UNIVERSITY OF CALIFORNIA, SAN DIEGO

Unequal Error Protection for Compressed Video over Noisy Channels

A dissertation submitted in partial satisfaction of the
requirements for the degree
Doctor of Philosophy

in

Electrical Engineering
(Signal and Image Processing)

by

Arash Vosoughi

Committee in charge:

Professor Pamela C. Cosman, Chair
Professor William S. Hodgkiss
Professor Laurence B. Milstein
Professor Truong Q. Nguyen
Professor Steven J. Swanson

2015

The dissertation of Arash Vosoughi is approved, and it is acceptable in quality and form for publication on microfilm and electronically:

_____

_____

_____

_____

_____
Chair

University of California, San Diego

2015

DEDICATION

To my dearest parents,
my sister Azadeh, my brother-in-law Alireza, and
my lovely wife Nazanin

EPIGRAPH

You can never cross the ocean until
you have the courage to lose sight of the shore.
—Christopher Columbus

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGEMENTS

First I would like to thank my advisor Professor Pamela Cosman for all her support, guidance, and advice throughout my PhD studies. She has been always generous with her time and forthcoming with her broad knowledge. I have learned many technical as well as life lessons from her. I thank her for being both advisor and mentor.

I also express my deepest gratitude to my other advisor Professor Laurence Milstein for all his support, advice, and friendly attitude. I believe it has been a great chance for me to have him as my advisor, as I have learned tremendously from him, lessons that I have used in my personal and professional lives. I greatly appreciate his punctiliousness, humbleness, and being respectful to the students.

My special thanks are due to the committee members of my dissertation, Professor William Hodgkiss, Professor Truong Nguyen, and Professor Steven Swanson for their invaluable time invested in reviewing my thesis and their constructive feedbacks.

I want to thank my mother and father who are the most precious people in my life. None of my achievements have been possible without their support and help. Raising four kids all with high level academic education is not an easy task; it requires lots of devotion, self-sacrifice, and patience. Thank you mom and dad for every thing you have done for me. Thank you for enduring all the hardships when raising us and we may have been ignorant and not appreciative to you. Thank you for all the moments that you chose to sacrifice your lives to create happier memories for us. Thank you for all the painful long journeys you have taken so far to visit and make sure we have good lives. You are my everything and I want nothing but your happiness. I hope this thesis is a little gift to you, my dearest real angels.

I want to thank my sister Azadeh who has always guided me and supported me throughout all my years of education. I owe lots of my success to Azadeh, as I always felt I am backed with a kind sister at home who can help me in learning math and science. Thank you my dear Azadeh for all your support when I was far from home. You have always been my idol that I learned from you a lot. You

taught me how one can generously be a help not only to his family members but also to others.

I also want to thank my dearest brother-in-law Alireza who is not among us anymore. My dear Alireza, you will always be in my mind and my heart. I can never forget all the things you did for me. You always treated me as your brother; indeed you were a kind, supportive, and thoughtful brother to me. I always admired your intelligence and enthusiasm for living a happy life. Thank you for all your generous help and support, which always remind me to try to be a better human being.

Last, but not least, I want to thank my beautiful lovely wife Nazanin who has brought happiness and peacefulness to my life. Thank you my dear Nazanin for being patient and keeping your hope of having a better life in the gloomy days of our lives. Thank you for taking care of every thing around home, when I was extremely busy with my PhD studies.

Chapter 2 of this dissertation is a reprint of the material as it appears in A. Vosoughi, V. Testoni, P. Cosman, and L. Milstein, "Multiview coding and error correction coding for 3D video over noisy channels", *Signal Processing: Image Communication*, vol. 30, pp. 107-120, Jan 2015, and is, in part, based on the material as it appears in A. Vosoughi, V. Testoni, P. Cosman, and L. Milstein, "Joint source-channel coding of 3D video using multiview coding", in Proc. *ICASSP*, 2013. I was the primary author and the co-authors Prof. Cosman and Prof. Milstein directed and supervised the research. The co-author Dr. Testoni also contributed to the ideas in this work. This research was supported by the Intel/Cisco Video Aware Wireless Networks (VAWN) program, by InterDigital, Inc., and by the National Science Foundation under grant number CCF-1160832.

Chapter 3 of this dissertation is a reprint of the material as it appears in A. Vosoughi, P. Cosman, and L. Milstein, "Joint source-channel coding and unequal error protection for video plus depth", *IEEE Signal Processing Letters*, vol. 22, Jan 2015. I was the primary author and the co-authors Prof. Cosman and Prof. Milstein directed and supervised the research. This research was supported by the Intel/Cisco Video Aware Wireless Networks (VAWN) program and by the

National Science Foundation under grant number CCF-1160832.

Chapter 4 of this dissertation is a reprint of the material as it appears in A. Vosoughi, S.-H. Chang, S.-H. Kim, P. Cosman, and L. Milstein, "Digital video broadcasting of spatially scalable video with multiple antennas", *manuscript under preparation*. I was the primary author and the co-authors Prof. Cosman and Prof. Milstein directed and supervised the research. The co-authors Dr. Chang and Dr. Kim also contributed to the ideas in this work and helped with the simulation process. This work was partially supported by the Army Research Office under Grant #W911NF-14-1-0340, and by the Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (2013R1A1A2065143).

VITA

2000-2005	B. S. in Electrical Engineering, Khajeh Nasir University of Technology, Tehran, Iran
2006-2008	M. S. in Electrical Engineering, Sharif University of Technology, Tehran, Iran
2010-2015	Ph. D. in Electrical Engineering (Signal and Image Processing), University of California, San Diego

PUBLICATIONS

Journal Papers

A. Vosoughi, V. Testoni, P. Cosman, and L. Milstein, "Multiview coding and error correction coding for 3D video over noisy channels", *Signal Processing: Image Communication*, vol. 30, pp. 107-120, Jan 2015.

A. Vosoughi, P. Cosman, and L. Milstein, "Joint source-channel coding and unequal error protection for video plus depth", *IEEE Signal Processing Letters*, vol. 22, Jan 2015.

A. Vosoughi, S.-H. Chang, S.-H. Kim, P. Cosman, and L. Milstein, "Digital video broadcasting of spatially scalable video with multiple antennas", manuscript under preparation.

Q. Song, A. Vosoughi, P. Cosman, and L. Milstein, "Rate distortion optimization and unequal error protection", manuscript under preparation.

Conference Papers

A. Vosoughi, V. Testoni, P. Cosman, and L. Milstein, "Joint source-channel coding of 3D video using multiview coding", in Proc. *ICASSP*, 2013.

A. Vosoughi and P. Cosman, "Frame loss visibility modeling of stereoscopic video for H.264/AVC-MVC", in Proc. *ICIP*, 2012.

A. Vosoughi, M.B. Shamsollahi, and A. Vosoughi, "Nonsubsampled higher-density discrete wavelet transform: filter design and application in image contrast enhancement," in Proc. *ICIP*, 2009.

A. Vosoughi, A. Vosoughi, and M. B. Shamsollahi, "Nonsubsampled higher-density discrete wavelet transform for image denoising," in Proc. *ICASSP*, 2009.

A. Vosoughi and M. B. Shamsollahi, "Speckle noise reduction of ultrasound images using M-band wavelet transform and Wiener filter in a homomorphic framework," in Proc. *BMEI*, 2008.

ABSTRACT OF THE DISSERTATION

Unequal Error Protection for Compressed Video over Noisy Channels

by

Arash Vosoughi

Doctor of Philosophy in Electrical Engineering
(Signal and Image Processing)

University of California, San Diego, 2015

Professor Pamela C. Cosman, Chair

The huge amount of data embodied in a video signal is by far the biggest burden on existing wireless communication systems. Adopting an efficient video transmission strategy is thus crucial in order to deliver video data at the lowest bit rate and the highest quality possible. Unequal error protection (UEP) is a powerful tool in this regard, whose ultimate goal is to wisely provide a stronger protection for the more important data, and a weaker protection for the less important data carried by a video signal. The use of efficient video delivery techniques becomes more important when 3D video content is transmitted over a wireless channel, since it contains twice as much data as 2D video. In this dissertation, we consider the

UEP problem for transmission of 3D video over wireless channels. The proposed UEP techniques entail relatively high computational complexity which lend themselves to be more suitable for video-on-demand delivery, where the time-consuming computations are done offline at the transmitter/encoder side.

To adopt UEP for 3D video, we consider a general problem of joint source-channel coding (JSCC). Solving the JSCC problem yields the optimum amount of 3D video compression as well as the optimum FEC (forward error correction) code rates exploited for UEP. We first need to estimate the perceived quality of the reconstructed video at the receiver. The lack of a good objective metric for 3D video makes adopting UEP a more challenging and problematic task compared to 2D video. Fortunately, for 3D video, some quality thresholds are derived in the literature based on the PSNR (peak-signal-to-noise-ratio) metric through experimental tests. These thresholds allow us to formulate the JSCC optimization problem using the PSNR in a straightforward but different way from the typical counterpart optimization problems in the literature. More precisely, we put the constraints of the optimization problem on the quality of the reconstructed 3D video and set our goal to minimize the total bit rate. We adopt the multiview coding (MVC) extension of the H.264/AVC. We also propose a scalable variant of MVC and formulate and solve the JSCC optimization problem for it. We show that significant gains are obtained if the proposed UEP scheme is combined with asymmetric coding.

We also tackle the UEP problem for the video plus depth (V+D) format. We employ the SSIM (Structural SIMilarity) metric for designing UEP for V+D, since it has been shown that PSNR does not properly characterize the perceived quality of a 3D video represented in V+D format. Moreover, the synthesized right view always shows a huge PSNR loss (even in the absence of compression), which does not even allow us to use the asymmetric coding PSNR thresholds. This motivated us to adopt the classical JSCC problem formulation, where our goal is to maximize the quality of the reconstructed left and right views, given that there is a constraint on the sum of the number of source bits and the number of FEC bits. We show that UEP provides significant gains compared to equal error protection.

We also derive several interesting results; some of them are in accordance with what have already been published in the literature and some of them are not. We show that the reason for this inconsistency is that we are solving the UEP problem in a more general situation, which yields novel solutions.

Lastly, we focus on UEP for video broadcasting over wireless channels. Our goal here is to design a UEP-based video broadcasting system that well serves all the users within the service area of a base station. In a service area, there exist heterogeneous users with different display resolutions operating at different bit rates. Spatially scalable video is an excellent video compression format for this scenario, since it allows a user to decode that portion of the scalable bit stream that fits its operating bit rate as well as its display resolution. We tackle this problem for a MIMO (multi-input-multi-output) channel which enables us to exploit either spatial diversity or spatial multiplexing in a multipath fading channel to increase channel reliability or throughput, respectively. We employ spatial diversity techniques, in particular the Alamouti code, to encode the base layer. We also adopt spatial multiplexing techniques, in particular the V-BLAST, to encode the enhancement layer. By controlling the power allocation between the base layer and the enhancement layer, we can control the level of protection we provide to each of them. We also show that the adoption of scalable video in our system yields much higher gains compared to non-scalable video.

Chapter 1

Introduction

1.1 3D Video Compression

In Section 1.1.1, we describe 3D video compression using the multiview coding (MVC) extension of the H.264/AVC standard. We then introduce video plus depth (V+D) representation of 3D video in Section 1.1.2 and explain how V+D data is compressed.

1.1.1 Multiview Coding (MVC)

A stereo video is captured by a pair of cameras which mimic the way our eyes see the real objects around us. Video sequences captured by the left camera and the right camera are, respectively, referred to as left (primary) view and right (secondary) view. Figure 1.1 shows a few frames of a stereo video and a typical structure of predictive coding used to encode (compress) the frames. Arrows indicate which frames are used as reference frames for predictive encoding. For example, the first P-frame of the left view uses the I-frame for prediction. I-frames are coded without reference to any other pictures. Frames of the left view are coded with a typical hierarchical GOP (group of pictures) structure as provided by H.264/AVC. For encoding the P-frames and B-frames, *motion compensation* is done through temporal prediction and biprediction, respectively (blue arrows in Figure 1.1). Motion compensation is also utilized for predictive encoding of

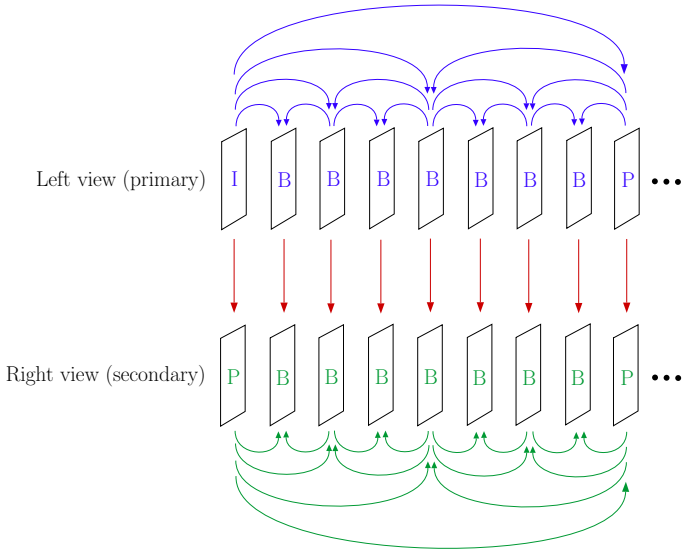


Figure 1.1: A typical MVC coding structure.

the right view frames (green arrows). MVC can achieve a better compression for right view by enabling *interview prediction* (red arrows). Interview prediction exploits similarities between the frames of the left view and right view to remove the redundancies of the right view.

Figure 1.2 depicts how an MVC encoder chooses a reference frame from either the primary view or the secondary view in order to compress a particular region in a given frame of the secondary view. The encoder starts by finding a region of pixels in a reference frame which can be a good predictor of the pixels of the region being coded (current region). The best match for the current region is found by searching in its spatial neighborhood in the reference frame, while minimizing a proximity measure such as the SAD (sum of absolute differences) or the SSD (sum of squared differences). The proximity measure is computed between the reference region and the current region. Once the best match is found, the encoder computes the difference between the current region and its reference region (the difference is referred to as residual), calculates the DCT (discrete cosine transform) of the residual, quantizes the DCT coefficients, and signals the quantized DCT coefficients (referred to as levels) to the decoder. Quantization of the DCT coefficients is the lossy part of video compression which makes retrieving the

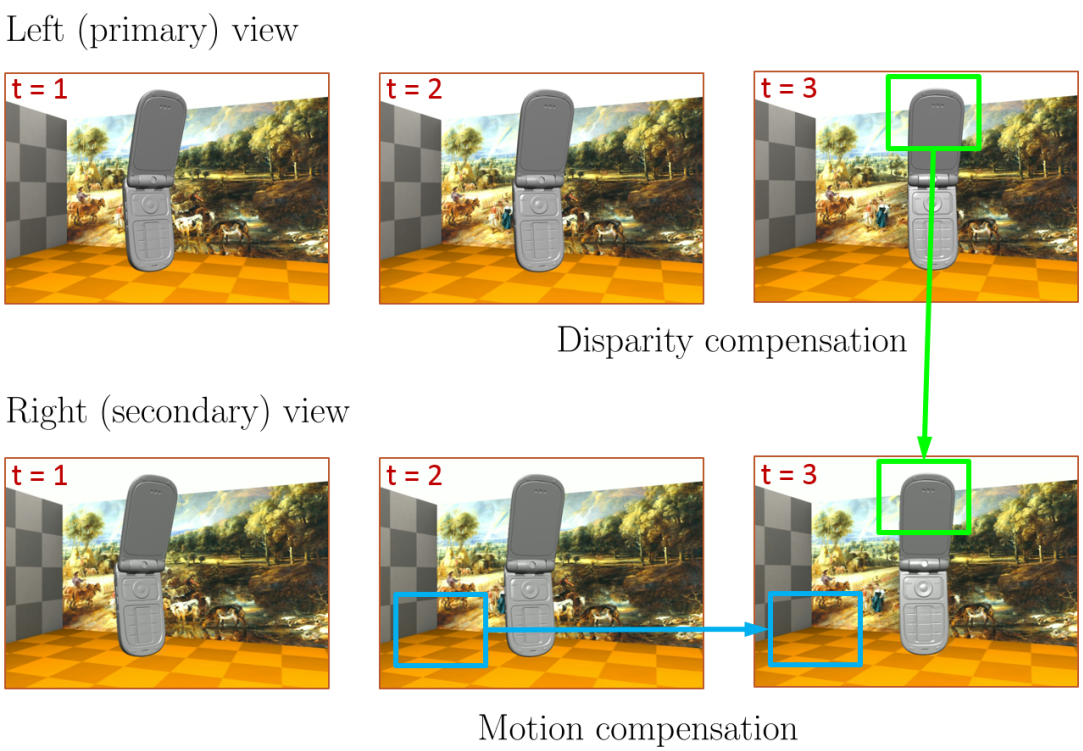


Figure 1.2: Illustration of motion compensation and disparity compensation in MVC.

exact original uncompressed video impossible. Now, suppose the current frame being coded is a frame of the secondary view at time $t = 3$. To encode the region bounded by the blue rectangle[1], the MVC encoder selects the frame at time $t = 2$ of the secondary view as the reference frame, since in that frame the encoder can find a region of pixels that can be used as an excellent predictor of the region being coded.

To encode the region bounded by the green rectangle in the secondary view frame at $t = 3$, the MVC encoder chooses to use the primary view frame at $t = 3$ as reference, since it can find a better match there. If both the current region and its reference region belong to the same view, the above procedure is called motion compensation. On the other hand, if the reference region belongs to another view

[1]The H.264/AVC standard and its MVC extension only support predictive coding for rectangular regions at the level of macroblocks (a macroblock is a region of size 16 pixels $\times$ 16 pixels) or smaller regions such as $16 \times 8, 8 \times 8, 4 \times 4$, etc. The large rectangular regions considered here are just for illustration of the concepts of motion compensation and disparity compensation.

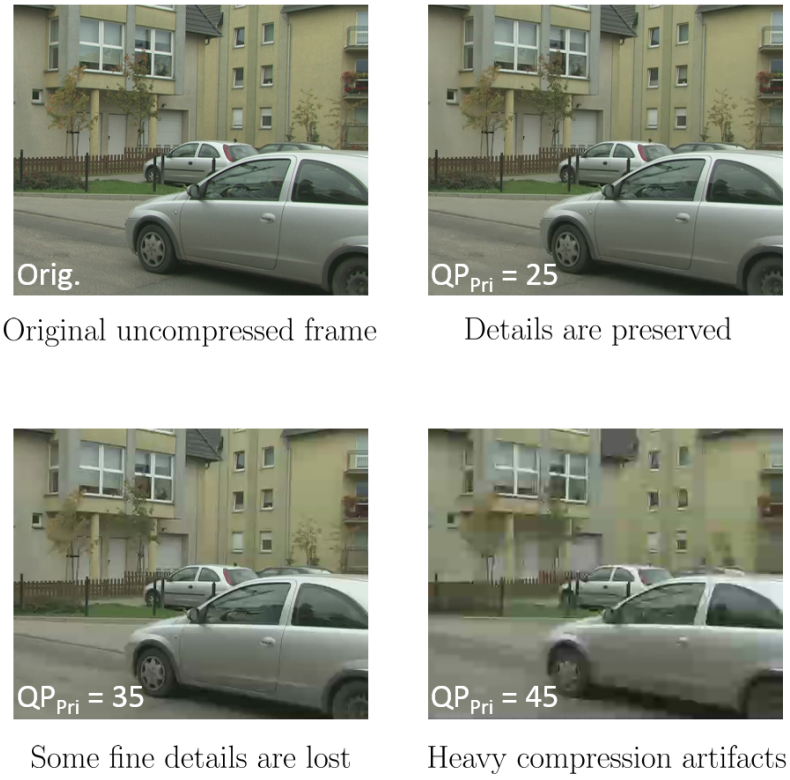


Figure 1.3: Effect of quantization parameter on the quality of the reconstructed (decompressed) video.

(called the reference view), the procedure is called disparity compensation. At the decoder, the reference region is first decoded to obtain the prediction. The inverse quantization and inverse DCT are then applied to the coded residuals. The result is then added to the prediction to reconstruct the coded region.

Quantization step size is one important factor that controls the amount of compression. The quantization step size is signaled by a particular syntax element referred to as the quantization parameter (QP). According to the H.264/AVC standard, a quantization parameter can only take a discrete value from 0 to 51, where QP = 0 corresponds to the lossless coding mode of compression. A higher quantization parameter corresponds to a larger quantization step size and heavier compression. Figure 1.3 compares the visual quality of a reconstructed (decoded) frame which has been compressed using different quantization parameters.

1.1.2 Video Plus Depth (V+D)

V+D is an efficient representation of 3D video, where a stereo pair is rendered at the decoder from a color video signal and a per-pixel depth map [1], [2] (see Figure 1.4). Depth map is a grayscale image whose pixel intensities are related to distances from cameras. In Figure 1.4, brighter regions are located closer to the cameras. Depth maps are built using *depth estimation* techniques, some of them presented in [3], [4], [5], [6]. V+D format has become popular due to several useful characteristics it possesses compared to the conventional 3D video representation exploited in MVC. The first benefit of V+D is that it allows one to synthesize novel views from a scene which are not captured by the cameras. The goal of any view synthesis algorithm is to generate a realistic right view with minimal visual artifacts using the left view and the corresponding depth map [7]. View synthesis is typically done by linear warping of the left image based on the local depth information. A view synthesis method should also incorporate ways to deal with occluded regions. Occlusion happens when some element of a scene is only captured by one camera and is unknown to the other camera. A simple approach to deal with occlusions is to mirror the intensities in the scanline adjacent to the hole. More complicated hole-filling techniques are also proposed in the literature. Investigating the details of view synthesis is out of the scope of this work and we refer to [7] for further details on this subject.

The other important feature of V+D is that a same stereo video content can be more compressed if it is represented by V+D format rather than by conventional formats exploited in simulcast coding and MVC. The V+D compression consists of compressing a left view video and a per-pixel depth map instead of compressing a left view and a right view in an MVC compression scenario. The depth map is a grayscale video which by itself has less data to be compressed compared to a color right view. In addition, the depth map typically consists of several regions each of which having very correlated grayscale values. The dramatic correlation between the pixels of the depth map can be compressed very efficiently using the state-of-the-art video compression tools such as H.264/AVC.

Although V+D brings higher compression ratios compared to MVC, it suf-

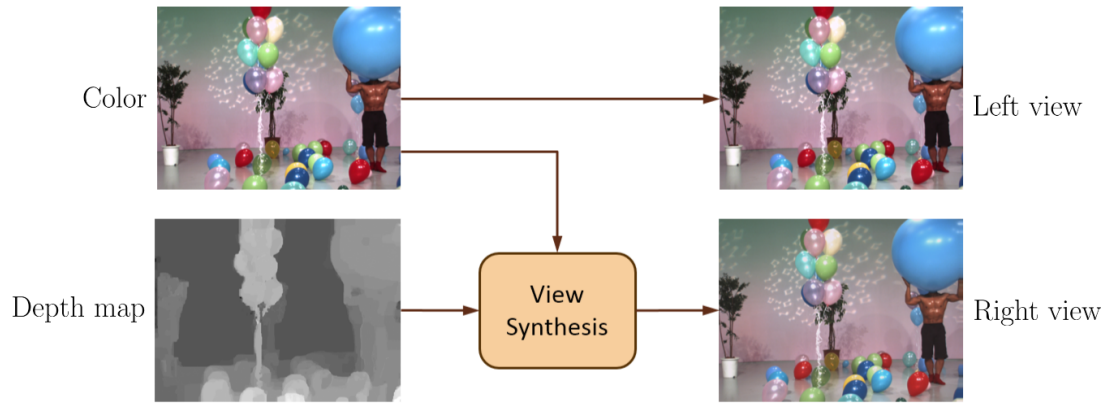


Figure 1.4: V+D representation of 3D video. Left view and color are the same. Right view is synthesized by a view synthesis algorithm which uses color and depth map.

fers from particular reconstruction errors in the synthesized right view, which are not present in a right view reconstructed by MVC. These errors may be due to having noisy/inaccurate depth maps, color mismatch between the left and right views, and occlusion. These errors are not well captured by the conventional objective metrics such as PSNR, in that these metrics do not respond to these errors as like the human visual system (HVS) does. For example, in case of an occlusion, the conventional quality metrics may indicate a huge quality loss (since pixel interpolation is used to fill the occluded region and the interpolated pixels may have dramatic different values compared to the original pixel values), while that occlusion may not be perceived by the HVS.

It has been shown in the literature that some objective quality metrics, such as SSIM (Structural SIMilarity) and VQM (video quality metric), are more suitable to measure the quality of a reconstructed video coded using the V+D format [8].

1.2 Scalable Video Coding

A scalable video bit stream consists of at least two substreams of which one is called base layer (BL) and the others are referred to as enhancement layers (ELs). The base layer is always coded independently from the other layers, while

an enhancement layer is coded using the information contained in the base layer or the lower enhancement layers. A set of prediction techniques, referred to as *interlayer prediction*, are used to capture and remove the redundancy between an enhancement layer and the lower layers. This implies that an efficient interlayer prediction is vital to increase the compression efficiency of a scalable video coder. Scalable video coding (SVC) is an extension of the H.264/AVC standard that enables transmission and decoding of a scalable video bit stream. It supports three types of scalability, namely, temporal scalability, quality scalability, and spatial scalability. For all these three types of scalability, the base layer is coded at a low bit rate such that decoding the base layer alone yields a video with a basic quality, while decoding the enhancement layers progressively enhances the quality of the reconstructed video. More details on scalable video coding are given in Sections 1.2.2, 1.2.3, and 1.2.1.

Several applications of SVC have been proposed in the literature. In this dissertation, we mainly exploit two features of SVC in designing UEP for video transmission. First, SVC lends itself to be very beneficial in a video communication scenario where there exist heterogeneous users operating at different bit rates. For example, a user with a low resolution display typically works at a low bitrate (which may be due to either having a smaller number of receive antennas or less available processing power), while a user with a high resolution display usually works at a higher bit rate (which may be due to either having a larger number of receive antennas or more available processing power). In that scenario, we encode the video content using the desired number of enhancement layers and send the same scalable video bit stream to all the users.

Consider a case where there are two types of users: a user with a low resolution display operating at a low bit rate, and a user with a high resolution display operating at a high bit rate. In that case, the spatially scalable bit stream has a base layer and only one enhancement layer. A user working at a low bit rate only decodes a low bit rate base layer, which results in reconstruction of a low resolution version of the original content that fits the small display screen of that user. On the other hand, a user working at a high bit rate can decode both

the low bit rate base layer and a high bit rate enhancement layer, which results in reconstruction a full-resolution video that fits the big display screen of that user. The use of SVC in the mentioned scenario clearly obviates the need to encode the same video content at two different bit rates and send them to all the users.

The second important benefit of SVC is that it can provide a graceful quality degradation in a video communication system if it is cleverly combined with UEP. This can be done by unequally protecting the layers according to the contribution they make in enhancing the quality of the reconstructed video. For a two-layer bit stream, this is done by providing a stronger protection for the base layer and a weaker protection for the enhancement layer. The reason for this choice is that, receiving and decoding the base layer is crucial in achieving a basic acceptable reconstruction quality and thus it should receive strong protection, while the enhancement layer can receive less protection.

1.2.1 Temporal Scalability

For temporal scalability, decoding the base layer of the scalable bit stream yields a low frame rate video with the same original spatial resolution. Video sequences at higher frame rates are reconstructed by first decoding the base layer and then by progressively decoding the enhancement layers.

1.2.2 Spatial Scalability

For spatial scalability, decoding the base layer alone produces a spatially reduced resolution video at the decoder. Any other higher resolution reconstructions are obtained by first decoding the base layer and then by progressively decoding the enhancement layers until the desired resolution is reconstructed (see Fig. 1.5).

Fig. 1.6 shows a block diagram of a two-layer spatially scalable encoder and decoder. In this figure, f represents an uncoded frame with the original spatial resolution $2N \times 2M$, where $2N$ and $2M$ denote the number of pixels in the vertical and horizontal directions, respectively. The uncoded BL frame g with spatial resolution $N \times M$ is obtained by lowpass filtering of f (not shown) and then

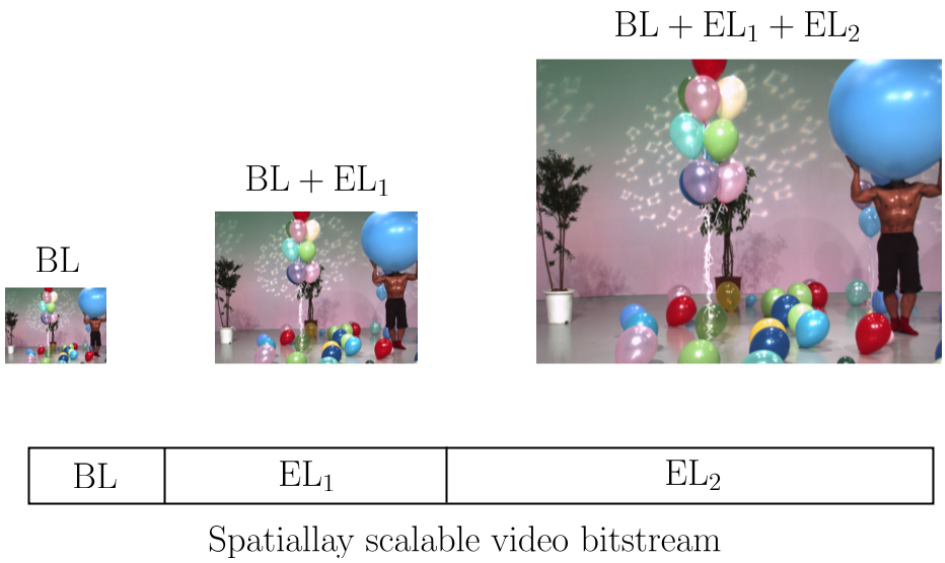


Figure 1.5: Spatially scalable video. Higher resolution videos are reconstructed at the decoder by progressively decoding more number of layers.

downsampling the result in both directions. The BL bitstream is then obtained by encoding g. To obtain the EL bit stream, the BL frame $\hat{g}$ is first reconstructed by decoding the BL bit stream. The result is then upsampled and interpolated in both directions which yields a frame with resolution $2N \times 2M$, which serves as a predictor of f. The difference between f and the prediction is then coded to build the EL bit stream. At the decoder, the half-resolution video is obtained by just extracting and then decoding the BL substream which yields $\hat{g}$. The full-resolution frame $\hat{f}$ is obtained by decoding the EL substream, upsampling the $\hat{g}$, and then adding them together. We refer to [9] for more details on how the SVC bit stream is built based on the interlayer prediction.

