is commonly known that a smaller slice size may degrade the coding efficiency due to the correlation broken between some neighboring MBs and the extra overhead information for small packets, but at the same time, it can provide the robustness of video streaming against packet losses. For example, if the picture is divided into a large number of slices and each slice fits in one separate packet, in the event of packet loss, only a very small portion of the picture will be lost and the errors will not propagate to any other slice in the picture due to independency

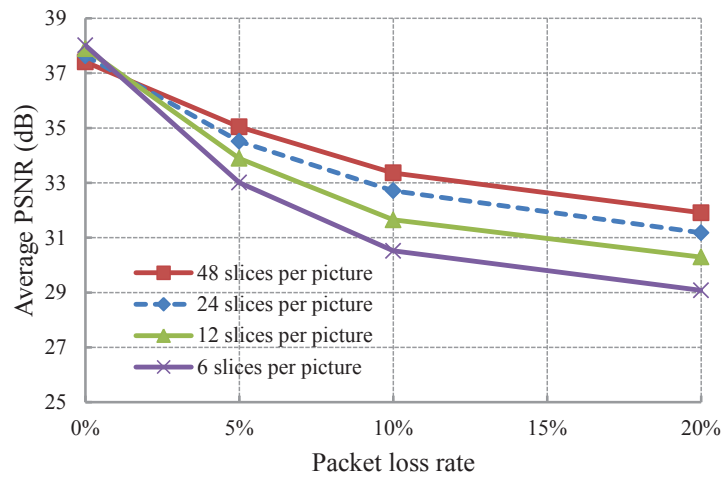
---

<sup>6</sup>If the transmission errors still exist after the proposed error correction algorithm, the remaining errors will continuously afflict the subsequent frames through the temporal prediction path within the same view. Nevertheless, this small intra-view error propagation can eventually be attributed to the errors from the synthetic reference frame.

of slices.



(a) *Effect of slice number on the coding efficiency.*



(b) *Effect of slice number on the efficiency of the error resilience.*

Figure 5.4: Impacts of slice number on the efficiency of the coding performance and error resilience.

In order to see how the slice size affects the coding efficiency and error robustness in the high resolution 3-D video sequences, we conduct some experiments on the BookArrival sequence, where each picture is divided into different number of slices. Figure 5.4 shows the effects of slice partitioning on both the efficiency of lossy coding and the efficiency of error concealment. As can be observed from Figure 5.4(a), only a small loss on the coding efficiency can be observed between 48 slices per picture and 6/12/24 slices per picture. This is due to the fact, in

the high resolution picture, even if the slice size is set to be small, the number of MBs in a slice is still higher than that in the standard resolution picture, thus making intra prediction and entropy coding within a slice still more effective. On the other hand, in terms of error resilience performance in Figure 5.4(b), the difference between 6 slices per picture and 48 slices per picture is approximately 3 dB at the packet loss rate of 10%, which is significant. The main reason for this is that, if the slice with larger size is lost, the error concealment will become much more difficult. Therefore, the simulation results indicate that an appropriate increase of the amount of slices per picture can improve the error resilience but does not reduce the coding efficiency greatly. Based on the above reasons, we choose smaller slice size for texture and depth map coding (i.e., 48 slices per texture picture and 12 slices per depth picture for  $1024 \times 768$  video sequence) in this work for better error resilience performance.

Nevertheless, we also checked some industrial applications such as video conferencing, and found 10~22 slices per texture picture for practical  $1024 \times 768$  video encoders is usually recommended [184]. As such, we also tested the proposed disparity vector correction algorithm with larger slice sizes (e.g., 12 slices per texture picture) to confirm that the slice size does not affect the performance of the proposed algorithm. Table 5.3 presents the PSNR comparison results using the proposed algorithm, where the slice numbers are chosen to be 48 and 12. As can be observed from Table 5.3, our proposed algorithm can achieve almost identical performances under these two different slice size settings.

Table 5.3:

PSNR comparison of the proposed algorithm with two different slice size settings (GOP = 30).

Sequence	Slice number	Y-PSNR (dB) at different loss rates		
		5%	10%	20%
Lovebird1	48	36.48	36.63	36.51
	12	36.37	36.53	36.42
BookArrival	48	35.99	36.10	35.72
	12	35.88	36.02	35.61
Newspaper	48	36.53	36.26	35.88
	12	36.42	36.17	35.76

### 5.7.2 Verification of Depth Error Estimation

In order to verify the accuracy of the proposed depth map error estimation approach, we compare the estimated and measured depth errors at the frame level with a packet loss rate of 10% for the depth map of the left view of the BookArrival and Newspaper sequences. The Mean Absolute Difference (MAD) is used as the error metric. As can be observed from Figure 5.5, the estimated depth error with the proposed method at the decoder is very close to its actual one along the whole sequence. The tests over other video sequences and packet loss rates yield similar results. Meanwhile, since the reconstructed depth error is used to compute the disparity error for block-based disparity compensation in decoding, we also evaluate the accuracy of the depth map error estimated at the MB level. Table 5.4 shows the correlation coefficients between the estimated and measured depth map errors for all the test sequences. As can be observed from the table, the correlation coefficients for the test packet loss rates are all greater than 0.90, which proves that the depth error can also be precisely estimated at the MB level. Therefore, the estimated depth map error can be efficiently utilized to determine the rendering position error.

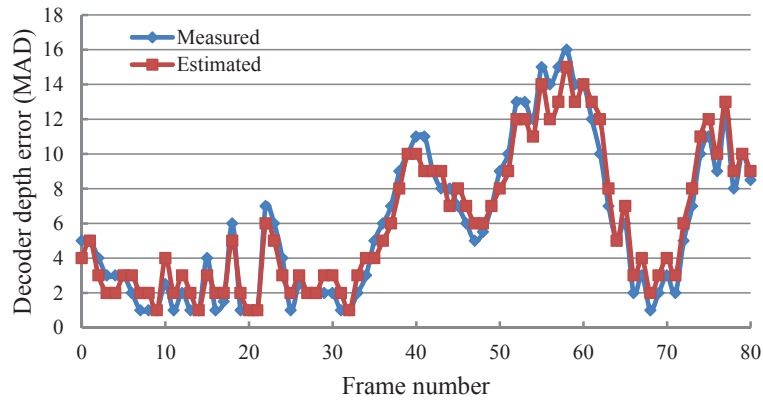
Table 5.4:

Correlation coefficient between the estimated and measured depth map errors at the MB level for each video sequence.

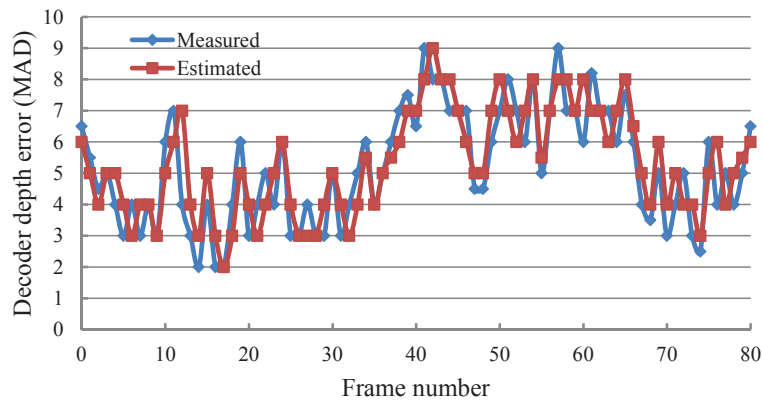
Sequence	Correlation coefficients at various loss rates		
	5%	10%	20%
Lovebird1	0.95	0.96	0.96
BookArrival	0.94	0.93	0.94
Newspaper	0.93	0.92	0.90
GT_Fly	0.96	0.97	0.96
Undo_Dancer	0.94	0.91	0.93

### 5.7.3 Performance Comparison of the Right View

In order to evaluate the performance of the proposed disparity vector correction algorithm, the proposed method with the estimated depth error, the proposed method with the actual depth error, and “JMVC” are compared. “JMVC” rep-



(a) "BookArrival" sequence.



(b) "Newspaper" sequence.

Figure 5.5: Comparison between the measured and estimated depth map errors at the frame level.

resents the basic scheme that only the aforementioned error concealment method is employed at the JMVC decoder to recover the erroneous region in the depth map, and no error recovery method is used for texture video decoding. For brevity, the proposed method with the estimated depth error (EDE) is denoted by “Proposed with EDE”, whereas the proposed method with the actual depth error (ADE) is denoted by “Proposed with ADE”. Since the proposed method focuses on mitigating error propagation occurred in the right view, only the comparative results for the texture video of the right view are given. Table 5.5 summarizes the comparison results of the average PSNRs for all the test sequences at various packet loss rates. GOP sizes of 100 and 30 are tested. As can be observed, the “Proposed with EDE” scheme yields significant and consistent gains over the “JMVC” scheme, and the performance of the “Proposed with EDE” scheme is very close to that of the “Proposed with ADE” scheme. When comparing the “Proposed with EDE” scheme with the “JMVC” scheme at these two different GOP size settings, it is also clear that the Newspaper sequence achieves the maximum average PSNR gains of about 3.7 dB, 5.0 dB, and 7.5 dB among the test sequences at the packet loss rates of 5%, 10%, and 20%, respectively. This is because that, the Newspaper sequence captures nearer scene and has a more complex depth map than the other sequences, which results in larger depth map errors, and consequently larger prediction position errors. Through our proposed method, the prediction position error can be effectively eliminated. Besides, the PSNR gains achieved by the “Proposed with EDE” method increase with the increase of the packet loss rate.

Note that in practice, the texture video may also be corrupted by channel errors. In order to evaluate the performance of the proposed algorithm in this practical environment, we additionally implement the proposed algorithm with a packet loss rate of 10% for both the texture video and depth streams. In this test, the error concealment strategy introduced in [167] is used in the proposed algorithms and the “JMVC” scheme to recover the lost information in the texture video. The resulting PSNR values obtained by different methods are presented in the last column in Table 5.5. As can be observed, the proposed algorithms with the estimated and actual depth errors still significantly outperform the “JMVC”

Table 5.5:

Average PSNR comparison for the right view video with a variety of packet loss rates.

