RAMIN GHAZNAVI YOUVALARI
360-DEGREE PANORAMIC VIDEO CODING

Master of Science Thesis

ABSTRACT

RAMIN GHAZNAVI YOUVALARI: 360-Degree Panoramic Video Coding
Tampere University of Technology
Master of Science Thesis, 56 pages
August 2016
Master's Degree Programme in Information Technology
Major: Signal Processing
Examiners: Prof. Moncef Gabbouj
　　　　　Dr. Miska Hannuksela
　　　　　Dr. Alireza Aminlou
Keywords: Video Coding, Virtual Reality, Omnidirectional Video, Equirectangular Projection, Pseudo-Cylindrical Projection, Quality Assessment

Virtual reality (VR) creates an immersive experience of real world in virtual environment through computer interface. Due to the technological advancements in recent years, VR technology is growing very fast and as a result industrial usage of this technology is feasible nowadays. This technology is being used in many applications for example gaming, education, streaming live events, etc.

Since VR is visualizing the real world experience, the image or video content which is used must represent the whole 3D world characteristics. Omnidirectional images/videos demonstrate such characteristics and hence are used in VR applications. However, these contents are not suitable for conventional video coding standards, which use only 2D image/video format content. Accordingly, the omnidirectional content are projected onto a 2D image plane using cylindrical or pseudo-cylindrical projections.

In this work, coding methods for two types of projection formats that are popular among the VR contents are studied: Equirectangular panoramic projection and Pseudo-cylindrical panoramic projection. The equirectangular projection is the most commonly used format in VR applications due to its rectangular image plane and also wide support in software development environments. However, this projection stretches the nadir and zenith areas of the panorama and as a result contain a relatively large portion of redundant data in these areas. The redundant information causes extra bitrate and also higher encoding/decoding time. Regional down-sampling (RDS) methods are used in this work in order to decrease the extra bitrate caused by over-stretched polar areas. These methods are categorized into persistent regional down-sampling (P-RDS) and temporal regional down-sampling (T-RDS) methods. In the P-RDS method, the down-sampling is applied to all frames of the video, but in the T-RDS method, only inter frames are down-sampled and the intra

frames are coded in full resolution format in order to maintain the highest possible quality of these frames.

The pseudo-cylindrical projections map the 3D spherical domain to a non-rectangular 2D image plane in which the polar areas do not have redundant information. Therefore, the more realistic sample distribution of 3D world is achieved by using these projection formats. However, because of non-rectangular image plane format, pseudo-cylindrical panoramas are not favorable for image/video coding standards and as a result the compression performance is not efficient. Therefore, two methods are investigated for improving the intra-frame and inter-frame compression of these panorama formats. In the intra-frame coding method, border edges are smoothed by modifying the content of the image in non-effective picture area. In the inter-frame coding method, gaining the benefit of 360-degree property of the content, non-effective picture area of reference frames at the border is filled with the content of the effective picture area from the opposite border to improve the performance of motion compensation.

As a final contribution, the quality assessment methods in VR applications are studied. Since the VR content are mainly displayed in head mounted displays (HMDs) which use 3D coordinate system, measuring the quality of decoded image/video with conventional methods does not represent the quality fairly. In this work, spherical quality metrics are investigated for measuring the quality of the proposed coding methods of omnidirectional panoramas. Moreover, a novel spherical quality metric (USS-PSNR) is proposed for evaluating the quality of VR images/video.

PREFACE

The research work in this thesis has been carried out from February 2015 - June 2016, at Nokia Technologies in collaboration with Department of Signal Processing, Tampere University of Technology (TUT), Tampere, Finland.

First and foremost, I would like to express my deepest gratitude to my supervisor Prof. Moncef Gabbouj for providing the opportunity for me to conduct my thesis research and his guidance during this project.

My sincere acknowledgment goes to Dr. Miska Hannuksela from Nokia Technologies for his endless support, technical and academic guidance during this work.

I am also grateful to Dr. Alireza Aminlou not only for his excellent co-supervision of this work, but also helping me during the difficulties that I faced throughout this research work.

I would also like to thank my colleagues in Nokia Technologies, Emre Aksu, Jani Lainema, Alireza Zare, Kashyap Kammachi Sreedhar, Antti Hallapuro, Vinod Mala-malvadakital, Igor Curcio and Jari Hagqvist for their support and providing friendly office atmosphere.

Special thanks to my dearest friend Saber Kordestanchi for his endless support and friendship during my studies.

I have had very good and supporting friends whom I'd like to thank for all their help: Solmaz Hach, Masoud Malekzadeh, Mohammad Behgam, Saman Bahrampour, Sajjad Nouri, Pouria Hajiani and Sounak Bhattacharya.

And finally I deeply appreciate the support of my parents, my father Ali Ghazna-vi and my late mother Effat Heidarzadeh. The people who always supported every decision that I made in my life with their unbelievable kindness and respect.

Tampere, August 2016

Ramin Ghaznavi Youvalari

I dedicate this thesis to my mother Effat Heidarzadeh. (1956 - 2015)

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS AND SYMBOLS

2D	Two-Dimensional
3D	Three-Dimensional
AMVP	Advanced Motion Vector Prediction
BDBR	Bjøntegaard Delta Bit Rate
BR	Bit Rate
CABAC	Context Adaptive Binary Arithmetic Coding
CB	Coding Block
CODEC	Coding-Decoding
CTB	Coding Tree Block
CTU	Coding Tree Unit
CU	Coding Unit
DBF	De-Blocking Filter
DCT	Discrete Cosine Transform
DST	Discrete Sine Transform
FOV	Field Of View
H.265/HEVC	High Efficiency Video Coding
H.264/AVC	Advanced Video Coding
HMD	Head Mounted Display
ITU-T	International Telecommunication Union-Telecommunication standardization sector
JCT-VC	Joint Collaborative Team on Video Coding
MC	Motion Compensation
ME	Motion Estimation
MPEG	Moving Picture Experts Group
MSE	Mean Square Error
MV	Motion Vector
P-RDS	Persistent Regional Down-Sampling
PSNR	Peak Signal-to-Noise Ratio
PU	Prediction Unit
QP	Quantization Parameter
RA	Random Access
RD	Rate-Distortion
RDS	Regional Down-Sampling
SAO	Sample Adaptive Offset
T-RDS	Temporal Regional Down-Sampling
TU	Transform Unit

USS-PSNR	Uniformly Sampled Spherical Peak Signal-to-Noise Ratio
VCEG	Video Coding Experts Group
VQEG	Video Quality Experts Group
VR	Virtual Reality

1. INTRODUCTION

Virtual reality (VR) creates a virtual environment to visualize the real world medium. This technology allows user to have perception of presence by immersion and intuitive interaction through the computer interface. The emergence of this technology goes back to 1960s where the early computer interface created for various applications. However, due to the technological limitations, the so called VR technology was not able to be used in industrial practices [1]. Nowadays, the advancement of technology provided required infrastructure for developing the VR to be used in practice e.g. streaming live events, education, gaming, medical applications, etc.

The VR content is usually acquired by using multiple camera setup or a camera device with multiple lenses and image sensors to cover the whole 360-degree scene with high resolution and high frame rate. For example some of the content that is used in this work are captured using Nokia's virtual reality camera OZO [2], which consists of eight fisheye lenses with field of view of 195 degrees, each. This setup allows the system to record the whole 360-degree scene by stitching multiple views from the camera.

In order to bring the immersive experience to the end user, using stereoscopic panoramas with high resolution and high frame rate is an important factor. As a consequence, these requirements create challenges in storage and transmission of VR content. Therefore, using efficient compression algorithms to overcome such constraints is inevitable. For this purpose, several coding standards are available for compressing VR video sequences, such as Advanced Video Coding (H.264/AVC) [3] and High Efficiency Video Coding (H.265/HEVC) [4]. However, neither of these compression standards are targeting VR video content that demands the above-mentioned requirements. Hence, the need for efficient compression tools which can cope with these requirements is a critical factor in virtual reality applications.

This thesis aims to study novel compression methods for efficient coding of VR video content using existing compression standards. Along with the compression methods, the quality assessment metrics for VR applications are studied for measuring the coding distortions in this work.