1.2.3 Quality Scalability

For quality scalability, decoding the base layer yields a low quality (SNR) video with the same original spatial resolution. Other reconstructions with higher qualities are obtained by first decoding the base layer and then by progressively decoding the enhancement layers until the desired quality is obtained (see Fig. 1.7).

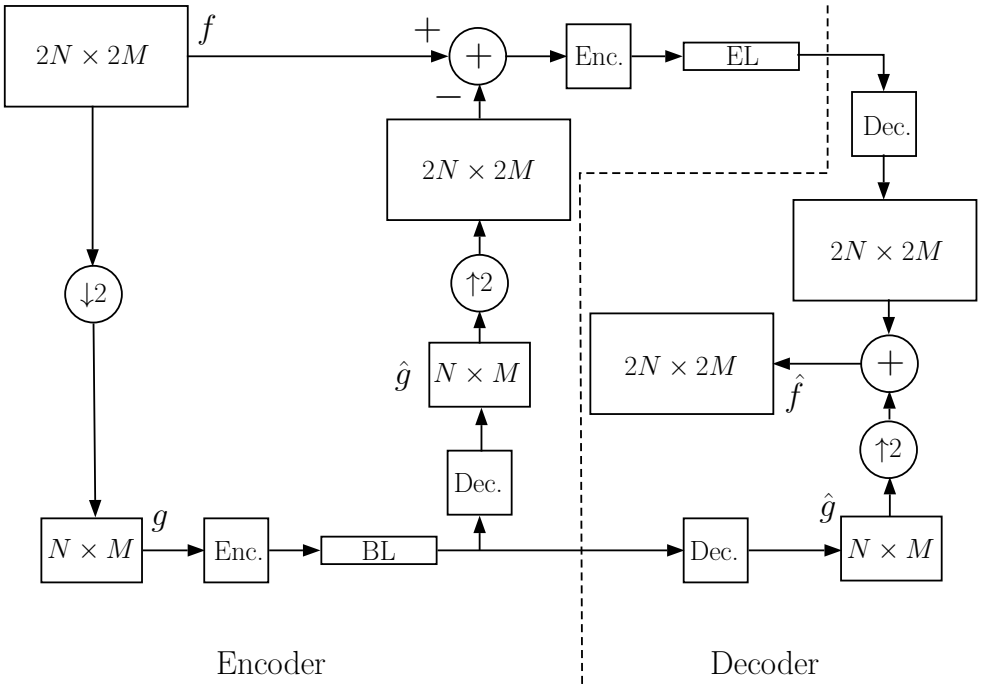


Figure 1.6: Block diagram of spatial scalability with two layers.

Fig. 1.8 shows a block diagram of a two-layer quality scalable encoder and decoder. Here, f represents an uncoded frame with the original spatial resolution $2N \times 2M$. The BL frame is obtained by encoding the original frame at a low bit rate (low quality), while the original resolution is preserved. To obtain the EL bit stream, the BL frame f' is first reconstructed by decoding the BL bit stream, which is used as a predictor of f. The difference between f and the prediction f' is then coded to build the EL bit stream. At the decoder, the low-quality video is obtained by extracting and decoding the BL substream which yields f'. The high-quality frame $\hat{f}$ is obtained by decoding the EL substream and adding the result to f'.

1.2.4 Scalability for 3D Video

Although the SVC extension of H.264/AVC supports all the three types of scalability, the MVC extension only supports temporal scalability. Results such as those presented in [10] and [11] show that temporal scalability in either just one

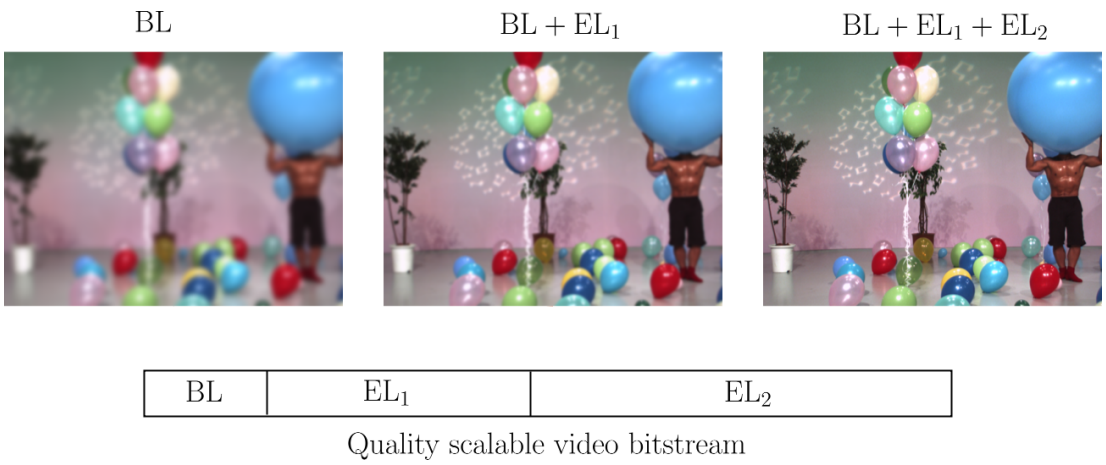


Figure 1.7: Quality scalable video. Higher quality videos are reconstructed at the decoder by progressively decoding more number of layers.

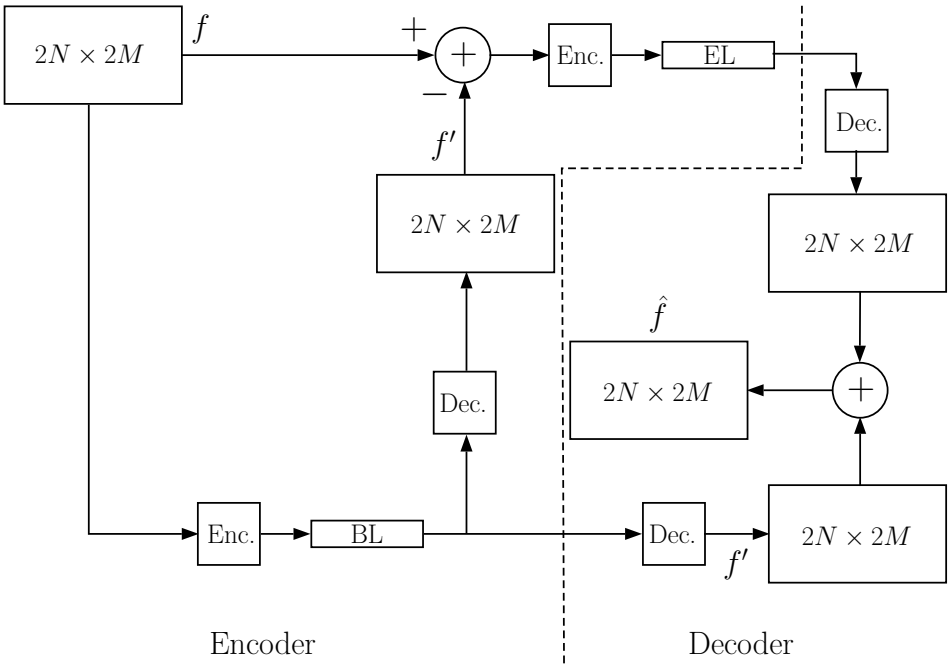


Figure 1.8: Block diagram of quality scalability with two layers.

or both views gives good results for low motion video, but for medium to high motion video, it may be unacceptable due to visible jumping effects. Although there is no standard-compliant spatial or quality scalable MVC bit stream, several non-standard variants have been proposed [12], [13], [14], [15].

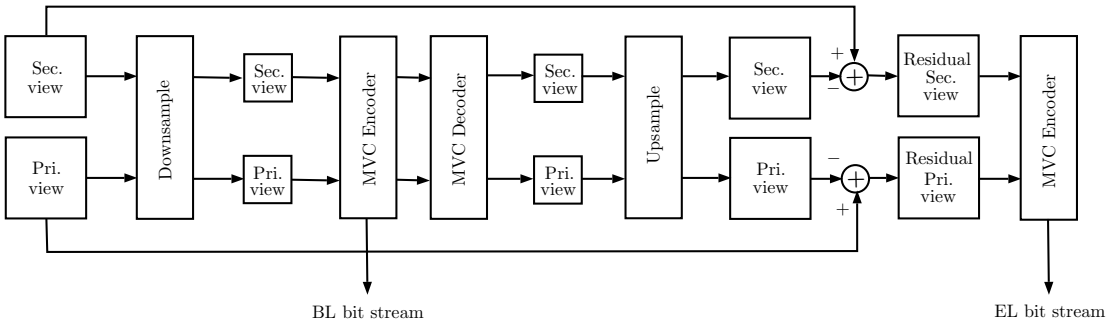


Figure 1.9: Proposed spatially scalable MVC. One MVC bit stream is generated for the base layer and another MVC bit stream is generated for the enhancement layer.

Other works proposed for 3D scalable coding attempt to define the best mode of scalability for 3D video. Early subjective tests with MPEG-2 in [16] and [17] show that spatial scalability is preferred over quality scalability. The reason is that in overall stereoscopic perception, especially for low bit rates, blocking artifacts produced by quality scalability implementations are more disturbing than the blurring effect produced by spatial scalability implementations. However, newer results in [18], [19], [20], indicate that the perceived quality depends on the 3D display and also that MPEG-2 may cause different artifacts than H.264/AVC on coded video. According to these results, users prefer quality scalability for polarized projection displays and spatial scalability for autostereoscopic parallax barrier displays. Results in [20] also show that, if the primary view is encoded at sufficiently high quality and the secondary view is encoded at low quality, users prefer spatial scalability over quality scalability.

In this dissertation, we adopt spatial scalability for MVC. Since spatial scalability is not supported by the standard, we propose a spatially scalable variant of MVC. Figure 1.9 shown a block diagram of the proposed spatially scalable scheme. The primary view and secondary view frames of a GOP are each lowpass filtered (not shown) and downsampled by a factor of 2 in both directions. These are encoded with MVC and constitute the base layer MVC bit stream. The enhancement layer bit stream is generated through upsampling, interpolation (not shown), and computing the residual views. These residual views are also encoded by MVC.

1.3 Human Visual System Considerations

In this section, we first describe *binocular suppression* theory. We then explain how binocular suppression can be exploited for *asymmetric coding* of 3D video in order to reduce the bit rate of the compressed 3D video. We then study the objective quality metrics that are usually used for video quality assessment.

1.3.1 Binocular Suppression

The binocular suppression theory [21], [22], [23], [24], [25], [26] says that the HVS is insensitive to errors which occur in one view only (see Figure 1.10). This result, determined experimentally, can be explained by the ability of the HVS to compensate for missing information. Because the visual cortex does not always receive perfect information from both eyes, it must infer some information given what is provided. That can mean suppressing errors which occur in a single view, while obtaining the necessary information from the other. Binocular suppression theory has given rise to asymmetric video coding, in which one view is coded with higher quality than the other.

1.3.2 Asymmetric Coding

Asymmetric coding refers to adopting different coding approaches for encoding the left and the right images of a pair of stereo images. Examples of asymmetric coding are adopting different QPs, different resolutions, or different frame rates (or a combination of them) to encode the left and the right images. Following binocular suppression theory, asymmetric coding may provide similar perceived 3D quality with a significant decrease in bit rate. Several papers propose asymmetric coding schemes [23], [27], [28], [29], [30], where one of the views is significantly more coarsely quantized than the other, or is coded with a reduced spatial resolution, generating blurring at the upsampling procedure. In [31], subjective experiments showed that in the asymmetric coding case, where one view is coded at very high quality (40dB) and the other view is coded at any level down to a threshold value of approximately 33dB, the resulting stereo video is indistin-

(a)

(b)

Figure 1.10: Binocular suppression. First row is a stereo pair with left image at high quality (40dB) and right image at low quality (33dB). Second row is a stereo pair with both left and right images at high quality (40dB). Both pairs are perceived the same to the HVS if they are displayed on a stereo TV.

guishable from the symmetric high quality case of both views coded at 40dB. It was found that when both views are coded above their corresponding thresholds, asymmetric coding is preferable to symmetric coding at the same total bit rate, whereas when one or both views are coded below its threshold, symmetric coding is generally preferable. These thresholds are described and employed in Section 2.4. References [32] [33] [34] introduced scalability and asymmetry into MVC.

1.3.3 Video Quality Metrics

To design a UEP method for compressed video, we need to estimate the perceived quality of reconstructed video at the receiver. Several quality metrics have been used for this purpose including the PSNR, SSIM, and VQM. Although these metrics work relatively well for quality assessment of 2D video, developing a

quality metric for 3D video is a more challenging task, since such a metric needs to incorporate the complex perceptual attributes of 3D such as depth, overall image quality, presence, naturalness, and visual comfort. 3D video quality assessment is still an open challenge and there are no objective metrics which are widely recognized as reliable predictors of human 3D quality perception [8]. The 2D video quality metrics mentioned above are also usually adopted for 3D video quality assessment [8], [35], [36], [37], [38]. In the following, we describe PSNR and SSIM which are widely used in the literature and we also use them in this dissertation.

PSNR is the most common objective metric that is used to evaluate the quality of a reconstructed 2D video. It is computed by

$$\mathrm{PSNR} = 10 \log_{10} \left(\frac{255^2}{\mathrm{MSE}} \right), \tag{1.1}$$

where MSE (mean squared error) is calculated from

$$\mathrm{MSE} = \frac{1}{T \times W \times H} \sum_{t=1}^{T} \sum_{y=1}^{W} \sum_{x=1}^{H} \big(f(x,y,t) - \hat{f}(x,y,t) \big)^2. \tag{1.2}$$

In (1.2), x and y represent the coordinates of a pixel, t is the time index of a frame, f represents the original video signal, $\hat{f}$ denotes the reconstructed video, T denotes the number of frames, W is the frame width, and H is the frame height. For MVC and simulcast coding, it is very common to use a weighted average MSE of left view and right view, and compute the PSNR as

$$\mathrm{PSNR} = 10 \log_{10} \left(\frac{\alpha \mathrm{MSE}_L + (1-\alpha) \mathrm{MSE}_R}{2} \right), \tag{1.3}$$

where α is typically set to $\frac{1}{2}$. In (1.3), MSE_X denotes the MSE that is computed over the frames of view $X \in \{L, R\}$.

It has been shown that PSNR is not able to acceptably model the view synthesis errors [8] in V+D format. Subjective experiments indicate that SSIM

is better correlated to perceived quality than PSNR [8]. The SSIM between two images f and g is obtained from

$$\mathrm{SSIM}(f,g) = \frac{(2\mu_f\mu_g + C_1)(2\sigma_{fg} + C_2)}{(\mu_f^2 + \mu_g^2 + C_1)(\sigma_f^2 + \sigma_g^2 + C_2)}, \tag{1.4}$$

where μ_X and σ_X^2, respectively, denote the mean and the variance of the pixels of image X, and σ_{XY} represents the cross-correlation between images X and Y. C_1 and C_2 are two constants introduced to avoid instability when either $(\mu_f^2 + \mu_g^2)$ or $(\sigma_f^2 + \sigma_g^2)$ is very close to zero. More precisely, we set $C_1 = (K_1 L)^2$ and $C_2 = (K_2 L)^2$, where $K_1 \ll 1$ and $K_2 \ll 1$ are two small constants, and L is the dynamic range of pixel values (256 for 8-bit grayscale images). The SSIM is usually applied locally rather than globally. The local statistics μ_X, σ_X, and σ_{XY} are usually computed within a local square window, which moves pixel-by-pixel over the entire image. Local SSIM is computed for each window and then averaged over the entire image. SSIM varies between -1 and 1, where larger values correspond to lower distortion. The SSIM between two GOPs x and y is calculated as the average of SSIMs between the corresponding frames of x and y, and we denote it by $\mathrm{SSIM}(x, y)$. Let N denote the number of frames in a GOP, then

$$\mathrm{SSIM}(x,y) = \frac{1}{N}\sum_{i=1}^{N} \mathrm{SSIM}(x_i, y_i), \tag{1.5}$$

where x_i and y_i represent the ith frame of GOP x and GOP y, respectively. For 3D video, SSIM is first obtained for each view and then the average is taken over the views.

1.4 Error Concealment

In an error-prone channel, the coding structure of H.264/AVC and its SVC and MVC extensions can create significant reconstruction errors that may propagate throughout the entire GOP of a reconstructed video. We discuss these errors and show that how *error concealment* (EC) attempts to reduce the adverse effects

of them.

1.4.1 EC for 2D Non-Scalable Video

The H.264/AVC standard allows dividing a frame into several groups of consecutive macroblocks, where each group of macroblocks is referred to as a *slice*. Each slice of a frame is encoded independently from the other slices of that frame, and, thus, each slice is decoded independently from the others. Although the slicing strategy reduces the coding efficiency of a video encoder, it is tremendously beneficial in improving the error resiliency of a compressed video transmitted over an error-prone channel. The fact that the slices of a frame can be decoded independently implies that if a few of the slices in a frame are lost, the decoder is still able to decode the slices that are received correctly and so some portions of the frame can be reconstructed correctly. In an attempt to further improve the quality of the decoded video, some techniques referred to as error concealment are usually applied at the decoder to reduce the effect of errors due to losing the slices in transmission. A common error concealment is called *frame copying* in which any pixel of a lost slice is recovered by copying from the co-located pixel in the nearest (in terms of display order) previous reference frame that is already decoded and available at the decoder buffer.

Although the frame copying error concealment can alleviate the adverse effects of errors due to slice losses, it cannot successfully conceal all kinds of errors that may happen in transmission. For example, it readily fails in concealing an error that happen by losing a slice with a high motion content. In that case, copying from a reference frame may lead to a big error, and even worse, the generated error can propagate to the other frames. The propagating error stems from the temporal predictive coding and motion compensation.

Figure 1.13 shows how the errors are propagated throughout a set of consecutive frames of a decoded video sequence. It also makes a qualitative comparison between the errors generated by losses of different slices with different motion content. Here, a car is moving from right to left. Figure 1.13(a) shows a reconstructed video with no losses, and Figure 1.13(b) shows a reconstructed video in which two

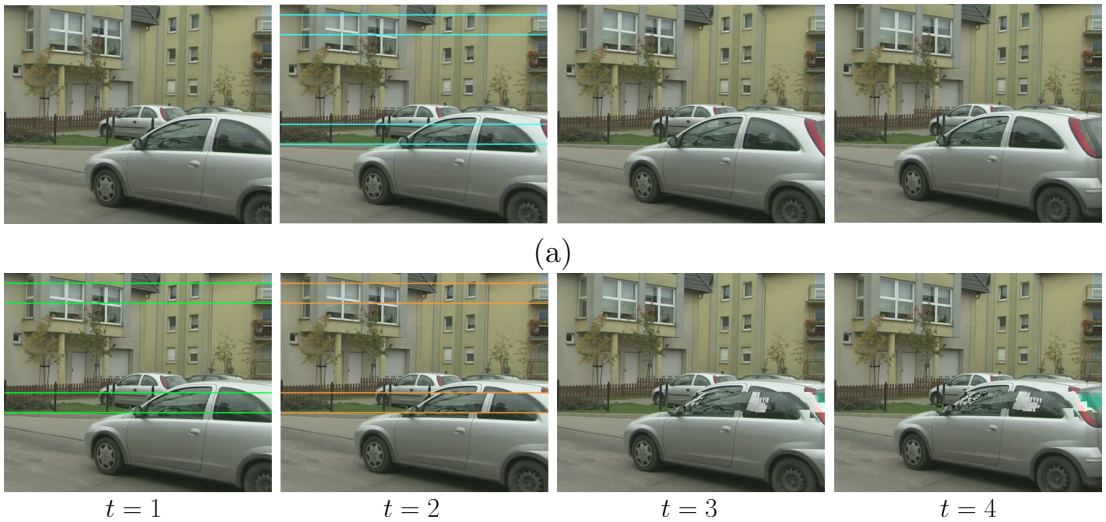


Figure 1.11: Channel distortion and error propagation for non-scalable 2D video. (a) Reconstructed video when no losses happen. (b) Reconstructed video when two slices are lost at $t = 2$. Slices marked by orange lines at $t = 2$ are lost and they are concealed by copying from the regions marked by green lines at $t = 1$. Rectangular regions marked in blue at $t = 2$ depict the reconstructed pixels in the absence of losses and error concealment.

slices at $t = 2$ (marked by lines in orange color) are lost and concealed. The top slice belongs to a static background, while the bottom slice includes a region with some motion. The two rectangular regions marked by green lines in frame $t = 1$ are the co-located pixels from which the frame copying is performed for the two slices lost at $t = 2$. Reconstructed pixels in the absence of losses and error concealment are marked by the blue lines at $t = 2$ in Figure 1.13(a). Comparing Figure 1.13(a) and Figure 1.13(b) shows that the error due to losing the top slice is perfectly concealed by frame copying while, on the other hand, the bottom slice loss is not successfully concealed and its error propagates to the other frames.

1.4.2 EC for 3D Non-Scalable Video

The MVC extension of H.264/AVC also supports frame slicing for the sake of error resiliency [39]. We use frame copying for losses both in the primary view and the secondary view. Errors due to losses in secondary view do not propagate

to the primary view, however, errors in primary view may propagate to the frames of secondary view because of the use of interview prediction in MVC. We show this in Figure 1.12. The situation is similar to the one illustrated in Figure 1.11, with a difference that now the two lost slices belong to the primary view. We see that again the error due to losing the top slice is perfectly concealed by frame copying. However, frame copying is not successful in concealing the bottom slice loss and the error propagates to the frames of the primary view with $t > 2$, and to the frames of the secondary view with $t \leq 2$.

1.4.3 EC for Scalable Video

For base layer of 2D scalable video, we can use any error concealment we use for non-scalable 2D video, since the BL should be decodable by a compliant non-scalable decoder. For enhancement layers, however, we show that we can adopt a better strategy which uses the BL for concealing the EL errors.

We continue our discussion by examining an example. In our example, we encode the video sequence 'Foreman' into a base layer and an enhancement layer. Figure 1.13 shows the decoded frames from time $t = 25$ to $t = 32$, where only the base layer is decoded and frame copying is used for error concealment. Here, each frame is divided into 9 slices, where all the slices from slice number 3 to slice number 7 of frame $t = 25$ are lost and concealed by frame copying. We see that frame copying is not able to perform well due to presence of some motion in the region covered by the slices lost in transmission, and the error generated in reference frame $t = 25$ propagates to the other following frames.

Although frame copying may not perform well in concealing the errors occurred in the BL, we can adopt a better error concealment strategy for losses that occur in the EL. Instead of copying from a previously decoded full-resolution reference frame, for an enhancement layer slice lost from a frame at time t_0, we perform copying from the upsampled pixels of the decoded base layer frame at time t_0. We note that the decoded base layer is not in full-resolution and we need to do upsampling before copying. Since copying is done from a base layer reference frame with the same time index, the motion is preserved at the final full-resolution

Figure 1.12: Channel distortion and error propagation for 3D video encoded using MVC. Reconstructed (a) primary view and (b) secondary view when no losses happen. Reconstructed (c) primary view and (d) secondary view when two slices are lost in primary view at $t = 2$. Errors due to losses in primary view propagate to the frames of both the primary view and the secondary view.

decoded output, which significantly reduces the error propagation due to losses in the enhancement layer. In Figures 1.14 and 1.15, we compare the effect of losses in the base layer and enhancement layer, respectively, where in both figures both the base layer and enhancement layers are decoded. In Figure 1.14, all the base layer slices from slice number 3 to slice number 7 of the frame at time $t = 25$ are

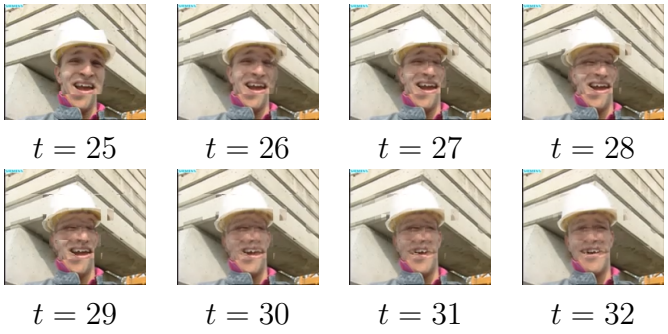


Figure 1.13: Frames 25 to 35 of video sequence 'Foreman' where only BL is decoded and slices 3 to 7 of frame 25 are lost. Frame copy error concealment is applied at the decoder.

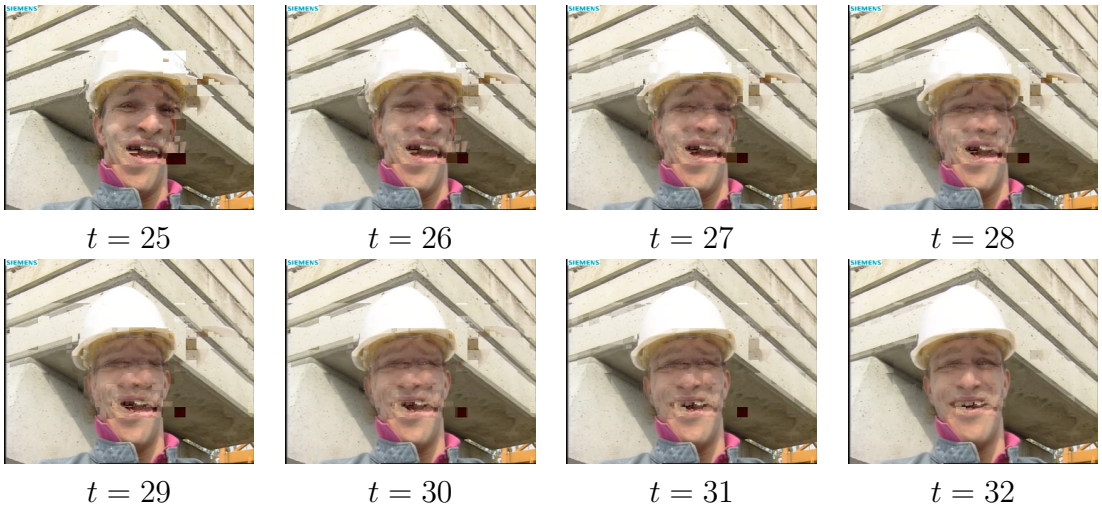


Figure 1.14: Frames 25 to 35 of video sequence 'Foreman' where both BL and EL are decoded and slices 3 to 7 of frame 25 of BL are lost. Frame copy error concealment is applied at the decoder.

lost. We see that the decoded video severely suffers from the losses that occur in the base layer. In Figure 1.15, all the slices of the enhancement layer from slice number 6 to slice number 14 of the frame at time $t = 25$ are lost. It is clearly seen that the errors in the enhancement layer are successfully concealed. Figure 1.16 illustrates the decoded video without any losses. By comparing the results of Figures 1.15 and 1.16, we notice some blurring at the locations of losses in Figure 1.15 that is attributed to the upsampling of the base layer.

For scalable 3D video, we only consider spatial scalability for MVC in this

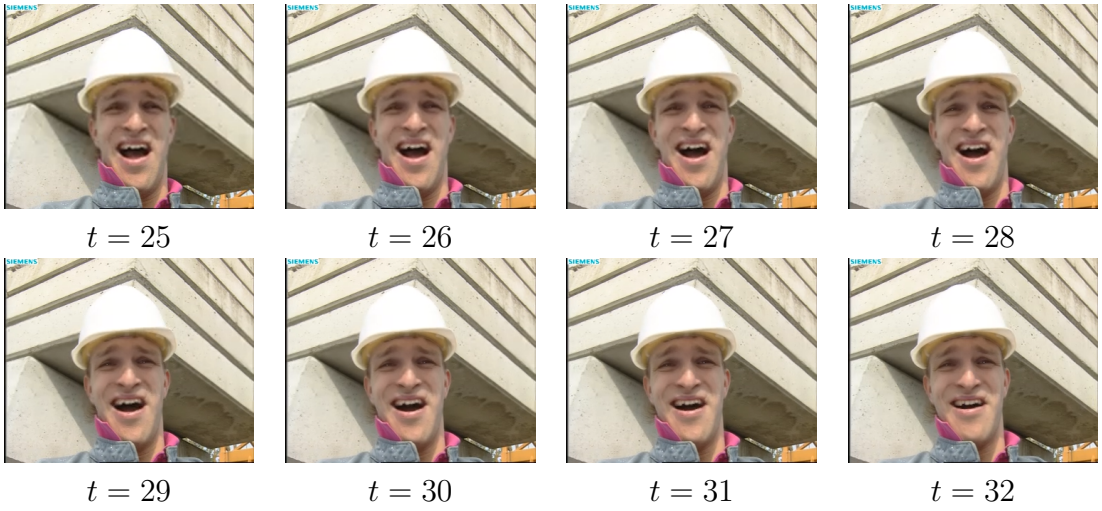


Figure 1.15: Frames 25 to 35 of video sequence 'Foreman' where both BL and EL are decoded and slices 6 to 14 of frame 25 of EL are lost. Slices 3 to 7 of frame 25 of BL are upsampled and used for error concealment.