Sequence	Scheme	Y-PSNR (dB) at different loss rates			
		5%	10%	20%	10% for both texture and depth
Lovebird1 (GOP = 100)	JMVC	35.55	35.27	34.65	34.01
	Proposed with EDE	36.24	36.31	36.27	34.89
	Proposed with ADE	36.69			35.12
Lovebird1 (GOP = 30)	JMVC	35.76	35.41	34.88	34.27
	Proposed with EDE	36.48	36.63	36.51	35.14
	Proposed with ADE	36.91			35.39
BookArrival (GOP = 100)	JMVC	33.79	32.81	29.41	30.23
	Proposed with EDE	35.78	35.86	35.48	32.47
	Proposed with ADE	36.61			32.97
BookArrival (GOP = 30)	JMVC	33.92	33.03	29.74	30.48
	Proposed with EDE	35.99	36.10	35.72	32.70
	Proposed with ADE	36.96			33.25
Newspaper (GOP = 100)	JMVC	32.35	30.71	27.94	28.79
	Proposed with EDE	36.12	35.78	35.45	32.41
	Proposed with ADE	36.53			32.75
Newspaper (GOP = 30)	JMVC	32.79	31.24	28.43	29.28
	Proposed with EDE	36.53	36.26	35.88	32.86
	Proposed with ADE	36.91			33.25
GT_Fly (GOP = 100)	JMVC	37.04	36.71	36.11	34.87
	Proposed with EDE	37.56	37.34	37.25	35.74
	Proposed with ADE	37.95			35.92
GT_Fly (GOP = 30)	JMVC	37.35	36.93	36.34	35.06
	Proposed with EDE	37.81	37.67	37.49	35.95
	Proposed with ADE	38.21			36.13
Undo_Dancer (GOP = 100)	JMVC	33.46	32.53	31.21	29.95
	Proposed with EDE	34.56	34.45	34.51	31.83
	Proposed with ADE	34.98			32.02
Undo_Dancer (GOP = 30)	JMVC	33.78	32.84	31.53	30.25
	Proposed with EDE	34.88	34.81	34.82	32.14
	Proposed with ADE	35.42			32.37

scheme. This is due to the fact that with the proposed algorithms, the transmission errors caused by packet losses in depth map can be readily concealed.

Although we have already argued that the translational disparity compensation prediction (TDCP) is disabled for the analysis of the transmission errors caused by VSP, TDCP can nonetheless be enabled under our proposed disparity vector correction algorithm. If the TDCP is enabled, the performance gain of the proposed method will be smaller since the number of blocks encoded with VSP will decrease. Table 5.6 lists the percentages of the VSP based inter-view coding mode among all the prediction modes and the performance comparison results



when both VSP and TDCP are enabled. In this test, the GOP size is set to 30. As can be observed, the average percentage of the VSP based inter-view coding mode is usually around 34.12% compared with those of both TDCP and motion compensation prediction. Even in this case, the “TDCP+Proposed with EDE” method still outperforms the method of “TDCP+JMVC” by 0.27 dB  $\sim$  1.01 dB at the packet loss rate of 10%.

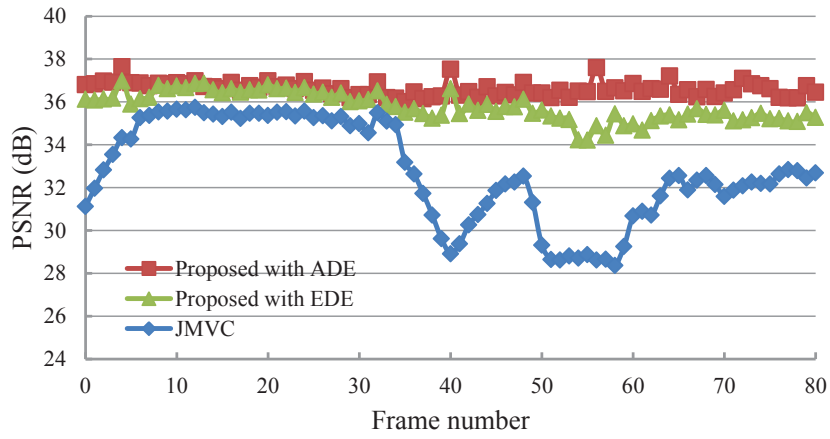
Table 5.6:

Performance comparison for the texture at the packet loss rate of 10% when both VSP and TDCP are enabled.

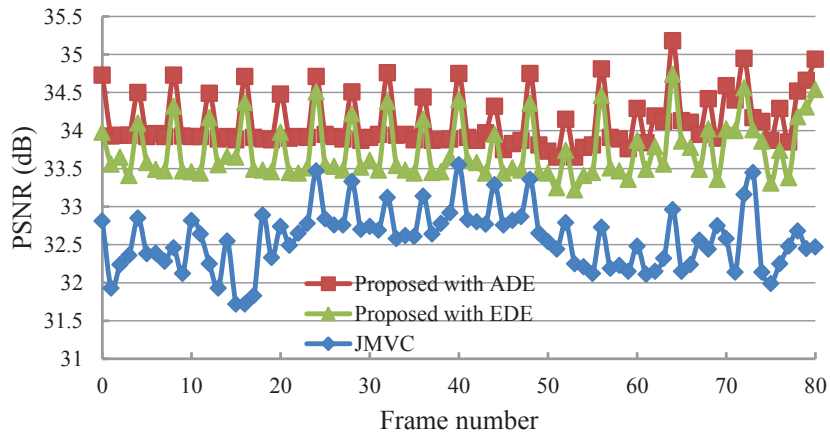
Sequence	Percentage of VSP mode	Average PSNR (dB)		
		JMVC	TDCP+JMVC	TDCP+Proposed with EDE
Lovebird1	30.15%	35.41	35.54	35.86
BookArrival	34.57%	33.03	33.23	34.05
Undo_Dancer	38.28%	32.84	32.91	33.47
GT_Fly	35.78%	36.93	37.24	37.51
Newspaper	31.84%	31.24	32.82	33.83

Figure 5.6 shows the frame-by-frame PSNR comparison between our proposed methods and the “JMVC” scheme for the right view video of the BookArrival and Undo\_Dancer sequences at the packet loss rate of 10%. From the performance comparison, it can be seen that the “Proposed with EDE” method achieves considerable improvements over the “JMVC” scheme, and the performance gap between the “Proposed with EDE” and “Proposed with ADE” in each frame is also relatively small. As for some specific frames of the BookArrival and Undo\_Dancer sequences, the PSNRs of the “Proposed with EDE” can be up to 7.7 dB and 2.65 dB higher than those of the “JMVC” scheme, respectively.

To subjectively evaluate the performance of the proposed method, Figure 5.7 shows a comparison of the 45th frames of the “BookArrival” sequence obtained by the “JMVC” scheme, the “Proposed with EDE” scheme and the “Proposed with ADE” scheme. Figure 5.7(a) shows the original decoded texture image of the left view, which are not contaminated by transmission errors. Figure 5.7(b) shows the reconstructed depth image of the left view, which has been corrupted by channel errors. Figure 5.7(c) shows the texture image of the virtual view synthesized by the reconstructed texture image and depth image of the left view.



(a) View “10” of the “BookArrival” sequence.



(b) View “5” of the “Undo\_Dancer” sequence.

Figure 5.6: PSNR comparison of each frame with a packet loss rate of 10%.

Figures 5.7(d), 5.7(e), and 5.7(f) represent the decoded texture images of the right view with the “JMVC”, “Proposed with EDE”, and “Proposed with ADE” schemes, respectively. As can be observed from Figures 5.7(a)-5.7(d), due to the reconstructed depth errors, the texture pixels in the left view are projected to the wrong spatial locations in the synthesized image, and consequently result in prediction position errors in the decoded texture image of the right view. Through the “Proposed with EDE” scheme, it is obvious that the prediction position errors can be faithfully recovered as shown in Figure 5.7(e), and the subjective visual qualities of the “Proposed with EDE” and “Proposed with ADE” schemes are almost identical.

#### 5.7.4 Performance Comparison of the Decoder-Side Synthesized View

As mentioned earlier, one of the biggest advantages of MVD based 3-D video representation is the support for synthesis of additional perspective views at the receiver. Therefore, in order to further validate the performance of the proposed method in this context, we provide the PSNR comparison results for decoder-side view synthesis quality of different input video sequences in Table 5.7. The intermediate virtual view is synthesized using two pairs of transmitted textures and depth maps from two neighbouring coded views, and the PSNR of virtual view synthesis is measured between the virtual view images synthesized by the uncompressed texture and depth images and the decoded texture and depth images. As can be observed from Table 5.7, both the “Proposed with EDE” and “Proposed with ADE” methods still achieve better view synthesis performance than the “JMVC” method. This is not surprising as the view synthesis quality is always strongly correlated with the quality of the texture videos and depth maps of the coded views. It is also easy to see that the Newspaper sequence is among the sequences that benefits the most from the proposed algorithm. We also implement the proposed algorithm on top of the 3D-AVC reference software 3D-ATM v6.0 [185] and test using the two-view scenario.  $T_0D_0D_1T_1$  coding order is used, where  $T_i$  and  $D_i$  are the texture and depth components respectively from the  $i^{th}$  view, corresponding to the left or right adjacent views. The depth map is



(a) *Decoded texture image of the left view.*



(b) *Reconstructed depth image of the left view.*

Figure 5.7: Subjective quality comparison for frame 45 of the BookArrival sequence with the packet loss rate 10%.



(c) *Rendered texture image of the virtual view.*



(d) *Decoded texture image of the right view with the “JMVC” method.*

Figure 5.7: Subjective quality comparison for frame 45 of the BookArrival sequence with the packet loss rate 10% (con’t).



(e) *Decoded texture image of the right view with the “Proposed with EDE” method.*



(f) *Decoded texture image of the right view with the “Proposed with ADE method”.*

Figure 5.7: Subjective quality comparison for frame 45 of the BookArrival sequence with the packet loss rate 10% (con’t).

selected to be coded at full resolution. The remaining coding parameters are set to the same values as those used for JMVC. As can be observed from the table, the proposed algorithm on 3D-ATM v6.0 shows similar performance as that on JMVC platform in the event of depth packet losses.

Table 5.7:

Average PSNR comparison for the synthesized view video with a variety of packet loss rates.