Block diagram of a simple virtual reality system is illustrated in Figure 1.1. The figure consists of capturing, pre-processing, encoding, transmission, decoding and
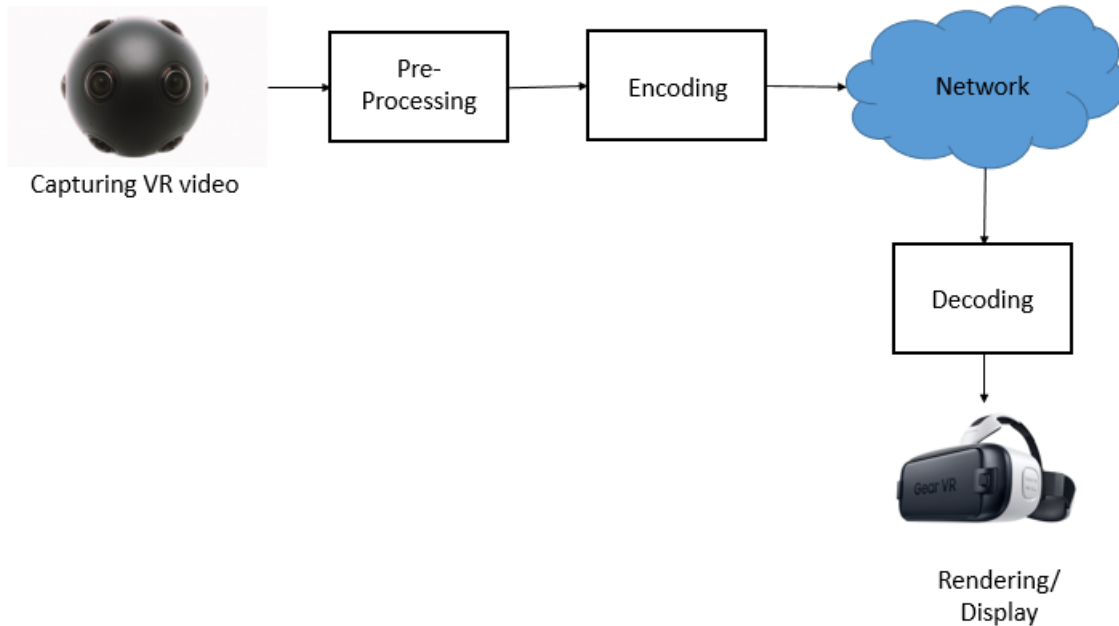
Figure *1.1 Block diagram of a virtual reality system*

displaying the video sequences.

- **Capturing:** the VR video capturing process includes multi-camera setup (e.g. Nokia's VR camera OZO [2]) in order to record the whole 360 degree scene in raw format.

- **Pre-processing:** the captured video content is pre-processed in this step prior to encoding operation. The process may include filtering, color correction, stitching, format conversion, etc.

- **Encoding:** compression operation on the pre-processed video is applied in this step for efficient storing or streaming purposes. The state of the art compression standards used in this process e.g. H.264/AVC and H.265/HEVC.

- **Transmission:** the compressed data is transmitted to the end user through the network to be consumed in the VR devices.

- **Decoding:** the end user receives the bitstream through the network on his/her device (e.g. mobile phone) and the transmitted video is decoded using the implemented decoder in the device.

- **Rendering/display:** the decoded video content is rendered in this step and displayed in the head mounted displays (e.g. Samsung gear VR [5]). The rendering and displaying process may include some post-processing operations prior to displaying e.g. post-filtering, stitching, re-sampling, etc.

1.1 Objectives and Scope of the Thesis

The conducted research work in this thesis was done in a collaboration of Tampere University of Technology (TUT) and Nokia Technologies. The goal of this thesis is to investigate the efficient compression algorithms for panoramic video content for VR applications. As mentioned earlier, the VR applications consume stereoscopic high quality panoramic contents with high frame rate and high resolution. Therefore, using the current coding standards for compressing these contents are not beneficial for VR applications.

The coding standards require the two-dimensional (2D) representation of 3D world to be able to compress the content. Therefore, various 2D projection formats exist to map the spherical coordinates onto 2D image plane. The common formats used in compression domain are equirectangular, pseudo-cylindrical, cube map, equal area projection, etc. In this work, we considered equirectangular and pseudo-cylindrical projection formats which are among the popular coding formats in VR applications. Compression algorithms are usually lossy and introduce some distortions to the coded video content. Measuring the coding distortions are also investigated in this work. Hence the VR content are displayed in HMDs which uses 3D coordinate system for visualization of the video, using conventional quality measurement will not represent the compression artifacts properly. Therefore, assessing the quality in spherical coordinate is investigated in this work and the proposed coding algorithms were analyzed by using various spherical quality metrics.

1.2 Thesis Outline

The rest of the thesis is organized as follows:

- **Chapter 2:** gives an overview of the H.265/HEVC standard. The core algorithms that are used in this standard are briefly described.

- **Chapter 3:** the regionally down-sampled methods for coding equirectangular panoramas are discussed in this chapter.

- **Chapter 4:** this chapter investigates the efficient compression algorithms for pseudo-cylindrical panoramas.

- **Chapter 5:** spherical quality assessment metrics are described in this chapter.

- **Chapter 6:** this chapter includes the experimental results for coding techniques that are discussed in previous chapters.

- **Chapter 7:** gives a conclusion and summary of the implemented methods and the potential future work.

2.   HIGH EFFICIENCY VIDEO CODING STANDARD OVERVIEW

This chapter provides a brief description of the coding algorithms used in High Efficiency Video Coding (H.265/HEVC) standard.

2.1   Introduction

The High Efficiency Video Coding (HEVC) standard was developed by Joint Collaborative Team on Video Coding (JCT-VC) of the ITU-T Video Coding Experts Group (VCEG) and the ISO/IEC Motion Picture Experts Group (MPEG) [6]. HEVC is capable of efficient coding of stereo or multiview video with high resolutions (e.g. 4k x 2k or 8k x 4k) compared to the previous standards such as Advanced Video Coding (AVC). Therefore, this codec is suitable for virtual reality applications which use stereoscopic videos with high resolutions and high frame rates [4] [7].

The HEVC standard uses hybrid coding approach similar as earlier standards such as H.264/AVC [3], but with higher compression gain, support for higher video resolutions, data loss resilience and ability of parallel processing. Figure 2.1, illustrates the block diagram of a typical hybrid video encoder such as HEVC, in which:

- I_n: Image/video to be encoded

- P_{inter}: Inter prediction

- P_{intra}: Intra prediction

- MS: Mode selection

- MEM: Reference frame memory

- F: Filtering

- T, T^{-1}: Transform and inverse transform
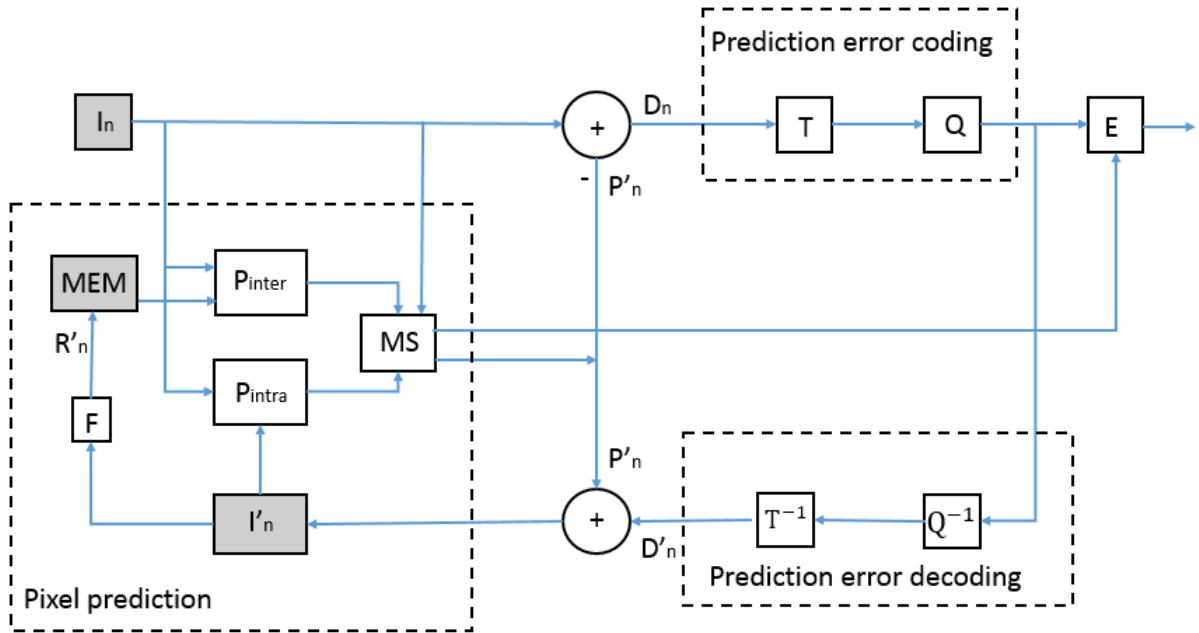
Figure   2.1 *Hybrid block diagram of an encoder*

- Q, Q^{-1}: Quantization and inverse quantization

- E: Entropy encoding

HEVC uses flexible structure of quad-tree partitioned coding tree unit (CTU), which consists of variable-sized coding units (CUs), prediction units (PUs) and transform units (TUs). The CTU has flexible size in which the encoder selects the size of it. Each CTU consists of coding tree block (CTB) for each component (luma and chroma). CTBs have flexible sizes (64x64, 32x32 or 16x16) as CTUs and the HEVC has the ability to partition them into smaller blocks. The size of coding blocks (CBs) is decided by the quadtree syntax of the CTU for each component. The coding unit (CU) is a combination of luma and chroma CBs together in which each CTB can contain one or multiple CUs. The partitioning can spread to CUs and prediction units (PUs) and transform units (TUs) result from the corresponding partitioning [8] [9]. An example of quad-tree partitioning in HEVC is illustrated in Figure 2.2 The partitioning of PUs which are basic units for prediction are shown in Figure 2.3 for intra-picture coding and inter-picture coding.