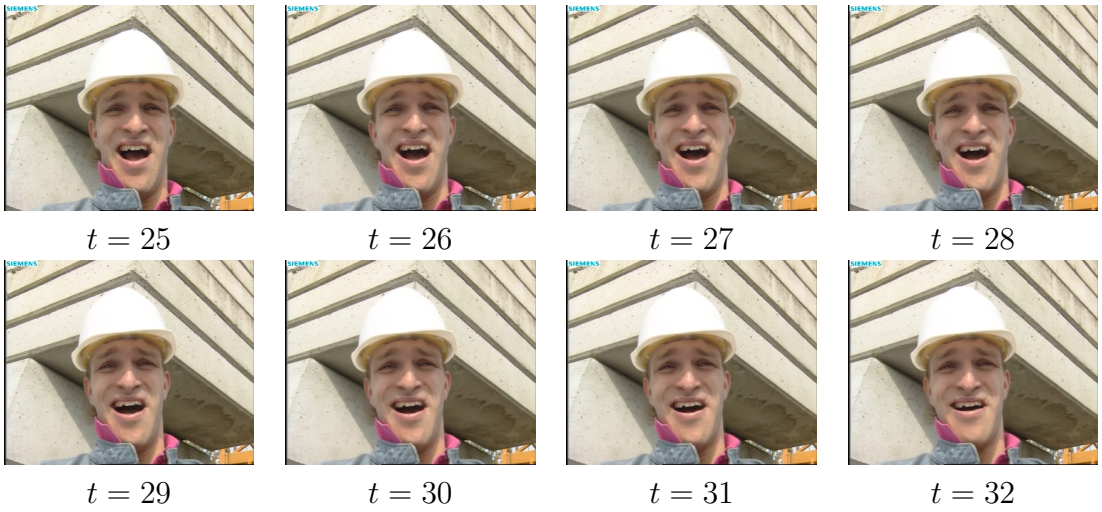


Figure 1.16: Frames 25 to 35 of video sequence 'Foreman' where both BL and EL are decoded and no packets are lost.

dissertation. The EC we adopted for MVC is similar to the one introduced in Section 1.4.3. We apply frame copying for BL losses, and perform BL upsampling from the same view for EL losses.

1.5 UEP for Video

Video signals carry a huge amount of data which is certainly a big burden on wireless communication systems. It thus requires adoption of efficient video transmission strategies which attempt to reach a good trade-off between the data bit rate and the quality of the reconstructed video at the receiver. Unequal error protection is a powerful tool in this regard, which aims to wisely provide a stronger protection for the more important data, and a weaker protection for the less important data carried by video signal. A stronger (weaker) protection is typically provided by using a larger (lower) amount of forward error correction (FEC) bits. UEP tries to efficiently allocate different amounts of FEC to different parts of a compressed video according to the contribution they make in enhancing the quality of the reconstructed video.

1.5.1 Prior Work for 2D Video

Compressed video bit stream needs to be protected by FEC before transmission over a noisy channel. Redundancy bits introduced by FEC reduce the number of bits available for source coding, and hence limit the accuracy of source coding. Increasing the source coding accuracy forces us to use weaker FEC codes which increases the distortion due to channel errors. This implies that both source coding and channel coding should be applied in a clever manner. The trade-off between source coding accuracy and channel error protection in error-prone channels is a joint source channel coding (JSCC) problem and is a well-studied area for single view video sequences. A comprehensive review on this topic is presented in [40]. The work in [41] applies JSCC specifically for video transmission over additive white Gaussian noise (AWGN) channels using rate compatible punctured convolutional (RCPC) codes [42]. The optimal point found by JSCC varies over different AWGN channel signal-to-noise-ratios (SNRs). JSCC for single view video sequences is also studied for several wireless environments in [43].

One UEP approach for video transmission is to employ different FEC code rates for each video packet according to its importance. The importance of each

packet can be determined by the estimation of the distortion in the reconstructed video produced by each packet loss separately. The distortion of the reconstructed video should be reduced when compared to the reconstructed video protected with equal error protection (EEP), where all video packets are coded with the same channel code rate. The distortion estimation can rely on traditional quality metrics, such as MSE, or on metrics based on human visual perception, such as the packet loss visibility model presented in [44].

1.5.2 Prior Work for 3D Video

3D video signal carries significantly more amount of data compared to 2D video. This justifies the importance of adoption of JSCC for 3D video delivery over noisy channels.

1.5.2.1 MVC

The performance and transmission of MVC bit streams in error-prone channels have been studied in [45], [46], [47], [48], [49]. Some of the works on multiview streaming optimization, as in [45], propose end-to-end distortion models taking into account estimated packet loss probabilities for multiview video packets, but do not include channel error protection schemes. The work in [46] has the same characteristics, but includes a form of UEP by simply setting a smaller packet loss rate for the packets in the base view as well as the packets in the first 20 frames of the other views. Another work [47] that studied the transmission of multiview video sequences over error-prone channels considered UEP through a selective packet discard mechanism. Several error resilience techniques for multiview video sequences are described in [48] and [49].

A typical JSCC optimization approach is to fix a total rate of B bits and then determine the optimal division of B between source and FEC, where the objective function could be the average MSE to be minimized. An example of this type of optimization for 3D video can be found in [50], where a weighted average MSE of the left view and of the right view is used as the objective function to be minimized. Formulating the optimization in this way is problematic for 3D video

because although MSE is well-defined for each of the individual left view and right view, there is not yet any well-accepted way to quantify the quality of the combined 3D video [51],[52]. Minimizing the average MSE subject to a rate constraint would imply that left/right MSEs of $(\mathrm{MSE}_L, \mathrm{MSE}_R) = (\epsilon, 3\epsilon)$ and of $(\mathrm{MSE}_L, \mathrm{MSE}_R) = (2\epsilon, 2\epsilon)$ produce equivalent average MSEs, although the subjective visual quality might be very different. This issue motivated us to formulate the JSCC problem in a different way as we describe in the following.

Our alternative approach to the optimization is to fix the distortion or PSNR of each view to some level, and then attempt to minimize the number of bits required to achieve it. Putting the distortion in the constraint, rather than in the objective function, allows one to choose two separate constraints (one for each view). Therefore, the particular goal of our JSCC scheme is to minimize the total bit rate, composed of source and error-correction bits, while both reconstructed views achieve predetermined PSNR values. Fortunately, some quality thresholds are derived for the reconstructed stereo video based on the PSNR metric through experimental tests [19]. These quality thresholds, which are derived according to binocular suppression theory, enable us to formulate the JSCC problem using the PSNR as we discussed above. Details of the proposed JSCC scheme is given in Chapter 2.

1.5.2.2 V+D

We are interested in the delivery of V+D data over mobile devices [1]. The quality of a received 3D video in V+D format is affected by both the source coding accuracy of the color video and the depth map, and the amount of redundancy introduced by FEC to protect them over the channel. Therefore, for a fixed bit rate, it is crucial to design a clever method to divide the bits between the source and the FEC such that the quality is maximized at the receiver. Our goal here is to maximize the average quality of the reconstructed left view and right view, given that there is a constraint on the sum of the number of source bits and the number of FEC bits.

In [53], two different protection levels are considered for V+D, and the

authors concluded that color should be protected more strongly than depth. Following this conclusion, a UEP method is proposed in [54] for V+D data over WiMAX communication channels based on unequal power allocation. In [55], it is concluded that depth can be compressed more compared to color, and downsampling the depth by a factor of two is recommended to increase coding efficiency, although the effect of a channel is not investigated.

We consider both downsampled and full-resolution depth scenarios. Both the color and depth are encoded by an H.264/AVC encoder [56] and then protected by FEC using UEP such that each individual packet is protected according to its importance. The importance of packets is based on the SSIM index [57]. The JSCC yields the optimum color and depth quantization parameters as well as the UEP code rates that jointly maximize the quality at the receiver. Turbo codes [58] are used for FEC, and simulation results are given for flat Rayleigh fading channels. The performances of different scenarios are compared, and UEP performance is compared to EEP. We show that the adopted UEP provides significant gains compared to the EEP. We also derive several interesting results. Some of these results are in accordance with what have already been published in the literature and some of them are not. We show that the reason of this inconsistency is that we are solving the UEP problem in a more general situation which yields novel solutions. For example, we show that although the depth map should be compressed more compared to color (which is in agreement to prior works in the literature), it should be protected more compared to color (which is in contrast to prior works in the literature). We also propose to use a depth map that is downsampled by a factor of 4 instead of 2, which the latter is proposed in the literature.

1.6 UEP for MIMO Video Broadcasting

In Chapter 4 of this dissertation, we consider UEP for video broadcasting over wireless channels. Our goal here is to design a UEP-based video broadcasting system that benefits all types of users within a service area of a transmitter in an optimum way. We assume that heterogeneous users with different display

resolutions and different operating data rates are present in the service area. We tackle this problem for a MIMO (multi-input-multi-output) channel. We propose to use scalable video coding for video compression. For MIMO communication, we propose to use spatial diversity techniques for BL transmission and spatial multiplexing techniques for EL transmission. We superpose the BL and EL in a way that a stronger protection is provided for the BL compared to the EL.

1.6.1 MIMO Communications

MIMO refers to a collection of signal processing techniques that have been developed to enhance the performance of wireless communication systems using multiple antennas at the receiver, the transmitter, or both [59]. MIMO techniques can be used either to combat multipath fading to improve the link reliability, or to exploit multipath fading to increase the data rate. Improving the channel reliability is made possible by creating *spatial diversity*, and increasing the data rate is provided by *spatial multiplexing*. Spatial diversity techniques extract a diversity gain to combat fading, and they thus improve the link reliability. A popular example of these techniques are orthogonal space-time block codes (OSTBCs) [60], [61], which achieve full diversity with a simple linear receiver. Spatial multiplexing techniques use a layered approach to increase the channel data rate [62], [63]. One popular example is the vertical Bell Laboratories layered space-time (V-BLAST) architecture, where independent data streams are transmitted over different antennas to increase the data rate. Although spatial multiplexing increases the data rate, it cannot usually achieve the full spatial diversity. However, some space-time block codes have been studied in the literature which can yield the benefits of both spatial diversity and multiplexing [64], [65], [66], [67], [68], [69].

1.6.2 Hierarchal Constellations for UEP

In mobile video broadcasting systems such as Digital Video Broadcasting (DVB), there exist various types of user equipment which usually have different number of receive antennas. For example, a tiny mobile phone may have a single

antenna due to its limited hardware space. On the other hand, a tablet or a notebook computer usually has more than one antenna, since it has a larger hardware space and a higher computational capability.

Theoretical investigation of efficient communication from a single source to multiple receivers established the fundamental idea that optimal broadcast transmission could be achieved by a hierarchical transmission scheme [70], [71]. In addition, it has been shown [72]–[73] that a practical UEP method is achieved by using a constellation of nonuniformly spaced signal points that is referred to as *hierarchical modulation*. In this constellation, the more important bits of a symbol have a larger minimum Euclidean distance compared to the less important bits of that symbol. Hierarchical constellations have been intensively studied for digital broadcasting systems [72][73], and the Digital Video Broadcasting-Terrestrial (DVB-T) standard [74] has incorporated hierarchical QAM (quadrature amplitude modulation) for scalable video transmission.

1.6.3 SVC-MIMO Video Broadcasting

Higher diversity and/or spectral efficiency gains can be achieved if a MIMO system employs a larger number of antennas. Let N_t and N_r denote the number of transmit and receive antennas, respectively. By using spatial diversity techniques, we can achieve a diversity gain of up to $N_t \times N_r$ [59]. On the other hand, we can achieve a spectral efficiency gain of up to $\min(N_t, N_r)$ by using spatial multiplexing techniques [59]. This indicates that the users with one receive antenna cannot achieve a spectral efficiency larger than 1, since for them we have $\min(N_t, 1) = 1$. This implies that the base station needs to broadcast video using spatial diversity rather than spatial multiplexing, so that the data is decodable by all types of users. We also note that, for high data rates, spatial multiplexing techniques such as V-BLAST outperform spatial diversity techniques such as OSTBC [75][76]. That means forcing the base station to adopt only spatial diversity techniques for all types of users may lead to a significant performance loss, particularly for users which are able to achieve spectral efficiencies larger than one by exploiting more than one receive antennas.

In Chapter 4 of this dissertation, we propose an efficient broadcasting strategy that combines space-time coding and scalable video coding to tackle this problem. We suppose that the number of antennas of a device and the size of its screen is mainly limited by the hardware space affordable by the device. This means that a user with more receive antennas can have a higher-resolution screen. With these suppositions, we consider a MIMO video broadcasting system where the base station possesses two transmit antennas, and two different types of user devices reside in the service area: i) a *big user* with two receive antennas and a high-resolution screen, and ii) a *small user* with a single receive antenna and a low-resolution screen.

We propose an efficient video broadcasting scheme which combines spatial diversity and spatial multiplexing techniques with spatially scalable video coding. The base layer of the scalable video is encoded using spatial diversity techniques, such as the Alamouti code, while the enhancement layer is encoded using spatial multiplexing techniques, such as the V-BLAST.

1.7 Thesis Outline

In Chapter 2, we tackle the UEP and the JSCC problem of a 3D stereo video transmitted over a noisy wireless channel. We first model the end-to-end distortion of a stereo video compressed by the MVC extension of the H.264/AVC. The model captures the distortion due to the compression of both the primary and secondary views as well as the distortion due to the channel losses in both views. We show that the model is accurate enough in estimating the end-to-end distortion of both views. We then use these estimates to predict the average end-to-end distortion over a lossy channel. We formulate the UEP/JSCC optimization problem based on two distortion thresholds; one for the reconstructed left view and another for the reconstructed right view. Finally, we validate the performance of the designed UEP/JSCC scheme for several video sequences by performing many channel realizations and measuring the average distortion values. The measured distortion values show that the quality thresholds determined in the UEP design

are indeed satisfied on the average over many channel realizations.

In Chapter 3, we consider the UEP/JSCC problem of a 3D video represented by the V+D format. We investigate a downsampled depth map as well as a full-resolution one. In doing that, we first derive a measure to quantify the quality of the left view and the synthesized right view based on assigning individual quality scores to individual slices of both the compressed color and depth. We compute the scores using the SSIM metric. We use these scores to quantify the average end-to-end distortion of a V+D content compressed by the AVC/H.264 and transmitted over a noisy wireless channel. We show that the proposed UEP/JSCC scheme performs much better compared to the EEP/JSCC scheme. We also derive some interesting results and discuss them.

In Chapter 4, we consider UEP for video broadcasting over wireless channels. Our goal is to design a video broadcasting system that well serves all types of users within the service area of a base station. Users have different display resolutions as well as different operating data rates. We consider this problem for a MIMO channel. We use spatial scalable video coding for video compression. We propose to use spatial diversity techniques for encoding the base layer and spatial multiplexing techniques for encoding the enhancement layer. We superpose the BL and EL bit streams in a way that the BL is protected stronger than the EL. We show that our proposed UEP scheme significantly outperforms the baseline schemes in terms of the PSNR.

Chapter 2

Unequal Error Protection for Multiview Coding

In this chapter, we propose a UEP method for 3D video transmission over noisy channels. To compress 3D video, we use the MVC extension of H.264/AVC. We also propose a type of spatially scalable MVC. To design UEP for 3D video, we consider a general problem of joint source-channel coding. Solving the JSCC problem yields the optimum quantization parameters of an MVC-encoded stereo video as well as the optimum FEC code rates used for UEP. Our goal is to minimize the total number of bits, which is the sum of the number of source bits and the number of FEC bits, under the constraints that the quality of the left and right views must each be greater than predetermined PSNR thresholds at the receiver. We first consider symmetric coding, for which the quality thresholds are equal. Following binocular suppression theory, we also consider asymmetric coding, for which the quality thresholds are unequal. The optimization problem is solved using both EEP and the proposed UEP scheme. An estimate of the expected end-to-end distortion of the two views is formulated for a packetized MVC bit stream over a noisy channel. The UEP algorithm uses these estimates for packet rate allocation. Results for various scenarios, including non-scalable/scalable MVC, symmetric/asymmetric coding, and UEP/EEP, are provided for both AWGN and flat Rayleigh fading channels. The UEP bit savings compared to EEP are given, and the performances of different scenarios are compared for several stereo video

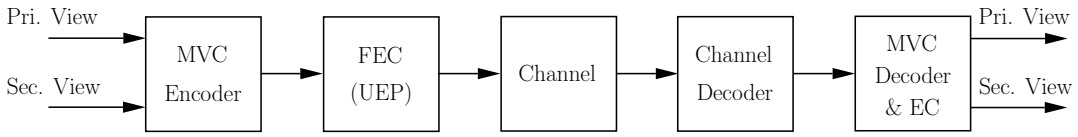


Figure 2.1: Block diagram of a 3D video communication system employing the proposed JSCC scheme.

sequences.

This chapter is organized as follows: Section 2.1 shows how the proposed JSCC scheme is adopted in a video transmission system. In Section 2.2 and Section 2.3, we derive estimates to predict the end-to-end distortion in a 3D video transmission scenario. Sections 2.4 and 2.5, respectively, describe the JSCC problem formulation and how we solve it using integer programming. Section 2.6 gives simulation results for various coding schemes for both AWGN and flat Rayleigh fading channels, and Section 2.7 concludes this chapter.

2.1 Overview of the System Design

The system block diagram is shown in Figure 2.1. The primary and secondary views are jointly compressed by an MVC encoder. The amounts of compression of the primary view and the secondary view are controlled by two separate quantization parameters q_1 and q_2, respectively. The MVC bit stream is protected by adding FEC bits and then transmitted over a channel. UEP provides different levels of protection at the packet level through allocating different FEC rates. At the receiver, channel decoding is applied to detect the erroneous bits. Our assumption is that if even one bit of a transmitted packet is erroneous, the whole packet is marked as undecodable and not decoded by the video decoder. The primary and secondary views are then decompressed by an MVC decoder, where error concealment is done for the lost packets. We use frame copying error concealment (see Sections 1.4.1 and 1.4.2).

2.2 Modeling the End-to-End Distortion

A UEP design for compressed video requires accurate estimation of the end-to-end distortion in a video communication system. An end-to-end distortion measure should incorporate both the distortion due to source compression and the distortion due to packet losses. In this section, we model the end-to-end distortion of a GOP of an MVC-encoded 3D video that is sent over a noisy channel. We show by simulation that the model we adopt is accurate in predicting the actual end-to-end distortion measured for different packet loss ratios. The model is then used in Section 2.3 to derive an estimate of the expected end-to-end distortion. Later on in Section 2.4, we use the estimates for UEP.

We first consider the non-scalable MVC case. Let $f^{(v)}$ represent the original pixel values of view v of a GOP, and $\hat{f}^{(v)}$ be the reconstructed values at the encoder, where $v = 1$ represents the primary view and $v = 2$ represents the secondary view. We denote the pixel values of view v of the GOP at the decoder as $\tilde{f}^{(v)}$. The distortion of view v is the sum of distortions of all its pixels. It is common in the literature to approximate the source quantization distortion and the channel distortion as being uncorrelated [50], [46], [77], [78]. With this approximation, the expected distortion of view v of the GOP can be written as

$$\begin{aligned} D^{(v)} &= E\{\text{cmse}(f^{(v)}, \hat{f}^{(v)})\} + E\{\text{cmse}(\hat{f}^{(v)}, \tilde{f}^{(v)})\} \\ &= D^{(v)}_{Src} + D^{(v)}_{Loss}, \end{aligned} \tag{2.1}$$

where $\text{cmse}(x^{(v)}, y^{(v)})$ is the cumulative mean squared error (CMSE) between the pixels of view v of GOP x and view v of GOP y, $D^{(v)}_{Src}$ represents the source distortion over the entire view v of the GOP, and $D^{(v)}_{Loss}$ denotes the distortion introduced by the channel due to packet losses. In (2.1), we model the end-to-end distortion such that the source distortion and channel distortion are additive, where the precise value of $D^{(v)}_{Src}$ is computed at the encoder. To compute $D^{(v)}_{Loss}$ for a set of lost packets, we assume that the error signals due to individual losses from either the primary or the secondary view are separate throughout the GOP. For example, if a slice is lost at the top of a frame and another slice is lost at the bottom

of that frame (or another frame), the error signals due to the loss of these packets are generally independent. Using this assumption, the CMSE contributions of the individual packets to the CMSE of either the primary view or the secondary view of the GOP are additive. To compute the channel distortion $D_{Loss}^{(v)}$ for a set of lost packets, the model adds up the CMSE values due to the individual lost packets. This additivity assumption is also used for example in [50], [79], [80], [81], [82], [83], [84]. A CMSE value represents the precise error propagated throughout a view of the GOP, and we assume that it has already been computed offline at the encoder for each packet of the GOP.

Now, we investigate the accuracy of the model in estimating the end-to-end distortion of a GOP. In our experiment, packets of an encoded 3D video are randomly dropped with different packet loss ratios (PLRs), where 1000 random realizations are done for each PLR. For non-scalable MVC, for error concealment we implemented linear interpolation for lost I slices, and slice copy for lost P slices, such that a lost P slice is concealed from its reference frame in the same view. Figs. 2.2(a)-(d) show histograms of $\Delta\text{PSNR} \stackrel{\triangle}{=} \text{PSNR}_m - \text{PSNR}_{est}$ for PLRs 0.5% and 2% for the sequence 'Oldtimers', where PSNR_m is the actual PSNR measured at the receiver (that is computed between the original uncompressed video and the lossy decoded video) and PSNR_{est} is computed by the model. The model computes the end-to-end distortion using (2.1), that accounts for both the source distortion and the channel distortion. We see that the model is accurate in estimating the end-to-end distortion if few packets of a GOP are lost in transmission. If the channel gets bad such that the number of losses after the channel decoder becomes large, the accuracy of the model decreases. However, our JSCC scheme allows us to add as many parity bits as needed to meet the quality constraints for a bad channel condition.

The model adopted for non-scalable MVC can also be used for estimating the end-to-end distortion of scalable MVC. That is, we assume that the source distortion and channel distortion are additive, and that the CMSE contributions of lost packets are additive. This can again be verified by realizing many channel realizations and different PLRs. For scalable MVC, error concealment was imple-

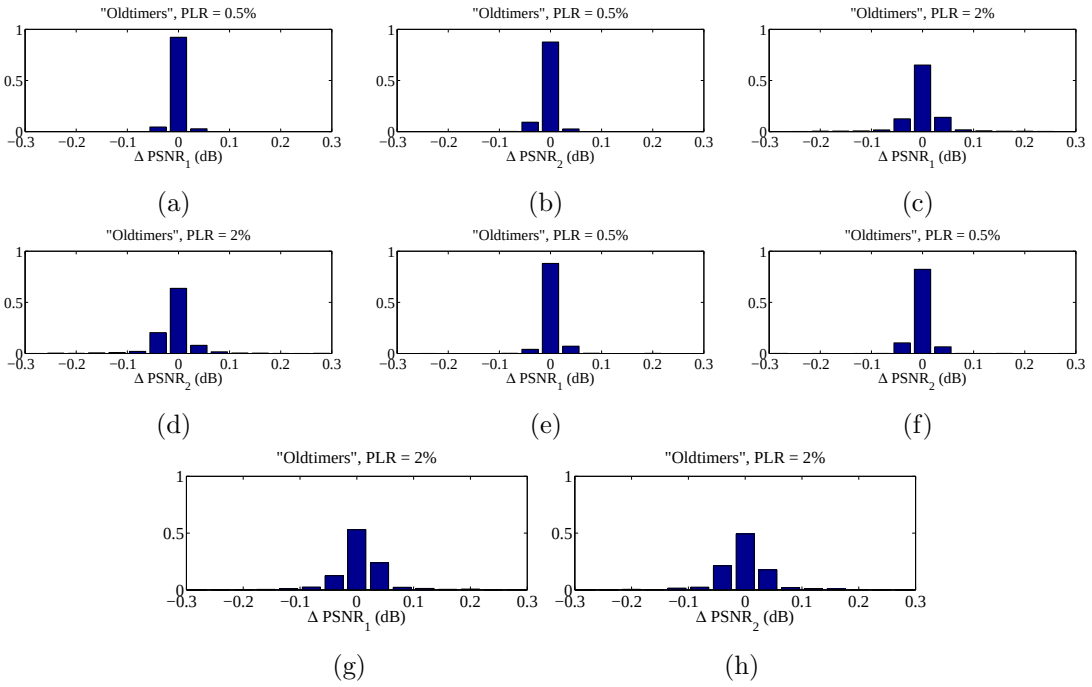


Figure 2.2: Histograms of error $\Delta\text{PSNR} \stackrel{\triangle}{=} \text{PSNR}_m - \text{PSNR}_{est}$ for packet loss ratios 0.5% and 2%, and video sequence 'Oldtimers'. (a), (b), (c), and (d) Non-scalable MVC, and (e), (f), (g), and (h) scalable MVC. ΔPSNR_1 and ΔPSNR_2 correspond to primary and secondary view respectively.

mented such that, when a BL packet is lost, frame copying is used for the BL, and EL information is preserved (linear interpolation is used for lost I slices of the BL); when an EL packet is lost, an upsampled version of the co-located slice of the BL is used for error concealment [85], and if two co-located BL and EL slices are lost simultaneously, frame copying is used for both. Figs. 2.2(e)-(h) show histograms of the errors for PLRs 0.5% and 2% for the sequence 'Oldtimers' for the scalable coder.

Table 2.1 shows the mean absolute value of ΔPSNR, which is defined as $\overline{|\Delta\text{PSNR}|} = \frac{1}{N}\sum_{i=1}^{N}|\Delta\text{PSNR}_i|$, where N is the number of realizations. The small $\overline{|\Delta\text{PSNR}|}$ values indicate that the model is accurate in estimating the measured PSNR values at the receiver.

We also investigate the accuracy of the additivity approximation for particular packet loss patterns where two packets are lost at the same time. We consider

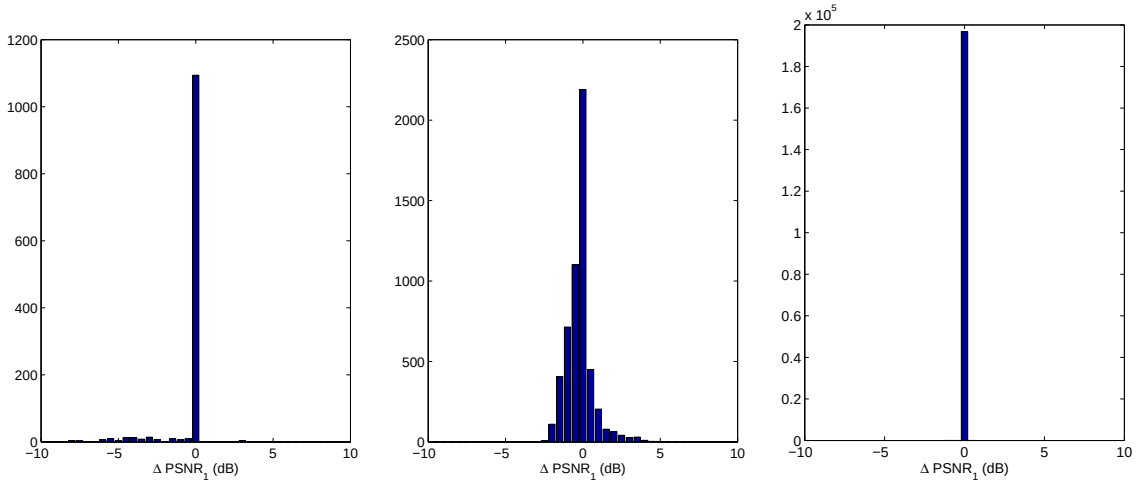
Table 2.1: Mean absolute value of $\Delta\text{PSNR} \stackrel{\triangle}{=} \text{PSNR}_m - \text{PSNR}_{est}$ in dB for packet loss ratios 0.5%, 1%, 2%, and 5%.

	PLR	non-scalable		scalable	
		$\overline{\lvert\Delta\text{PSNR}\rvert}_{\text{Pri}}$	$\overline{\lvert\Delta\text{PSNR}\rvert}_{\text{Sec}}$	$\overline{\lvert\Delta\text{PSNR}\rvert}_{\text{Pri}}$	$\overline{\lvert\Delta\text{PSNR}\rvert}_{\text{Sec}}$
Oldtimers	0.5%	0.007	0.01	0.009	0.013
	1%	0.016	0.017	0.017	0.023
	2%	0.028	0.029	0.034	0.047
	5%	0.056	0.067	0.059	0.092
Race	0.5%	0.017	0.018	0.029	0.030
	1%	0.036	0.045	0.079	0.073
	2%	0.089	0.105	0.142	0.150
	5%	0.226	0.278	0.237	0.239

the following packet loss patterns: (1) packets which are in adjacent rows within the same frame, (2) packets which are located in the same row in different frames (spaced apart from 1 to $N_F - 1$ frames, where N_F is the number of frames in a view of a GOP), and (3) all other possible combinations of two packets. Figure 2.3(a) shows a histogram of all possible adjacent combinations, which comprise 0.6% of all possible combinations, Figure 2.3(b) depicts a histogram of all combinations of two packets located in the same row but in different frames, which comprise 6% of all the possible combinations, and Figure 2.3(c) is a histogram of all the other combinations, which comprise 93.4% of all the possible combinations. We observe that the model is not very accurate for some combinations of packet loss patterns (1) and (2). On the other hand, the model is highly accurate for all other combinations. These observations show that the model is inaccurate only for adjacent losses and losses from the same row (which together comprise a small percentage of all possible combinations) because in such cases, the propagated errors may affect each other.

2.3 Expected End-to-End Distortion

In this section, we derive an estimate of the expected end-to-end distortion of a GOP of an MVC-encoded video sent over a noisy channel using the model



(a) (b) (c)

Figure 2.3: Histograms of error $\Delta\text{PSNR} \stackrel{\triangle}{=} \text{PSNR}_m - \text{PSNR}_{est}$ for two packets lost in a GOP (see text for description).

developed in Section 2.2. We first derive the estimate for non-scalable MVC and then consider scalable MVC.