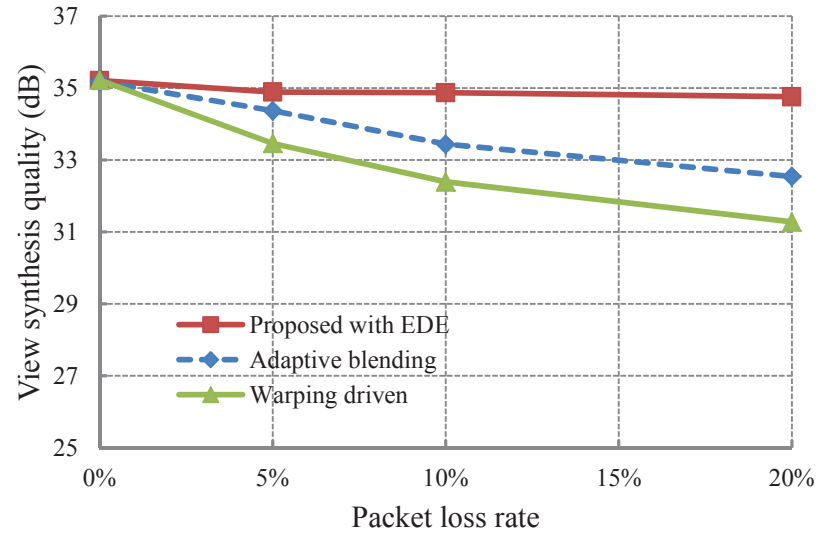
Sequence	View ID	JMVC Platform			3D-ATM Platform		
		Scheme	5%	10%	Scheme	5%	10%
Lovebird1	“7”	JMVC	35.89	35.62	3D-ATM	36.10	35.73
		Proposed with EDE	36.62	36.46	Proposed with EDE	36.89	36.75
		Proposed with ADE	36.68		Proposed with ADE	36.94	
BookArrival	“9”	JMVC	34.81	33.97	3D-ATM	35.01	34.18
		Proposed with EDE	36.29	36.27	Proposed with EDE	36.51	36.48
		Proposed with ADE	36.45		Proposed with ADE	36.69	
Newspaper	“5”	JMVC	34.01	33.26	3D-ATM	34.24	33.51
		Proposed with EDE	36.34	36.02	Proposed with EDE	36.61	36.36
		Proposed with ADE	36.46		Proposed with ADE	36.87	
GT_Fly	“6”	JMVC	37.21	37.04	3D-ATM	37.44	37.27
		Proposed with EDE	37.71	37.63	Proposed with EDE	37.96	37.86
		Proposed with ADE	37.98		Proposed with ADE	38.25	
Undo_Dancer	“3”	JMVC	34.12	33.51	3D-ATM	34.41	33.80
		Proposed with EDE	34.89	34.87	Proposed with EDE	35.29	35.19
		Proposed with ADE	34.98		Proposed with ADE	35.37	

In order to further reveal the effectiveness of our proposed algorithm, we choose the more related work [140], [186] for comparison, which is so far the latest work in this area. In [140], the authors proposed an adaptive blending error concealment strategy for the synthesized view. Firstly, they proposed an error model to track the reliability of each coded block in each transmitted view given observed packet loss events. Then, based on the estimated reliability, the source pixel with higher reliability is weighted more heavily during the two-view-based merging process. This adaptive blending error concealment strategy is referred to as “Adaptive blending” in this comparison. In [186], the authors developed a warping driven based mode selection method for depth map error concealment, where INTRA, INTER, and MV sharing are chose as the candidate modes, and the optimal mode is then determined based on the distortion between the warped view and the coded view. This comparative approach is denoted by “Warping driven”. Although all these methods perform the error concealment strategy to

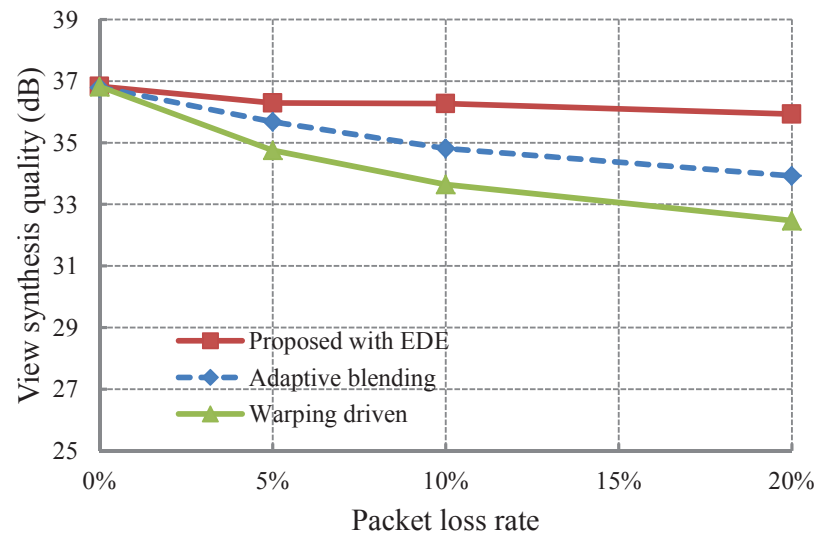
combat transmission errors of 3-D video at the decoder, the main objectives of these methods are quite different. The proposed method in this Chapter aims to mitigate the transmission errors occurred at the texture video, whereas [140] intends to improve the quality of synthesized view and the “Warping driven” algorithm mainly optimizes the reconstructed depth map. For fair comparison, we use the PSNR of rendered views along with the total bit rate of all multi-view textures and depths to quantitatively evaluate the performances of these methods. This is reasonable since the quality of synthesized views can simultaneously reflect the reconstructed quality of texture and depth map. In this simulation, packet loss rates of 5%, 10%, and 20% are emulated for both textures and depth maps.

The average PSNR performance and overall bit rate comparisons are illustrated in Figure 5.8 and Table 5.8, respectively. As can be observed from Figure 5.8, the performance of “Warping driven” algorithm is generally inferior to those of the proposed algorithm and the “Adaptive blending” algorithm at all test packet loss rates. This is mostly due to the fact, the “Warping driven” approach is designed to reduce the transmission error in the depth map, and the depth distortion in itself is not as important as distortion of texture because depth information is only supplementary data for view synthesis. When comparing the proposed algorithm with the “Adaptive blending” algorithm, it can be seen that the proposed algorithm still consistently outperforms the “Adaptive blending” algorithm. This is because, although “Adaptive blending” algorithm can adaptively blend corresponding pixels in the two captured views during DIBR, some transmission errors still exist in the texture and depth of the coded views and then the rendering errors in the synthesized are inevitable. On the other hand, in terms of the overall bit rate of textures plus depths in Table 5.8, the proposed algorithm achieves about 20% bit rate saving on average compared to those two reference algorithms. These bit rate reduction can be explained by the intrinsic property of the encoding structure adopted in our proposed strategy, i.e., exploiting inter-view dependency through VSP during encoding allows us to achieve the same level of reconstruction quality for texture and depth with a smaller bit rate. To sum up, the proposed algorithm not only improves the rendered view quality by disparity vector correction for the coded texture, but also saves the total bit





(a) “Undo\_Dancer” sequence.



(b) “BookArrival” sequence.

Figure 5.8: Decoder-side synthesized view quality versus packet loss rate.

rate having to be transmitted.

Table 5.8:  
Total bit rate comparison.

Sequence	Bit rate (Mbps)		
	Proposed	Adaptive blending	Warping driven
BookArrival	1.43	1.80	1.68
Undo_Dancer	2.28	2.91	2.74

In addition, our experiments also reveal the proposed algorithm consumes much less computational complexity than those two reference methods. This is because, in the “Adaptive blending” scheme, the decoder needs to additionally calculate the worst-case distortion for each texture pixel of the two coded views with a certain range, and every pixel within the range needs to be considered during each worst distortion computation. In the “Warping driven” algorithm, the decoder performs the actual computationally-heavy view synthesis processes three times at the decoder to drive the block-wise mode selection for each lost depth block, which requires a significantly higher computational complexity.

### 5.7.5 Computational Complexity Analysis

The proposed disparity vector correction algorithm introduces additional complexity for the MVC decoder. However, the additional computational costs are very modest and well justified by the notable error resilience performance improvements achieved. As can be observed from (5.3) and (5.5), for each correctly received pixel (inter-coded) in the depth map, we need one multiplication to calculate the disparity error. As for each lost pixel, it requires two additions and one multiplication for this calculation. Then, in order to generate a representative corrected disparity for each block in the right view by means of (5.13), another  $N$  additions (or subtractions) are needed to calculate the deviation, and  $N$  additions are required to compute the summation for each pixel, where  $N$  is the number of pixels in a block as defined in Section 5.6. Therefore, when the depth pixel is received correctly, the proposed algorithm requires a total of one multiplication and  $2N$  additions. In contrast, when incorrectly received, a total

of one multiplication and  $2N + 2$  additions is needed. For these two cases, the average number of arithmetic operations per pixel represents a modest complexity increase. For the experiments, the hardware platform is a laptop computer equipped with 2.40 GHz Intel (R) Core (M) 2 Duo CPU and 3G memory running Microsoft Windows 7 Professional. The average percent increase in complexity for different test sequences are listed in Table 5.9. According to the results, an average increase of only 2.1% in execution time with respect to the VSP-enabled JMVC decoder is registered. This small increase in computational complexity is well paid off by the significant quality improvement.

Table 5.9:

Computational complexity comparison between the proposed method and the VSP-enabled JMVC.

Sequence	Decoding time comparison		Time increment
	VSP-enabled JMVC (s)	Proposed (s)	
Lovebird1	100.12	102.24	2.1%
BookArrival	111.73	114.20	2.2%
Newspaper	105.63	107.11	1.4%
GT_Fly	150.46	153.42	1.9%
Undo_Dancer	164.73	168.61	2.4%
<b>Average</b>	<b>126.53</b>	<b>129.11</b>	<b>2.1%</b>

## 5.8 Summary

In this chapter, we have proposed an efficient disparity vector correction algorithm to improve the performance of VSP based 3-D video transmission. Based on the analysis of the error propagation behaviour of 3-D video, a new prediction position error is derived with respect to the depth map error due to channel losses. With the aim of mitigating error propagation of prediction position error, the monotonic relationship between the disparity error and depth map error is established, in which the depth error is accurately estimated at the decoder by considering the depth map smooth properties. Especially, the approximation of the innovation term invoked in depth error estimation is proved through theoretical derivation and experimental observations.

After the disparity vector is corrected with the derived disparity change, the predicted pixel can find the optimum matching pixel from the synthetic reference frame. The corrected disparity vector representative for all the pixels within the corresponding block is obtained using a median filter. Experimental results show that the proposed method can significantly improve the performance on both objective and subjective visual qualities. The proposed algorithm has very low computational complexity and implementation cost, and is therefore suitable for wireless 3-D video applications.

Although the proposed algorithm only considers the forward VSP, i.e., mapping the texture pixels to the virtual image plane using the depth map of the reference view, it can be extended and adapted to the backward VSP based 3-D video transmission. Furthermore, the proposed method can also be applied to the VSP-support depth map coding and transmission.