This hierarchical block structure of HEVC provides an efficient methods to code the different texture pattern of the image. Like other standards, the block based coding
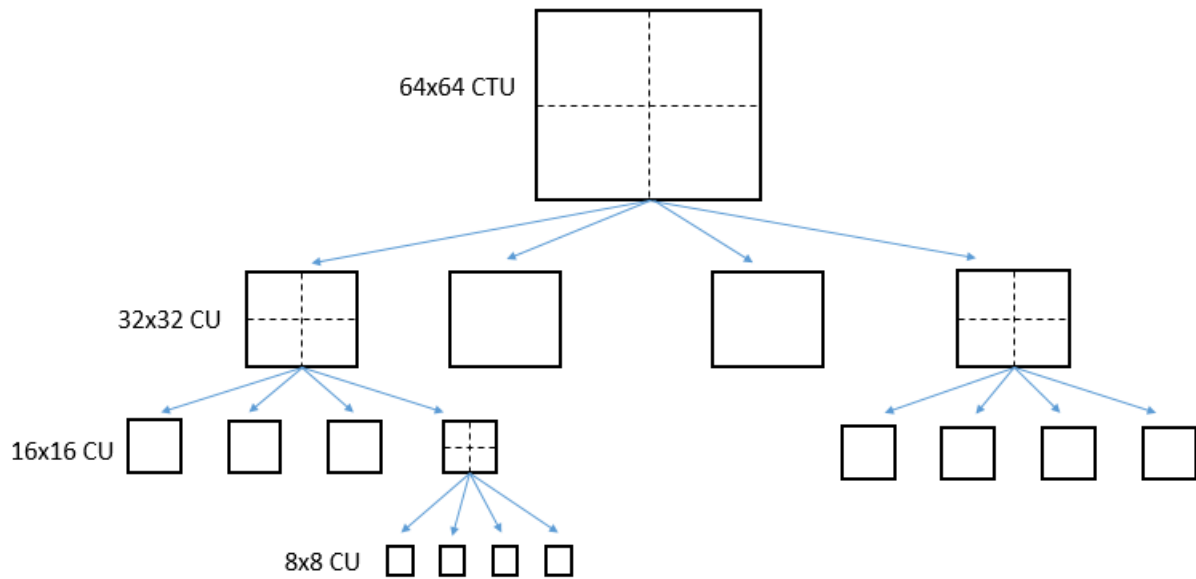
Figure 2.2 *Example of quad-tree splitting in HEVC*

(a)

(b)
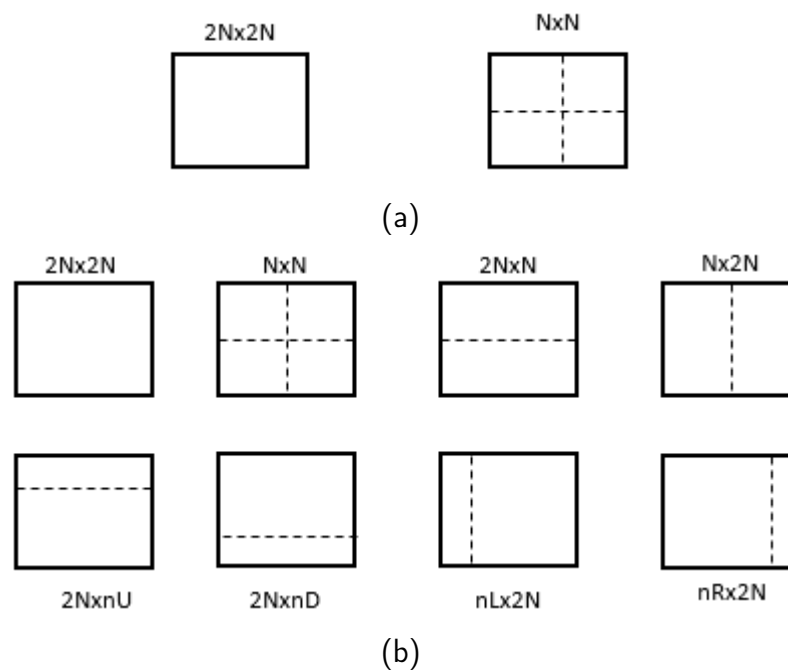
Figure 2.3 *PU partitioning structure in a) Intra and b) Inter prediction*

procedures applied include different phases:

- Spatial (Intra-picture) prediction

- Temporal (Inter-picture) prediction

- Transform

- Quantization

- Entropy Coding

- In-loop Filtering

The intra-picture prediction uses spatial prediction from the data within the same picture in order to predict the coding block. However, the inter-picture prediction includes predicting motion information from the reference picture(s) which are encoded beforehand.

2.2 Spatial Prediction

Spatial (a.k.a. intra-picture) prediction is applied to encode each frame separately from the others. The process is applied according to the transform block (TB) size in the picture and predicts the sample values of the block spatially by subtracting them from the neighboring TBs that are already encoded and reconstructed. The process includes DC, planar and directional predictions. Unlike H.264/AVC which has 8 modes for angular prediction, HEVC uses 33 directional prediction modes in order to predict the samples of the block efficiently. Figure 2.4 shows the coding block (predicted samples $P_{x,y}$) and the neighbor samples (reference samples $R_{x,y}$) that are used for intra-picture prediction process [10].

2.3 Temporal Prediction

Temporal or inter-picture prediction uses the temporally neighboring pictures of the video in order to compress the frames which have temporal redundancy. The process includes motion estimation from the reference picture and the motion vectors (MVs) used for sample prediction in each block. The process of choosing best match for motion information from reference frame in a particular search range from the co-located block position is called motion compensation (MC). The motion compensation uses quarter-sample precision for motion vectors, and 7-tap or 8-tap filtering for fractional-sample interpolation. Figure 2.5 demonstrates the block to be encoded in the current frame and the corresponding predicted block in the reference frame.

$R_{0,0}$	$R_{1,0}$	$R_{2,0}$	...	$R_{N,0}$	$R_{N+1,0}$	...	$R_{2N,0}$
$R_{0,1}$	$P_{1,1}$	$P_{2,1}$	...	$P_{N,1}$			
$R_{0,2}$	$P_{1,2}$	⋱		⋮			
⋮	⋮						
$R_{0,N}$	$P_{1,N}$	...		$P_{N,N}$			
$R_{0,N+1}$							
⋮							
$R_{0,2N}$							

Figure 2.4 *Intra prediction from neighbor samples in HEVC*
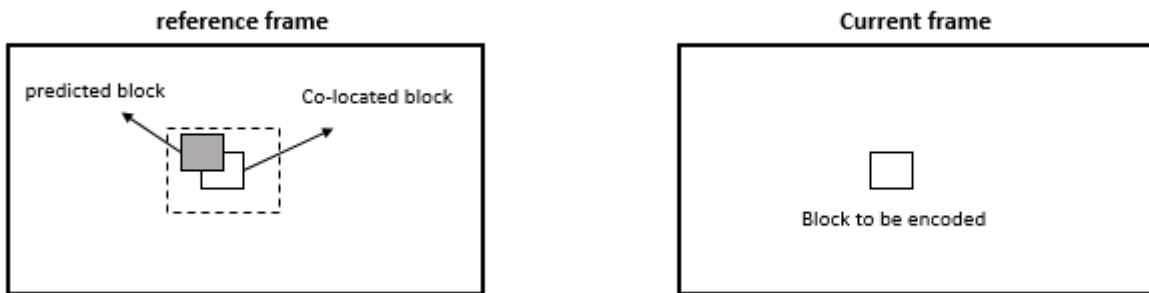
Figure 2.5 *Temporal prediction from reference picture in HEVC*

2.3.1 Motion Vector Prediction

For motion vector prediction, HEVC may use spatially adjacent vectors and/or motion vectors from reference pictures. The process includes certain motion vector candidates for both cases (spatial and temporal) [9]. Spatial and temporal motion vector candidates in HEVC are shown in Figure 2.6.