2.3.1 Non-Scalable MVC

In the following, $d_{m,v}^{(v')}$ denotes the CMSE contribution of the mth packet of view $v \in \{1,2\}$ to the CMSE of view $v' \in \{1,2\}$, and $\tilde{f}_{m,v}^{(v')}$ represents the reconstructed view v' of the GOP at the decoder when the mth packet of view v is lost. Also, $p_i^{(v)}$ is the probability that the ith packet of view v is lost in transmission.

The CMSE contribution of the ith packet of the primary view to the CMSE of the primary view is zero if the packet is not lost, and is equal to $d_{i,1}^{(1)}(q_1)$ if the packet is lost, where q_1 is the quantization parameter used to encode the primary view of the GOP. Thus, following the model assumptions, the average end-to-end distortion of the primary view can be estimated as

$$D^{(1)}(q_1, r_1^{(1)}, \ldots, r_K^{(1)}, \Theta) = D_{Src}^{(1)}(q_1) + \sum_{i=1}^{K} p_i^{(1)} \Big(r_i^{(1)}, s_i^{(1)}(q_1), \Theta \Big) d_{i,1}^{(1)}(q_1), \qquad (2.2)$$

where K is the number of primary view packets in a GOP (which is the same as the number of secondary view packets in the GOP), and $d_{i,1}^{(1)}(q_1)$ is equal to $\text{cmse}(\hat{f}^{(1)}(q_1), \tilde{f}_{i,1}^{(1)}(q_1))$. Packet loss probability $p_i^{(1)}$ depends on the packet size $s_i^{(1)}$ in bits, the code rate $r_i^{(1)}$ by which the packet is protected, and Θ, which represents the channel characteristics; $\Theta = \text{SNR}$ for an AWGN channel and $\Theta = (\text{SNR}, T_c)$ for a flat Rayleigh fading channel, where T_c is the channel coherence time, defined in Section 2.6. It is assumed that the coded packets are lost independently. This assumption holds for an AWGN channel. For flat Rayleigh fading channels, independent losses within a GOP are obtained for an archival video by interleaving GOPs such that each interleaved block contains at most one packet from a particular GOP. In this work, the quantity $d_{i,1}^{(1)}(q_1)$ is computed at the encoder. In addition, since there is no closed-form expression to compute the packet loss probability $p(r, s, \Theta)$ for RCPT (rate compatible punctured turbo) codes, a lookup table is made by simulation, which yields $p(r, s, \Theta)$ for different ranges of packet sizes. The probability $p(r, s, \Theta)$ is obtained for packet sizes 250, 750, 1500, 2500, 3500, and 5000 in bits, and respectively used for all the packet sizes in the ranges $[0, 500)$, $[500, 1000)$, $[1000, 2000)$, $[2000, 3000)$, $[3000, 4000)$, and $[4000, \infty)$.

The distortion generated in the secondary view can be formulated in a similar manner. However, since the error due to a lost packet in the primary view propagates in both the primary and secondary views, for the secondary view, the CMSE contribution of lost primary packets should be considered as well as the CMSE contribution of lost secondary packets. Therefore, the average end-to-end distortion of the secondary view can be estimated as:

$$D^{(2)}(q_1, q_2, r_1^{(1)}, \ldots, r_K^{(1)}, r_1^{(2)}, \ldots, r_K^{(2)}, \Theta) = D_{Src}^{(2)}(q_1, q_2) + \sum_{i=1}^{K} p_i^{(1)}\Big(r_i^{(1)}, s_i^{(1)}(q_1), \Theta\Big) d_{i,1}^{(2)}(q_1, q_2) + \sum_{j=1}^{K} p_j^{(2)}\Big(r_j^{(2)}, s_j^{(2)}(q_1, q_2), \Theta\Big) d_{j,2}^{(2)}(q_1, q_2), \tag{2.3}$$

where $d_{i,1}^{(2)}(q_1, q_2) = \text{cmse}(\hat{f}^{(2)}(q_1, q_2), \tilde{f}_{i,1}^{(2)}(q_1, q_2))$, and $D_{j,2}^{(2)}(q_1, q_2) = \text{cmse}(\hat{f}^{(2)}(q_1, q_2), \tilde{f}_{j,2}^{(2)}(q_1, q_2))$. The quantities $d_{i,1}^{(2)}(q_1, q_2)$ and $d_{j,2}^{(2)}(q_1, q_2)$ are com-

puted at the encoder and used in the simulations. Computing the distortion values at the encoder side requires decoding the whole GOP for each slice of the GOP. The computational complexity of our algorithm at the encoder side is high and it can be done offline.

2.3.2 Scalable MVC

Similar to the case of non-scalable MVC, an estimate of the expected end-to-end distortion of the primary view for the scalable MVC case is given by

$$\begin{aligned} &D^{(1)}(q_{\mathrm{BL}_1}, r_1^{(\mathrm{BL}_1)}, \ldots, r_{\frac{K}{2}}^{(\mathrm{BL}_1)}, q_{\mathrm{EL}_1}, r_1^{(\mathrm{EL}_1)}, \ldots, r_K^{(\mathrm{EL}_1)}, \Theta) \\ &\quad = D_{Src}^{(1)}(q_{\mathrm{BL}_1}, q_{\mathrm{EL}_1}) + \sum_{i=1}^{\frac{K}{2}} p_i^{(\mathrm{BL}_1)}\Big(r_i^{(\mathrm{BL}_1)}, s_i^{(\mathrm{BL}_1)}(q_{\mathrm{BL}_1}), \Theta\Big) d_{i,\mathrm{BL}_1}^{(1)}(q_{\mathrm{BL}_1}, q_{\mathrm{EL}_1}) + \\ &\qquad\qquad \sum_{j=1}^{K} p_j^{(\mathrm{EL}_1)}\Big(r_j^{(\mathrm{EL}_1)}, s_j^{(\mathrm{EL}_1)}(q_{\mathrm{BL}_1}, q_{\mathrm{EL}_1}), \Theta\Big) d_{j,\mathrm{EL}_1}^{(1)}(q_{\mathrm{BL}_1}, q_{\mathrm{EL}_1}). \end{aligned} \tag{2.4}$$

For the secondary view, we have

$$\begin{aligned} &D^{(2)}(q_{\mathrm{BL}_1}, r_1^{(\mathrm{BL}_1)}, \ldots, r_{\frac{K}{2}}^{(\mathrm{BL}_1)}, q_{\mathrm{EL}_1}, r_1^{(\mathrm{EL}_1)}, \ldots, r_K^{(\mathrm{EL}_1)}, \\ &q_{\mathrm{BL}_2}, r_1^{(\mathrm{BL}_2)}, \ldots, r_{\frac{K}{2}}^{(\mathrm{BL}_2)}, q_{\mathrm{EL}_2}, r_1^{(\mathrm{EL}_2)}, \ldots, r_K^{(\mathrm{EL}_2)}, \Theta) = D_{Src}^{(2)}(q_{\mathrm{BL}_1}, q_{\mathrm{EL}_1}, q_{\mathrm{BL}_2}, q_{\mathrm{EL}_2}) + \\ &\qquad \sum_{i=1}^{\frac{K}{2}} p_i^{(\mathrm{BL}_1)}\Big(r_i^{(\mathrm{BL}_1)}, s_i^{(\mathrm{BL}_1)}(q_{\mathrm{BL}_1}), \Theta\Big) d_{i,\mathrm{BL}_1}^{(2)}(q_{\mathrm{BL}_1}, q_{\mathrm{EL}_1}, q_{\mathrm{BL}_2}, q_{\mathrm{EL}_2}) + \\ &\qquad \sum_{j=1}^{K} p_j^{(\mathrm{EL}_1)}\Big(r_j^{(\mathrm{EL}_1)}, s_j^{(\mathrm{EL}_1)}(q_{\mathrm{BL}_1}, q_{\mathrm{EL}_1}), \Theta\Big) d_{j,\mathrm{EL}_1}^{(2)}(q_{\mathrm{BL}_1}, q_{\mathrm{EL}_1}, q_{\mathrm{BL}_2}, q_{\mathrm{EL}_2}) + \\ &\qquad \sum_{m=1}^{\frac{K}{2}} p_m^{(\mathrm{BL}_2)}\Big(r_m^{(\mathrm{BL}_2)}, s_m^{(\mathrm{BL}_2)}(q_{\mathrm{BL}_1}, q_{\mathrm{BL}_2}), \Theta\Big) d_{m,\mathrm{BL}_2}^{(2)}(q_{\mathrm{BL}_1}, q_{\mathrm{EL}_1}, q_{\mathrm{BL}_2}, q_{\mathrm{EL}_2}) + \\ &\sum_{n=1}^{K} P_n^{(\mathrm{EL}_2)}\Big(r_n^{(\mathrm{EL}_2)}, s_n^{(\mathrm{EL}_2)}(q_{\mathrm{BL}_1}, q_{\mathrm{EL}_1}, q_{\mathrm{BL}_2}, q_{\mathrm{EL}_2}), \Theta\Big) d_{n,\mathrm{EL}_2}^{(2)}(q_{\mathrm{BL}_1}, q_{\mathrm{EL}_1}, q_{\mathrm{BL}_2}, q_{\mathrm{EL}_2}), \end{aligned} \tag{2.5}$$

where in (2.4) and (2.5) $d_{t,l}^{(v')}$ denotes the CMSE contribution of the tth packet of layer $l \in \{\mathrm{BL}_1, \mathrm{BL}_2, \mathrm{EL}_1, \mathrm{EL}_2\}$ to the CMSE of view $v' \in \{1, 2\}$, and $p_t^{(l)}$ is the probability that the tth packet of layer l is lost in transmission. In (2.4) and (2.5), $d_{i,l}^{(v)} = \mathrm{cmse}(\hat{f}^{(v)}, \tilde{f}_{i,l}^{(v)})$, where $\tilde{f}_{i,l}^{(v)}$ represents the reconstructed view v of a GOP at the decoder when the ith packet of layer l is lost, and $\hat{f}^{(v)}$ denotes the reconstructed view v of the GOP when there are no packet losses. The quantity $d_{i,l}^{(v)}$ is computed at the encoder.

2.4 JSCC Problem Formulation for MVC

The objective of our JSCC problem is to minimize the total number of bits, which is the sum of the numbers of source bits and FEC bits of both the primary and secondary views. For non-scalable MVC, the objective function is formulated as

$$\min_{\substack{q_1 \in \mathcal{Q}_1 \\ q_2 \in \mathcal{Q}_2 \\ r_1^{(1)}, \ldots, r_K^{(1)} \in \mathcal{R} \\ r_1^{(2)}, \ldots, r_K^{(2)} \in \mathcal{R}}} \left(\sum_{i=1}^{K} \frac{s_i^{(1)}(q_1)}{r_i^{(1)}} + \sum_{j=1}^{K} \frac{s_j^{(2)}(q_1, q_2)}{r_j^{(2)}} \right), \tag{2.6}$$

where $\mathcal{R} = \{R_1, R_2, \ldots, R_N\}$ is the set of available RCPT code rates, and $\mathcal{Q}_1$ and $\mathcal{Q}_2$ are the sets of quantization parameters. The optimization is done jointly over a primary view and the corresponding secondary view of a GOP. Quantization parameters q_1 and q_2 are applied for all macroblocks of the primary and secondary views of a GOP.

In minimizing the objective function (2.6), quality constraints must be satisfied. For the symmetric coding case, we require that the expected distortions of the primary and secondary views be less than or equal to a predetermined threshold T_1 at the receiver. However, for the asymmetric coding case, we require that the expected distortion of the primary view and the expected distortion of the secondary view be less than or equal to two different predetermined thresholds, T_1 and T_2. Using (2.2) and (2.3), this can be expressed for both the symmetric and

asymmetric coding cases as:

$$D^{(1)}(q_1, r_1^{(1)}, \ldots, r_K^{(1)}, \Theta) \leq T_1$$
$$D^{(2)}(q_1, q_2, r_1^{(1)}, \ldots, r_K^{(1)}, r_1^{(2)}, \ldots, r_K^{(2)}, \Theta) \leq T_2, \tag{2.7}$$

where for the symmetric coding case $T_1 = T_2 = 10^{(-\frac{40dB}{10})}$, and for the asymmetric coding case $T_1 = 10^{(-\frac{40dB}{10})}$ and $T_2 = 10^{(-\frac{33dB}{10})}$. The reason for choosing the particular PSNR values 40dB and 33dB is explained in Section 1.3.2.

The objective function of the JSCC problem for scalable MVC can be formulated as was done for the non-scalable case. For scalable MVC, we have

$$\min_{\substack{(q_{\mathrm{BL}_1}, q_{\mathrm{BL}_2}, q_{\mathrm{EL}_1}, q_{\mathrm{EL}_2}) \in \mathcal{Q}_S \\ r_i^{(\mathrm{BL}_1)},\ r_j^{(\mathrm{BL}_2)},\ r_m^{(\mathrm{EL}_1)},\ r_n^{(\mathrm{EL}_2)} \in \mathcal{R}}} \left(\sum_{i=1}^{\frac{K}{2}} \frac{s_i^{(\mathrm{BL}_1)}(q_{\mathrm{BL}_1})}{r_i^{(\mathrm{BL}_1)}} + \sum_{j=1}^{K} \frac{s_j^{(\mathrm{EL}_1)}(q_{\mathrm{BL}_1}, q_{\mathrm{EL}_1})}{r_j^{(\mathrm{EL}_1)}} + \right.$$
$$\left. \sum_{m=1}^{\frac{K}{2}} \frac{s_m^{(\mathrm{BL}_2)}(q_{\mathrm{BL}_1}, q_{\mathrm{BL}_2})}{r_m^{(\mathrm{BL}_2)}} + \sum_{n=1}^{K} \frac{s_n^{(\mathrm{EL}_2)}(q_{\mathrm{BL}_1}, q_{\mathrm{EL}_1}, q_{\mathrm{BL}_2}, q_{\mathrm{EL}_2})}{r_n^{(\mathrm{EL}_2)}} \right), \tag{2.8}$$

where $\mathcal{Q}_S$ is a set of 4-tuple quantization parameters.
Similar to the non-scalable MVC case, two constraints must be satisfied in minimizing the objective function (2.8). Using (2.4) and (2.5), these constraints are written as

$$D^{(1)}(q_{\mathrm{BL}_1}, r_1^{(\mathrm{BL}_1)}, \ldots, r_{\frac{K}{2}}^{(\mathrm{BL}_1)}, q_{\mathrm{EL}_1}, r_1^{(\mathrm{EL}_1)}, \ldots, r_K^{(\mathrm{EL}_1)}, \Theta) \leq T_1$$
$$D^{(2)}(q_{\mathrm{BL}_1}, r_1^{(\mathrm{BL}_1)}, \ldots, r_{\frac{K}{2}}^{(\mathrm{BL}_1)}, q_{\mathrm{EL}_1}, r_1^{(\mathrm{EL}_1)}, \ldots, r_K^{(\mathrm{EL}_1)},$$
$$q_{\mathrm{BL}_2}, r_1^{(\mathrm{BL}_2)}, \ldots, r_{\frac{K}{2}}^{(\mathrm{BL}_2)}, q_{\mathrm{EL}_2}, r_1^{(\mathrm{EL}_2)}, \ldots, r_K^{(\mathrm{EL}_2)}, \Theta) \leq T_2, \tag{2.9}$$

where for the symmetric coding case $T_1 = T_2 = 10^{(-\frac{40dB}{10})}$, and for the asymmetric coding case $T_1 = 10^{(-\frac{40dB}{10})}$ and $T_2 = 10^{(-\frac{33dB}{10})}$.

In the two optimization problems introduced in (2.6) and (2.7), and (2.8) and (2.9), different code rates are typically assigned to different packets. The code rate assigned to a particular packet depends on 1) the size of the source packet as determined by the quantization parameters q_1 and q_2 for the non-scalable MVC

source encoder, and by q_{BL_1}, q_{BL_2}, q_{EL_1}, and q_{EL_2} for the scalable MVC source encoder, 2) the distortion the packet generates if it is lost in transmission, and 3) the probability that the packet is lost, which depends on channel characteristics specified by Θ. To find the quantization parameters and code rates that minimize the objective functions (2.6) and (2.8), we search over a grid of QPs, where for non-scalable MVC the search is done over a two-dimensional grid specified by vector (q_1, q_2) and for scalable MVC is done over a four-dimensional grid specified by vector $(q_{\mathrm{BL}_1}, q_{\mathrm{BL}_2}, q_{\mathrm{EL}_1}, q_{\mathrm{EL}_2})$. The solution is obtained as a quantization vector in the grid and a set of code rates, which together produce the smallest total number of bits and, at the same time, meet the quality constraints. The sets $\mathcal{Q}_1$ and $\mathcal{Q}_2$ in (2.6), and QP_s in (2.8), are determined for each GOP of a given video sequence. To do that, for the MVC-non-scalable case, we first perform a binary search over q_1 and q_2 to rule out the QPs for which the noise-free encoded video does not meet the quality constraints, and find the largest possible QPs, $q_1^{(max)}$ and $q_2^{(max)}$, that satisfy the constraints. The ruled out QPs are not considered for optimization since they do not meet the quality constraints even in the absence of channel distortion. The sets $\mathcal{Q}_1$ and $\mathcal{Q}_2$ are then defined as the sets whose members are QPs less than or equal to $q_1^{(max)}$ and $q_2^{(max)}$, respectively. For the scalable case, we perform an exhaustive search over the QPs q_{BL_1}, q_{BL_2}, q_{EL_1}, and q_{EL_2}, to rule out the ones for which the noise-free encoded video does not meet the quality constraints.

2.5 Integer Optimization

The optimization problems introduced in Section 2.4 are nonlinear integer programming problems, which can be solved by the branch-and-bound (BnB) method [86]. The BnB method is based on binary variables [86]. For non-scalable MVC, we transform each variable $r_i^{(1)}$ to N binary variables $x_{i,l}$ $(1 \leq l \leq N)$, and each variable $r_j^{(2)}$ to N binary variables $y_{j,l}$, where x and y take values from the set $\{0, 1\}$. We then substitute $r_i^{(1)}$ with $\sum_{l=1}^{N} x_{i,l} R_l$ and $r_j^{(2)}$ with $\sum_{l=1}^{N} y_{j,l} R_l$ in (2.6) and (2.7). With these transformations, $2K$ equality constraints are considered

along with the inequalities given in (2.7), which are

$$\sum_{l=1}^{N} x_{i,l} = 1, \quad 1 \le i \le K$$
$$\sum_{l=1}^{N} y_{j,l} = 1, \quad 1 \le j \le K. \tag{2.10}$$

Now, we consider the scalable MVC case. We transform each variable $r_i^{(\mathrm{BL}_1)}$ to N binary variables $x_{i,l}$ $(1 \le l \le N)$, each variable $r_j^{(\mathrm{BL}_2)}$ to N binary variables $y_{j,l}$, each variable $r_m^{(\mathrm{EL}_1)}$ to N binary variables $z_{m,l}$, and each variable $r_n^{(\mathrm{EL}_2)}$ to N binary variables $t_{n,l}$, where x, y, z, and t take values from the set $\{0, 1\}$. We then make the following substitutions in (2.8) and (2.9): $r_i^{(\mathrm{BL}_1)}$ is substituted with $\sum_{l=1}^{N} x_{i,l} R_l$, $r_j^{(\mathrm{BL}_2)}$ is substituted with $\sum_{l=1}^{N} y_{j,l} R_l$, $r_m^{(\mathrm{EL}_1)}$ is substituted with $\sum_{l=1}^{N} z_{m,l} R_l$, and $r_n^{(\mathrm{EL}_2)}$ is substituted with $\sum_{l=1}^{N} t_{n,l} R_l$. From these transformations, $3K$ equality constraints must be considered in conjunction with the inequalities in (2.9), which are:

$$\sum_{l=1}^{N} x_{i,l} = 1, \quad 1 \le i \le \frac{K}{2}$$
$$\sum_{l=1}^{N} y_{j,l} = 1, \quad 1 \le j \le \frac{K}{2}$$
$$\sum_{l=1}^{N} z_{m,l} = 1, \quad 1 \le m \le K$$
$$\sum_{l=1}^{N} t_{n,l} = 1, \quad 1 \le n \le K. \tag{2.11}$$

2.6 Simulation Results and Discussion

Simulation results for AWGN and flat Rayleigh fading channels are given in this section. Binary phase shift keying (BPSK) modulation/demodulation is employed for data transmission over the channel. Samples of the received signal, at a given signal-to-noise ratio E_b/N_0, can be represented by $y = \alpha x + n$, where E_b is energy-per-bit, N_0 is the one-sided power spectral density of the noise, n

is a zero-mean Gaussian random variable with standard deviation $\sqrt{N_0/2E_b}$, and $x \in \{-1, 1\}$. For an AWGN channel, α is unity, and for a Rayleigh fading channel, α has Rayleigh distribution with $E\{\alpha^2\} = 1$. The coherence time of a fading channel, T_c, represents the number of symbols affected by the same fade level, and assuming a block-fading channel, each fade is considered to be independent of the others. An interleaver is used to mitigate the effect of error bursts due to the fading channels, and we used a fixed size block interleaver with depth 500 and width 100.

Results are presented for two video sequences, 'Race' and 'Oldtimers', with resolution 640×480, where 'Race' is a high-motion video that contains moving objects and camera panning, and 'Oldtimers' is low-motion. We used the JM 18.2 reference software (stereo profile) for encoding the sequences, where each row of macroblocks of either the primary or secondary view is encoded as a slice. We used the JMVC 8.2 reference software for decoding the MVC bit stream. The primary view frames of a GOP are coded as IPPP..., the secondary view frames are coded as PPPP..., and the GOP size is 20 frames.

We used turbo codes for channel coding. The turbo encoder is composed of two recursive systematic convolutional encoders with constraint length 4, which are concatenated in parallel [87]. The feedforward and feedback generators are 15 and 13, respectively, both in octal. The mother code rate of the RCPT code is $\frac{1}{3}$, the puncturing period $P = 8$, and the set of available rates is $\{\frac{P}{P+l} | l = 1, 2, \ldots, 2P\}$. An iterative soft-input/soft-output (SISO) decoding algorithm is used for turbo decoding. We considered eight iterations to compute the decoded word error rates.

Figure 2.4 shows a scatter plot that depicts how the UEP allocates code rates to different packets of an encoded video. This scatter plot is for 'Race', where the video is encoded by a non-scalable MVC encoder, SNR = 11dB, and the channel experiences flat Rayleigh fading with $T_c = 2000$. Each point on the scatter plot corresponds to a packet that belongs either to the primary or secondary view. The x-axis represents the normalized packet size, and the y-axis represents the inverse of the allocated code rate (higher inverse of code rate corresponds to more protection of the code). For a primary view packet, the z-axis indicates the

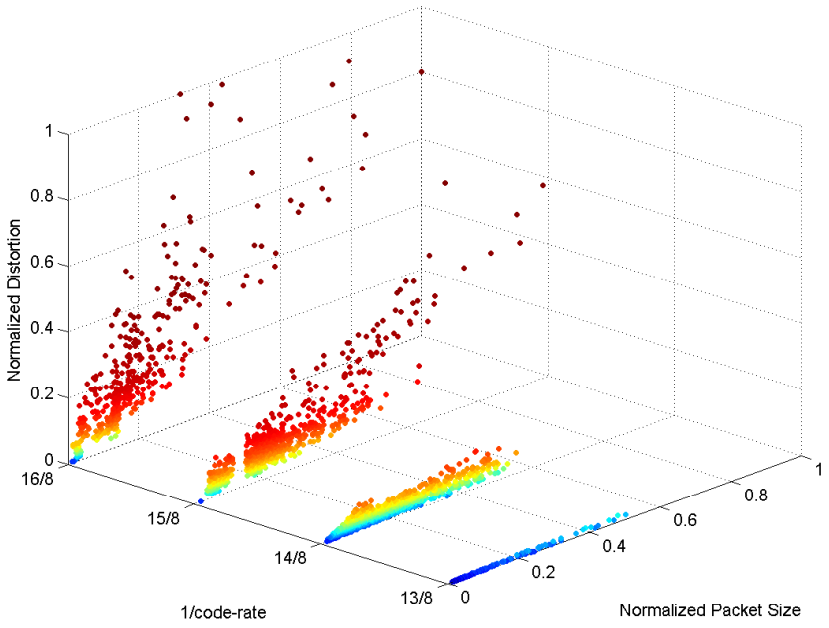
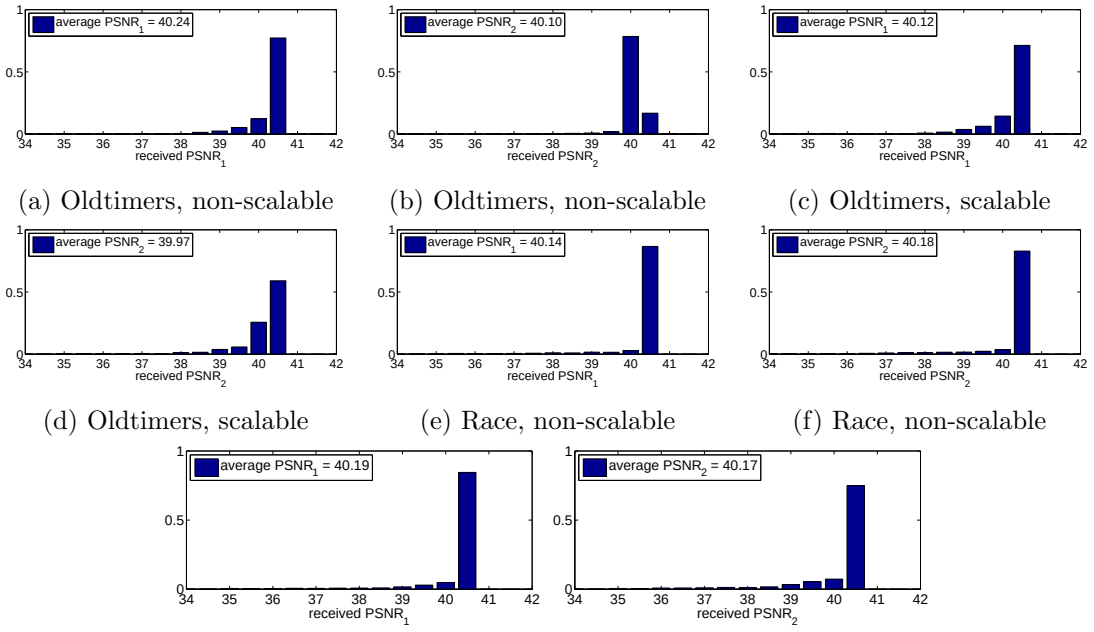


Figure 2.4: Scatter plot of the code rates allocated by UEP to different packets of 'Race', where SNR = 10dB, and $T_c = 2000$.

normalized sum of distortions in the primary and secondary views if that packet is lost, and for a secondary view packet, the z-axis represents the normalized distortion generated in the secondary view if the packet is lost. In this scatter plot, similar levels of distortion are depicted with similar ranges of colors. Packets which generate high distortions are protected with strong codes. These packets are typically the particular slices that generate significant error propagation if they are lost. We also notice that for packets with similar levels of distortion, larger packets are protected less than smaller packets. If two packets generate the same level of distortion, the larger packet might receive less protection than the smaller packet in order to minimize the total number of bits.

We performed validation tests with 2000 channel realizations to see if the UEP solution obtained using the model and the expected end-to-end distortion estimates (developed in Sections 2.2 and 2.3) meets the quality constraints for realistic channel realizations. Figure 2.5 shows the received PSNR histograms of 'Race' and 'Oldtimers', where symmetric coding is considered, and UEP is used for protection of the encoded video over the channel. We see that the average



(a) Oldtimers, non-scalable (b) Oldtimers, non-scalable (c) Oldtimers, scalable

(d) Oldtimers, scalable (e) Race, non-scalable (f) Race, non-scalable

(g) Race, scalable (h) Race, scalable

Figure 2.5: Received PSNR of the primary view, PSNR_1, and the secondary view, PSNR_2, for symmetric coding and both non-scalable and scalable MVC. Results are obtained for 2600 channel realizations of the tested SNR values and T_c's.

PSNRs generally meet the specified 40dB constraints. In Figs. 2.5(a) to (h), the percentage of received PSNRs which are larger than 40dB are 87%, 93%, 82%, 82%, 89%, 85%, 88%, 80%.

In the following, we compare the total number of bits required by UEP and EEP for different scenarios. By EEP, we mean that all of the packets are protected by the best single code rate, which is determined by exhaustive simulation over all possible EEP rates. Each scenario is specified by 1) UEP or EEP, 2) channel is AWGN or fading, 3) non-scalable MVC or scalable MVC encoder/decoder is used, and 4) symmetric coding or asymmetric coding is utilized. In all of the comparisons, the percentage of bit savings of scenario A compared to scenario B is defined as

$$e = \frac{\#bits^{(B)} - \#bits^{(A)}}{\#bits^{(B)}} \times 100\%. \tag{2.12}$$

Figure 2.6 shows the results of non-scalable MVC and symmetric coding for 100 frames of video sequences 'Race' and 'Oldtimers', and both AWGN and fading

channels. Figs. 2.6(a), (c), (e), and (g) show the total number of bits, and Figs. 2.6(b), (d), (f), and (h) depict the percentage of UEP bit savings compared to EEP. UEP always requires fewer bits than EEP. As expected, fewer bits are required when the channel SNR increases. For the fading channel, for a particular SNR, more bits are required for a larger coherence time. This is because when the coherence time gets larger, the diversity order becomes smaller which reduces the capability of a code to protect a packet, and thus, packets need to be protected with a stronger code. The average gains of UEP over EEP for AWGN and fading channels are 11.6% and 13.4%, respectively, for 'Race', and 13.7% and 16.2% for 'Oldtimers'.