# Chapter 6

## Conclusion and Future Work

This thesis focuses on enhancing the robustness of multi-view video data and multi-view depth data transmission over error-prone packet-switched networks. By virtue of an in-depth analysis of the propagating behaviour of transmission errors due to packet losses, three different error-resilient multi-view coding algorithms have been proposed in Chapters 3, 4, and 5, respectively, all of which joint source coding and transmission efficiently.

In the proposed WZ-based error-resilient multi-view video coding scheme, the MVC standard is firstly employed to compress all the views, and WZ encoding is then applied to the key frames of some selected views to produce an auxiliary bit stream. At the decoder, the auxiliary parity bits are used to correct channel errors of the key frames of the selected views. The major contribution of this scheme is that this approach gives an explicit indication about how to allocate the WZ bits based on the estimation of the channel-induced distortion. One advantage of the proposed method is that it naturally allows for rate adaptivity and unequal error protection with fine granularity at the frame and bit-plane levels.

In Chapter 4, a method has been proposed for optimal mode switching, which enhances the robustness of MVD video coders against packet losses. This method estimates the view synthesis distortion considering the compound impact of the transmission distortions of both the texture video and the depth map, and then incorporates the estimated overall distortion into a rate-distortion framework to jointly optimize the encoding modes of the texture MBs and depth MBs. In addition, to allow for a fine tuning of the bit rate, a new Lagrange multiplier is

derived through a fast convex search algorithm. The superiority of the proposed approach over other state-of-the-art techniques is demonstrated through simulation results. The proposed method requires only modification of the encoder parametric decisions, and is thus standard-compatible.

Finally, this thesis has advanced research on reliable transmission of VSP-based 3-D video. A new approach to correct view synthesis prediction errors caused by packet losses in depth maps has been developed. This scheme firstly analyzes how the incorrect depth information gradually affects the dependent views through the VSP-based warping path. The major novelty of this analysis lies in the definition of a new geometry error for the pixels whose disparity vectors are correctly received. Based on the newly defined geometry errors, a novel disparity vector correction algorithm is proposed to locate the matching synthesized pixels with negligible extra computational complexity. Simulation results show that the estimated depth error and actual depth error are highly correlated, and the proposed technique using the estimated error performs closely to the one using the actual error. This work is the first of its kind to optimize the performance of VSP-based 3-D video in consideration of potential packet losses.

The first two error-resilient algorithms are implemented at the encoder side, while the last one is carried out at the decoder. In general, these three error-resilient algorithms can work collaboratively to make the output multi-view bit stream more resilient to transmission errors. In our simulation, all the proposed error-resilient algorithms were built upon a codec based on H.264/AVC framework. This is because, H.264/AVC has already been widely deployed in industry, especially in SoC chips, reusing its framework would make the corresponding decoder easier to implement. However, it should be noted that the principal idea of our proposed schemes is not limited to the H.264/AVC framework.

It should be noted that, this thesis studied packet switching networks, in which the packet loss rate is assumed for various schemes. The channel in this dissertation can be simulated as erasure channel. However, when wireless channels (e.g., Gaussian channels and Rayleigh fading channels) are used in the real video delivery systems, the conclusions drawn from the erasure channel case can still hold. This is because that, when variable length coding (VLC) is used in

the multi-view video compression, a single bit error can lead to many following bits being undecodable and hence useless. Therefore, the bit error in VLC in 3-D video streaming can be regarded as effective erasure errors. Moreover, as most Internet-based multimedia services employ User Datagram Protocol (UDP) as their transport protocol, the bit error within a packet would erase the whole packet as well.

In addition, in this thesis, we have implicitly considered the end-to-end delay in the main chapters. Chapter 4 designs a low-complexity error resilient multi-view video plus depth coder, while Chapter 5 implements a light-weight decoder for error concealment of disparity vector in VSP. Although, in Chapter 3, we use a feedback channel to transmit the parity bits in the proposed WZ-based algorithm, this feedback channel would not induce many network delay as this feedback channel is only invoked when the encoder parity bit rate is underestimated by the proposed WZ bit rate estimation algorithm. Therefore, our findings from this thesis can be applied to the video communications system with stringent delay requirement.

## 6.1 Future Work

This discussion concludes with recommendations of future works that are natural extensions of the problems considered in this thesis:

- **Robust Distributed Multi-view Video Coding with Low-Complexity**

**Inter-Camera Communications:** In Chapter 3, WZ coded bits are employed as an auxiliary stream to protect the MVC-standard coded bit stream. Although this scheme can achieve a better performance than conventional solutions based on FEC codes, it results in a very high encoder complexity since the complicated temporal and inter-view correlation exploration process is still performed during encoding. However, in some applications such as wireless multimedia sensor networks, it is desirable to have low energy consumption in the smart cameras. This places a stringent constraint on the complexity of the multi-view video or depth data encoding process [187], [188]. Therefore, we would like to develop an ap-

proach which has low encoding complexity, is robust while satisfying tight latency constraints, and requires no inter-camera communications. Toward this end, we will address two main issues by resorting to results in information theory and computer vision. First, since the encoder at each camera does not have access to views observed from the other cameras, some noise models are needed at the time of encoding that capture both the statistical relationship and the geometrical constraints between multiple camera views. Secondly, as the SI is essential to ensuring a good quality of reconstructed multi-view videos, how to generate high-quality SI by making use of overlapping camera views is also worthwhile for future investigations.

- Adaptive Mode Switching for Loss-Resilient Depth Map Coding:** In Chapter 4, the view synthesis distortion is incorporated into rate-distortion optimization for optimal depth mode switching. However, in practice, a number of depth pixels will not cause distortion in the synthesized view even if the depth map is corrupted by channel errors [189]. For example, if the original and distorted depth values are mapped to the same disparity, there will not be any numerical difference in the warping process. In this case, rate-distortion optimization with view synthesis cost is no longer applicable for depth map transmission since the depth distortion itself can propagate to the subsequent and neighboring frames by temporal and inter-view prediction schemes. As a result, the depth coding mode should be optimized in accordance with the depth distortion itself when there are no synthesis errors. In view of this fact, a new mode switching scheme that adaptively employs the expected view synthesis distortion or depth distortion as the distortion metric is needed to further improve the error-resilient performance of depth map streaming.
- Efficient Warping Error Correction Algorithm for Backward View Synthesis Prediction-based 3-D video Using Virtual Disparity Vector Estimation:** The proposed disparity vector correction algorithm presented in Chapter 5 is built up on forward VSP (F-VSP). However, F-VSP is usually considered to be demanding in terms of memory use and processing power due to hole and occlusion handling. As a result, the original F-VSP



design is replaced by the backward VSP approach (B-VSP) utilizing the texture first coding order for non-base views. The B-VSP design is found to offer comparable compression efficiency to F-VSP with very low decoder complexity [190]. Therefore, it is of great significance to further research into error concealment strategies for B-VSP-based 3-D video transmission. In B-VSP, the disparity vector of the neighboring block is used to locate a virtual depth block from the reference view, and then the virtual depth block is converted to a disparity vector to fetch the corresponding texture block from the reference view. It can be seen that the neighboring block disparity vector plays a very important role in establishing a connection between the virtual depth block and the current non-base view. If the neighboring block disparity vector is damaged during transmission, it will render that the current non-base view cannot be correctly reconstructed regardless whether or not the corresponding depth map of the reference view is correct. Therefore, in comparison to F-VSP-based 3-D video transmission, the key challenging issue is to develop a new algorithm for virtual disparity vector estimation rather than depth error estimation as presented in Chapter 5. One possible solution is to extrapolate the disparity vectors from the neighboring views. However, this disparity vector estimate may not be accurate enough since the inter-view correlation is very difficult to exploit. A more accurate virtual disparity vector estimation algorithm for B-VSP-based 3-D video is an ongoing research topic of great interest.



# References

- [1] P. Benzie, J. Watson, P. Surman, I. Rakkolainen, K. Hopf, H. Urey, V. Sainov, and C. v. Kopylow, “A survey of 3DTV display: techniques and technologies,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 17, no. 11, pp. 1647–1658, Nov. 2007.
- [2] J. Konard and M. Halle, “3-D displays and signal processing-An answer to 3-D ills?” *IEEE Signal Process. Mag.*, vol. 24, no. 6, pp. 97–111, Nov. 2007.
- [3] A. Smolic, P. Kauff, S. Knorr, A. Hornung, M. Kunter, M. Muller, and M. Lang, “Three-dimensional video postproduction and processing,” *Proceedings of the IEEE*, vol. 99, no. 4, pp. 607–625, Apr. 2011.
- [4] L. Jiang, J. He, N. Zhang, and T. Huang, “An overview of 3-D video representation and coding,” *3D Research Journal*, vol. 1, no. 1, pp. 43–47, Aug. 2010.
- [5] P. Ndjiki-Nya, M. Koppel, D. Doshkov, H. lakshman, P. Merkle, K. Muller, and T. Wiegand, “Depth image-based rendering with advanced texture synthesis for 3-D video,” *IEEE Trans. Multimedia*, vol. 13, no. 3, pp. 453–465, Jun. 2011.
- [6] E. Bosc, R. Pepion, P. L. Callet, M. Koppel, P. Ndjiki-Nya, M. Pressigout, and L. Morin, “Towards a new quality metric for 3-D synthesized view assessment,” *IEEE Journal on Selected Topics in Signal Process.*, vol. 5, no. 7, pp. 1332–1343, Nov. 2011.
- [7] G. Tech, Y. Chen, K. Muller, J.-R. Ohm, A. Vetro, and Y.-K. Wang, “Overview of the multiview and 3D extensions of high efficiency video cod-

- ing,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 26, no. 1, pp. 35–49, Jan. 2016.
- [8] A. Vetro, P. Pandit, H. kimata, A. Smolic, and Y.-K. Wang, Joint draft 8 of multiview video coding, Hannover, Germany, Joint Video Team (JVT) Doc. JVT-AB2014, Jul. 2008.
- [9] R. Martins, C. Brites, J. Ascenso, and F. Pereira, “Refining side information for improved transform domain Wyner-Ziv video coding,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 19, no. 9, pp. 1327–1341, Sept. 2009.
- [10] S. Yea and A. vetro, “RD-optimized view synthesis prediction for multiview video coding,” in *Proc. IEEE Int. Conf. Image Process.*, San Antonio, US, Sept. 2007, pp. 209-212.
- [11] J. Y. Lee, J.-L. Lin, Y.-W. Chen, Y.-L. Chang, I. Kovliga, A. Fartukov, M. Mishurovskiy, H.-C. Wey, Y.-W. Huang, and S.-M. Lei, “Depth-based texture coding in AVC-compatible 3D video coding,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 25, no. 8, pp. 1347–1361, Aug. 2015.
- [12] S. Wenger, “Common condition for wire-line, low delay IP/UDP/RTP packet loss resilient testing,” ITU-T VCEG document VCEG-N79r1, Sept. 2001.
- [13] G. Bjontegaard, *Calculation of Average PSNR Differences Between RD Curves*, document VCEG-M33, ITU-T SG16/Q6 (VCEG), Austin, TX, Apr. 2001.
- [14] *Video Codec for Audiovisual Services at px64 kbit/s*, ITU-T Rec. H.261, version 1: Nov. 1990, version 2: Mar. 1993.
- [15] *Video Coding for Low Bit Rate Communicaiton*, ITU-T Rec. H.263, Nov. 1995 (and subsequent editions).
- [16] *Coding of Moving Pictures and Associated Audio for Digital Storage Media at up to About 1.5 Mbit/s—part 2: Video*, ISO/IEC 11172-2 (MPEG-1), ISO/IEC JTC 1, 1993.