For the case of spatial candidates, only top and left neighboring blocks are used as reference considering the fact that the right and bottom blocks have not been decoded yet. Co-located block and the bottom-right block in the reference pictures used for predicting the motion information. The process of selecting spatial and temporal motion vectors for the coding block is known as advanced motion vector prediction (AMVP). In this step, among the motion vector candidates, as illustrated in Figure 2.7, two spatial MV and one temporal MV are derived for the final AMVP
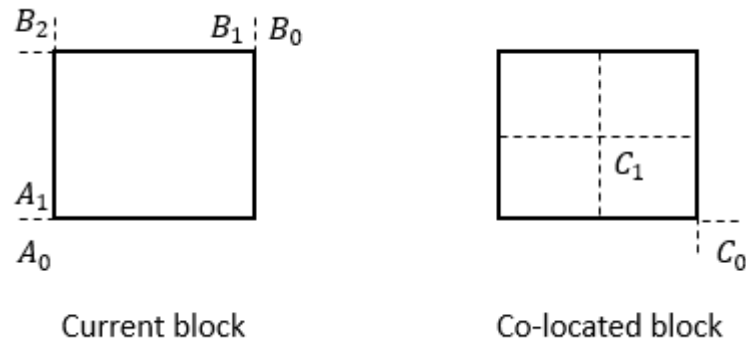
Figure *2.6* Spatial and temporal motion vector candidates
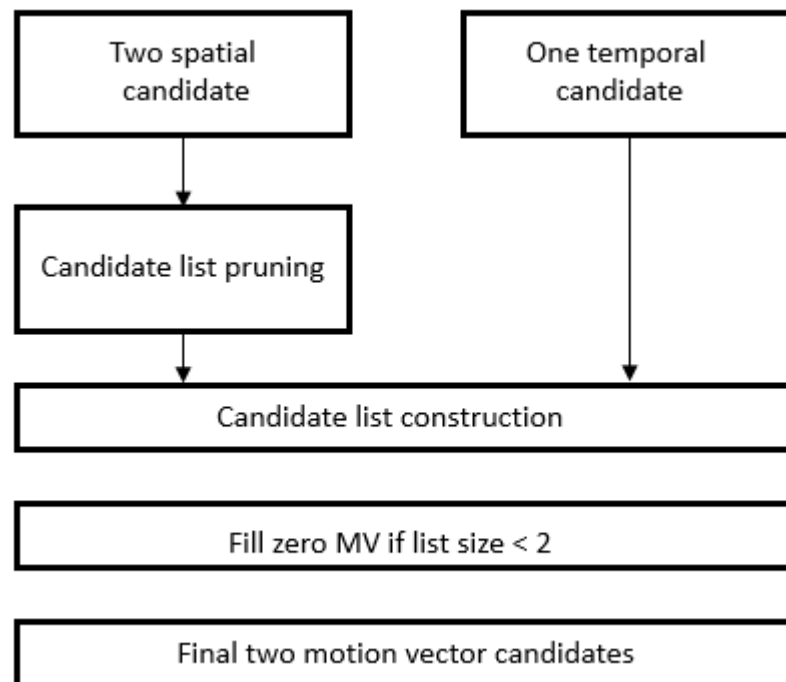
Figure *2.7* Motion vector selection process in HEVC

candidate list.

In the quad-tree structure of HEVC, each block is split to four child blocks and hence result in ineffective borders. In order to compensate this problem, HEVC uses block merging approach for coding the motion parameters efficiently. Merge candidate list contains all the motion information from the reference picture lists, a reference index and motion vector for each list. The merge candidate list is produced as following: up to four spatial merge candidates from the five spatial neighboring blocks; one candidate from the two temporal candidates in co-located blocks and two additional merge candidates from the bi-predictive candidates and zero MV candidates.

2.4 Transform and Quantization

The residual signals that are resultant of spatial and temporal prediction are highly correlated. In order to reduce this correlation of samples, HEVC uses spatial two-dimensional transform and quantization on the residual values in TUs before coding them with entropy encoder.

Two types of transform matrices used in HEVC:

- Discrete sine transform (DST) applied only on 4x4 luma residual blocks in intra-picture prediction.

- Discrete cosine transform (DCT) applied on the other residual blocks of luma and chroma components.

The above-mentioned transforms are applied as 1-D transforms in horizontal and vertical directions for each block.

The transform coefficients later quantized using quantization parameter (QP) by dividing them with the integer values of QP.

2.5 Entropy Coding

The core coding scheme in the HEVC is improved context adaptive binary arithmetic coding (CABAC) which is used in H.264/AVC standard. The evolved coding scheme enables high compression ratio, parallel processing due to the lower dependencies between coded data and lower context memory requirements in the codec [11].

2.6 In-loop Filtering

Two filtering operations are applied on the reconstructed frames before storing them in the reference frame memory. The purpose of filtering the reconstructed frames is to reduce the coding artifacts mainly caused by quantization and fractional sample interpolation processes. These artifacts can appear as blocking and ringing effects in the decoded video [13].

2.6.1 De-blocking Filter

The block-based coding scheme causes discontinuities in the reconstructed frame which appears as visible blocking artifacts in the frame. De-blocking filter is applied to the block boundary samples to reduces the resulted artifacts [12].

2.6.2 Sample Adaptive Offset

Sample adaptive offset (SAO) is a new in-loop filtering technique that is introduced in HEVC. The SAO filtering includes a categorization process for reconstructed samples in each region and obtaining an offset value for each category. The SAO filter reduces the mean sample distortion of each reconstructed region by adding the obtained offset values to the samples [13].

2.7 Decoding Process

The decoding process of a video in HEVC includes the encoder side algorithms in reverse order. Figure 2.8 illustrates the decoding procedure in HEVC.
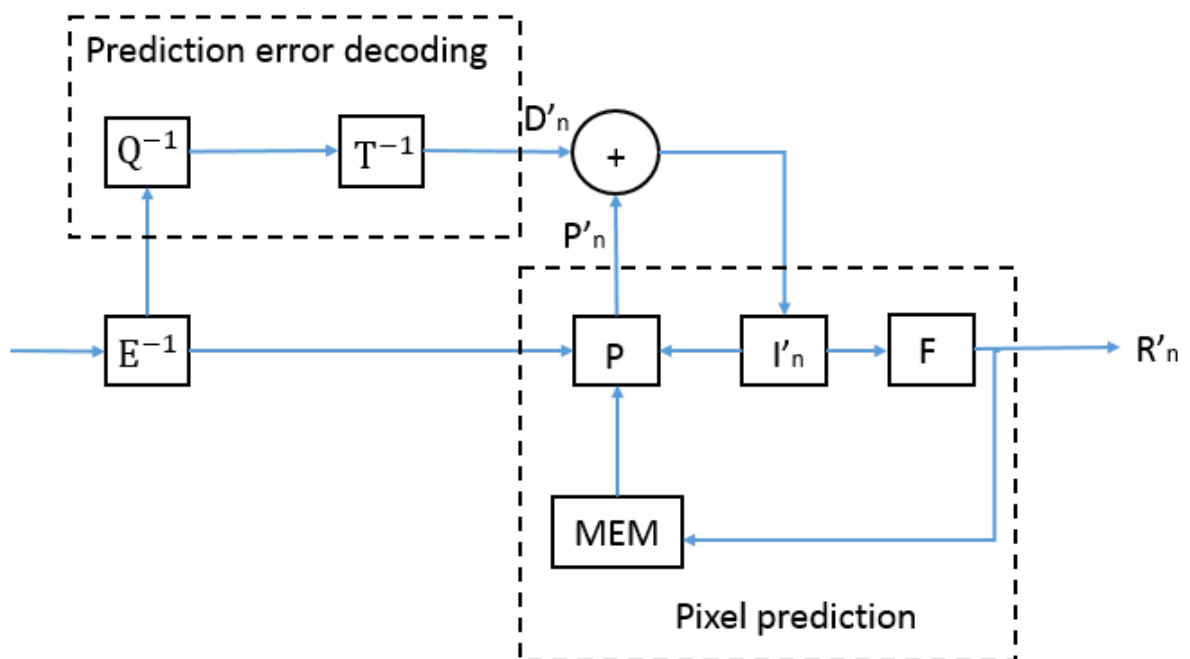
Figure **2.8** *Hybrid block diagram of a decoder*

3. REGIONAL DOWN-SAMPLING METHODS IN OMNIDIRECTIONAL VIDEO CODING

This chapter describes the regional down-sampling (RDS) methods in coding of omnidirectional video content. Section 3.1 discusses the problems in coding of equirectangular panoramas. The persistent regional down-sampling (P-RDS) method for compressing the equirectangular panoramas is described in section 3.2. The temporal regional down-sampling (T-RDS) method is proposed in section 3.3, in order to efficiently compress equirectangular panoramas in the cases where the P-RDS method fails to improve the rate-distortion (RD) performance compared to the conventional coding of equirectangular panoramas.