From Figure 2.6, we see that the UEP bit savings decreases for higher SNR values, which indicates that the UEP and EEP performances become close for higher SNR values. This is because, as the SNR increases, packets do not need much protection, so both EEP and UEP can use high code rates. For fading channels, we also observe that, for a particular SNR, the UEP bit savings is higher for larger coherence times. As shown in Figure 2.6(a) and (e), and discussed above, for a larger coherence time, EEP needs to protect the data with a lower code rate. UEP can flexibly select from many available code rates, leading to a higher bit savings over EEP.

Comparing the 'Race' and 'Oldtimers' results reveals that the number of bits required for 'Race' (high-motion content) is always higher than that of 'Oldtimers' (low-motion content), which is expected. An interesting observation is that the percentage of bit savings of UEP is slightly higher for 'Oldtimers' compared to 'Race'. For low-motion video, there are fewer packets that should be protected using the strong code rates, and these are the ones that contain high motion and their error propagation can not be concealed efficiently. A larger number of low-motion video packets can be protected with weak codes, which are the ones that belong to the static background or very low-motion regions. These packets generate an insignificant amount of distortion if they are lost in transmission, since their error propagation can be efficiently concealed.

Now we present the results for scalable MVC and symmetric coding. Figs.

2.7(a) and (b) show the number of bits required by UEP and EEP, and Figs. 2.7(c) and (d) illustrate the percentage of bit savings of UEP compared to EEP for fading channels. Comparing the results of Figure 2.6 and Figure 2.7, we see that all the observations made for the non-scalable case are also made for the scalable case. The average gains of UEP over EEP for the fading channels are 17.5% and 19.5% for 'Race' and 'Oldtimers', respectively.

So far, we have presented results for symmetric coding. Asymmetric coding results are presented in Figure 2.8 for non-scalable MVC, and in Figure 2.7(e) and (f) for scalable MVC. The percentages of bit savings of UEP over EEP are comparable to the symmetric coding case.

Figure 2.9 compares the results of symmetric and asymmetric coding for non-scalable video. In this figure, the percentage of bit savings of asymmetric/UEP is compared to both symmetric/UEP and symmetric/EEP. The average gain of asymmetric/UEP over symmetric/UEP and symmetric/EEP for fading channels is 36.4% and 45.2% for 'Race', and 36.8% and 47.1% for 'Oldtimers', respectively. For AWGN, the average gains are 38.3% and 45.4% for 'Race', and 36.1% and 45.0% for 'Oldtimers', respectively. We made similar comparisons between scalable/asymmetric and scalable/symmetric and obtained similar results.

It is also interesting to compare the performance of non-scalable and scalable scenarios to see how much overhead (coding inefficiency) is caused by scalability. By comparing the non-scalable and scalable results, we observe that the number of required bits for scalable MVC is always higher than that of non-scalable MVC. Figure 2.10 depicts the percentage of overhead of scalable MVC compared to non-scalable MVC for 'Race' and 'Oldtimers', and for symmetric coding. Comparable results are obtained for asymmetric coding. Although scalable MVC has an overhead penalty, scalability has an advantage if the subjective quality of the lossy decoded bit stream is considered at the receiver. When a BL packet is lost through transmission, frame copying error concealment is used at the decoder, which generates a noticeable error propagated throughout the GOP [85], specifically for slices possessing high motion content. However, when an EL packet is lost, an upsampled version of the BL is used for error concealment at the decoder,

Table 2.2: Percentage of packet losses of the tested video bit streams protected by UEP over the flat Rayleigh fading channel.

	non-scalable	scalable		
		BL	EL	BL & EL
Race	0.10%	0.02%	0.08%	0.0000%
Oldtimers	0.34%	0.12%	0.24%	0.0006%

which perhaps causes a less noticeable error [85].

Table 2.2 shows the percentages of packets that are lost from either the BL, EL, or both layers, for 2600 flat Rayleigh fading channel realizations of all the tested SNR values and coherence times. Results are given for both scalable MVC and non-scalable MVC, where the encoded bit streams are protected using the code rates obtained by the UEP approach. Considering 'Race' for example, we observe that 0.02% and 0.08% of the packets are respectively lost from the BL and the EL. This indicates that the majority of losses occur in the EL, whose errors are concealed more effectively than ones in the BL. In addition, we see that the percentage of BL losses is considerably lower than that of the non-scalable losses. These observations suggest that scalability can perform better than non-scalability in terms of subjective quality.

2.7 Conclusions

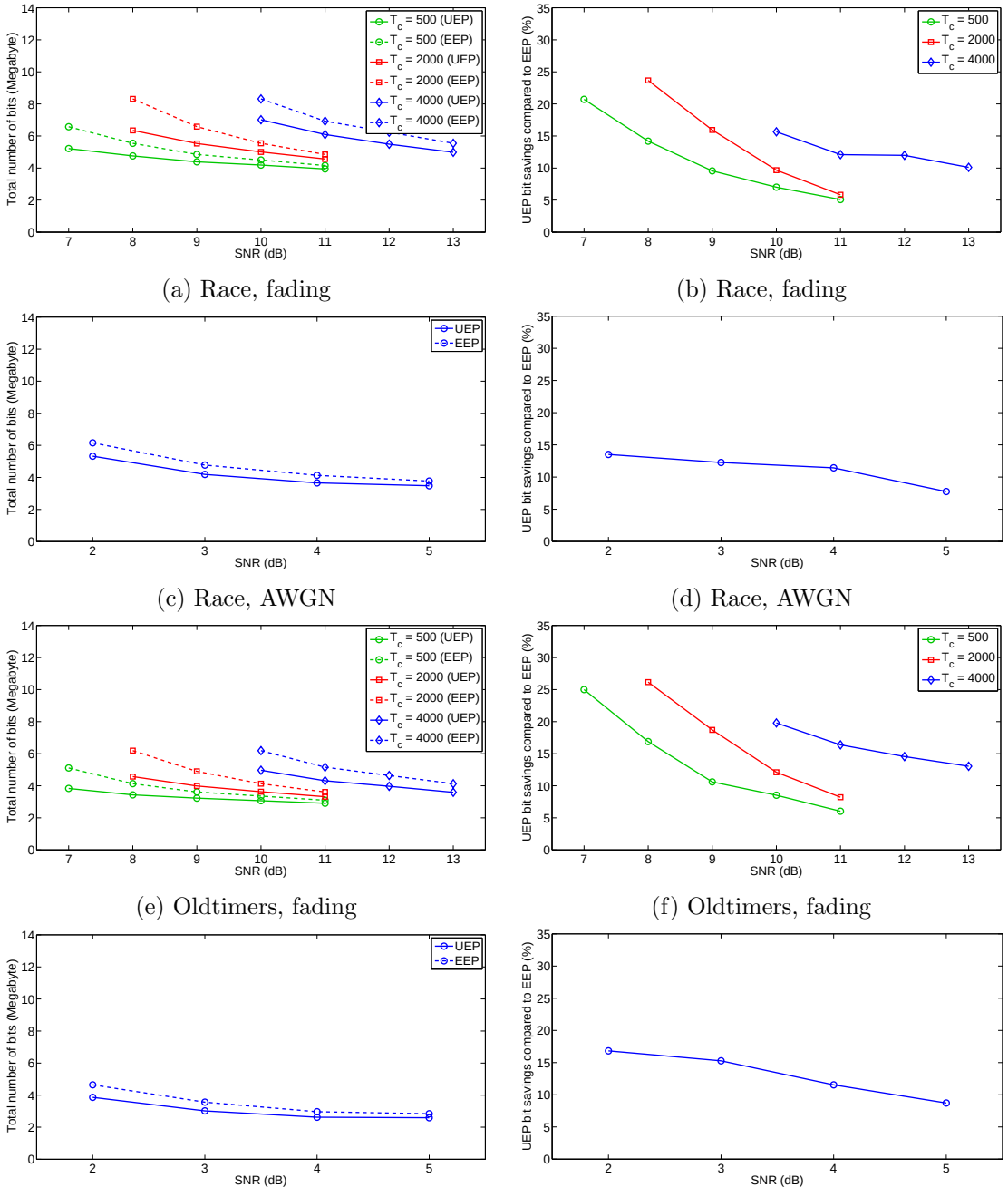
In this chapter, we addressed the JSCC problem of a 3D video sent over AWGN and fading channels with the goal of minimizing the total number of transmitted bits while subject to video quality constraints. We considered non-scalable MVC and a type of spatial-scalable MVC, and both symmetric and asymmetric coding. The UEP approach proposed here proved to be efficient at achieving this goal when compared to EEP for all the scenarios considered, where the average gains vary from 11.6% to 19.5%. Asymmetric coding was also compared to symmetric coding. Comparable gains were obtained for non-scalable MVC and scalable MVC, where the asymmetric/UEP gain over symmetric/UEP and symmetric/EEP vary, respectively, from 36.1% to 38.3% and from 45.0% to 47.1%.

We also showed that although using scalability leads to an overhead compared to non-scalable MVC, it may have an advantage in terms of the subjective quality of the received video, since most of the lost packets occur in the enhancement layer whose errors are less noticeable to the human visual system compared to the errors due to packets lost in the base layer.

2.8 Acknowledgment

This research was supported by the Intel/Cisco Video Aware Wireless Networks (VAWN) program, by InterDigital, Inc., and by the National Science Foundation under grant number CCF-1160832.

Chapter 2 of this dissertation is a reprint of the material as it appears in A. Vosoughi, V. Testoni, P. Cosman, and L. Milstein, "Multiview coding and error correction coding for 3D video over noisy channels", *Signal Processing: Image Communication*, vol. 30, pp. 107-120, Jan 2015, and is, in part, based on the material as it appears in A. Vosoughi, V. Testoni, P. Cosman, and L. Milstein, "Joint source-channel coding of 3D video using multiview coding", in Proc. *ICASSP*, 2013. I was the primary author and the co-authors Prof. Cosman and Prof. Milstein directed and supervised the research. The co-author Dr. Testoni also contributed to the ideas in this work.



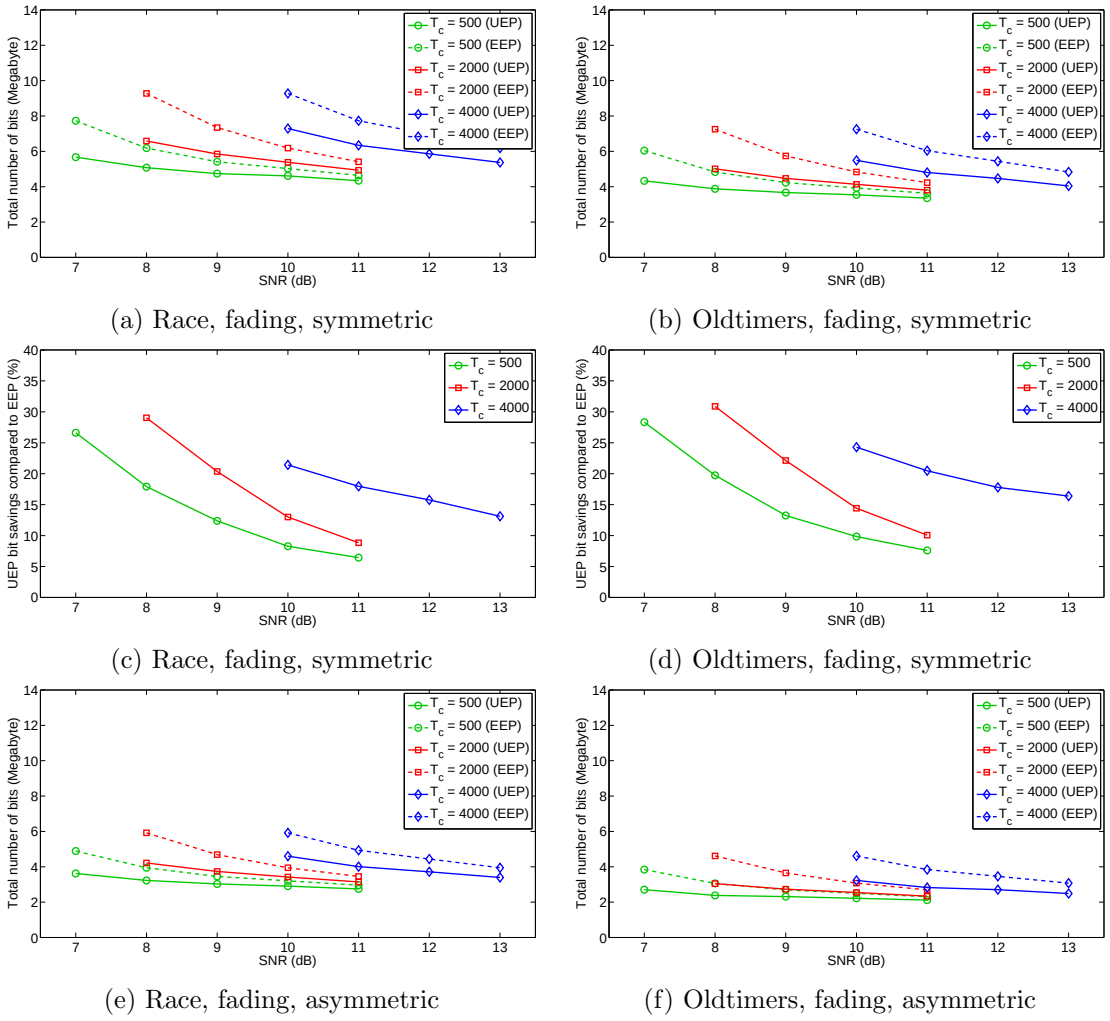
(a) Race, fading (b) Race, fading

(c) Race, AWGN (d) Race, AWGN

(e) Oldtimers, fading (f) Oldtimers, fading

(g) Oldtimers, AWGN (h) Oldtimers, AWGN

Figure 2.6: Results for non-scalable MVC, symmetric coding, and AWGN and fading channels. (a), (c), (e), (g) Total number of bits required by UEP and EEP, and (b), (d), (f), and (h) percentage of bit savings of UEP compared to EEP.



(a) Race, fading, symmetric

(b) Oldtimers, fading, symmetric

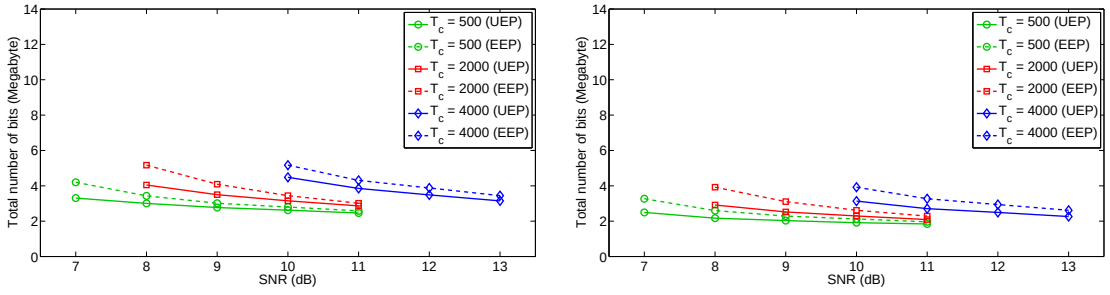
(c) Race, fading, symmetric

(d) Oldtimers, fading, symmetric

(e) Race, fading, asymmetric

(f) Oldtimers, fading, asymmetric

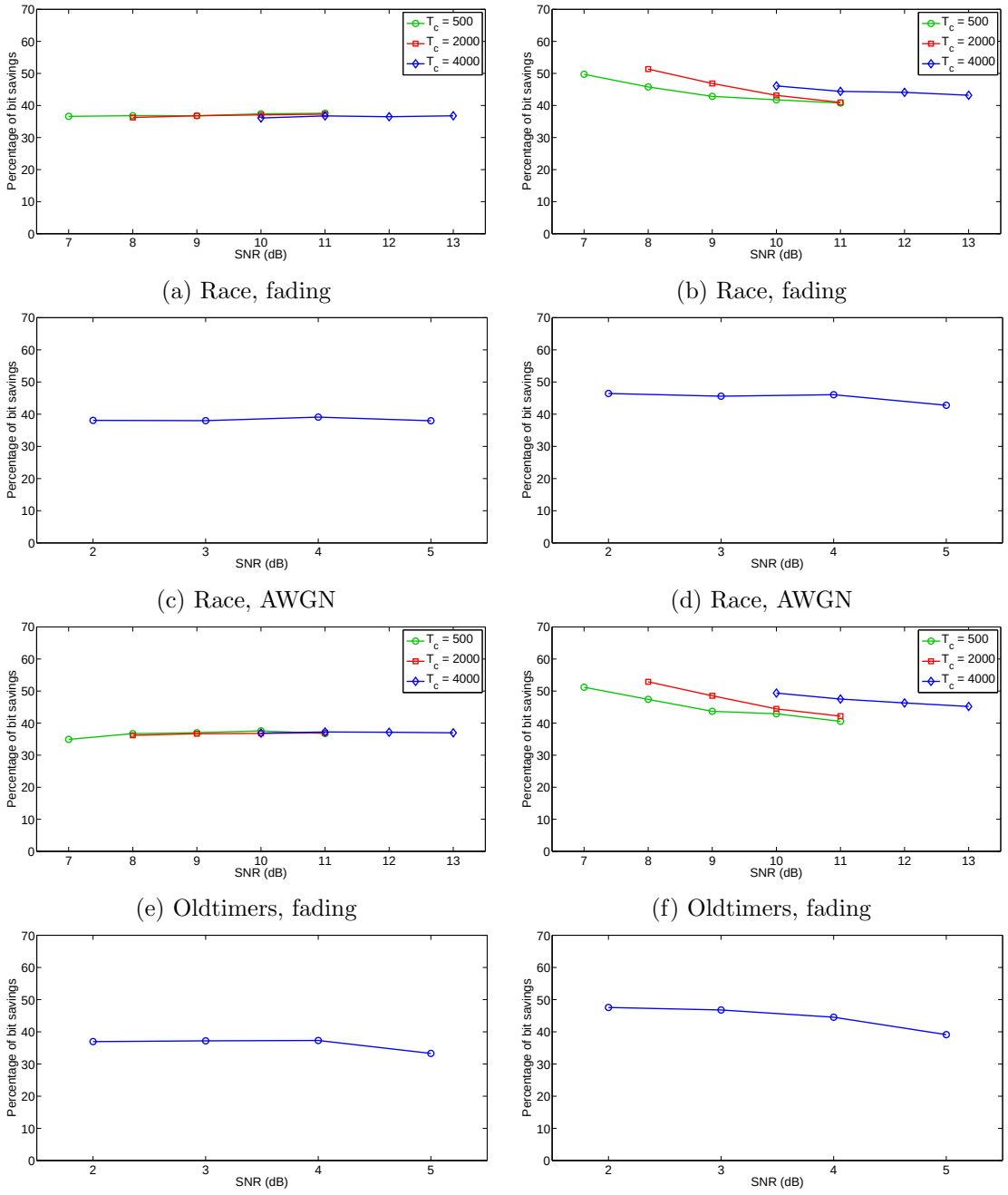
Figure 2.7: Results for scalable MVC and fading channels.



(a) Race, fading

(b) Oldtimers, fading

Figure 2.8: Results for non-scalable MVC, asymmetric coding, and fading channels.



(a) Race, fading

(b) Race, fading

(c) Race, AWGN

(d) Race, AWGN

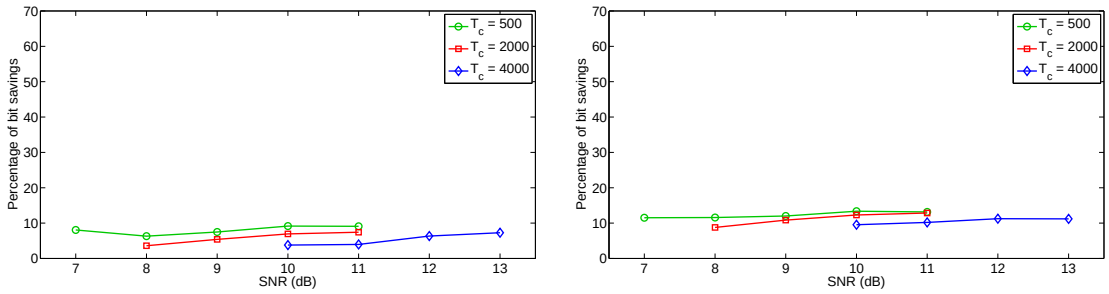
(e) Oldtimers, fading

(f) Oldtimers, fading

(g) Oldtimers, AWGN

(h) Oldtimers, AWGN

Figure 2.9: Percentage of bit savings of asymmetric coding compared to symmetric coding for non-scalable MVC and both AWGN and fading channels. (a), (c), (e), (g) Bit savings of asymmetric/UEP over symmetric/UEP, (b), (d), (f), and (h) bit savings of asymmetric/UEP over symmetric/EEP.



(a) Race (b) Oldtimers

Figure 2.10: Percentage of bit savings of non-scalable MVC compared to scalable MVC for symmetric/UEP and fading channels.

Chapter 3

Unequal Error Protection for Video Plus Depth

In the previous chapter, we considered UEP for 3D video compressed using MVC. In this chapter, we consider UEP for 3D video compressed using the video plus depth (V+D) format. We propose a JSCC scheme for V+D data. Full-resolution and downsampled depth maps are considered. The proposed JSCC scheme yields the optimum color and depth quantization parameters, as well as the optimum FEC code rates used for UEP at the packet level. Different coding scenarios are compared, and the UEP gain over EEP is quantified for flat Rayleigh fading channels. We show that the proposed UEP scheme significantly outperforms EEP. We also derive several interesting results on how the color and depth are compressed and protected compared to each other.

This chapter is organized as follows: Section 3.1 describes V+D encoder and decoder. Section 3.2 shows how the proposed JSCC scheme is adopted in a video transmission system. Section 3.3 derives an end-to-end distortion measure based on the SSIM for both the left view and the synthesized right view. Section 3.4 gives the JSCC problem formulation for V+D using the end-to-end distortion measure introduced in Section 3.3. Section 3.5 discusses simulation results, and Section 3.6 concludes this chapter.

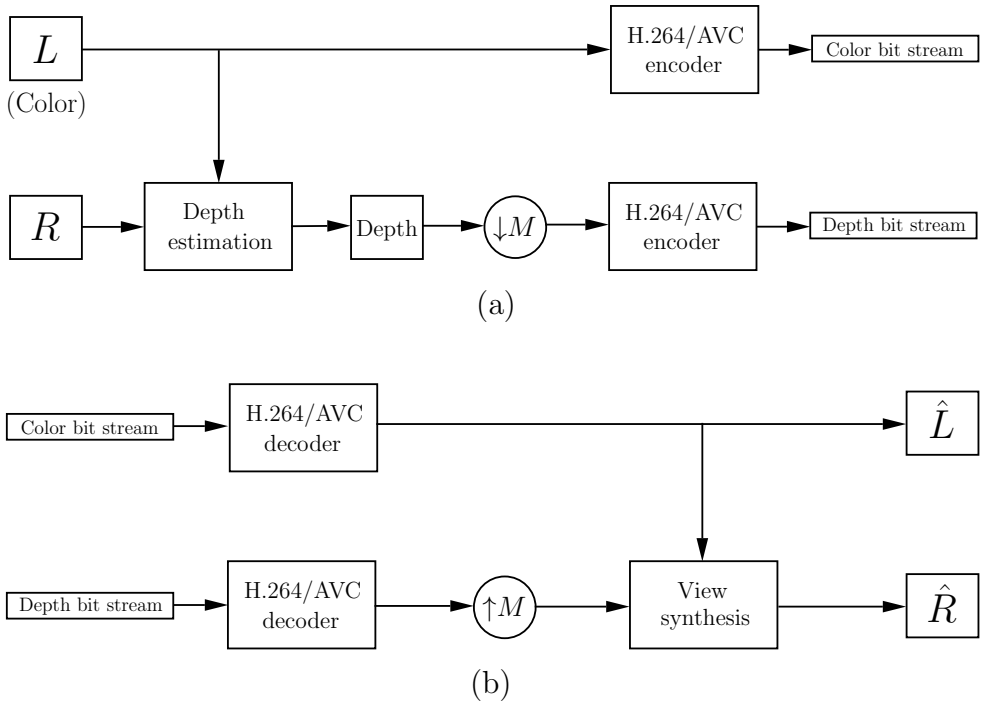


Figure 3.1: Block diagram of (a) V+D encoder, and (b) V+D decoder.

3.1 V+D Encoder and Decoder

V+D is an efficient representation of 3D video, where a stereo pair is rendered at the decoder from a color video signal and a per-pixel depth map. V+D data is usually compressed by independently compressing the color and depth using a compression tool such as H.264/AVC. Figure 3.1(a) and Figure 3.1(b) depict V+D encoder and decoder, respectively. We may downsample depth by a factor of M before the compression (for example, downsampling by 2 is recommended in [55]). At the decoder, left view is obtained by decompressing the compressed color video. Depth map is decompressed and right view is synthesized using both the decompressed color and depth. We note that depth should be upsampled by a factor of M if it is downsampled at the encoder. We assume that precomputed depth maps are available at the encoder and our focus will be on the compression and protection of the color and depth and their impacts on the quality of the reconstructed left and right views.

3.2 Overview of the System Design

Our goal is to solve the JSCC problem for V+D data. We consider both the downsampled and full-resolution depth map scenarios. In formulating the JSCC for V+D, we follow the typical JSCC problem formulation in which the bit rate is set as constraint and the goal is to maximize the quality at the receiver. It is worth mentioning that we are not able to apply the MVC quality thresholds 40dB/40dB (for symmetric coding) and 40dB/33dB (for asymmetric coding) to the V+D case. The quality of the synthesized right video in terms of the PSNR always shows a great loss even in the absence of any compression (30dB on the average over different video sequences). Although this PSNR loss is mostly tolerated by the HVS, it does not even meet the 33dB quality threshold of the asymmetric coding. The PSNR loss is due to occlusion, color mismatch between the right view and left view, and noisy/inaccurate depth maps, which are unavoidable in any view synthesis algorithm.

The system block diagram is shown in Figure 3.2. The color video and depth map are first both compressed by an H.264/AVC encoder. A UEP technique is used to assign FEC to packets of the compressed color and depth. The protected color and depth bit streams are then transmitted over the channel. At the receiver, channel decoding is performed and the erroneous packets are detected. The color (left view) and depth bit streams are decoded by an H.264/AVC decoder, where error concealment is done for the erroneous packets that are already detected by the channel decoder. The right view is then synthesized using the decoded color and depth. The depth map may be downsampled by a factor of M at the encoder before the compression. Having the depth map downsampled by a factor of M, we should upsample the depth map by a factor of M before performing the view synthesis. We consider full-resolution and downsampled depth by factors of 2 and 4, which are represented by $\downarrow$No, $\downarrow$2, and $\downarrow$4, respectively.

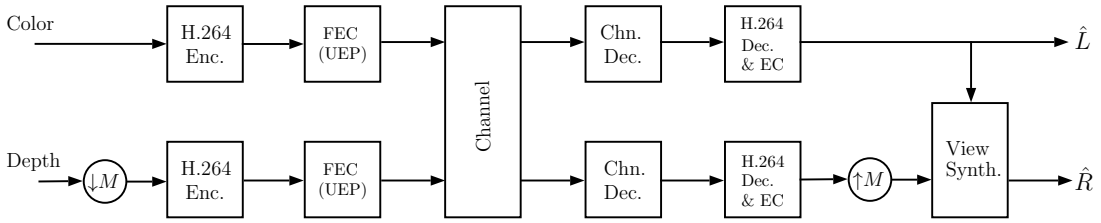


Figure 3.2: Block diagram of a V+D transmission system employing the proposed JSCC scheme.

3.3 End-to-End Distortion Based on SSIM

To formulate the JSCC problem within a V+D framework, we need an objective metric that is able to measure the end-to-end distortion of the reconstructed left and right views. It has been shown that the conventional objective metric PSNR is not able to model the distortions that are due to view synthesis errors [8]. We use SSIM to quantify the end-to-end distortion, since it has been shown that perceived quality is better correlated to SSIM than to PSNR [8]. The SSIM between two GOPs x and y is calculated as the average of SSIMs between the corresponding frames of x and y, and we denote it by $\mathrm{SSIM}(x, y)$. In contrast to the MSE, the SSIM is highly nonlinear and thus we cannot model the end-to-end distortion as in Section 2.2. Instead, we define a quality score based on the SSIM, that is computed for each individual slice of the compressed color and depth.