- [17] *Coding of Audio-Visual Objects–Part 2: Visual*, ISO/IEC 14496-2 (MPEG-4 Visual version 1), ISO/IEC JTC 1, Apr. 1999 (and subsequent editions).
- [18] *Generic Coding of Moving Pictures and Associated Audio Information–Part 2: Video*, ITU-T Rec. 262 and ISO/IEC 13818-2 (MPEG 2 Video), ITU-T and ISO/IEC JTC 1, Nov. 1994.
- [19] T. Wiegand, G. J. Sullivan, G. Bjontegaard, and A. Luthra, “Overview of the H.264/AVC video coding standard,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 13, no. 7, pp. 560–576, Jul. 2003.
- [20] G. J. Sullivan, J.-R. Ohm, W.-J. Han, and T. Wiegand, “Overview of the high efficiency video coding (HEVC) standard,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 22, no. 12, pp. 1649–1668, Dec. 2012.
- [21] D. Slepian and J. K. Wolf, “Noiseless coding of correlated information sources,” *IEEE Trans. Inform. Theory*, vol. IT-19, no. 4, pp. 471-480, Apr. 1973.
- [22] A. Wyner and J. Ziv, “The rate-distortion function for source coding with side information at the decoder,” *IEEE Trans. Inform. Theory*, vol. IT-22, no. 1, pp. 1-10, Jan. 1976.
- [23] R. Puri, A. Majumdar, P. Ishwar, and K. Ramchandran, “Distributed video coding in wireless sensor networks,” *IEEE Signal Process. Mag.*, vol. 23, no. 4, pp. 94-106, July 2006.
- [24] B. Girod, A. M. Aaron, S. Rane, and D. Rebollo-Monedero, “Distributed video coding,” *Proc. IEEE*, vol. 93, no. 1, pp. 71-83, Jan. 2005.
- [25] 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 Trans. Circuits Syst. Video Technol.*, vol. 21, no. 12, pp. 1859–1868, Dec. 2011.
- [26] *Text of ISO/IEC 14496-10:200X/FDAM 1 Multiview Video Coding*, Document N9978, ISO/IEC JTC1/SC29/WG11, Jul. 2008.

- [27] G. J. Sullivan, J. M. Boyce, Y. Chen, J.-R. Ohm, C. A. Segall, and A. Vetro, "Standardized extensions of high efficiency video coding (HEVC)," *IEEE J. Sel. Topics Signal Process.*, vol. 7, no. 6, pp. 1001-1016, Dec. 2013.
- [28] *Call for Proposal on 3D Video Coding Technology*, ISO/IEC JTC1/SC29/WG11, MPEG, Doc. N12036, Geneva, Switzerland, Mar. 2011.
- [29] 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 Trans. Image Process.*, vol. 22, no. 9, pp. 3449–3458, Sep. 2013.
- [30] K. Muller, H. Schwarz, D. Marpe, C. Bartnik, S. Bosse, H. Brust, T. Hinz, H. Lakshman, P. Merkle, F. H. Rhee, G. Tech, M. Winken, and T. Wiegand, "3D high-efficiency video coding for multi-view video and depth data," *IEEE Trans. Image Process.*, vol. 22, no. 9, pp. 3366–3378, Sep. 2013.
- [31] W. Su, D. Rusanovskyy, M. M. Hannuksela, and H. Li, "Depth-based motion vector prediction in 3D video coding," in *Proc. Picture Coding Symp.*, Krakow, Poland, May 2012, pp. 37–40.
- [32] E. Ekmekcioglu, V. Velisavljevic, and S. T. Worrall, "Content adaptive enhancement of multi-view depth maps for free viewpoint video," *IEEE J. Sel. Topics Signal Process.*, vol. 5, no. 2, pp. 352–361, Apr. 2011.
- [33] I. Daribo, C. Tillier, and B. Pesquet-Popescu, "Adaptive wavelet coding of the depth map for stereoscopic view synthesis," in *Proc. IEEE Int. Workshop Multimedia Signal Process.*, Cairns, Australia, Oct. 2008, pp. 34–39.
- [34] S.-Y. Kim and Y.-S. Ho, "Mesh-based depth coding for 3D video using hierarchical decomposition of depth maps," in *Proc. IEEE Int. Conf. Image Process.*, San Antonio, US, Sept. 2007, pp. V-117–V-120.
- [35] P. Merkle, Y. Morvan, A. Smolic, D. Farin, K. Muller, P. H. N. De With, and T. Wiegand, "The effects of multiview depth video compression on multiview

- rendering,” *Signal Process.: Image Commun.*, vol. 24, no. 1–2, pp. 73–88, Jan. 2009.
- [36] I. Daribo, G. Cheung, and D. Florencio, “Arithmetic edge coding for arbitrarily shaped sub-block motion prediction in depth video compression,” in *Proc. IEEE Int. Conf. Image Process.*, Orlando, US, Oct. 2012, pp. 1541–1544.
- [37] S. Grewatsch and E. Muller, “Sharing of motion vectors in 3D video coding,” in *Proc. IEEE Int. Conf. Image Process.*, Singapore, Oct. 2004, pp. 3271–3274.
- [38] Y. Wang and Q.-F. Zhu, “Error control and concealment for video communication: A review,” *Proceedings of the IEEE*, vol. 86, no. 5, pp. 974–997, May 1998.
- [39] Y. Wang, S. Wenger, J. Wen, and A. K. Katsaggelos, “Error resilient video coding techniques,” *IEEE Signal Process. Mag.*, vol. 17, no. 4, pp. 61–82, Jul. 2000.
- [40] P. Lambert, W. Neve, Y. Dhondt, and R. Walle, “Flexible macroblock ordering in H.264/AVC,” *J. Visual Commun. Image Represent.*, vol. 17, no. 2, pp. 358–375, Apr. 2006.
- [41] T. Stockhammer and M. Bystrom, “H.264/AVC data partitioning for mobile video communication,” in *IEEE Int. Conf. Image Process.*, Singapore, Oct. 2004, pp. 545–548.
- [42] T. Troger and A. Kaup, “Inter-sequence error concealment techniques for multi-broadcast TV reception,” *IEEE Trans. Broadcast.*, vol. 57, no. 4, pp. 777–793, Dec. 2011.
- [43] P. Haskell and D. Messerschmitt, “Resynchronization of motion compensated video affected by ATM cell loss,” in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP)*, San Francisco, California, Mar. 1992, pp. 545–548.

- [44] S. Rane, P. Baccichet, and B. Girod, *Progress Report on CE6: Systematic Lossy Error Protection Based on H.264/AVC Redundant Slices and Flexible Macroblock Ordering*, JVT-T093, Jul. 2006.
- [45] T. Turetti and C. Huitema, "Video conferencing on the Internet," *IEEE/ACM Trans. Networking*, vol. 4, no. 3, pp. 340–351, Jun. 1996.
- [46] G. Cote and F. Kossentini, "Optimal intra coding of blocks for robust video communication over the Internet," *Signal Process.: Image Commun.*, vol. 15, no. 1, pp. 25–34, Sept. 1999.
- [47] R. Zhang, S. L. Regunathan, and K. Rose, "Optimal intra/inter mode switching for robust video communication over the Internet," presented at the 33rd Asilomar Conf. Signals, Syst., Computer, Pacific Grove, California, Oct. 1999.
- [48] A. Leontaris and P. C. Cosman, "Video compression for lossy packet networks with mode switching and a dual-frame buffer," *IEEE Trans. Image Process.*, vol. 13, no. 7, pp. 885–897, Jul. 2004.
- [49] H. Yang and K. Rose, "Recursive end-to-end distortion estimation with model-based cross-correlation approximation," in *Proc. IEEE Int. Conf. Image Process.*, Barcelona, Spain, Sept. 2003, pp. 469–472.
- [50] A. Majumdar, J. Wang, and K. Ramchandran, "Drift reduction in predictive video transmission using a distributed source coded side channel," in *Proc. 12th Ann. ACM Int. Conf. Multimedia*, New York, 2004, pp. 404–407.
- [51] S. Ekmekci and T. Sikora, "Recursive decoder distortion estimation based on AR(1) source modeling for video," in *Proc. Int. Conf. Image Process.*, Singapore, 2004, pp. 187–190.
- [52] T. Stockhammer, T. Wiegand, and S. Wenger, "Optimized transmission of H.26L/JVT coded video over packet-lossy networks," in *Proc. Int. Conf. Image Process.*, Rochester, NY, 2002, pp. 173–176.