3.1 Introduction

Virtual reality (VR) applications have become popular in recent years, hence also increasing the importance of encoding and streaming video content for VR devices as efficiently as possible. In order to provide a full immersive experience, using 360-degree omnidirectional video content with high resolution and high frame rate is inevitable. For compressing the omnidirectional video clips, a projection onto a two-dimensional image plane is necessary. Panoramic images cover the whole 360-degree scene horizontally and up to 180-degree vertically around the capturing position and can be represented by a sphere that has been mapped onto a two dimensional image plane using a cylindrical projection. Coding of omnidirectional content using different projections are widely studied in the literature [14] [15] [16] [17] [18].
Among the cylindrical projections, the equirectangular projection is the most popular format for VR applications due to its ease of use and wide support in software development environments. The equirectangular projection maps the full 360-degree scene to a two-dimensional (2D) rectangular format, which is suitable for the current video coding standards, such as Advanced Video Coding (H.264/AVC) and High Efficiency Video Coding (H.265/HEVC). However, the problem with equirectangular panorama format is that it stretches the nadir and zenith areas of the spherical scene. Due to the stretching, the number of samples toward the nadir and zenith is proportionally greater compared to the equator areas. Consequently, the polar

areas contain a large number of redundant samples. Processing and encoding these extra samples result into a high bitrate and an increase of the encoding/decoding complexity of the codec.

3.2   Persistent Regional Down-Sampling Method (P-RDS)

In order to reduce the coding bitrate of equirectangular panoramas, it is beneficial to divide the panorama to multiple stripes and reduce the number of samples by down-sampling the polar stripes. As the redundant information is located in the nadir and zenith parts of an equirectangular image, the down-sampling ratio is higher in these areas. Moreover, since the samples are overstretched horizontally, the down-sampling is applied in the horizontal direction and the vertical pixel density is kept in the original resolution to avoid losing information. Figure 3.1 illustrates example divisions of an equirectangular panorama picture into multiple stripes and the corresponding regionally down-sampled version of that.

The resampling of stripes in equirectangular panoramas can improve the coding performance by decreasing the redundant samples in the nadir and zenith areas. However, in some sequences the down-sampling causes loss of information, i.e. smoothing, in the spherical domain. This smoothing also propagates in time due to inter prediction. Encoding the intra frames in the down-sampled format will decrease the bitrate significantly, but on the other hand, may result overall loss in rate-distortion (RD) performance due to the loss of information in the polar areas. The P-RDS method is suitable for coding still images where each image is coded independently. In the case of video content, the method may not be reliable due to the above-mentioned problem. Similar approach has been proposed in [19] [20]. As presented in the results achieved by [19], the method gives a significant compression gain for still images, but for the case of videos, the method may result into a compression loss depending on the content. Figure 3.2 illustrates the PSNR values of coding Lisboa sequence in conventional full resolution equirectangular format versus coding with P-RDS method. In the P-RDS method, the top and bottom parts of the equirectangular panorama with height of $\frac{1}{4}$ of the original height, down-sampled with ratio of 2. The random access configuration used in HEVC for the coding process. The PSNRs are calculated using USS-PSNR metric which will be introduced in chapter 5. The resulted luma PSNR differences are presented in Figure 3.3. As it can be observed from the figures, there is a big difference between conventional coding method and P-RDS method which can result in RD performance loss in overall.
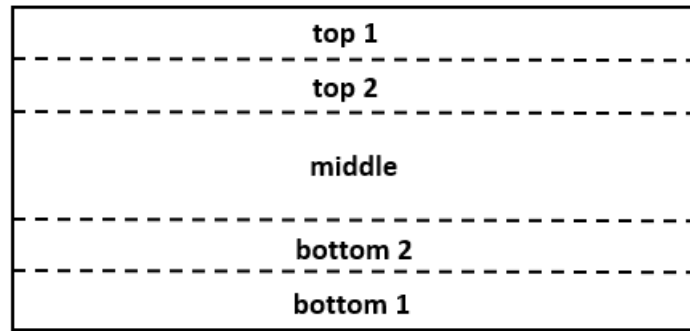
(a)

(b)
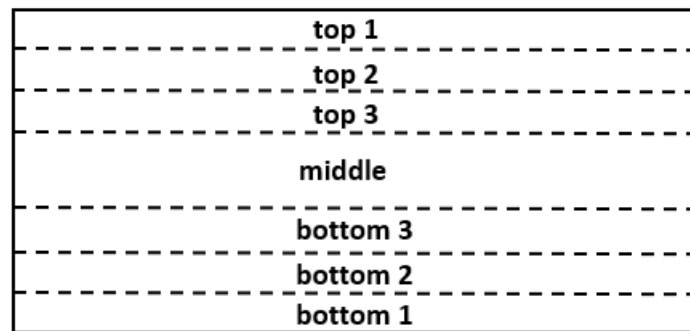
(c)
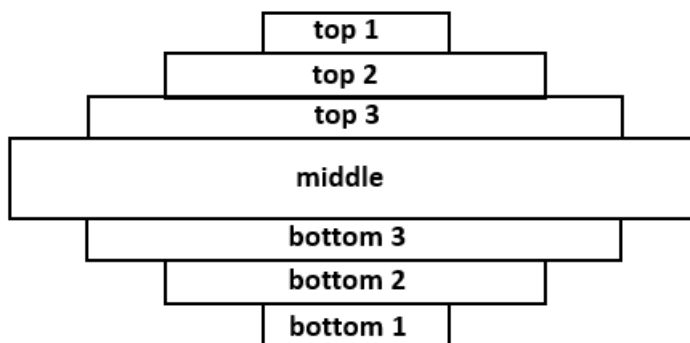
(d)

Figure 3.1 *Regionally down-sampled equirectungular panorama*
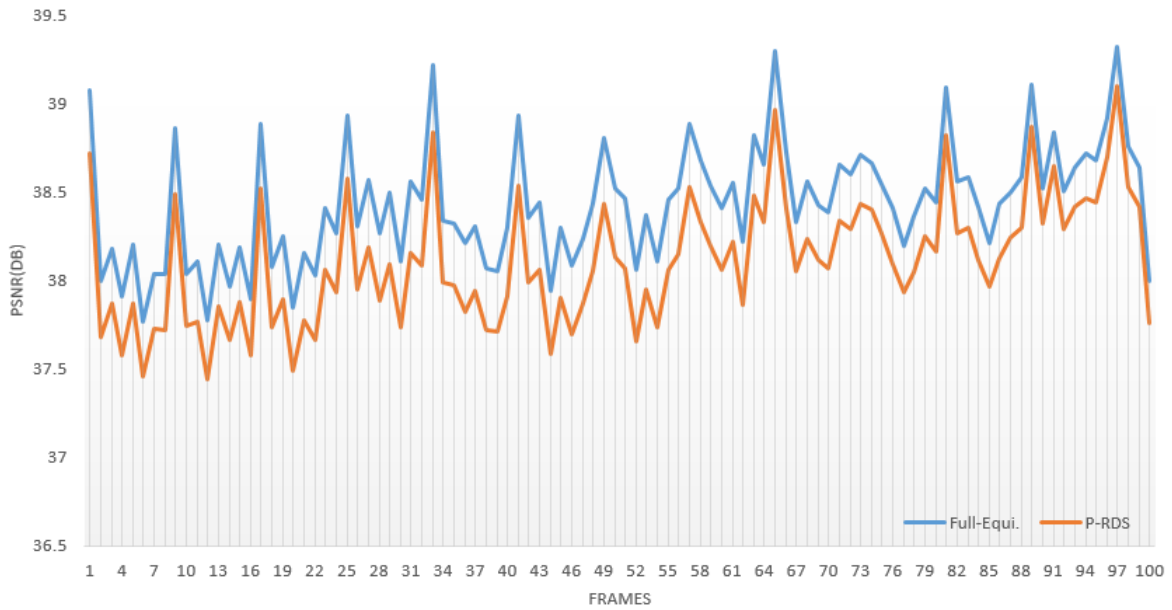
Figure **3.2** *PSNR values for Lisboa sequence*
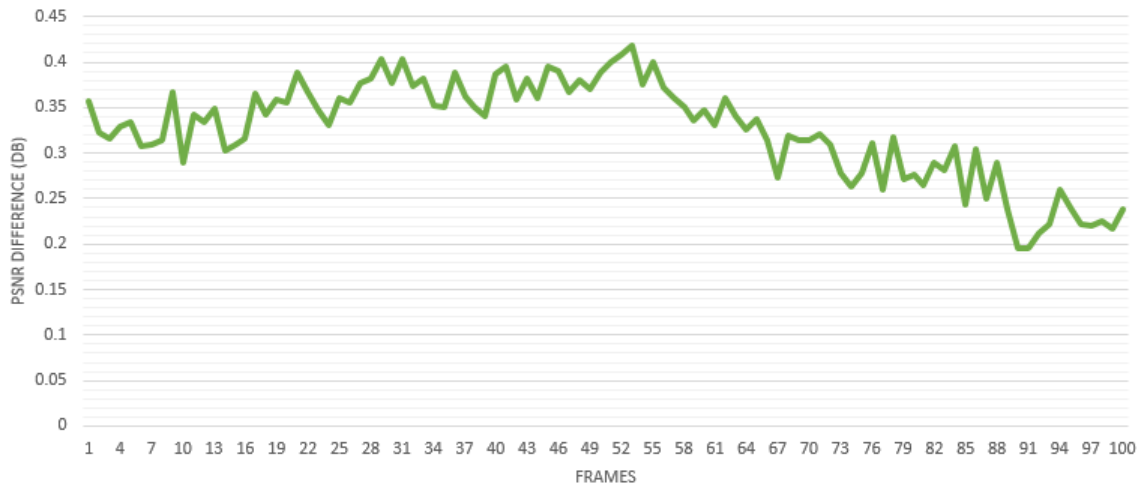
Figure **3.3** *PSNR difference for Lisboa sequence*