We first derive a measure of the end-to-end distortion for the left (color) view based on the SSIM. This measure should incorporate both the effects of the color source distortion and the color channel distortion. We assign a score to each packet of the color. The score depends on whether the packet is or is not lost. More precisely, if the ith packet of a color GOP is lost, the score assigned to that packet is computed by

$$d^L_{i,C}(q_C) = \mathrm{SSIM}\Big(f^L, \tilde{f}^L_{i,C}(q_C)\Big), \tag{3.1}$$

where q_C is the color quantization parameter, f^L denotes the original uncompressed left view GOP, and $\tilde{f}^L_{i,C}(q_C)$ represents the left view decoded GOP with error

concealment as if only the ith color packet were lost. We note that $d^L_{i,C}(q_C)$ reflects the quality throughout the left view GOP (including the effect of error propagation) due to losing the ith color packet; larger values of $d^L_{i,C}(q_C)$ correspond to lower distortion generated due to loss of the ith packet. If the ith color packet is not lost, the score assigned to that packet is computed by

$$d^L_s(q_C) = \text{SSIM}\Big(f^L, \hat{f}^L(q_C)\Big), \tag{3.2}$$

where $\hat{f}^L(q_C)$ denotes the left view error-free decoded GOP. Since each packet has two different scores (given in (3.1) and (3.2)), the score is a random variable. Let $\mathbf{D}_i$ be the random variable representing the score assigned to the ith packet. Thus, $\{\mathbf{D}_i = d^L_{i,C}(q_C)\}$ if the packet is lost, and $\{\mathbf{D}_i = d^L_s(q_C)\}$ if the packet is not lost. For a particular q_C, $d^L_{i,C}(q_C)$ and $d^L_s(q_C)$ can be computed offline at the encoder for $1 \leq i \leq N_C$, where N_C is the number of packets in a color GOP. Now, we take the expected value of the average of the scores of all the color packets as the quality of the left view:

$$E^L = E\Big\{\frac{1}{N_C}\sum_{i=1}^{N_C}\mathbf{D}_i\Big\}. \tag{3.3}$$

Let $p_{i,C}\Big(s_{i,C}(q_C), r_{i,C}, \Theta\Big)$ be the probability of losing the ith color packet, where $p_{i,C}$ depends on the source packet size in bits, $s_{i,C}(q_C)$, the code rate allocated to that packet, $r_{i,C}$, and the channel characteristics Θ. For a flat Rayleigh fading channel, Θ=(SNR, T_c), where T_c is the coherence time. Following (3.3), we have

$$E^L = \frac{1}{N_C}\sum_{i=1}^{N_C}\left(d^L_s(q_C) + p_{i,C}\Big(s_{i,C}(q_C), r_{i,C}, \Theta\Big)\Big(d^L_{i,C}(q_C) - d^L_s(q_C)\Big)\right). \tag{3.4}$$

For the synthesized right view, scores are defined for both the color and depth packets, since both contribute to the quality of the synthesized right view. If the ith color packet is lost, we compute the score as

$$d^R_{i,C}(q_C, q_D) = \text{SSIM}\Big(f^R, \tilde{f}^R_{i,C}(q_C, q_D)\Big), \tag{3.5}$$

and if the ith depth packet is lost, we compute the score from

$$d^R_{i,D}(q_C, q_D) = \text{SSIM}\Big(f^R, \tilde{f}^R_{i,D}(q_C, q_D)\Big). \tag{3.6}$$

In (3.5) and (3.6), q_D is the depth quantization parameter, f^R is the GOP synthesized from the original left view ([88], [89]), $\tilde{f}^R_{i,C}(q_C, q_D)$ denotes the right view GOP after decoding, error concealment, and view synthesis as if only the ith packet were lost from the color, and $\tilde{f}^R_{i,D}(q_C, q_D)$ denotes the right view GOP after decoding, error concealment, and view synthesis as if only the ith packet were lost from the depth. If the ith packet of the color or the depth is not lost, the score assigned to that packet is computed by

$$d^R_s(q_C, q_D) = \text{SSIM}\Big(f^R, \hat{f}^R(q_C, q_D)\Big), \tag{3.7}$$

where $\hat{f}^R(q_C, q_D)$ denotes the error-free decoded synthesized right view GOP. Similar to (3.3), we consider the expected value of the average of scores of color and depth packets as the quality of the synthesized right view:

$$\begin{aligned} E^R = &\frac{1}{N_C+N_D}\left(\sum_{i=1}^{N_C}\left(d^R_s(q_C, q_D)+p_{i,C}\Big(s_{i,C}(q_C), r_{i,C}, \Theta\Big)\Big(d^R_{i,C}(q_C, q_D) - d^R_s(q_C, q_D)\Big)\right)+\right. \\ &\left.\sum_{i=1}^{N_D}\left(d^R_s(q_C, q_D) + p_{i,D}\Big(s_{i,D}(q_D), r_{i,D}, \Theta\Big)\Big(d^R_{i,D}(q_C, q_D) - d^R_s(q_C, q_D)\Big)\right)\right), \end{aligned} \tag{3.8}$$

where N_D is the number of packets of a depth GOP, $s_{i,D}(q_D)$ is the size of the ith depth source packet in bits, and $r_{i,D}$ and $p_{i,D}$ are, respectively, the code rate allocated to that packet and the probability of losing that packet.

We now define the objective function of the JSCC problem as

$$\frac{E^L + E^R}{2}, \tag{3.9}$$

that is to be maximized. An interpretation of this objective function is as follows:

Let us consider the ith and the jth packets of the color, where $i \neq j$ and $1 \leq i, j \leq N_C$. The contribution of the ith packet and the jth packet to the E^L term of the objective function is equal to $f_i = \frac{d_s^L(q_C)+p_{i,C}\times\left(d_{i,C}^L(q_C)-d_s^L(q_C)\right)}{N_C}$ and $f_j = \frac{d_s^L(q_C)+p_{j,C}\times\left(d_{j,C}^L(q_C)-d_s^L(q_C)\right)}{N_C}$, respectively. We note that $d_{k,C}^L(q_C) - d_s^L(q_C) \leq 0$ for $1 \leq k \leq N_C$. Thus, if $p_{i,C} = p_{j,C}$ and $d_{i,C}^L(q_C) > d_{j,C}^L(q_C)$, or, if $p_{i,C} < p_{j,C}$ and $d_{i,C}^L(q_C) = d_{j,C}^L(q_C)$, then $f_i > f_j$. This means that a packet with a lower distortion value (larger score) or a smaller loss probability has a larger contribution to the objective function, that is to be maximized. Further, note that if $d_{i,C}^L(q_C) = d_s^L(q_C)$, then $f_i = \frac{d_s^L(q_C)}{N_C}$, meaning that the contribution of a packet with no channel distortion due to error concealment is equal to the source distortion averaged over all the packets. The interpretation given above is for the E^L term of the objective function; a similar interpretation can be made for the E^R term.

3.4 JSCC Problem Formulation for V+D

We formulate the JSCC problem as: given a bit budget B, we seek to maximize the overall quality of the reconstructed 3D video that is measured by the objective function defined in (3.9). We note that the total number of bits, which is the sum of the number of source bits and FEC bits, is equal to

$$\sum_{i=1}^{N_C} \frac{s_{i,C}(q_C)}{r_{i,C}} + \sum_{i=1}^{N_D} \frac{s_{i,D}(q_D)}{r_{i,D}}. \tag{3.10}$$

Let $\mathcal{R}$ be the set of available code rates, and $\mathcal{Q}_C$ and $\mathcal{Q}_D$ represent the sets of quantization parameters used to encode the color and depth, respectively. Let $\mathbf{R_C} \triangleq (r_{1,C}, r_{2,C}, \ldots, r_{N_C,C})$, and $\mathbf{R_D} \triangleq (r_{1,D}, r_{2,D}, \ldots, r_{N_D,D})$. To maximize the quality of the received 3D video, we maximize the objective function

$$\max_{\substack{(q_C,q_D)\in\mathcal{Q}_C\times\mathcal{Q}_D \\ \mathbf{R_C}\in\mathcal{R}^{N_C},\mathbf{R_D}\in\mathcal{R}^{N_D}}} \frac{E^L + E^R}{2}, \tag{3.11}$$

subject to the bit constraint

$$\sum_{i=1}^{N_C} \frac{s_{i,C}(q_C)}{r_{i,C}} + \sum_{i=1}^{N_D} \frac{s_{i,D}(q_D)}{r_{i,D}} \leq B, \tag{3.12}$$

where B is the bit budget. The optimization problem introduced in (3.11) and (3.12) is a discrete optimization problem that is solved using the branch and bound method [86].

3.5 Simulation Results and Discussion

We present simulation results for flat Rayleigh fading channels with BPSK modulation/demodulation. We use block-fading model and employ a block interleaver with depth 500 and width 100. We use UMTS turbo codes for FEC [87]. The available code rates we considered are $\{\frac{8}{9}, \frac{4}{5}, \frac{2}{3}, \frac{4}{7}, \frac{1}{2}, \frac{4}{9}, \frac{2}{5}, \frac{4}{11}, \frac{1}{3}\}$, obtained by puncturing a mother code of rate $\frac{1}{3}$. An iterative SISO decoding algorithm is used for turbo decoding. Each row of macroblocks is encoded as a packet, the GOP structure is IPPP, and the GOP size is 10 frames.

Figure 3.3 shows the trajectories of the optimum QPs obtained for a GOP of 'Balloons' video sequence (1024×768) as the bitrate constraint increases, where SNR=8dB and T_c=4000. For the ↓No scenario, when the bitrate constraint is 3.1 Mbps, QP_{depth}=50 and QP_{depth}=35. When the bitrate constraint increases to 12.3 Mbps, QP_{depth} goes to 40 and QP_{color} goes to 22. This shows that over a range of rates, the depth can be significantly compressed compared to the color. When the depth is downsampled, QP_{depth} is still larger than QP_{color}, but the gap is smaller than for the ↓No scenario, showing that when depth has lost spatial resolution, the optimization does not penalize it so much on compression.

Figures 3.4, 3.5, and 3.6 show the color and depth average code rates versus the bitrate constraint for video sequence 'Balloons' for scenarios ↓No, ↓2, and ↓4, respectively. The average code rates decrease when the bitrate constraint increases. Considering the ↓No scenario, we see that although the depth is significantly compressed compared to the color (see Figure 3.3), it is protected more since the depth

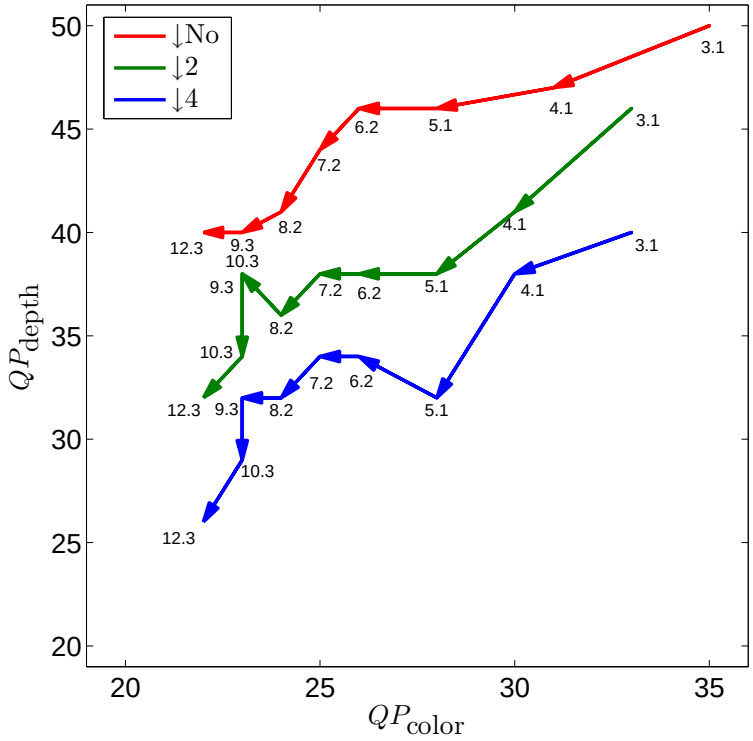


Figure 3.3: Trajectories of the optimum QPs for ↓No, ↓2, and ↓4 for a flat Rayleigh fading channel with SNR=8dB and T_c=4000. Numbers next to the trajectories denote the bitrate constraints in Mb/sec.

average code rate is lower than that of the color. In [53], the authors concluded that color should be protected more than depth. That conclusion was made for the symmetric coding case, where QP_{color}=QP_{depth}=30. We also solved the JSCC problem with the additional symmetric coding constraint, i.e., we set q_C=q_D in (3.11) and (3.12), and our results showed that, indeed, the color should be protected more than the depth, in agreement with [53]. In other words, we are in agreement with the results of [53] for the special case of identical quantization parameters, but the general case of unequal quantization parameters yields the result that depth should be compressed more severely than color, but that then depth should be protected more. Results for ↓2 and ↓4 also indicate that the JSCC tends to protect the depth slightly more than the color.

We now compare the scenarios ↓No, ↓2, and ↓4 in terms of FEC protection.

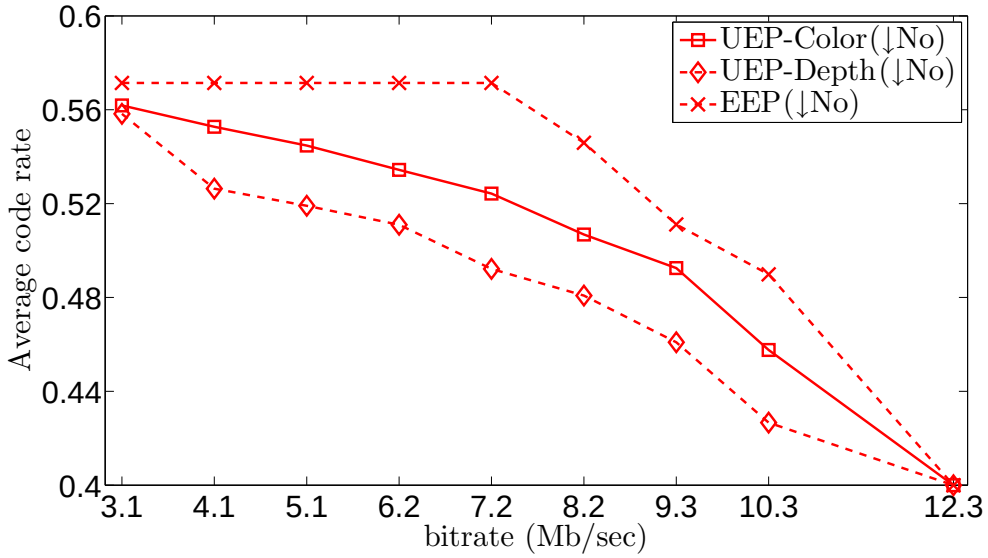


Figure 3.4: Average color and depth code rates for a flat Rayleigh fading channel with SNR=8dB for video sequence 'Balloons' for ↓No.

We compute the average code rate $\overline{R}$:

$$\overline{R} = \frac{\#\text{color source bits} + \#\text{depth source bits}}{\#\text{color source+FEC bits} + \#\text{depth source+FEC bits}}.$$

Figure 3.7 shows $\overline{R}$ versus the bitrate constraint for 'Balloons'. For a particular bitrate, $\overline{R}$ decreases when the downsampling factor increases, meaning that for the same bitrate constraint, a stronger protection is needed for a larger downsampling factor.

Different scenarios are also compared for channel realizations using the PSNR and SSIM metrics. Following [88] and [89], in computing the full-reference metrics PSNR and SSIM, the reference of the right view is obtained by view synthesis from the original uncompressed left view. For each channel realization, the left and right view SSIMs are averaged and then the average is taken over all the channel realizations, which is denoted by $\overline{\text{SSIM}}_{LR}$. The average PSNR for each channel realization is calculated by $10\log_{10}\left(\frac{255^2}{(\text{MSE}_L+\text{MSE}_R)/2}\right)$, where MSE_L and MSE_R are the mean-squared errors obtained for the left and right views, respectively. The average is then taken over all the channel realizations, which is denoted by $\overline{\text{PSNR}}_{LR}$. Figs. 3.8 shows $\overline{\text{PSNR}}_{LR}$ for 200 channel realizations for video sequence 'Balloons',

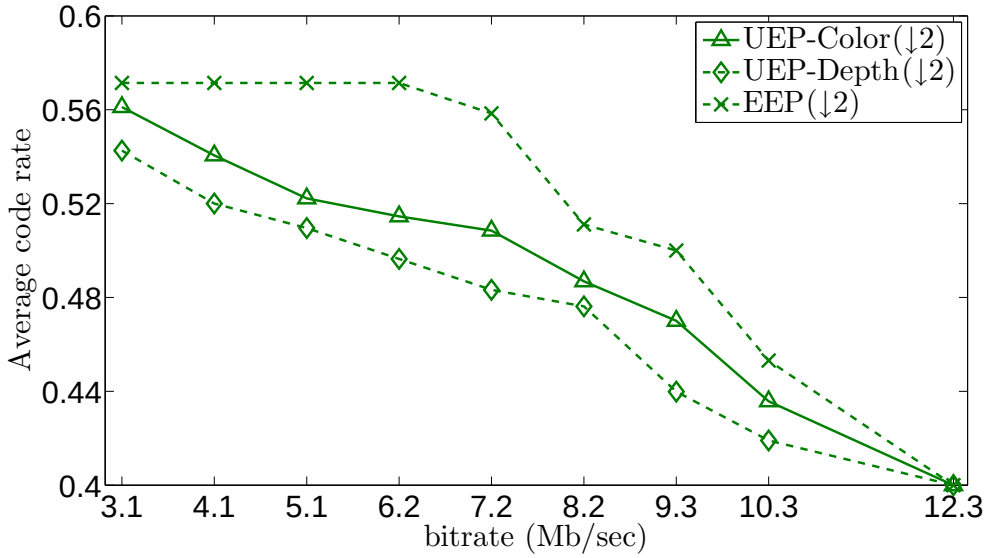


Figure 3.5: Average color and depth code rates for a flat Rayleigh fading channel with SNR=8dB for video sequence 'Balloons' for ↓2.

where SNR=8dB and T_c=4000. Results for $\overline{\text{SSIM}}_{LR}$ are given in Figure 3.9. We see that the ↓4 scenario outperforms the other scenarios except for high bitrates, for which the ↓2 scenario slightly outperforms the others.

Lastly, we compare the performance of UEP to that of EEP. Results are given for scenario ↓4, which was the best for most of the bitrates and channel conditions considered. Fig. 3.10 shows $\overline{\text{PSNR}}_{LR}$ for video sequence 'Balloons'. UEP outperforms EEP by up to 4.3dB. Figure 3.11 shows $\overline{\text{SSIM}}_{LR}$, where we see that the UEP outperforms EEP in terms of the SSIM as well.

3.6 Conclusions

In this chapter, we studied JSCC for video plus depth. Full-resolution and downsampled depth by factors of two and four were considered. Results show that the depth can be significantly compressed compared to the color (especially for ↓No and ↓2), although it needs to be protected more by FEC. We showed that when depth is downsampled, it should be less compressed and more protected to maximize the quality. In contrast to prior work which only considered equal quan-

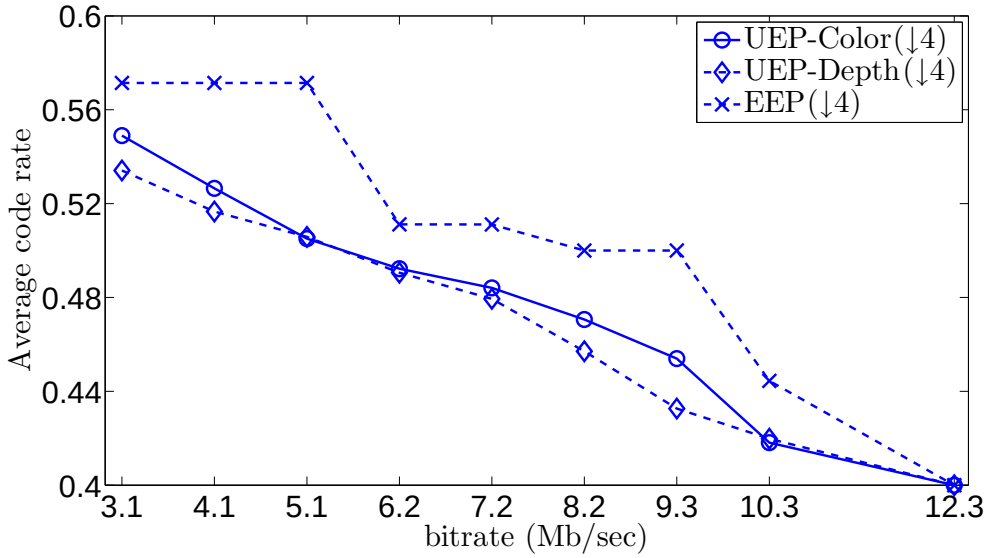


Figure 3.6: Average color and depth code rates for a flat Rayleigh fading channel with SNR=8dB for video sequence 'Balloons' for ↓4.

tization parameters and found that color should be more protected than depth, we found that depth should be compressed more severely than color and then protected more. We also showed that the downsampled depth by a factor of four outperforms the other scenarios except for high bitrates. The UEP approach proposed here was shown to yield up to 4.3dB gain in PSNR compared to EEP for flat Rayleigh fading channels.

3.7 Acknowledgment

This research was supported by the Intel/Cisco Video Aware Wireless Networks (VAWN) program and by the National Science Foundation under grant number CCF-1160832.

Chapter 3 of this dissertation is a reprint of the material as it appears in A. Vosoughi, P. Cosman, and L. Milstein, "Joint source-channel coding and unequal error protection for video plus depth", *IEEE Signal Processing Letters*, vol. 22, Jan 2015. I was the primary author and the co-authors Prof. Cosman and Prof. Milstein directed and supervised the research.

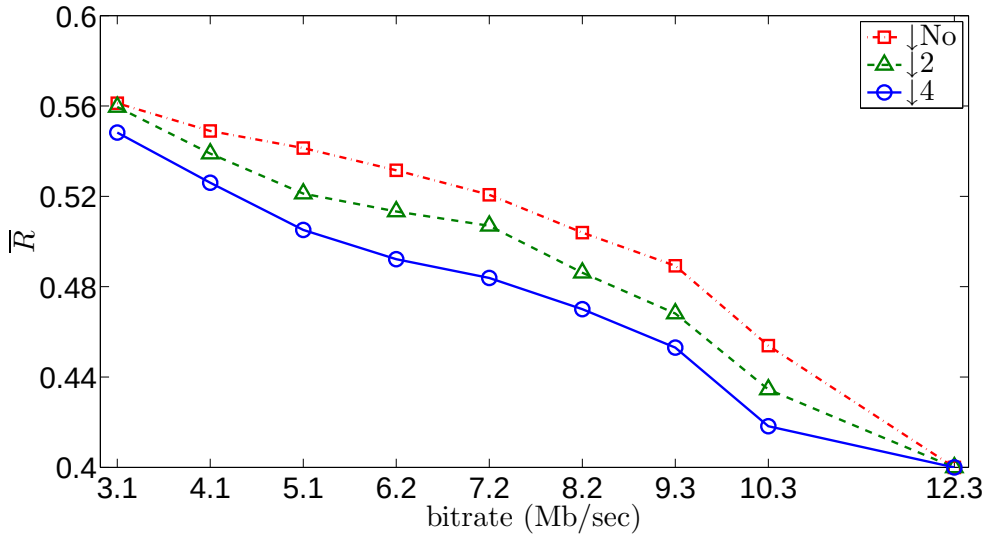


Figure 3.7: $\overline{R}$ for ↓No, ↓2, and ↓4 for a flat Rayleigh fading channel with SNR=8dB and T_c=4000.

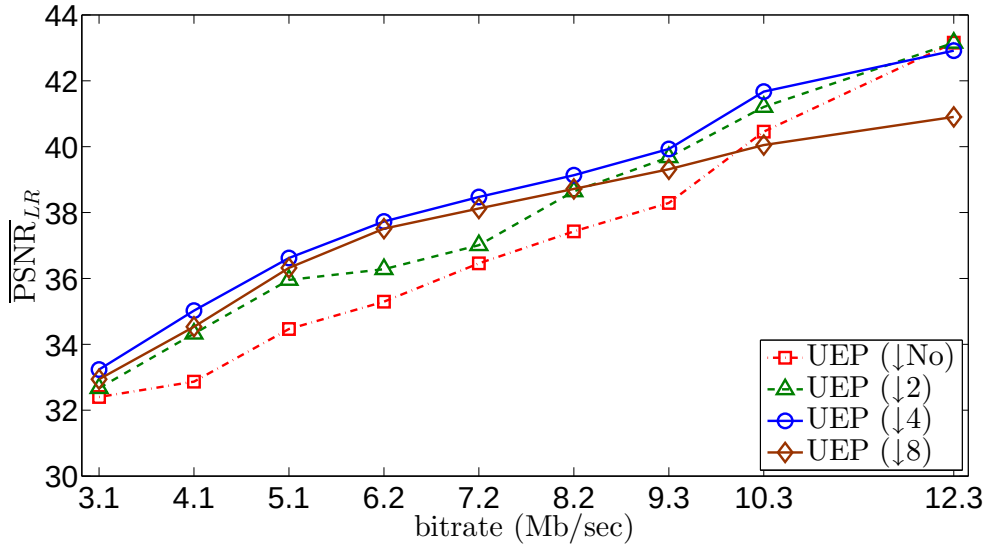


Figure 3.8: $\overline{\text{PSNR}}_{LR}$ obtained by using UEP for ↓No, ↓2, ↓4, and ↓8 for a flat Rayleigh fading channel with SNR=8dB and T_c=4000.

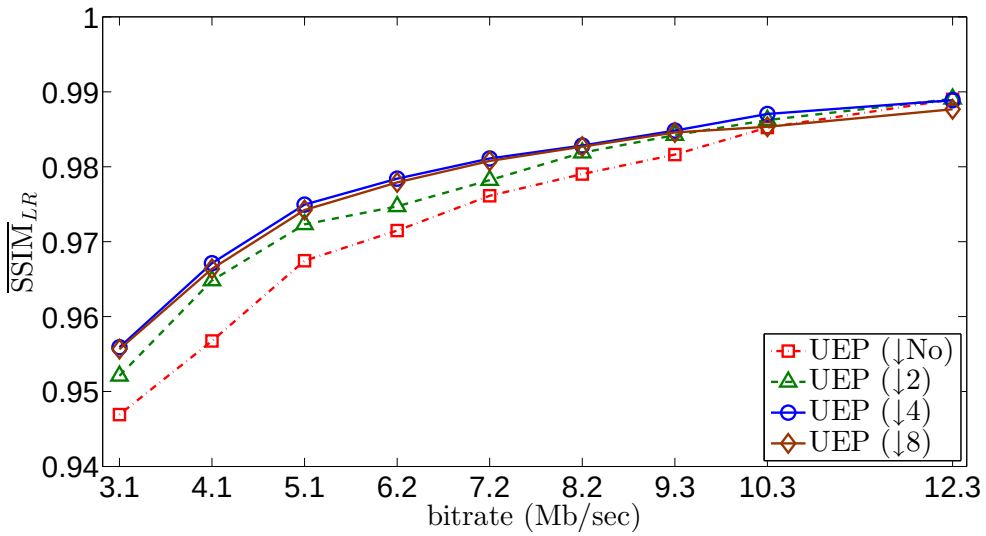


Figure 3.9: $\overline{\text{SSIM}}_{LR}$ obtained by using UEP for ↓No, ↓2, ↓4, and ↓8 for a flat Rayleigh fading channel with SNR=8dB and T_c=4000.

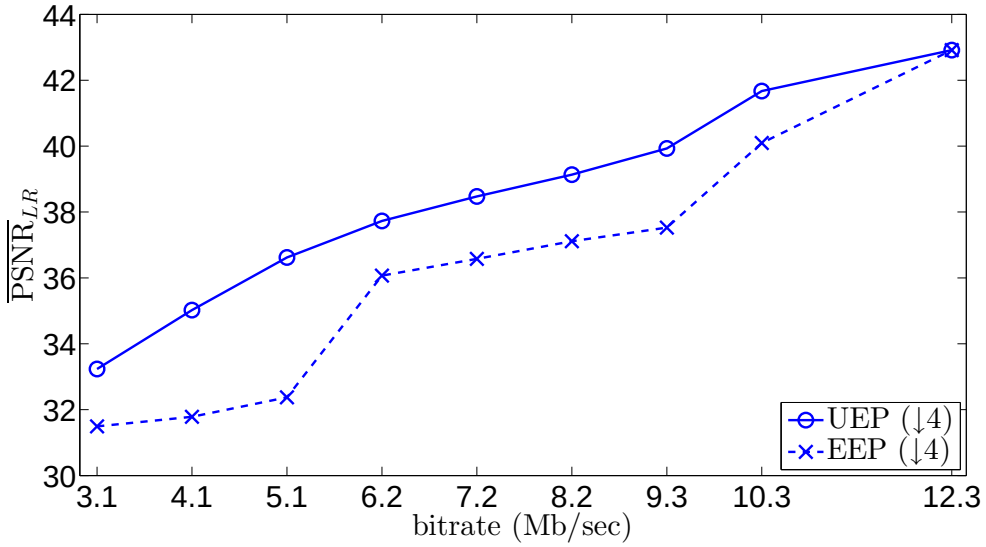


Figure 3.10: $\overline{\text{PSNR}}_{LR}$ of UEP and EEP for ↓4.

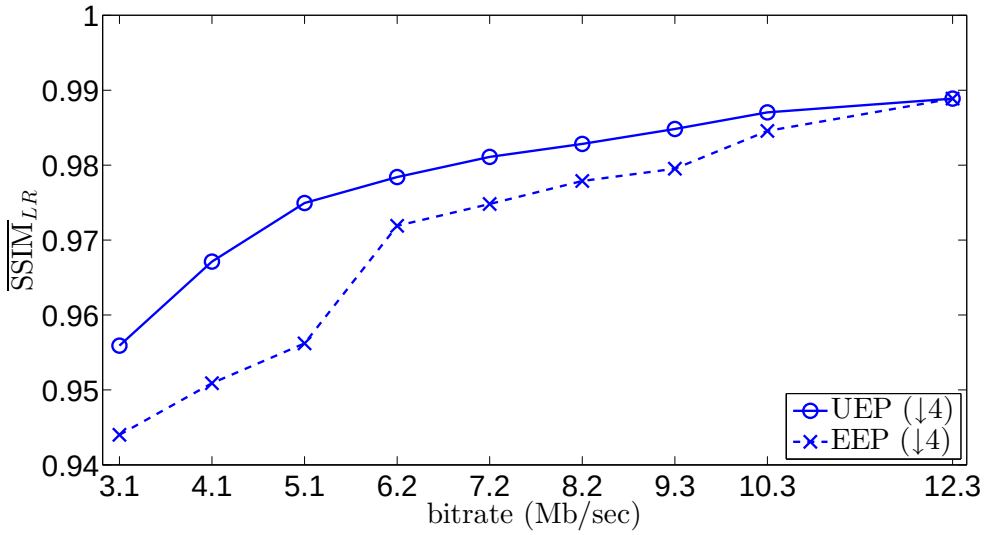


Figure 3.11: $\overline{\text{SSIM}}_{LR}$ of UEP and EEP for ↓4.