- [53] Y.-K. Wang, M. M. Hannuksela, and M. Gabbouj, "Error resilient video coding using unequally protected key pictures," in *Proc. Int. Workshop VL-BV03*, Madrid, Spain, Sep. 2003, pp. 290–297.
- [54] P. Baccichet, S. Rane, and B. Girod, "Systematic lossy error protection based on H.264/AVC redundant slices and flexible macroblock ordering," presented at the Packet Video Workshop (PV 2006), Hangzhou, China.
- [55] C. Zhu, Y.-K. Wang, M. M. Hannuksela, and H. Li, "Error resilient video coding using redundant pictures," in *Proc. Int. Conf. Image Process.*, Atlanta, US, Oct. 2006, pp. 801–804.
- [56] C. Zhu, Y.-K. Wang, M. M. Hannuksela, and H. Li, "Error resilient video coding using redundant pictures," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 19, no. 1, pp. 3–14, Jan. 2009.
- [57] Y. Wang and Q.-F. Zhu, "Signal loss recovery in DCT-based still and video image codecs" in *Proc. SPIE Conf. Visual Commun. Image Process. (VCIP)*, Boston, MA, Nov. 1991, pp. 667–678.
- [58] P. Salama, N. B. Shroff, E. J. Coyle, and E. J. Delp, "Error concealment techniques for encoded video streams," in *Proc. Int. Conf. Image Process.*, Washington, US, Oct. 1995, pp. 9–12.
- [59] Y. Wang, Q.-F. Zhu, and L. Shaw, "Maximally smooth image recovery in transform coding," *IEEE Trans. Commun.*, vol. 41, no. 10, pp. 1544–1551, Oct. 1993.
- [60] W. Zhu, Y. Wang, and Q.-F. Zhu, "Second-order derivative-based smoothness measure for error concealment in DCT-based codecs," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 8, no. 6, pp. 713–718, Oct. 1998.
- [61] S. D. Rane, G. Sapiro, and M. Bertalmio, "Structure and texture filling-in of missing image blocks in wireless transmission and compression," *IEEE Trans. Image Process.*, vol. 12, no. 3, pp. 296–303, Mar. 2003.

- [62] W. Y. Kung, C. S. Kim, and C. J. Kuo, "Spatial and temporal error concealment techniques for video transmission over noisy channel," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 16, no. 7, pp. 789–802, Jul. 2006.
- [63] X. Li and M. Orchard, "Novel sequential error concealment techniques using orientation adaptive interpolation," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 12, no. 10, pp. 857–864, Oct. 2002.
- [64] P. Salama, N. B. Shroff, and E. J. Delp, "Error concealment in MPEG video streams over ATM networks," *IEEE J. Sel. Areas Commun.*, vol. 18, no. 6, pp. 1129–1144, Jun. 2000.
- [65] M. J. Chen, L. G. Chen, and R. M. Weng, "Error concealment of lost motion vectors with overlapped motion compensation," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 7, no. 3, pp. 560–563, Jun. 1997.
- [66] W. M. Lam, A. R. Reibman, and B. Liu, "Recovery of lost erroneously received motion vectors," in *Proc. IEEE Int. Conf. Acoust. Speech Singal Process. (ICASSP)*, Minneapolis, USA, Apr. 1993, pp. 417–420.
- [67] S. Tsekeridou, F. A. Cheikh, M. Gabbouj, and I. Pitas, "Motion field estimation by vector rational interpolation for error concealment purposes," in *Proc. IEEE Int. Conf. Acoust. Speech Signal Process. (ICASSP)*, Phoenix, Arizona, Mar. 1999, pp. 3397–3400.
- [68] M. Al-Mualla, N. Canagarajahm, and D. R. Bull, "Error concealment using motion field interpolation," in *Proc. IEEE Int. Conf. Image Process.*, Chicago, Illinois, Oct. 1998, pp. 512–516.
- [69] J. H. Zheng and L. P. Chau, "A temporal error concealment algorithm for H.264 using Lagrange interpolation," in *Proc. IEEE Int. Symp. Circuits Syst.*, Vancouver, BC, Canada, May 2004, pp. 133–136.
- [70] W. N. Lie and Z. W. Gao, "Video error concealment by integrating greedy suboptimization and Kalman filtering techniques," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 16, no. 8, pp. 982–992, Aug. 2006.

- [71] Q. Peng, T. Yang, and C. Zhu, "Block-based temporal error concealment for video packet using motion vector extrapolation" in *Proc. IEEE Commun., Circuits and Syst. and West Sino Expositions*, Chengdu, China, Jul. 2002, pp. 10–14.
- [72] Y. Chen, K. Yu, J. Li, and S. Li, "An error concealment algorithm for entire frame loss in video transmission," presented at the IEEE Picture Coding Symp., San Francisco, USA, 2004.
- [73] B. Yan and H. Gharavi, "A hybrid frame concealment algorithm for H.264/AVC," *IEEE Trans. Image Process.*, vol. 19, no. 1, pp. 98–107, Jan. 2010.
- [74] X. Ji, D. Zhao, and W. Gao, "Concealment of whole-picture loss in hierarchical B-picture scalable video coding," *IEEE Trans. Multimedia*, vol. 11, no. 1, pp. 11–22, Jan. 2009.
- [75] B. Girod and N. Farber, "Feedback-based error control for mobile video transmission," *Proceedings of the IEEE*, vol. 87, no. 10, pp. 1707–1723, Oct. 1999.
- [76] Y. Wang, M. Claypool, and R. Kinicki, "Modeling RPS and evaluating video repair with VQM," *IEEE Trans. Multimedia*, vol. 11, no. 1, pp. 128–137, Jan. 2009.
- [77] G. Cheung, W.-T. Tan, and C. Chan, "Reference frame optimization for multiple-path video streaming with complexity scaling," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 17, no. 6, pp. 649–662, Jun. 2007.
- [78] A. Kubota, A. Smolic, M. Magnor, M. Tanimoto, T. Chen, and C. Zhang, "Multiview imaging and 3DTV," *IEEE Signal Process. Mag.*, vol. 24, no. 6, pp. 10–21, Nov. 2007.
- [79] ITU-T and ISO/IEC JTC 1, "Final Draft Amendment 3," Amendment 3 to ITU-T Recommendation H.262 and ISO/IEC 13818-2, ISO/IEC JTC 1/SC 29/WG 11 (MPEG) Doc. N1366, Sept. 1996.

- 
- [80] Advanced Video Coding for Generic Audiovisual Services, Standard ISO/IEC JTC 1, Mar. 2012.
- [81] M. Schier and M. Welzl, "Optimizing selective ARQ for H.264 live streaming: a novel method for predicting loss impact in real time," *IEEE Trans. Multimedia*, vol. 14, no. 2, pp. 415-430, April 2012.
- [82] W. Xiang, C. Zhu, C. K. Siew, Y. Xu, and M. Liu, "Forward error correction-based 2-D layered multiple description coding for error resilient H.264 SVC video transmission," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 19, no. 12, pp. 1730-1738, Dec. 2009.
- [83] J. Y. Liao and J. D. Villasenor, "Adaptive intra update for video coding over noisy channel," in *Proc. Int. Conf. Image Processing (ICIP)*, Lausanne, Switzerland, Sept. 1996, pp. 763-766.
- [84] H. Yang and K. Rose, "Optimizing motion compensated prediction for error resilient video coding," *IEEE Trans. Image Process.*, vol. 19, no. 1, pp. 108-118, Jan. 2010.
- [85] I. Radulovic, P. Frossard, Y.-K. Wang, M. M. Hannuksela, and A. Hallapuro, "Multiple description video coding with H.264/AVC redundant pictures," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 20, no. 1, pp. 144-148, Jan. 2010.
- [86] X. Xiang, D. Zhao, Q. Wang, S. Ma, and W. Gao, "Rate-distortion optimization with inter-view refreshment for stereoscopic video coding over error-prone networks," in *Proc. SPIE. Visual Communications and Image Process. (VCIP)*, San Jose, USA, Jan. 2009, pp. 72570K-1-72570K-8.
- [87] A. S. Tan, A. Aksay, G. B. Akar, and E. Arikan, "Rate-distortion optimization for stereoscopic video streaming with unequal error protection," *Eurasip Journal on Advances in Signal Processing*, pp. 14, Jan. 2009.
- [88] 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, May 2009.
- [89] S. Liu, Y. Chen, Y.-K. Wang, M. Gabbouj, M. M. Hannuksela, and H. Li, "Frame loss error concealment for multi-view video coding," in *Proc. IEEE Int. Sym. Circuits and System (ISCAS)*, Seattle, Washington, USA, May 2008, pp. 3470-3473.
- [90] Y. Zhou, C. Hou, and W. Xiang, "Modeling of transmission distortion for multi-view video in packet lossy networks", in *Proc. IEEE Global Telecommun. Conf. (GLOBECOM)*, Miami, Florida, USA, Dec. 2010, pp. 1-5.
- [91] M.B. Dissanaayake, D.V.S. X. D. Sevail, S.T. Worrall, and W.A.C. Fernando, "Error resilient technique for MVC using redundant disparity vectors," in *Proc. IEEE Int. Conf. Multimedia and Expo (ICME)*, Singapore, Jul. 2010, pp. 1712-1717.
- [92] D. Slepian and J. K. Wolf, "Noiseless coding of correlated information sources," *IEEE Trans. Inform. Theory*, vol. IT-19, no. 4, pp. 471-480, Apr. 1973.
- [93] A. Wyner and J. Ziv, "The rate-distortion function for source coding with side information at the decoder," *IEEE Trans. Inform. Theory*, vol. IT-22, no. 1, pp. 1-10, Jan. 1976.
- [94] R. Puri, A. Majumdar, P. Ishwar, and K. Ramchandran, "Distributed video coding in wireless sensor networks," *IEEE Signal Process. Mag.*, vol. 23, no. 4, pp. 94-106, July 2006.
- [95] B. Girod, A. M. Aaron, S. Rane, and D. Rebollo-Monedero, "Distributed video coding," *Proc. IEEE*, vol. 93, no. 1, pp. 71-83, Jan. 2005.
- [96] A. Sehgal, A. Jagmohan, and N. Ahuja, "Wyner-Ziv coding of video: An error resilient compression framework," *IEEE Trans. Multimedia*, vol. 6, no. 2, pp. 249-258, Apr. 2004.