3.3   Temporal Regional Down-Sampling Method (T-RDS)

In this section, temporal regional down-sampling (T-RDS) method is described, in order to alleviate the compression loss of the P-RDS method which is discussed in section 3.2. Since the intra frames affect the prediction of inter frames, encoding intra frames in the conventional full resolution equirectangular format and applying
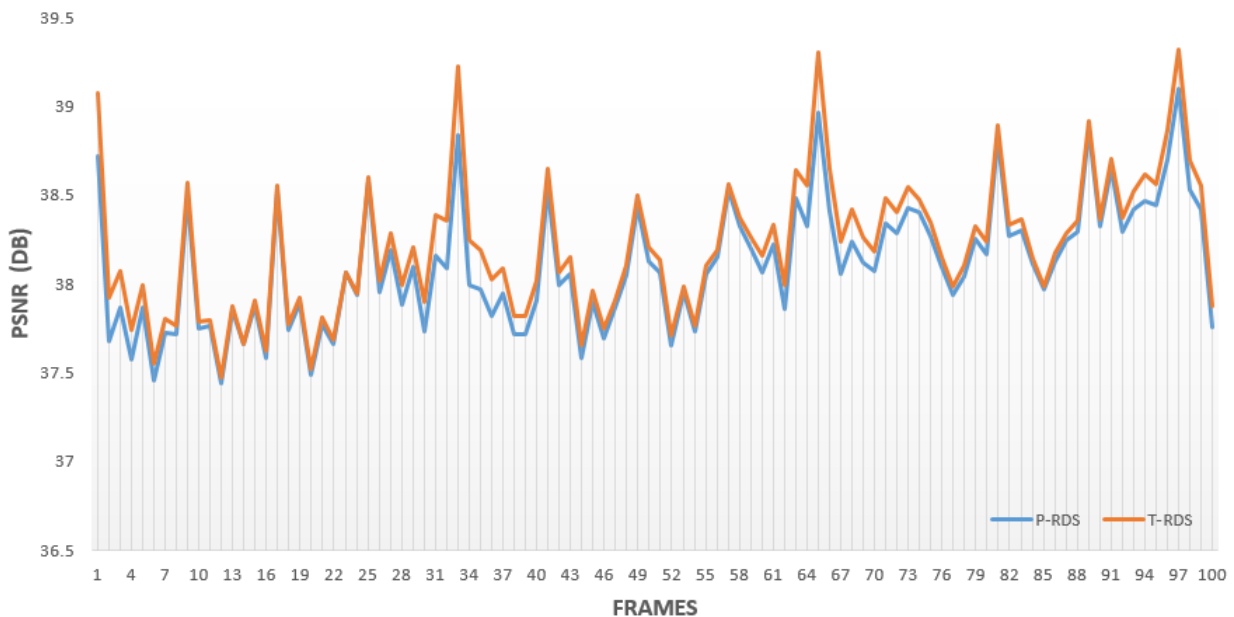
Figure 3.4 *PSNR values of T-RDS and P-RDS methods in Lisboa sequence*

regional down-sampling for only inter frames is proposed. This technique can boost the quality of the encoded frames and as a result the overall performance will increase.

The resulted PSNR values for Lisboa sequence using P-RDS and T-RDS methods are illustrated in Figure 3.4. As it can be observed from the figure, highest PSNR differences are in the intra frames (frame numbers 1, 33, 65 and 97). Figure 3.5 shows the resulted differences in the PSNR values for each frame. The figure also illustrates how the poor quality of intra frames can propagate in time in inter frames and cause poor quality for these frames as well.

3.3.1 Encoding Process in T-RDS Method

Persistent RDS method does not require changing the coding standard since the down-sampling process applies to all of the frames in a sequence and hence can be done as a pre-processing step before feeding the video to the codec. But the temporal RDS methods requires some high level modifications in the coding standard in order to make it suitable for the method.

The encoding algorithm includes two additional steps compared to conventional encoder:
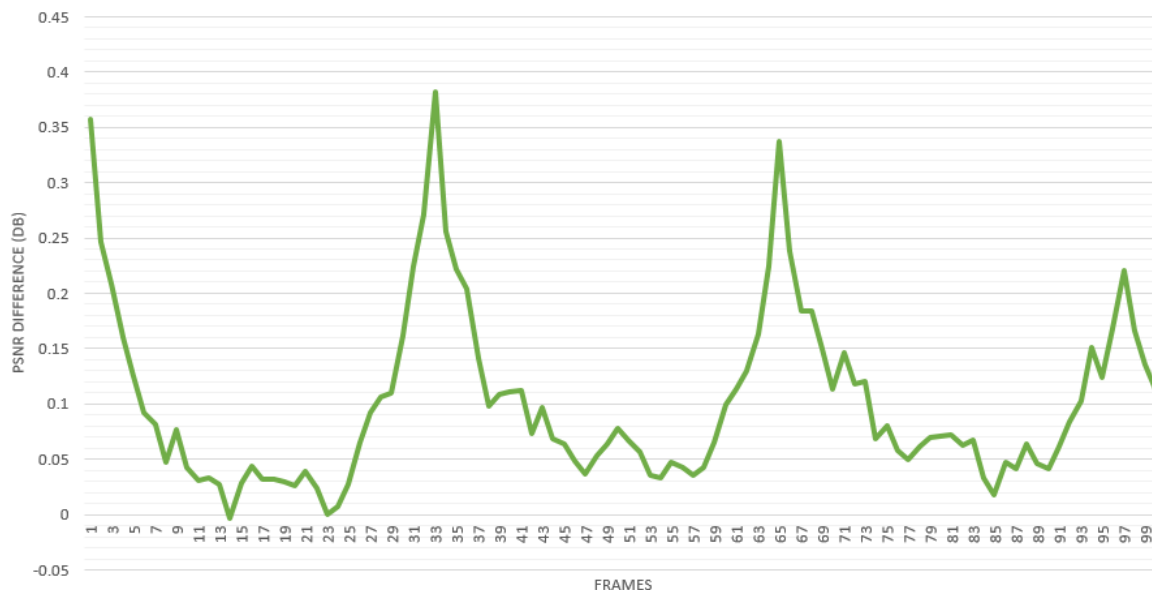
Figure 3.5 *PSNR difference between T-RDS and P-RDS methods in Lisboa sequence*

- Reference frame manipulation (which is only applied to intra frames) before storing in the reference frame memory.

- Inter frame manipulation prior to encoding.

Figure 3.6.a demonstrates the encoding algorithm for coding equirectangular panoramas with T-RDS method. As it can be observed, the video sequence fed to the codec without any pre-processing step in full resolution equirectangular format. The encoder encodes the intra frame which is presented as uncompressed picture U_0 in the figure. After reconstruction of intra frame in the encoder, the regional down-sampling process is applied to the preliminary reconstructed picture. The down-sampled regions in the picture are then relocated to form a reference frame (reconstructed picture R_0), which is stored in the decoded picture buffer and used subsequently as a reference for inter prediction.

For the case of inter frames (uncompressed picture U_n, n>0), the encoder applies the regional down-sampling and relocating process before encoding these frames as illustrated in Figure 3.6.a. The processed inter frames are then encoded in the RDS format. The reconstructed inter frames R_n (n>0) are stored in the decoded picture buffer without any resampling or relocating of stripes.

3.3.2   Decoding Process in T-RDS Method

The decoding process for the method is demonstrated in Figure 3.6.b. The process decodes the intra frames which have been encoded in full resolution equirectangular format. Then the decoder applies the same reference frame manipulation as performed in the encoder side in order to create the same picture format as that used in inter frames. The resampled and relocated intra frame (reconstructed picture R_0) is then stored in the decoded picture buffer and used as reference frame for inter prediction process. The inter frames are decoded without any extra pre-processing or post-processing step.