Chapter 4

UEP for Scalable Video Broadcasting over MIMO Channels

This chapter addresses UEP for video broadcasting over wireless channels. We assume that heterogeneous users with different display resolutions and different operating data rates are present in a service area of a base station. Our goal is to design a video broadcasting system that well serves all types of users within the service area. We tackle this problem for a MIMO (multi-input-multi-output) channel. We use spatial scalable video coding for video compression which readily enables us to provide UEP at the layer level. For MIMO communication, we propose to use spatial diversity techniques for base layer, and spatial multiplexing techniques for enhancement layer. The BL and EL bit streams are superposed in a way that the BL receives a stronger protection compared to the EL. We compare the performance of our proposed scheme with that of two baseline schemes which only adopt spatial diversity techniques. We show that our proposed scheme significantly outperforms both the baseline schemes. This chapter is organized as follows: The system model and technical preliminaries are provided in Section 4.1. In Section 4.2, we first present two baseline MIMO video broadcasting schemes and then introduce the proposed scheme. Section 4.3 provides simulation results and Section 4.4 concludes this chapter.

4.1 MIMO Preliminaries

We first introduce a mathematical description of a MIMO channel which describes the relationship between the transmitted and received signals. We assume that the signal bandwidth is less than the coherence bandwidth of the channel or, in other words, channel is frequency non-selective. This allows us to model the channel between a transmit and a receive antenna with a complex gain. Suppose N_t and N_r represent the number of transmit and receive antennas, respectively. We define the following parameters:

$h_{ij} \triangleq$ complex channel gain between the jth transmit antenna and the ith receive antenna;

$r_i \triangleq$ receive signal at the ith receive antenna;

$s_j \triangleq$ symbol transmitted from the jth transmit antenna;

$z_i \triangleq$ noise signal at the ith receive antenna.

Using the above notation, we can write the received signal as

$$r_i = \sum_{j=1}^{N_t} h_{ij} s_j + z_i, \qquad i = 1, \ldots, N_r. \tag{4.1}$$

We can write (4.1) in matrix form as

$$\mathbf{r} = \mathbf{H}\mathbf{s} + \mathbf{z}, \tag{4.2}$$

where

$$\mathbf{s} \triangleq [s_1, \ldots, s_{N_t}]^T, \tag{4.3}$$

$$\mathbf{r} \triangleq [r_1, \ldots, r_{N_r}]^T, \tag{4.4}$$

$$\mathbf{z} \triangleq [z_1, \ldots, z_{N_r}]^T, \text{ and} \tag{4.5}$$

$$\mathbf{H} \triangleq \begin{bmatrix} h_{11} & \dots & h_{1,N_t} \\ \vdots & \ddots & \vdots \\ h_{N_r,1} & \dots & h_{N_r,N_t} \end{bmatrix}. \tag{4.6}$$

The entries of $\mathbf{H}$ are modeled to be independent and identically distributed (i.i.d.) $\sim \mathcal{CN}(0,1)$, and they are assumed to be known at the receiver, but not known at the transmitter. It is assumed that $\mathbf{H}$ is constant over T symbol durations. $\mathbf{z}$ is a zero-mean complex white Gaussian noise vector such that $E\left\{\mathbf{z}(\mathbf{z}^*)^T\right\} = \sigma_z^2 \mathbf{I}_{N_r}$. SNR per symbol is defined as $\gamma_s \triangleq E\{|s_i|^2\}/\sigma_z^2$. Now, let $\mathbf{S} = [\mathbf{s}_1, \dots, \mathbf{s}_T]$ represent a space-time code that is obtained by concatenating T transmit vectors $\mathbf{s}_1, \dots, \mathbf{s}_T$. Note that the space-time code $\mathbf{S}$ consists of $N_t \times T$ symbols which are transmitted from N_t transmit antennas in T symbol durations. Also, let N_s denote the number of symbols which are packed in $\mathbf{S}$. The spatial multiplexing rate of $\mathbf{S}$ is defined as N_s/T.

4.2 Video Broadcasting over MIMO Channels

We consider a MIMO video broadcasting system which consists of a base station with two transmit antennas and two types of user devices: i) a big user with two receive antennas and a high-resolution screen, and ii) a small user with a single receive antenna and a low-resolution screen. In Section 4.2.1, we briefly describe a few benefits of using SVC for video broadcasting that motivated us to adopt SVC for video compression. Section 4.2.2 introduces hierarchical constellations which provide UEP for scalable video transmitted over noisy channels. We then introduce two baseline schemes in Sections 4.2.3 and 4.2.4. By baseline scheme we refer to a one in which video is broadcast only using spatial diversity techniques. Our proposed MIMO broadcasting scheme is detailed in Section 4.2.5.

4.2.1 SVC for Video Broadcasting

The use of scalable video coding is very beneficial in video broadcasting systems [9]. SVC lends itself to be very useful in a video communication scenario

where there exist heterogeneous users that operate at different data rates. In that scenario, we encode the video content using the desired number of enhancement layers and send the same scalable video bit stream to all the users. Each user can decode that portion of the scalable bit stream that fits its operating data rate. The use of SVC clearly obviates the need to encode the same video content at several different bit rates and send them to all the users. In addition, SVC can provide a graceful quality degradation in a video communication system if it is cleverly combined with unequal error protection. This can be done by unequally protecting the layers according to the contribution they make in enhancing the quality of the reconstructed video. For a two-layer bit stream, this is done by providing a stronger protection for the base layer and a weaker protection for the enhancement layer.

4.2.2 Hierarchical Constellations for UEP of SVC

Hierarchical constellations are power tools in providing UEP for transmission of a scalable video bit stream over an error-prone channel. Figure 4.1 shows an example of a hierarchical 64-QAM constellation. The 64 signal points are divided into four clusters (quadrants) and each cluster consists of 16 signal points. The minimum Euclidean distance between two clusters is d_M. Clusters represent the two most significant bits (MSBs) of a transmitted symbol. For example, the MSBs of any symbol transmitted from a cluster denoted by '00' are '00'. The four least significant bits (LSBs) determine which of the 16 signal points within a cluster is chosen, and their minimum distance is d_L. The distance ratio $\alpha = d_M/d_L (> 1)$ determines how much more the MSBs are protected compared to the LSBs against the channel errors. It is readily inferred that we provide a stronger protection for MSBs by increasing the value of α, since we make the distances between the clusters bigger by doing so. We note that, in bad channel conditions (e.g., deep fades), the decoder only needs to establish which cluster the received signal belongs to in order to determine the MSBs. On the other hand, in good channel conditions, the decoder not only can determine the cluster (i.e., MSBs) but it may also be able to recover the LSBs. The hierarchical 64-QAM constellation in Figure 4.1 (which is

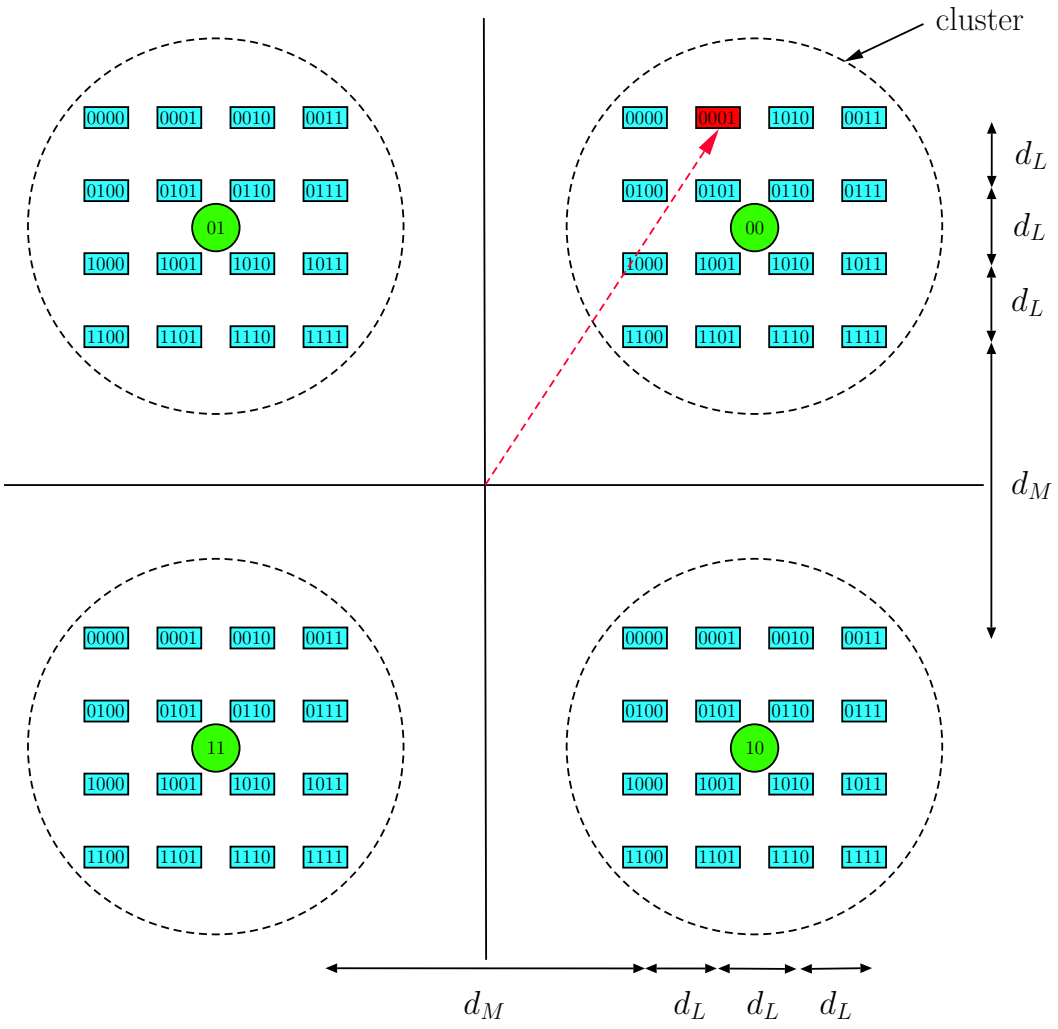


Figure 4.1: Hierarchical 4/64-QAM constellation. Six bits are transmitted per symbol. MSBs are displayed in green and LSBs are displayed in blue. Red arrow depicts the transmitted signal when the MSBs are '00' and the LSBs are '0001'.

denoted by 4/64-QAM) can be viewed as a superposition of quadrature phase shift keying (QPSK) and 16-QAM subconstellations. For low SNR, it operates as only a basic QPSK subconstellation having larger minimum distance. For high SNR, hierarchical 64-QAM can operate as both basic QPSK and secondary 16-QAM subconstellations.

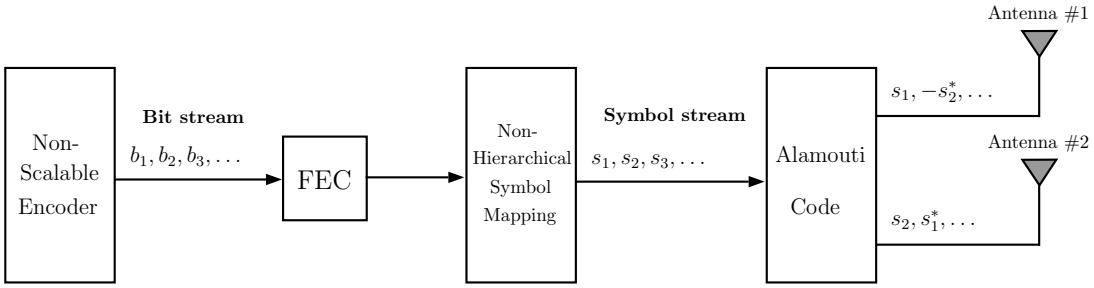


Figure 4.2: A baseline MIMO video broadcasting scheme with non-scalable video and non-hierarchical constellation.

4.2.3 Non-Scalable Baseline Scheme

Figure 4.2 depicts a baseline MIMO broadcasting scheme with non-scalable video coding. We refer to this baseline scheme as the *non-scalable baseline scheme* throughout this chapter. The system takes the compressed non-scalable bit stream and adds FEC. The coded bit stream is then mapped to constellation symbols. Note that a uniformly-spaced signal constellation (i.e., a non-hierarchical constellation) is employed for the non-scalable baseline scheme, and thus UEP is not provided for it. The constellation symbols are encoded by the Alamouti code and transmitted from two antennas.

4.2.4 Scalable Baseline Scheme

Figure 4.3 shows another baseline MIMO broadcasting scheme which employs spatial SVC and hierarchical constellations. We refer to this baseline scheme as the *scalable baseline scheme* throughout this chapter. The BL and EL bit streams are each transformed into a sequence of channel codewords. The two coded bit streams are then mapped to hierarchical constellation symbols. Note that the base layer is mapped to the MSBs of the hierarchical constellation, while the enhancement layer is mapped to the LSBs. This implies that the BL receives more protection compared to the EL. The hierarchical constellation symbols are encoded by the Alamouti code and transmitted from two antennas.

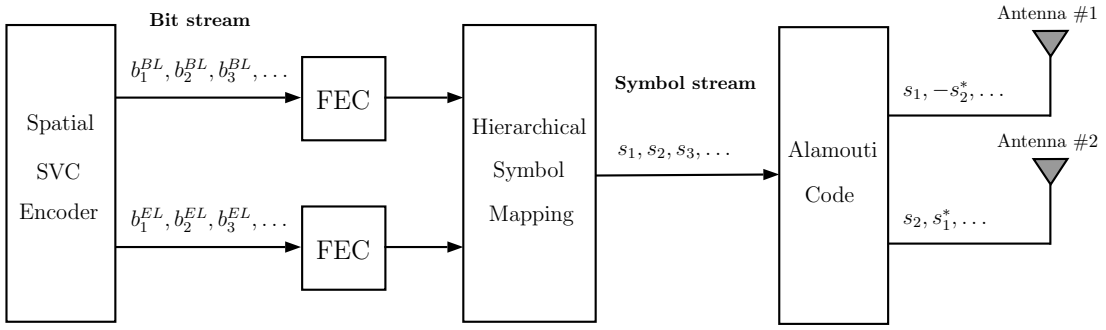


Figure 4.3: A baseline MIMO video broadcasting scheme with spatially scalable video and hierarchical constellation which provides UEP for scalable bit stream.

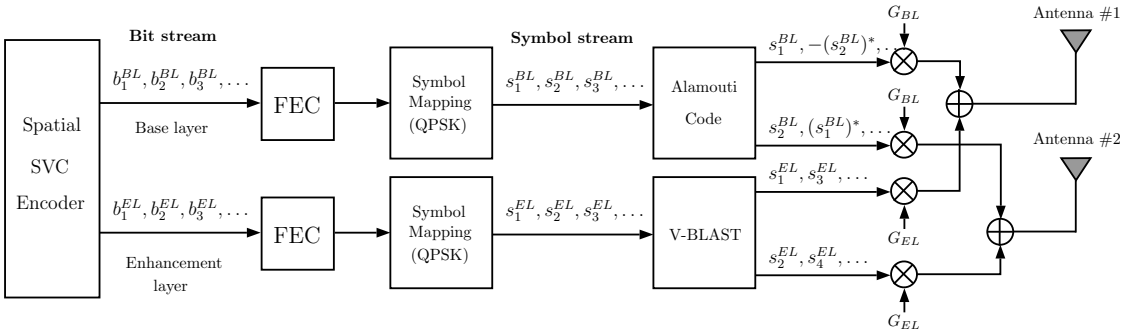


Figure 4.4: The proposed MIMO video broadcasting scheme with spatially scalable video and hierarchical constellation which provides UEP for scalable bit stream.

4.2.5 Proposed Scheme

Figure 4.4 shows a block diagram of our proposed MIMO video broadcasting scheme. The BL and EL bit streams are each transformed to a sequence of channel codewords. Each bit stream is mapped to a symbol bit stream using its own symbol constellation. The two different symbol streams are then coded using the space-time codes. We use Alamouti code to encode the BL symbol stream, and V-BLAST to encode the EL symbol stream. The resultant BL and EL symbols are multiplied by the transmit gains G_{BL} and G_{EL}, respectively, and superposed afterward. The transmit gain ratio $\beta = G_{BL}/G_{EL}$ (> 1) determines how much more the BL is protected compared to the EL. In Figure 4.4, $s_i^{BL}(i = 1, 2, 3, \ldots)$ denote the constellation symbols for the BL, and s_i^{EL} represent the constellation

symbols for the EL. The transmit matrix can be written as

$$\mathbf{G} = \begin{bmatrix} G_{BL}\, s_{2i-1}^{BL} + G_{EL}\, s_{4i-3}^{EL} & -G_{BL}\,(s_{2i}^{BL})^* + G_{EL}\, s_{4i-1}^{EL} \\ G_{BL}\, s_{2i}^{BL} + G_{EL}\, s_{4i-2}^{EL} & G_{BL}\,(s_{2i-1}^{BL})^* + G_{EL}\, s_{4i}^{EL} \end{bmatrix}, \tag{4.7}$$

where each row corresponds to a transmit antenna (spatial dimension) and each column corresponds to a symbol time (temporal dimension).

We note that the BL can be decoded by both the small user and the big user (the decoding procedure is elaborated in Section 4.3). However, the EL is only decodable by the big user. We recall that spatial multiplexing outperforms spatial diversity at high data rates. The fact that the EL has higher data rates compared to the BL motivated us to exploit spatial diversity techniques for BL and spatial multiplexing techniques for EL.

4.3 Simulation Results and Discussion

In this section, we evaluate the PSNR performance of the proposed MIMO video broadcasting scheme. For the proposed scheme depicted in Figure 4.4, we consider QPSK for BL, and 16-QAM for EL. During one symbol time duration, the proposed scheme transmits 2 bits of the BL and 8 bits of the EL. This is because the spatial multiplexing rate of the Alamouti code is equal to 1 (i.e., 1 QPSK symbol per symbol time), and that of V-BLAST is equal to 2 (i.e., 2 16-QAM symbols per symbol time).

We also evaluate the PSNR performances of the baseline schemes. For the scalable baseline scheme depicted in Figure 4.3, we use a hierarchical 4/1024-QAM constellation. A hierarchical 4/1024-QAM consists of a primary QPSK subconstellation and a secondary 256-QAM subconstellation. This implies that the scalable baseline scheme and the proposed scheme provide the same data rates. For the non-scalable baseline scheme depicted in Figure 4.2, we employ a 1024-QAM constellation. The data rate of this scheme (which corresponds to 10 bits per symbol time) is equal to that of the scalable baseline scheme and the proposed scheme.

For the baseline schemes, we use optimal maximum likelihood (ML) decod-

ing. For the proposed scheme, we use successive decoding [90] as is detailed in the following steps:

1. Alamouti decoding is performed on the received signal to decode the BL symbols;

2. The decoded BL symbols are subtracted from the received signal to obtain a residual signal;

3. MMSE (minimum-mean-squared-error) or ML decoding is performed on the residual signal in order to decode the EL symbols.

Although it has a suboptimal performance, successive decoding offers a much lower computational complexity compared to the ML decoding of the entire received signal. In step 3 above, we use ML decoding for V-BLAST.

We use H.264/SVC and evaluate the performance for video sequence 'Foreman'. The higher-resolution video reconstructed from decoding both the BL and EL has a resolution of 352×288, and the lower-resolution video reconstructed from decoding only the BL has a resolution of 176×144. We assume that the transmitted video signal experiences a slow fading channel such that the channel coefficients are nearly constant over a GOP. We use a B-frame hierarchical GOP structure, where each GOP has 16 frames. We also assume a perfect channel estimation at the receiver.

Figure 4.5 depicts the PSNR performance of a big user when the scalable baseline scheme is employed. We recall that a big user has a high-resolution screen. This means that when only the BL is decoded, a low-resolution video is reconstructed which needs to be upsampled in order to be displayed on a high-resolution screen. For a particular α in Figure 4.5, we observe that when SNR is low, PSNR reaches a plateau of about 30 dB. The reason is that, for low SNR values, only the BL is decodable, since it has been protected stronger compared to the EL. The low PSNR plateau of 30 dB is due to upsampling the BL which is needed for the big user. For a particular α, when channel SNR increases, the receiver gradually becomes able to decode the EL, and thus the quality of the decoded video gradually improves until it reaches the maximum PSNR value of

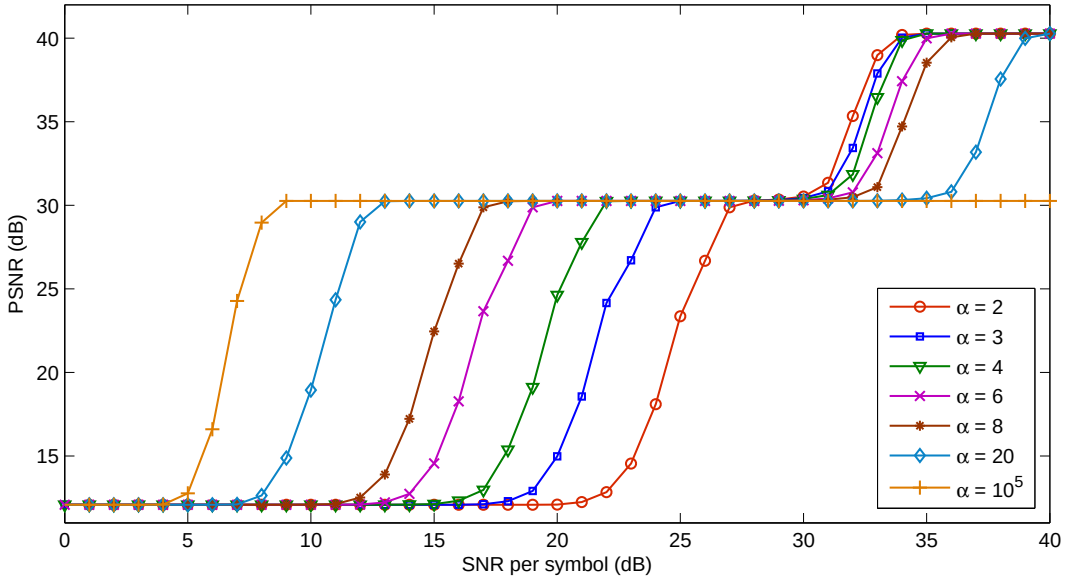


Figure 4.5: PSNR performance of a big user for the scalable baseline scheme.

about 40 dB. Another interesting observation is that, for a particular SNR value, when α increases, the performance in the range of low PSNRs improves. For a fixed channel SNR in that range of PSNRs, increasing α corresponds to providing a stronger protection for the BL, which leads to a better reconstruction quality. However, for a fixed channel SNR, when α increases, PSNR decreases in the range of high PSNRs. The reason is that a higher α translates to providing a weaker protection for the EL, which implies that we need a higher SNR to achieve the same reconstruction quality with a larger α.

Figure 4.6 depicts the PSNR performance of a big user when the MIMO broadcasting system employs the proposed scheme. The profile of the results is similar to the one presented in Figure 4.5. We see that as β increases, the performance in the range of low PSNRs improves, while the performance in the range of high PSNRs degrades.

Figures 4.5 and 4.6 indicate that the proposed scheme outperforms the scalable baseline scheme for a big user. Some of the curves in Figures 4.5 and 4.6 are plotted again in Figure 4.7 for better visual comparison. It is observed that in both low and high PSNR ranges, the proposed schemes with $\beta = 3, 3.5,$ and 4

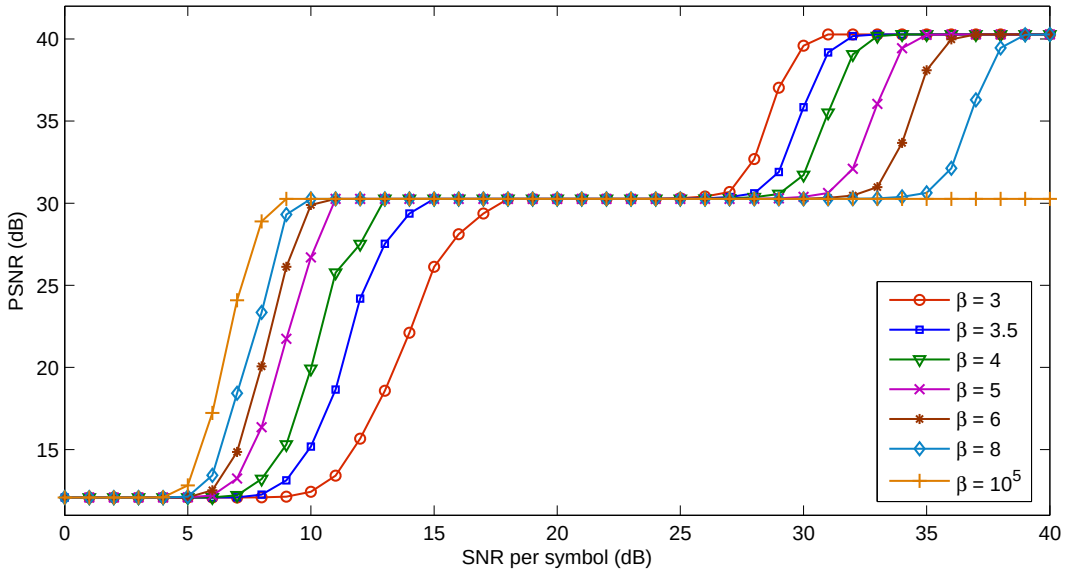


Figure 4.6: PSNR performance of a big user for the proposed scheme.

outperform the baseline schemes with $\alpha = 2, 4,$ and 8, respectively. We note that typical values of α are 2 and 4 [74]. Figure 4.7 also depicts the PSNR performance of the non-scalable baseline scheme. We see that the maximum PSNR the non-scalable baseline scheme can reach is slightly larger than the maximum PSNRs the others can reach. This is because the compression efficiency of non-scalable coding is slightly higher than that of the scalable coding. Disregarding this minor performance loss, we see that the proposed scheme significantly outperforms the non-scalable baseline scheme.

Figure 4.8 depicts the PSNR performance of a small user when we employ the scalable baseline scheme. We recall that a small user has a low-resolution screen. We also note that since we use the Alamouti code for both the BL and EL, a small user is able to decode both of them. In fact, it is possible that a small user achieves a better reconstruction quality by decoding both the BL and the EL and then downsampling the resulting full-resolution video, compared to the case where it decodes only the BL. We did extensive simulations with various video sequences, and concluded that if we let the small user decode the EL when at least 80% of the EL packets are received correctly, decoding the EL in conjunction with the BL is more beneficial than decoding the BL alone. We see in Figure 4.8 that a high

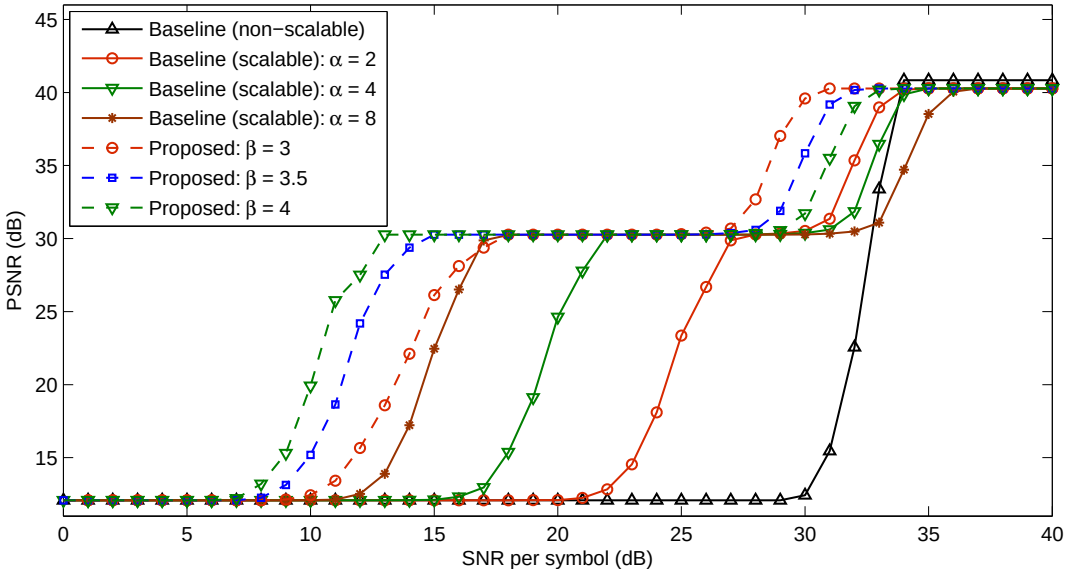


Figure 4.7: PSNR performance of a big user. The performance of the baseline scheme with non-scalable video coding is shown together with those of the proposed scheme and the baseline scheme with spatially scalable video coding and hierarchical constellation.