- [97] Y. Zhang, C. Zhu, and K. Yap, "A joint source-channel video coding scheme based on distributed video coding," *IEEE Trans. Multimedia*, vol. 10, no. 8, pp. 1648-1656, Dec. 2008.
- [98] Y. Zhang, H. Xiong, Z. He, S. Yu, and C. Chen, "An error resilient video coding scheme using embedded Wyner-Ziv description with decoder side non-stationary distortion modeling," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 21, no. 4, pp. 498-512, Apr. 2011.
- [99] Z. Xue, K. K. Loo, J. Cosmas, M. Tun, L. Feng, and P.-Y. Yip, "Error resilient scheme for wavelet video codec using automatic ROI detection and Wyner-Ziv coding over packet erasure channel," *IEEE Trans. Broadcast.*, vol. 56, no. 4, pp. 481-493, Dec. 2010.
- [100] O. Crave, B. Pesquet-Popescu, and C. Guillemot, "Robust video coding based on multiple description scalar quantization with side information," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 20, no. 6, pp. 769-779, Jun. 2010.
- [101] L. Qing, E. Masala, and X. He, "Practical distributed video coding in packet lossy channel," *Optical Engineering*, vol. 52, no. 7, pp. 1-18, Jul. 2013.
- [102] X. Guo, F. Wu, D. Zhao, and W. Gao, "Wyner-Ziv-based multiview video coding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 18, no. 6, pp. 713-724, June 2008.
- [103] Y. Li, S. Ma, D. Zhao, and W. Gao, "Modeling correlation noise statistics at decoder for multi-view distributed video coding," in *Proc. IEEE Symp. Circuits and Systems (ISCAS)*, Taipei, Taiwan, May 2009, pp. 2597-2600.
- [104] C. Guillemot, F. Pereira, L. Torres, T. Ebrahimi, R. Leonardi, and J. Ostermann, "Distributed monoview and multiview video coding," *IEEE Signal Process. Mag.*, vol. 24, no. 5, pp. 67-76, Sept. 2007.
- [105] C. Yeo and K. Ramchandran, "Robust distributed multiview video compression for wireless camera networks," *IEEE Trans. Image Process.*, vol. 19, no. 4, pp. 995-1008, Apr. 2010.

- [106] C. Brites and F. Pereira, "Correlation noise modeling for efficient pixel and transform domain Wyner-Ziv video coding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 18, no. 9, pp. 1177-1189, Sept. 2008.
- [107] "Description of core experiments in MVC," *ISO/IEC JTC1/SC29/WG11, MPEG2006/W7798*, Apr. 2006.
- [108] P. Merkle, A. Smolic, K. Mller, and T. Wiegand, "Efficient prediction structures for multiview video coding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 17, no. 11, pp. 1461-1473, Nov. 2007.
- [109] A. Vetro, T. Wiegand, and G. J. Sullivan, "Overview of the stereo and multi-view video coding extensions of the H.264/MPEG-4 AVC standard", *Proceedings of the IEEE*, vol. 99, no. 4, pp. 626-642, Apr. 2011.
- [110] Z. He and H. Xiong, "Transmission distortion analysis for real time video encoding and streaming over wireless networks," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 16, no. 9, pp. 1051-1062, Sept. 2006.
- [111] J. Wang, A. Majumdar, and K. Ramchandran, "Robust video transmission with distributed source coded auxiliary channel," *IEEE Trans. Image Process.*, vol. 18, no. 12, pp. 2695-2705, Dec. 2009.
- [112] S. Rane, P. Baccichet, and B. Girod, "Systematic lossy error protection of video signals," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 18, no. 10, pp. 1347-1360, Oct. 2008.
- [113] R. Zhang, S. L. Regunathan, and K. Rose, "Video coding with optimal inter/intra-mode switching for packet loss resilience," *IEEE J. Sel. Areas Commun.*, vol. 18, no. 6, pp. 966-976, Jun. 2000.
- [114] M. F. Sabir, R. W. Heath, Jr., and A. C. Bovik, "Joint source-channel distortion modeling for MPEG-4 video," *IEEE Trans. Image Process.*, vol. 18, no. 1, pp. 90-105, Jan. 2009.
- [115] D. Varodayan, A. Aaron, and B. Girod, "Rate-adaptive codes for distributed source coding," *Singal Process. (EURASIP)*, no. 86, pp. 3123-3130, Nov. 2006.

- [116] D. Kubasov, J. Nayak, and C. Guillemot, "Optimal reconstruction in Wyner-Ziv video coding with multiple side information," in *Proc. IEEE 9th Workshop Multimedia Signal Process.*, Chania, Crete, Greece, Oct. 2007, pp. 183-186.
- [117] C. Brites and F. Pereira, "An efficient encoder rate control solution for transform domain Wyner-Ziv video coding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 21, no. 9, pp. 1278-1292, Sept. 2011.
- [118] Z. He, Y. Liang, L. Chen, I. Ahmad, and D. Wu, "Power-rate-distortion analysis for wireless video communication under energy constraint," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 15, no. 5, pp. 645-657, May 2005.
- [119] ISO/IEC JTC1/SC29/WG11, WD 3 Reference Software for MVC, Doc. JVT-AC207. Busan, Korea, 2008.
- [120] A. Aarion, S. Rane, E. Setton, and B. Griod, "Transform-domain Wyner-Ziv codec for video," in *Proc. Visual Communications and Image Processing (VCIP)*, San Jose, CA, Jan. 2004.
- [121] S. Wenger, "Proposed error patterns for internet experiments," ITU-T VCEG document Q15-I-16r1, Oct. 1999.
- [122] C. Lambiri, "On the estimation and control of packet loss for VPN services," Ph.D. Dissertation, University of Ottawa, Ottawa, 2003.
- [123] M. Yajnik, S. B. Moon, J. Kurose, and D. Towsley, "Measurement and modeling of the temporal dependence in packet loss," in *Proc. IEEE INFOCOM*, New York, USA, vol. 1, Mar. 1999, pp. 345-352.
- [124] K. Muller, P. Merkle, and T. Wiegand, "3-D video representation using depth maps," *Proceedings of the IEEE*, vol. 99, no. 4, pp. 643-656, Apr. 2011.
- [125] A. Vetro, A. M. Tourapis, K. Muller, and T. Chen, "3D-TV content storage and transmission," *IEEE Trans. Broadcasting*, vol. 57, no. 2, pp. 384-394, Jun. 2011.



- [126] C. Fehn, "Depth-image-based rendering (DIBR), compression and transmission for a new approach on 3-D-TV," in *Proc. 11th SPIE Stereoscopic Displays Virtual Reality Syst.*, San Jose, CA, Jan. 2004, pp. 93-104.
- [127] ITU-T SG 16 WP 3 and ISO/IEC JTC 1/SC 29/WG 11, 3D-AVC Test Model 8, Doc. JCT3V-F1003, Oct. 2013.
- [128] ITU-T SG 16 WP 3 and ISO/IEC JTC 1/SC 29/WG 11, Test Model 6 of 3D-HEVC and MV-HEVC, Doc. JCT3V-F1005, Oct. 2013.
- [129] R. O. Hinds, T. N. Pappas, and J. S. Lim, "Joint block-based video source/channel coding for packet-switched networks," in *Proc. SPIE VCIP*, vol. 3309, San Jose, CA, Jan. 1998, pp. 124-133.
- [130] J. C. Schmidt and K. Rose, "Jointly optimized mode decisions in redundant video streaming," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 21, no. 4, pp. 513-518, Apr. 2011.
- [131] H. Yang and K. Rose, "Advance in recursive per-pixel end-to-end distortion estimation for robust video coding in H.264/AVC," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 17, no. 7, pp. 845-856, Jul. 2007.
- [132] Z. Chen, P. V. Pahalawatta, A. Michael Tourapis, and D. Wu, "Improved estimation of transmission distortion for error-resilient video coding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 22, no. 4, pp. 636-647, Apr. 2012.
- [133] G. Cote, S. Shirani, and F. Kossentini, "Optimal mode selection and synchronization for robust video communications over error-prone networks," *IEEE J. Sel. Areas Commun.*, vol. 18, no. 6, pp. 952-965, Jun. 2000.
- [134] Y. Zhang, W. Gao, Y. Lu, Q. Huang, and D. Zhao, "Joint source-channel rate-distortion optimization for H.264 video coding over error-prone networks," *IEEE Trans. Multimedia*, vol. 9, no. 3, pp. 445-454, Mar. 2007.
- [135] Y. Guo, Y. Chen, Y. Wang, H. Li, M. M. Hannuksela, and M. Gabbouj, "Error resilient coding and error concealment in scalable video coding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 19, no. 6, pp. 781-795, Jun. 2009.

- [136] S. Ekmekci, P. Frossard, and T. Sikora, "Distortion estimation for temporal layered video coding," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP)*, Philadelphia, USA, Mar. 2005, pp. 189–192.
- [137] F. Zhai, Y. Eisenberg, T. N. Pappas, R. Berry, and A. K. Katsaggelos, "Rate-distortion optimized hybrid error control for real-time packetized video transmission," *IEEE Trans. Image Process.*, vol. 15, no. 1, pp. 40–53, Jan. 2006.
- [138] Y. Zhou, C. Hou, W. Xiang, and F. Wu, "Channel distortion modeling for multi-view video transmission over packet-switched networks," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 21, no. 11, pp. 1679–1692, Nov. 2011.
- [139] B. Macchiavello, C. Dorea, E. M. Hung, G. Cheung, and W. Tan, "Reference frame selection for loss-resilient depth map coding in multiview video conferencing," in *Proc. SPIE Visual Inf. Process. Commun.*, Burlingame, USA, Feb. 2012, pp. 83050C-1–83050C-11.
- [140] B. Macchiavello, C. Dorea, E. M. Hung, G. Cheung, and W. Tan, "Reference frame selection for loss-resilient texture&depth map coding in multiview video conferencing," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Orlando, USA, Sep. 2012, pp. 1653–1656.
- [141] W.-S. Kim, A. Ortega, P. Lai, D. Tian, and C. Gomila, "Depth map coding with distortion estimation of rendered view," in *Proc. SPIE Visual Inf. Process. Commun.*, San Jose, California, Jan. 2010, pp. 75430B-1–75430B-10.
- [142] Y. Liu, Q. Huang, S. Ma, D. Zhao, and W. Gao, "Joint video/depth rate allocation for 3-D video coding based on view synthesis distortion model," *Signal Process.: Image Commun.*, vol. 24, no. 8, pp. 666–681, Jun. 2009.
- [143] F. Shao, G. Jiang, M. Yu, K. Chen, and Y. Ho, "Asymmetric coding of multi-view video plus depth based 3-D video for view rendering," *IEEE Trans. Multimedia*, vol. 14, no. 1, pp. 157–167, Feb. 2012.
- [144] H. Yuan, Y. Chang, J. Huo, F. Yang, and Z. Lu, "Model-based joint bit allocation between texture videos and depth maps for 3-D video coding,"

- IEEE Trans. Circuits Syst. Video Technol.*, vol. 21, no. 4, pp. 485-497, Apr. 2011.
- [145] Q. Wang, X. Ji, Q. Dai, and N. Zhang, "Free viewpoint video coding with rate-distortion analysis," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 22, no. 6, pp. 875-889, Jun. 2012.
- [146] Z. He, J. Cai, and C. W. Chen, "Joint source channel rate-distortion analysis for adaptive mode selection and rate control in wireless video coding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 12, no. 6, pp. 511-523, Jun. 2002.
- [147] S. Liu and C. W. Chen, "Scalable video transmission: packet loss induced distortion modeling and estimation," in *ACM NOSSDAV*, Vancouver, BC, Canada, Jun. 2011, pp. 111-116.
- [148] G. J. Sullivan and T. Wiegand, "Rate-distortion optimization for video compression," *IEEE Signal Process. Mag.*, vol. 15, no. 6, pp. 74-90, Nov. 1998.
- [149] P. A. Chou and Z. Miao, "Rate-distortion optimized streaming of packetized media," *IEEE Trans. Multimedia*, vol. 8, no. 2, pp. 390-404, Apr. 2006.
- [150] K. Ramchandran and M. Vetterli, "Best Wavelet packet bases in a rate-distortion sense," *IEEE Trans. Image Process.*, vo. 2, no. 2, pp. 160-175, Apr. 1993.
- [151] T. Stockhammer, D. Kontopodis, and T. Wiegand, "Rate-distortion optimization for JVT/H.26L video coding in packet loss environment," presented at the 12th Int. Packet Video Workshop, Pittsburgh, PA, Apr. 2002.
- [152] Y. Liu, J. Wang, and H. Zhang, "Depth image-based temporal error concealment for 3-D video transmission," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 20, no. 4, pp. 600-604, Apr. 2010.
- [153] B. Macchiavello, C. Dorea, E. M. Hung, G. Cheung, and W. Tan, "Loss-resilient coding of texture and depth for free-viewpoint video conferencing," *arXiv:1305.5464*, May 2013.