Some coding systems e.g. H.265/HEVC or H.264/AVC require the height of the coded pictures be identical throughout the coded video sequence and also the same as the height of the stored reference pictures. Hence, an empty stripe is included in the manipulated pictures to create the same picture format as the original video.

uncompressed picture U_0

top
middle
bottom

intra picture encoding

preliminary reconstructed picture

top
middle
bottom

regional resampling/ relocating

reconstructed picture R_0

top	bottom
middle	
Empty stripe	

uncompressed picture U_n, n > 0

top
middle
bottom

regional resampling/ relocating

processed uncompressed picture V_n, n > 0

top	bottom
middle	
Empty stripe	

reference picture(s) R_m, m < n

inter picture encoding

reconstructed picture R_n, n > 0

top	bottom
middle	
Empty stripe	

decoded picture buffer

(a)

bitstream

intra picture decoding

preliminary reconstructed picture

top
middle
bottom

regional resampling/ relocating

reconstructed picture R_0

top	bottom
middle	
Empty stripe	

reference picture(s) R_m, m < n

inter picture decoding

top	bottom
middle	
Empty stripe	

reconstructed picture R_n, n > 0

decoded picture buffer

(b)

Figure 3.6 *a) Encoding and b) Decoding algorithm, of temporal RDS method*

4. PSEUDO-CYLINDRICAL PANORAMIC VIDEO CODING

This chapter discusses compression methods for Pseudo-Cylindrical panoramas. The chapter includes coding problems in this picture format and the proposed solutions in intra-frame and inter-frame coding, which are discussed throughout the sections.

4.1 Introduction

Immersive virtual reality (VR) and its applications are growing very fast in the recent years. Hence, the need for wide field of view contents that can cover the whole 360-degree scene and make the interaction in this virtual environment feasible is an important factor. Panoramic content can be represented by a sphere that has been mapped to a two dimensional (2D) image plane using cylindrical or pseudo-cylindrical projections. In the case of cylindrical projections, as it has been discussed in chapter 3, where the spherical coordinates are mapped to the full rectangular 2D coordinates, the resulted image suffers from over stretching specially in the polar areas. Although cylindrical projections maintain the rectangular image format which is suitable for standard video codecs such as High Efficiency Video Coding (H.265/HEVC) and Advanced Video Coding (H.264/AVC), but projected images contain redundant information due to the over stretching.

A family of pseudo-cylindrical projections attempts to minimize the distortion of the polar areas of the cylindrical projections such as equirectangular projections, by bending the meridians toward the center of the map as a function of longitude while maintaining the cylindrical characteristics by preserving the parallel latitude lines, parallel [21][22]. These projections approximate equidistant sampling of 360-degree scene (which can be represented by 3D spherical coordinates). Hence, the pixel density is roughly equal regardless of the position on the sphere, providing spatially stable quality without the need of processing an excessive amount of pixels in compression.
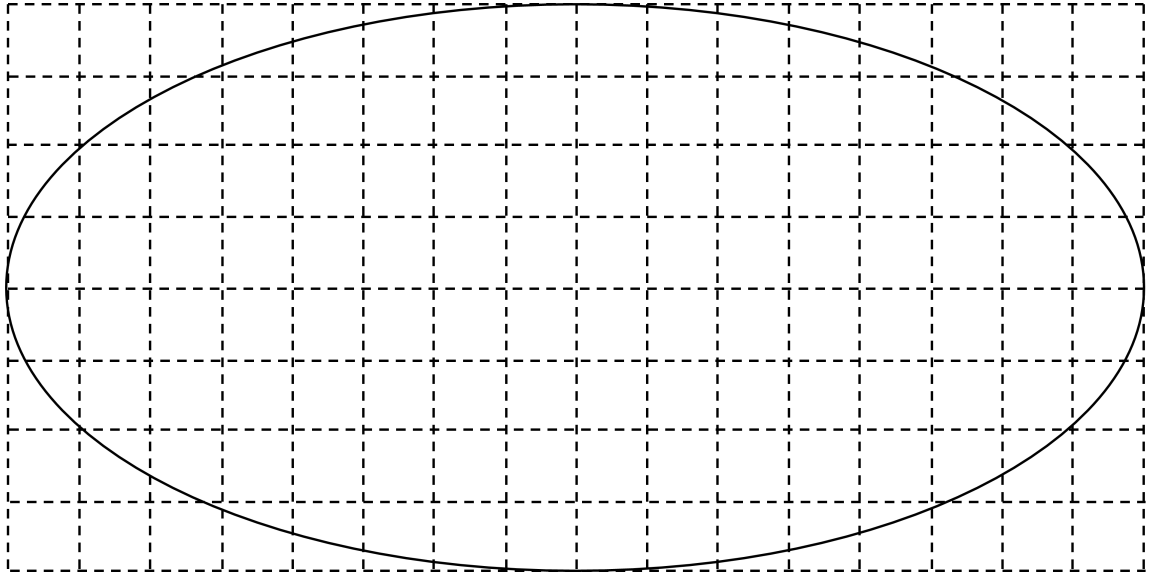
Figure *4.1 Illustration of a pseudo-cylindrical spherical image on a rectangular block grid.*

The benefits of pseudo-cylindrical projections include that they preserve the image content locally and avoid over stretching of polar areas. Moreover, images are represented by fewer pixels compared to respective cylindrically projected images (e.g. equirectangular panorama images) due to the fact that polar areas are not stretched. Due to fewer pixels, they may also compress better and are good candidates for panoramic image projection formats. Pseudo-cylindrical projections may be characterized based upon the shape of the meridians to sinusoidal, elliptical parabolic, hyperbolic, rectilinear and miscellaneous pseudo-cylindrical projections. Using Pseudo-cylindrical panoramas for rendering, navigation and user interaction purposes in virtual reality applications are studied in recent literatures [23][24][25]. Figure 4.1 represents a model of projected pseudo-cylindrical panorama. The effective picture area which contains the 360-degree panoramic data indicated by the solid line and the rectangular block grid is depicted with a dashed line. Examples of these pseudo-cylindrical panoramas are shown in Figure 4.2. Two types of pseudo-cylindrical projections are shown in the figure: sinusoidal and miscellaneous.
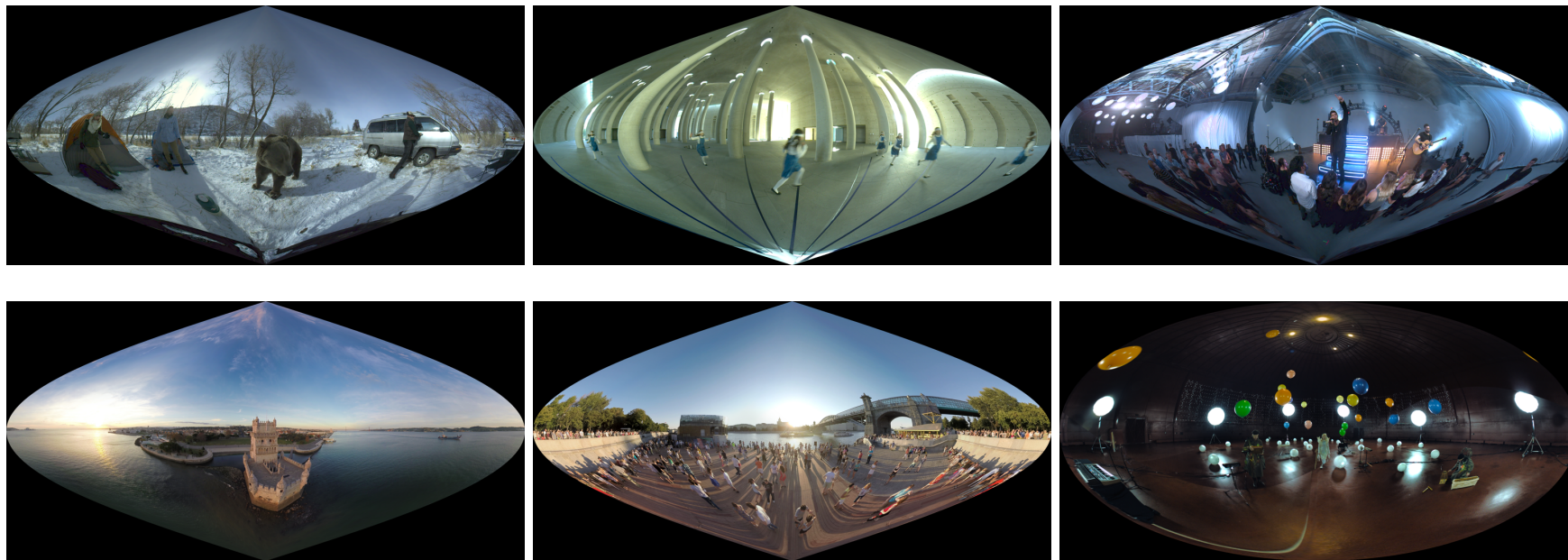
Figure 4.2 *Examples of Pseudo-cylindrical panoramas*

4.2 Problems in Coding of Pseudo-Cylindrical Panoramas

The boundary of the effective picture areas of pseudo-cylindrically projected spherical images is not rectangular and hence it is not aligned with the block partitioning grid used in the image and video encoding and decoding process. Hence, the blocks that include the boundary of the effective picture area contain sharp edges which are not favorable for image/video coding standards. The mentioned non-rectangular content format affects both intra-frame and inter-frame coding process. This is a well-known problem for object-based coding of MPEG-4 part 2 in which the shape of the moving object in the video is identified and separated from the stationary background and then coded separately [26]. Handling the non-rectangular object boundaries was studied in literature for only object-based coding purposes in MPEG-4 [27][28][29][30]. However, none of the previous works explored the efficient coding methods for non-rectangular panoramas. These methods improve the intra-frame coding but lack handling the non-effective picture areas in inter-frame prediction. This section analyzes this problem in intra-frame and inter-frame coding and later for each problem proposes solutions in order to improve the RD performance of these contents.