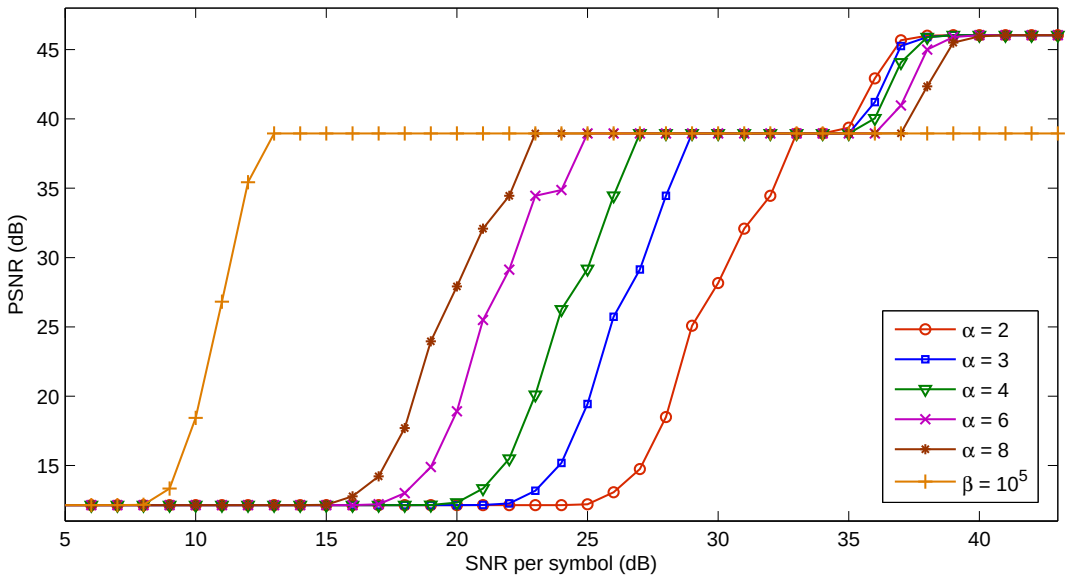


Figure 4.8: PSNR performance of a small user for the scalable baseline scheme.

enough PSNR (about 39 dB) is achieved even when the BL is decoded alone. We observe that when both the BL and EL are decoded, PSNR improves to about 46

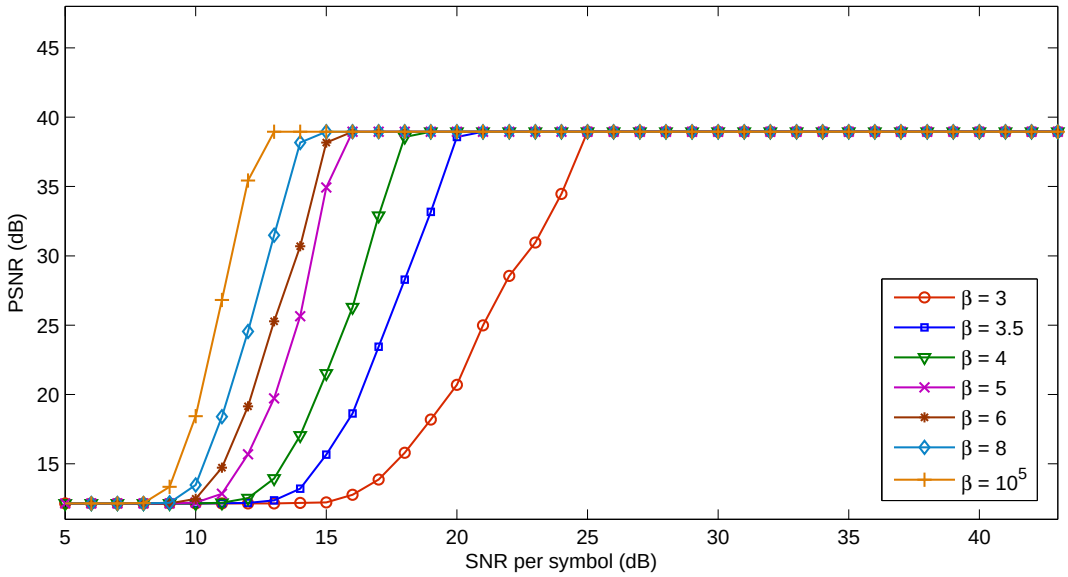


Figure 4.9: PSNR performance of a small user for the proposed scheme.

dB. We note that such high PSNR values are usually perceived identically by the human visual system (HVS), and thus a PSNR of 46 dB is not considered as a major performance improvement compared to the PSNR of 39 dB. We also observe that as α increases, the performance in the range of low PSNRs and high PSNRs improves and degrades, respectively.

Figure 4.9 depicts the PSNR performance of a small user when the proposed scheme is employed. We recall that the proposed scheme employs V-BLAST to encode the enhancement layer. We note that a small user with a single receive antenna cannot decode data that is encoded by V-BLAST. Thus, the small user is able to decode only the BL and it can achieve the maximum PSNR value of about 39 dB.

Some of the curves in Figures 4.8 and 4.9 are plotted again in Figure 4.10 for better visual comparison. For PSNRs lower than 40 dB, the proposed schemes with $\beta = 3$, 3.5, and 4 outperform the baseline schemes with $\alpha = 2$, 4, and 8, respectively. On the other hand, for PSNRs higher than 40 dB, the baseline scheme outperforms the proposed scheme. Such high PSNR values are usually perceived identically by the HVS, and thus they are not usually of interest. Figure 4.10 also

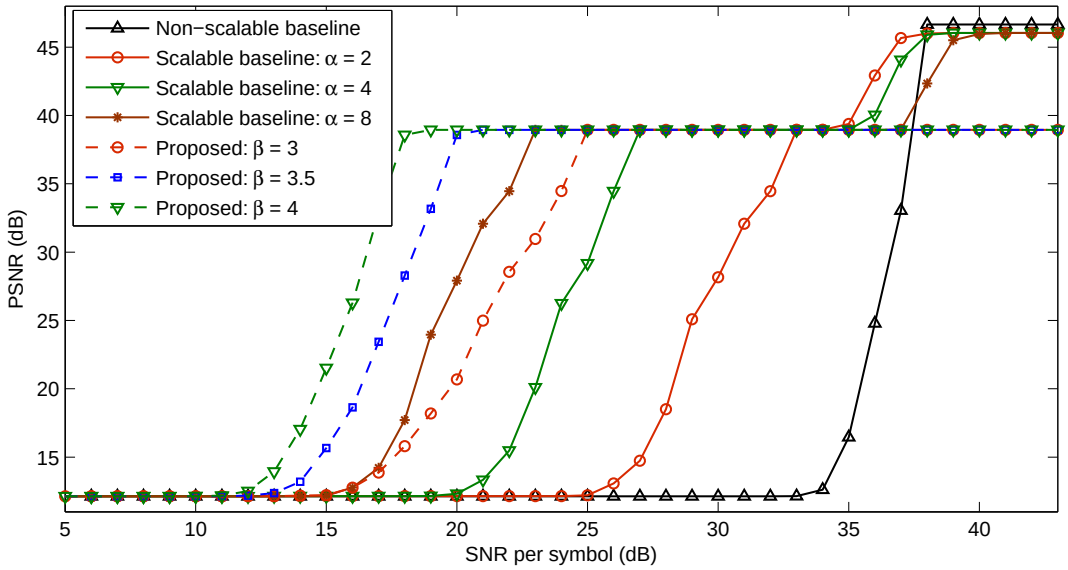


Figure 4.10: PSNR performance of a small user. The performance of the baseline scheme with non-scalable video coding is shown together with those of the proposed scheme and the baseline scheme with spatially scalable video coding and hierarchical constellation.

depicts the PSNR performance of the non-scalable baseline scheme. We see that the non-scalable baseline can provide a slightly higher maximum PSNR compared to the other schemes. However, the proposed scheme outperforms the non-scalable baseline scheme over the entire range of PSNRs we are interested in.

4.4 Conclusions

In this chapter, we considered UEP for video broadcasting over wireless channels. We assumed that two types of users are present within the service area of the transmitter: a user with a low-resolution screen and a user with a high-resolution screen. We used spatially scalable video that readily enables us to adopt UEP at the layer level. We proposed a UEP scheme for broadcasting of scalable video over MIMO channels. We employed spatial diversity techniques to encode the base layer and utilized spatial multiplexing to encode the enhancement layer. The BL and EL are then superposed in a way that BL receives more

protection compared to the EL. We compared the performance of our proposed scheme to that of two baseline schemes both of them exploiting only spatial diversity techniques. We showed that the proposed scheme significantly outperforms the baseline schemes for both types of users considered.

4.5 Acknowledgment

This work was partially supported by the Army Research Office under Grant #W911NF-14-1-0340, and by the Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (2013R1A1A2065143).

Chapter 4 of this dissertation is a reprint of the material as it appears in A. Vosoughi, S.-H. Chang, S.-H. Kim, P. Cosman, and L. Milstein, "Digital video broadcasting of spatially scalable video with multiple antennas", *manuscript under preparation*. I was the primary author and the co-authors Prof. Cosman and Prof. Milstein directed and supervised the research. The co-authors Dr. Chang and Dr. Kim also contributed to the ideas in this work and helped with the simulation process.

Chapter 5

Conclusions

In this dissertation, we have proposed UEP schemes for compressed 3D video. The UEP schemes are proposed within a framework of joint source-channel coding and are proposed for both MVC and V+D. We have also proposed unequal error protection for video broadcasting over the MIMO channels based on the hierarchal modulation and spatially scalable video.

In Chapter 2, we addressed the joint-source channel coding problem of a 3D video sent over AWGN and fading channels with the goal of minimizing the total number of transmitted bits while subject to video quality constraints. We considered non-scalable MVC, proposed a type of spatially scalable MVC, and addressed both symmetric and asymmetric coding. The UEP approach proposed here proved to be efficient at achieving this goal when compared to EEP for all the scenarios considered, where the average gains vary from 11.6% to 19.5%. Asymmetric coding was also compared to symmetric coding. Comparable gains were obtained for non-scalable and scalable MVC. The asymmetric/UEP gain over symmetric/UEP and symmetric/EEP vary, respectively, from 36.1% to 38.3% and from 45.0% to 47.1%. We also showed that, although using scalability leads to an overhead compared to non-scalable MVC, it may have an advantage in terms of the subjective quality of the received video, since most of the lost packets occur in the enhancement layer whose errors are less noticeable to the human visual system compared to the errors due to packets lost in the base layer.

In Chapter 3, we studied joint source-channel coding for video plus depth.

Full-resolution and downsampled depth by factors of two and four were considered. Results show that the depth can be significantly compressed compared to the color, although it needs to be protected more by FEC. We showed that when depth is downsampled, it should be less compressed and more protected to maximize the quality. In contrast to prior work which only considered equal quantization parameters and found that color should be more protected than depth, we found that depth should be compressed more severely than color and then protected more. We also showed that the downsampled depth by a factor of four outperforms the other scenarios except for high bitrates. The UEP approach proposed here was shown to yield up to 4.3dB gain in PSNR compared to EEP for flat Rayleigh fading channels.

In Chapter 4, we studied UEP for video broadcasting over wireless channels. We assumed that two types of users are present within the service area of a transmitter: a user with a low-resolution screen operating at a low data rate, and a user with a high-resolution screen operating at a high data rate. We used spatially scalable video for video compression. We proposed an efficient UEP scheme for scalable video broadcasting over wireless MIMO channels. We employed spatial diversity techniques (in particular the Alamouti codes) to encode the base layer and utilized spatial multiplexing techniques (in particular the V-BLAST) to encode the enhancement layer. The BL and EL are then superposed in a way that BL receives a stronger protection compared to the EL. We compared the performance of our proposed scheme to that of two baseline schemes both of them exploiting only spatial diversity techniques. We showed that the proposed scheme significantly outperforms the baseline schemes for both types of users considered.

Bibliography

[1] A. Gotchev, G. Akar, T. Capin, D. Strohmeier, and A. Boev, "Three-dimensional media for mobile devices," *Proceedings of the IEEE*, vol. 99, no. 4, pp. 708–741, 2011.

[2] A. Smolic, K. Mueller, P. Merkle, P. Kauff, and T. Wiegand, "An overview of available and emerging 3D video formats and depth enhanced stereo as efficient generic solution," in *Picture Coding Symposium, 2009. PCS 2009*, 2009, pp. 1–4.

[3] J. Williams and M. Bennamoun, "A non-linear filtering approach to image matching," in *Proc. International Conference on Pattern Recognition*, vol. 1, 1998, pp. 1–3.

[4] D. Scharstein and R. Szeliski, "A taxonomy and evaluation of dense two-frame stereo correspondence algorithms," *International Journal of Computer Vision*, vol. 47, pp. 7–42, 2002.

[5] C. Zitnick and T. Kanade, "A cooperative algorithm for stereo matching and occlusion detection," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 22, no. 7, pp. 675–684, 2000.

[6] J. Kack, "Robust stereo correspondence using graph cuts," Master's thesis, Royal Institute of Technology, 2004.

[7] D. Scharstein, *View Synthesis Using Stereo Vision*. Berlin, Heidelberg: Springer-Verlag, 1999.

[8] P. Hanhart and T. Ebrahimi, "Quality assessment of a stereo pair formed from decoded and synthesized views using objective metrics," in *3DTV-Conference: The True Vision-Capture, Transmission and Display of 3D Video (3DTV-CON), 2012*, Oct 2012, pp. 1–4.

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

[10] A. Aksay, C. Bilen, E. Kurutepe, T. Ozcelebi, G. B. Akar, M. R. Civanlar, and A. M. Tekalp, "Temporal and spatial scaling for stereoscopic video compression," in *Proceedings EUSIPCO*, vol. 6, no. 8, 2006.

[11] M. Drose, C. Clemens, and T. Sikora, "Extending single-view scalable video coding to multi-view based on H.264/AVC," in *ICIP*, 2006, pp. 2977–2980.

[12] N. Ozbek and A. Murat Tekalp, "Quality layers in scalable multi-view video coding," in *IEEE Int. Conf. on Multimedia and Expo*, 2009, pp. 185–188.

[13] M.-W. Park and G.-H. Park, "Realistic multi-view scalable video coding scheme," *IEEE Trans. on Consumer Electronics*, vol. 58, no. 2, pp. 535–543, 2012.

[14] Y. Chen, R. Zhang, and M. Karczewicz, "MVC based scalable codec enhancing frame-compatible stereoscopic video," in *IEEE Int. Conf. on Multimedia and Expo*, 2011.

[15] Y. Lei, S. Xiaowei, H. Chunping, G. Jichang, L. Sumei, and Z. Yuan, "An improved multiview stereo video FGS scalable scheme," in *3DTV Conference: The True Vision-Capture, Transmission and Display of 3D Video, 2009*, 2009.

[16] L. B. Stelmach and W. J. Tam, "Stereoscopic image coding: Effect of disparate image-quality in left-and right-eye views," *Signal Processing: Image Communication*, vol. 14, no. 1, pp. 111–117, 1998.

[17] L. B. Stelmach, W. J. Tam, D. V. Meegan, A. Vincent, and P. Corriveau, "Human perception of mismatched stereoscopic 3D inputs," in *ICIP*, vol. 1, 2000, pp. 5–8.

[18] H. Kalva, L. Christodoulou, L. M. Mayron, O. Marques, and B. Furht, "Design and evaluation of a 3D video system based on H.264 view coding," in *Proc. of the Int. Workshop on Network and Operating Systems Support for Digital Audio and Video*, 2006, p. 12.

[19] G. Saygili, C. Gurler, and A. M. Tekalp, "Quality assessment of asymmetric stereo video coding," in *ICIP*, 2010, pp. 4009–4012.

[20] G. Saygili, C. G. Gurler, and A. M. Tekalp, "3D display dependent quality evaluation and rate allocation using scalable video coding," in *ICIP*, 2009, pp. 717–720.

[21] B. Julesz, *Foundations of Cyclopean Perception.* Chicago: The University of Chicago Press, 1971.

[22] V. De Silva, H. Arachchi, E. Ekmekcioglu, A. Fernando, S. Dogan, A. Kondoz, and S. Savas, "Psycho-physical limits of interocular blur suppression and its application to asymmetric stereoscopic video delivery," in *Packet Video Workshop (PV), 2012 19th International*, May 2012, pp. 184–189.

[23] L. Stelmach, W. J. Tam, D. Meegan, and A. Vincent, "Stereo image quality: effects of mixed spatio-temporal resolution," *IEEE Trans. on Circuits and Systems for Video Tech.*, vol. 10, no. 2, pp. 188–193, 2000.

[24] W. A. IJsselsteijn, H. de Ridder, and J. Vliegen, "Subjective evaluation of stereoscopic images: effects of camera parameters and display duration," *IEEE Trans. on Circuits and Systems for Video Tech.*, vol. 10, no. 2, pp. 225–233, 2000.

[25] P. Seuntiens, L. Meesters, and W. Ijsselsteijn, "Perceived quality of compressed stereoscopic images: Effects of symmetric and asymmetric JPEG coding and camera separation," *ACM Trans. on Applied Perception*, vol. 3, no. 2, pp. 95–109, 2006.

[26] S. Yasakethu, W. Fernando, B. Kamolrat, and A. Kondoz, "Analyzing perceptual attributes of 3D video," *IEEE Trans. on Consumer Electronics*, vol. 55, no. 2, pp. 864–872, 2009.

[27] C. Fehn, P. Kauff, S. Cho, H. Kwon, N. Hur, and J. Kim, "Asymmetric coding of stereoscopic video for transmission over T-DMB," in *3DTV Conference*, 2007.

[28] A. Aksay, S. Pehlivan, E. Kurutepe, C. Bilen, T. Ozcelebi, G. B. Akar, M. R. Civanlar, and A. M. Tekalp, "End-to-end stereoscopic video streaming with content-adaptive rate and format control," *Signal Processing: Image Communication*, vol. 22, no. 2, pp. 157–168, 2007.

[29] Y. Chen, S. Liu, Y.-K. Wang, M. M. Hannuksela, H. Li, and M. Gabbouj, "Low-complexity asymmetric multiview video coding," in *IEEE Int. Conf. on Multimedia and Expo*, 2008, pp. 773–776.

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

[31] G. Saygili, C. G. Gurler, and A. M. Tekalp, "Evaluation of asymmetric stereo video coding and rate scaling for adaptive 3D video streaming," *IEEE Trans. on Broadcasting*, vol. 57, no. 2, pp. 593–601, 2011.

[32] J. Quan, M. M. Hannuksela, and H. Li, "Asymmetric spatial scalability in stereoscopic video coding," in *3DTV Conference: The True Vision-Capture, Transmission and Display of 3D Video*, 2011.

[33] N. Ozbek, A. M. Tekalp, and E. T. Tunali, "A new scalable multi-view video coding configuration for robust selective streaming of free-viewpoint tv," in *IEEE Int. Conf. on Multimedia and Expo*, 2007, pp. 1155–1158.

[34] E. Kurutepe, M. R. Civanlar, and A. M. Tekalp, "Client-driven selective streaming of multiview video for interactive 3dtv," *IEEE Trans. on Circuits and Systems for Video Tech.*, vol. 17, no. 11, pp. 1558–1565, 2007.

[35] H.-T. Quan, P. Callet, and M. Barkowsky, "Video quality assessment: from 2D to 3D- challenges and future trends," in *Int. Conf. on Image Proc. (ICIP)*. IEEE, Sept. 2010, pp. 4025–4028.

[36] J. Starch, J. Kilner, and A. Hilton, "Objective quality assessment in free-viewpoint video production," in *3DTV Conference: The True Vision - Capture, Transmission and Display of 3D Video*, May 2008, pp. 225–228.

[37] Y. Zhang, P. An, Y. Wu, and Z. Zhang, "A multiview video quality assessment method based on disparity and ssim," in *Int. Conf. Signal Processing (ICSP)*, Oct. 2010, pp. 1044–1047.

[38] Z. Zhu, Y. Wang, Y. Bai, and Q. Shi, "New metric for stereo video quality assessment," in *Symposium Photonics and Optoelectronics (SOPO)*, Aug. 2009, pp. 1–4.

[39] A. Vetro, T. Wiegand, and G. J. Sullivan, "Overview of the stereo and multiview video coding extensions of the H. 264/MPEG-4 AVC standard," *Proceedings of the IEEE*, vol. 99, no. 4, pp. 626–642, 2011.

[40] F. Zhai and A. Katsaggelos, "Joint source-channel video transmission," *Synthesis Lectures on Image, Video, and Multimedia Processing*, vol. 3, no. 1, pp. 1–136, 2007.

[41] M. Bystrom and J. W. Modestino, "Combined source-channel coding schemes for video transmission over an additive white Gaussian noise channel," *IEEE Journal on Selected Areas in Communications*, vol. 18, no. 6, pp. 880–890, 2000.

[42] J. Hagenauer, "Rate-compatible punctured convolutional codes (RCPC codes) and their applications," *IEEE Trans. on Communications*, vol. 36, no. 4, pp. 389–400, 1988.

[43] T. Stockhammer, M. M. Hannuksela, and T. Wiegand, "H.264/AVC in wireless environments," *IEEE Trans. on Circuits and Systems for Video Tech.*, vol. 13, no. 7, pp. 657–673, 2003.

[44] T.-L. Lin and P. Cosman, "Optimal RCPC channel rate allocation in AWGN channel for perceptual video quality using integer programming," in *Quality of Multimedia Exp., Int. Workshop on*, 2009, pp. 198–203.

[45] Y. Liu, Q. Huang, S. Ma, D. Zhao, and W. Gao, "RD-optimized interactive streaming of multiview video with multiple encodings," *Journal of Visual Communication and Image Representation*, vol. 21, no. 5, pp. 523–532, 2010.

[46] Y. Zhou, C. Hou, W. Xiang, and F. Wu, "Channel distortion modeling for multi-view video transmission over packet-switched networks," *IEEE Trans. on Circuits and Systems for Video Tech.*, vol. 21, no. 11, pp. 1679–1692, 2011.

[47] D. M. Bento and J. M. Monteiro, "A QoS solution for three-dimensional full-HD H.264/MVC video transmission over IP networks," in *Iberian Conf. on Info. Systems and Tech.*, 2012.

[48] B. W. Micallef and C. J. Debono, "An analysis on the effect of transmission errors in real-time H. 264-MVC Bit-streams," in *IEEE Mediterranean Electrotechnical Conf.*, 2010, pp. 1215–1220.

[49] Y. Su, A. Vetro, and A. Smolic, "Common conditions for multiview video coding. JVT-U211," 2006.

[50] 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 Applied Signal Proc.*, vol. 2009, pp. 7:1–7:14, 2008.

[51] Q. Huynh-Thu, P. Le Callet, and M. Barkowsky, "Video quality assessment: From 2D to 3D challenges and future trends," in *ICIP*, 2010, pp. 4025–4028.

[52] N. Ozbek, A. M. Tekalp, and E. T. Tunali, "Rate allocation between views in scalable stereo video coding using an objective stereo video quality measure," in *ICASSP*, vol. 1, 2007.

[53] C. T. E. R. Hewage, S. Worrall, S. Dogan, H. Kodikaraarachchi, and A. Kondoz, "Stereoscopic TV over IP," in *Visual Media Production, 2007. IETCVMP. 4th European Conference on*, Nov 2007, pp. 1–7.

[54] C. T. E. R. Hewage, Z. Ahmad, S. Worrall, S. Dogan, W. A. C. Fernando, and A. Kondoz, "Unequal Error Protection for backward compatible 3-D video transmission over WiMAX," in *Circuits and Systems, 2009. ISCAS 2009. IEEE International Symposium on*, May 2009, pp. 125–128.

[55] K. Klimaszewski, K. Wegner, and M. Domanski, "Influence of views and depth compression onto quality of synthesized view," ISO/IEC JTC1/SC29/WG11 MPEG2009/M16758, London, UK, 2009.

[56] T. Wiegand, G. Sullivan, G. Bjontegaard, and A. Luthra, "Overview of the H.264/AVC video coding standard," *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 13, no. 7, pp. 560–576, July 2003.

[57] Z. Wang, A. Bovik, H. Sheikh, and E. Simoncelli, "Image quality assessment: from error visibility to structural similarity," *Image Processing, IEEE Transactions on*, vol. 13, no. 4, pp. 600–612, April 2004.

[58] C. Berrou, A. Glavieux, and P. Thitimajshima, "Near Shannon limit error-correcting coding and decoding: Turbo-codes(1)," in *Communications, IEEE International Conference on*, vol. 2, May 1993, pp. 1064–1070.

[59] J. R. Hampton, *Introduction to MIMO Communications*. Cambridge University Press, 2013.

[60] S. M. Alamouti, "A simple transmit diversity technique for wireless communications," *IEEE J. Select. Areas Commun.*, vol. 16, no. 8, pp. 1451–1458, 1998.

[61] V. Tarokh, H. Jafarkhani, and A. R. Calderbank, "Space-time block codes from orthogonal designs," *IEEE Trans. Info. Theory*, vol. 45, no. 5, pp. 1456–1467, 1999.

[62] G. J. Foschini, "Layered space-time architecture for wireless communication in a fading environment when using multiple antennas," *Bell Labs. Tech. J.*, vol. 1, no. 2, pp. 41–59, 1996.

[63] R. Louie, M. McKay, and I. Collings, "Open-loop spatial multiplexing and diversity communications in ad hoc networks," *IEEE Trans. Inf. Theory*, vol. 57, pp. 317–344, 2011.

[64] B. Hassibi and B. Hochwald, "High-rate codes that are linear in space and time," *IEEE Trans. Inf. Theory*, vol. 48, pp. 1804–1824, 2002.

[65] R. W. H. Jr. and A. J. Paulraj, "Linear dispersion codes for MIMO systems based on frame theory," *IEEE Trans. Signal Process.*, vol. 50, pp. 2429–2441, 2002.

[66] J.-C. Belfiore, G. Rekaya, and E. Viterbo, "The golden code: A 2×2 full-rate space-time code with non-vanishing determinants," *IEEE Trans. Info. Theory*, vol. 51, pp. 1432–1436, 2005.

[67] F. Oggier, G. Rekaya, J.-C. Belfiore, and E. Viterbo, "Perfect space-time block codes," *IEEE Trans. Inf. Theory*, vol. 52, pp. 3885–3902, 2006.

[68] P. Elia, K. R. Kumar, S. A. Pawar, P. V. Kumar, and H.-F. Lu, "Explicit, minimum-delay space-time codes achieving the diversity-multiplexing gain tradeoff," *IEEE Trans. Inf. Theory*, vol. 52, pp. 3869–3884, 2006.

[69] B. A. Sethuraman, B. S. Rajan, and V. Shashidhar, "Full-diversity, high-rate space-time block codes from division algebras," *IEEE Trans. Inf. Theory*, vol. 49, pp. 2596–2616, 2003.

[70] T. Cover, "Broadcast channels," *IEEE Trans. Inform. Theory*, vol. 18, no. 1, pp. 2–14, 1972.

[71] E. van der Meulen, "A survey of multi-way channels in information theory: 1961-1976," *IEEE Trans. Inform. Theory*, vol. 23, no. 1, pp. 1–37, 1977.

[72] K. Ramchandran, A. Ortega, K. Uz, and M.Vetterli, "Multiresolution broadcast for digital hdtv using joint source/channel coding," *IEEE J. Select. Areas Commun.*, vol. 11, no. 1, pp. 6–23, 1993.

[73] A. R. Calderbank and N. Seshadri, "Multilevel codes for unequal error protection," *IEEE Trans. Inform. Theory*, vol. 39, no. 4, pp. 1234–1248, 1993.

[74] "Digital Video Broadcasting (DVB); Framing Structure, Channel Coding and Modulation for Digital Terrestrial Television," ETSI EN 300 744 V1.5.1, Nov. 2004.

[75] S.-H. Chang, P. C. Cosman, and L. B. Milstein, "Optimal transmission of progressive sources based on the error probability analysis of sm and ostbc," *IEEE Trans. Veh. Technol.*, vol. 63, pp. 94–106, 2014.

[76] H. Yang, "A road to future broadband wireless access: MIMO-OFDM based air interface," *IEEE Commun. Mag.*, vol. 43, pp. 53–60, 2005.

[77] Y. Wang, Z. Wu, and J. M. Boyce, "Modeling of transmission-loss-induced distortion in decoded video," *IEEE Trans on Circuits and Systems for Video Tech.*, vol. 16, no. 6, pp. 716–732, 2006.

[78] 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. on Circuits and Systems for Video Technology*, vol. 12, no. 6, pp. 511–523, 2002.

[79] Y.-Z. Huang and J. Apostolopoulos, "A joint packet selection/omission and FEC system for streaming video," in *ICASSP*, vol. 1, 2007, pp. 845–848.

[80] Y. zong Huang and J. Apostolopoulos, "Making packet erasures to improve quality of FEC-protected video," in *ICIP*, 2006, pp. 1685–1688.

[81] W.-Y. Kung, C.-S. Kim, and C.-C. Kuo, "Packet video transmission over wireless channels with adaptive channel rate allocation," *Journal of Visual Communication and Image Representation*, vol. 16, pp. 475–498, 2005.

[82] K. Stuhlmuller, N. Farber, M. Link, and B. Girod, "Analysis of video transmission over lossy channels," *Selected Areas in Communications, IEEE Journal on*, vol. 18, no. 6, pp. 1012–1032, 2000.

[83] W.-T. Tan and A. Zakhor, "Video multicast using layered FEC and scalable compression," *Circuits and Systems for Video Tech., IEEE Trans. on*, vol. 11, no. 3, pp. 373–386, 2001.

[84] T.-L. Lin and P. C. Cosman, "Efficient optimal RCPC code rate allocation with packet discarding for pre-encoded compressed video," *Signal Proc. Letters, IEEE*, vol. 17, no. 5, pp. 505–508, 2010.

[85] Y. Chen, K. Xie, F. Zhang, P. Pandit, and J. Boyce, "Frame loss error concealment for SVC," *Journal of Zhejiang University SCIENCE A*, vol. 7, no. 5, pp. 677–683, 2006.

[86] D. P. Bertsekas, *Nonlinear programming.* Athena Scientific, 1999.

[87] European Telecommunications Standards Institute, "Universal mobile telecommunications system (UMTS): Multiplexing and channel coding (FDD)," *3GPP TS 125.212 version 3.4.0*, pp. 14–20, Sept. 23 2000.

[88] A. Tikanmaki, A. Gotchev, A. Smolic, and K. Miller, "Quality assessment of 3D video in rate allocation experiments," in *Consumer Electronics, IEEE International Symposium on*, April 2008, pp. 1–4.

[89] G. Tech, A. Smolic, H. Brust, P. Merkle, K. Dix, Y. Wang, K. Muller, and T. Wiegand, "Optimization and comparison of coding algorithms for mobile 3DTV," in *3DTV Conference: The True Vision-Capture, Transmission and Display of 3D Video*, May 2009, pp. 1–4.

[90] S.-H. Chang, M. Rim, P. C. Cosman, and L. B. Milstein, "Superposition MIMO coding for the broadcast of layered sources," *IEEE Trans. Commun.*, vol. 59, no. 12, pp. 3240–3248, 2011.