- [154] B. T. Oh, J. Lee, and D.-S. Park, "Depth map coding based on synthesized view distortion function," *IEEE Journal of Selected Topics in Signal Process.*, vol. 5, no. 7, pp. 1344-1352, Nov. 2011.
- [155] Z. Li, W. Gao, F. Pan, S. Ma, K. P. Lim, G. Feng, X. Lin, S. Rahardja, H. Lu, and Y. Lu, "Adaptive rate control with HRD consideration," Joint Video Team document JVT-H014, May 2003.
- [156] Y. Zhang, S. Kwong, L. Xu, S. Hu, G. Jiang, and C. C. J. Kuo, "Regional bit allocation and rate distortion optimization for multi-view depth video coding with view synthesis distortion model," *IEEE Trans. Image Process.*, vol. 22, no. 9, pp. 3497-3512, Sept. 2013.
- [157] *Applications and Requirements on 3-D Video Coding*, ISO/IEC JTC1/SC29/WG11, MPEG Doc. N12035, Mar. 2011.
- [158] P. Merkle, K. Muller, A. Smolic, and T. Wiegand, "Efficient compression of multi-view video exploiting inter-view dependences based on H.264/MPEG4-AVC," in *Proc. IEEE Int. Conf. Multimedia and Expo (ICME)*, Toronto, Ontario, Canada, Jul. 2006, pp. 1717-1720.
- [159] E. Martinian, A. Behrens, J. Xin, and A. Vetro, "View synthesis for multiview video compression," in *Picture Coding Symposium (PCS)*, Beijing, China, Apr. 2006, pp. 38-39.
- [160] S. Yea and A. Vetro, "View synthesis prediction for multiview video coding," *Signal Process: Image Commun.*, vol. 24, no. 1-2, pp. 89-100, Jan. 2009.
- [161] S. Shimizu, M. Kitahara, H. Kimata, K. Kamikura, and Y. Yashima, "View scalable multiview video coding using 3-D warping with depth map," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 17, no. 11, pp. 1485-1495, Nov. 2007.
- [162] F. Zou, D. Tian, A. Vetro, and A. Ortega, "View synthesis prediction using adaptive depth quantization for 3D video coding," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Melbourne, Australia, Sept. 2013, pp. 1694-1698.

- [163] D. Tian, F. Zhou, and A. Vetro, "CE1.h: Backward View Synthesis Prediction using Neighboring Blocks," *ITU-T SG 16 WP 3 and ISO/IEC JTC 1/SC 29/WG 11*, Doc. JCT3V-C0152, Geneva, Switzerland, Jan. 2013.
- [164] C. Bal and T. Q. Nguyen, "Multi-view video plus depth coding with depth-based prediction mode," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 24, no. 6, pp. 995-1005, Jun. 2014.
- [165] M. Domanski, O. Stankiewicz, K. Wegner, M. Kurc, J. Konieczny, J. Siast, J. Stankowski, R. Ratajczak, and T. Grajek, "High efficiency 3D video coding using new tools based on view synthesis," *IEEE Trans. Image Process.*, vol. 22, no. 9, pp. 3517-3526, Sep. 2013.
- [166] B. Yan and J. Zhou, "Efficient frame concealment for depth image-based 3-D video transmission," *IEEE Trans. Multimedia*, vol. 14, no. 3, pp. 936-941, Jun. 2012.
- [167] V.-H. Doan, V.-A. Nguyen, and M. N. Do, "Efficient view synthesis based error concealment method for multiview video plus depth," in *Proc. Int. Symposium Circuits and Systems (ISCAS)*, Beijing, China, May 2013, pp. 2900-2903.
- [168] B. Macchiavello, C. Dorea, E. M. Hung, G. Cheung, and W. Tan, "Loss-resilient coding of texture and depth for free-viewpoint video conferencing," *IEEE Trans. Multimedia*, vol. 16, no. 3, pp. 711-725, Apr. 2014.
- [169] P. Gao and W. Xiang, "Rate-distortion optimized mode switching for error-resilient multi-view video plus depth based 3-D video coding," *IEEE Trans. Multimedia*, vol. 16, no. 7, pp. 1797-1808, Nov. 2014.
- [170] D. Tian, P. Lai, P. Lopez, and C. Gomila, "View synthesis techniques for 3D video," in *Proc. SPIE*, vol. 7443, pp. 74430T-1-74430T-11, Aug. 2009.
- [171] Y. Gao, G. Cheung, T. Maugey, P. Frossard, and J. Liang, "Encoder-driven inpainting strategy in multiview video compression," *IEEE Trans. Image Process.*, vol. 25, no. 1, pp. 134-149, Jan. 2016.

- [172] C. Lee and Y.-S. Ho, "A framework of 3D video coding using view synthesis prediction," in *Picture Coding Symposium (PCS)*, Krakow, Poland, May 2012, pp. 9-12.
- [173] L. Fang, N.-M. Cheung, D. Tian, A. Vetro, H. Sun, and O. C. Au, "An analytical model for synthesis distortion estimation in 3D video," *IEEE Trans. Image Process.*, vol. 23, no. 1, pp. 185-199, Jan. 2014.
- [174] H. Yuan, S. Kwong, J. Liu, and J. Sun, "A novel distortion model and Lagrangian multiplier for depth map coding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 24, no. 3, pp. 443-451, Mar. 2014.
- [175] F. Shao, G. Jiang, W. Lin, M. Yu, and Q. Dai, "Joint bit allocation and rate control for coding multi-view video plus depth based 3D video," *IEEE Trans. Multimedia*, vol. 15, no. 8, pp. 1843-1854, Dec. 2013.
- [176] K. Stuhlmuller, N. Farber, M. Link, and B. Girod, "Analysis of video transmission over lossy channels," *IEEE J. Sel. Areas Commun.*, vol. 18, no. 6, pp. 1012-1032, Jun. 2000.
- [177] T. Wiegand, N. Farber, K. Stuhlmuller, and B. Girod, "Error-resilient video transmission using long-term memory motion-compensated prediction," *IEEE J. Sel. Areas Commun.*, vol. 18, no. 6, pp. 1050-1062, Jun. 2000.
- [178] Z. Chen and D. Wu, "Prediction of transmission distortion for wireless video communication: analysis," *IEEE Trans. Image Process.*, vol. 21, no. 3, pp. 1123-1137, Mar. 2012.
- [179] Y. J. Liang, J. G. Apostolopoulos, B. Girod, "Analysis of packet loss for compressed video: effect of burst losses and correlation between error frames," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 18, no. 7, pp. 861-874, Jul. 2008.
- [180] H. Wang and S. Kwong, "Hybrid model to detect zero quantized DCT coefficients in H.264," *IEEE Trans. Multimedia*, vol. 9, no. 4, pp. 728-735, Jun. 2007.

- [181] S. Wan and E. Izquierdo, "Rate-distortion optimized motion-compensated prediction for packet loss resilient video coding," *IEEE Trans. Image Process.*, vol. 16, no. 5, pp. 1327-1338, May 2007.
- [182] ISO/IEC JTC1/SC29/WG11, 3DV/FTV EE2: Report on VSRS Extrapolation, Doc. M18356, Guangzhou, China, 2010.
- [183] S. Wenger, "H.264/AVC over IP," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 13, no. 7, pp. 645-656, Jul. 2003.
- [184] Application parameter settings for TMS320DM365 H.264 encoder [Online]. Available: <http://www.ti.com/lit/an/spraba9/spraba9.pdf>
- [185] 3D-ATM reference software version 6.0 [Online]. Available: <http://mpeg3dv.nokiaresearch.com/svn/mpeg3dv/tags/3DV-ATMv6.0/>
- [186] X. Zhang, Y. Zhao, C. Lin, H. Bai, C. Yao and A. Wang, "Warping-driven mode selection for depth error concealment," in *2nd IEEE Global Conference on Signal and Information Processing*, Atlanta, Georgia, Dec. 2014, pp. 302-306.
- [187] N.-M. Cheung, A. Ortega, and G. Cheung, "Rate distortion based reconstruction optimization in distributed source coding for interactive multiview video stream," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Hong Kong, China, Sept. 2010, pp. 3721-3724.
- [188] G. Petrazzuoli, T. Maugey, M. Cagnazzo, B. Pesquet-Popescu, "Depth-based multi-view distributed video coding," *IEEE Trans. Multimedia*, vol. 16, no. 7, pp. 1834-1848, Nov. 2014.
- [189] S. Ma, S. Wang, and W. Gao, "Low complexity adaptive view synthesis optimizaiton in HEVC based 3D video coding," *IEEE Trans. Multimedia*, vol. 16, no. 1, pp. 266-271, Jan. 2014.
- [190] F. Zou, D. Tian, A. Vetro, H. Sun, O. C. Au, and S. Shimizu, "View synthesis prediction in the 3-D video coding extensions of AVC and HEVC," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 24, no. 10, pp. 1696-1708, Oct. 2014.

- [191] D. Zhang and D. Ionescu, “Reactive estimation of packet loss probability for IP-based video services,” *IEEE Trans. Broadcasting*, vol. 55, no. 2, pp. 375-385, Jun. 2009.