4.2.1 Intra-Frame Coding Problem

In intra-frame coding, sharp edges in the boundary areas of pseudo-cylindrical panoramas create blocks with non-homogeneous texture which contain both actual picture content and pixels that are outside the effective picture area of the image. These non-homogeneous blocks will produce many high-frequency components after Discrete Cosine Transform (DCT) and quantization processes in the block compared to blocks with homogeneous texture, which typically have very few high-frequency values after DCT and quantization. The main problems that occur with the blocks containing sharp edges are as below:

- Intra prediction signal is typically not able to reproduce the sharp edge, causing the prediction error signal to be substantial and comprise a sharp edge too.

- The high-frequency components cause an increase in bitrate. However, many coding schemes, such as zig-zag scan of DCT coefficients, have been tuned with the expectation that the high-frequency components are less likely and/or with a smaller magnitude than the low-frequency components.

- The quantization of high-frequency components causes visible artifacts, such as ringing effect, for the entire decoded block particularly in the proximity of the sharp edges.

4.2.2 Inter-Frame Coding Problem

The inter-frame prediction of pseudo-cylindrical panoramas in the boundary areas is not efficient due to the fact that the samples in the reference frame are not available in the non-effective areas close to the boundaries of the effective picture. The reconstructed reference pictures that have non-rectangular effective picture area cause a sub-optimal inter prediction performance when:

- The prediction block or block to be encoded is in the boundary areas of the image and hence partially filled with non-effective picture area samples.

- Both the prediction block and the block to be encoded cover a boundary of the effective picture area, therefore both include some data from the non-effective picture area.

This mismatch between block being encoded and prediction block in the reference picture causes extra error samples in the prediction error block and hence incur some bitrate. Particularly, this happens in the following cases:

- Figure 4.3.a and Figure 4.3.b represent the block to be encoded and the prediction block respectively and the object motion in them. The gray area illustrates the effective picture area and as it can be seen, the motion of the object is toward the inside of the effective picture area. As it can be seen from the prediction block, the object is partially inside the effective area and this missing parts lead to huge residual values. The resulted extra residuals are shown in Figure 4.3.c.

- The predicted block in the reference picture contains more samples than the block to be encoded in the current picture. The motion in this case is towards outside the effective picture area. Figure 4.3.d and Figure 4.3.e represent the situation where the predicted block in the reference picture contains more samples than the block to be encoded in the current picture. The extra prediction error samples will occur in the prediction error block as shown in Figure 4.3.f.
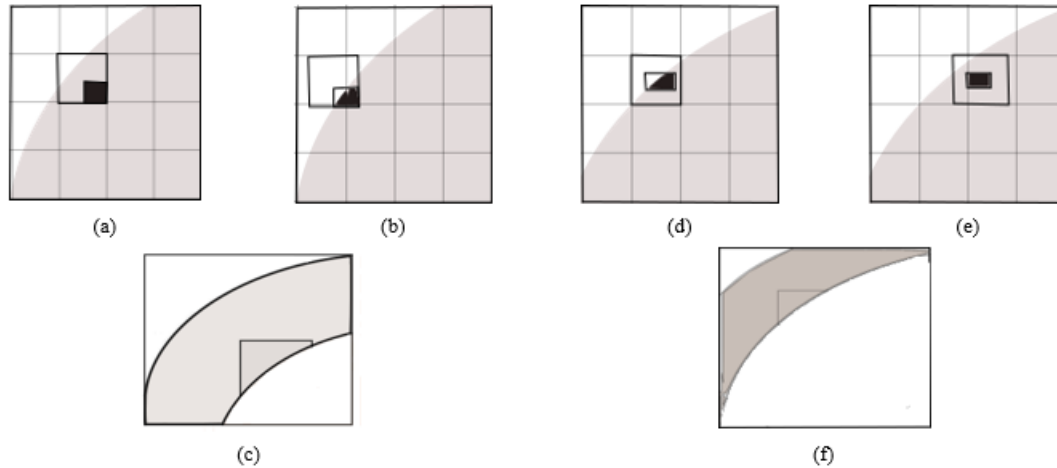
Figure 4.3 *Boundary block in current picture (a and d), prediction block in reference picture (b and e), and prediction error samples (c and f); when the motion is toward the effective picture area and toward the outside of effective picture area, are shown in (a-c) and (d-f), respectively. The black rectangle is a moving object in the video.*

- Another problem arises when a block in the current frame is inter predicted from a boundary region with fractional-pixel motion vector, in which case a motion compensation filter is applied to generate the prediction samples. Close to the boundary of the effective picture area, the motion compensation filter may use as input sample values from locations that are outside the effective picture area. The values of sample locations at different sides of the boundary may differ a lot, hence the motion compensation filter generates pixel values that has overshooting and undershooting effect because of the boundary edge. This overshooting and undershooting predicted values increase the values of residuals, which also increase the bitrate or distortion.

4.3   Proposed Methods

This section, proposes two methods for intra-frame coding in order to overcome the sharp edge problem in the boundary areas of the pseudo-cylindrical panoramas. Along with the methods for intra-frame coding, a method used for enhancing the performance in inter-frame coding of these panoramas.

4.3.1   Intra-frame Coding

As it has been discussed in section 4.2.1, the high-frequency components resulted in the boundaries caused by sharp edges are not favorable to the current video coding

standards such as HEVC and AVC. In order to avoid these high-frequency components, the boundary blocks which contain samples from the non-effective picture area must be filled with samples that are more correlated with the effective picture area samples. As a result, this correlation of pixel values can be easily handled with DCT transform and quantization in the encoding process.

Padding the Boundary Samples

Filling the boundary blocks which are partially containing the samples of non-effective picture area, are done by using the boundary samples of effective picture area. In this method, the first and last pixel of the effective picture boundary are replicated to the boundary blocks in the left and right side of the effective picture area respectively. Padding the boundary pixels row-wise in the neighbor blocks makes the boundary block samples high correlated and this high sample correlation helps the encoder to encode these blocks efficiently.

The results of *padding* method for Bear Attack and MyShelter Stationary Camera sequences in the boundary block areas are shown in two left images in Figure 4.5. As it can be observed, the texture in boundary areas is uniform and hence this enables the codec to compress these areas efficiently.

Copying and Padding the Boundary Samples from the Opposite-Side

Since the left-most pixels of the left boundary and the right-most samples of the right boundary of the 360-degree panoramic images are adjacent to each other, the samples from the opposite side of the effective picture can be used in order fill the boundary blocks. This can be effective particularly when there are significant amount of samples within the boundary block that are also within the effective picture area. These empty parts of the block may be filled with data from the other side to make the content of the block smooth to be efficiently compressed.

The copying method can be applied easily to the boundary areas; however, the polar areas of the pseudo-cylindrical panoramas are the problematic parts, since there are not many samples in these areas to be copied to boundary blocks, so the boundary blocks in the poles will be partially filled. The partially filled blocks will create high-frequency components in the encoding process, hence it would be more efficient to fill the boundary blocks which are partially filled with samples that are copied from the opposite side with some data that can preserve the correlation of samples inside the block. The *padding* method which is used in first part of section 4.3.1 can

Figure 4.4 *Block diagram of the encoding and decoding process of intra-frame coding methods*

be helpful in this situation. After copying samples from the opposite side to fill the boundary blocks, partially filled blocks are detected, and then the rest of the block are filled by replicating the first and the last pixel of each row to the remaining areas inside the block. The resulted images of this method are shown in Figure 4.5. The two images on the right of the figure belongs to *copying plus padding* method.

Encoding and Decoding Side

Figure 4.4 illustrates the whole process of encoding and decoding of proposed intra-frame methods. The *padding* or *copying plus padding* methods can be applied either as a pre-processing step or can be implemented as an in-loop process inside the codec. We considered this process as a pre-processing step. The benefit of doing pre-processing is that, it does not require to change the coding standard and the whole process takes only one pre-processing before encoding and one post-processing after decoding. As can be seen from block diagram in the figure, the pre-processed video is fed to the encoder and then the bitstream sent through network for receiver. In the receiver side, the video will be decoded; but the decoded video contains extra information in the boundary areas resulted from pre-processing state. The post-processing step is applied in order to extract the video in original format by detecting the effective picture area using a pre-defined mask.
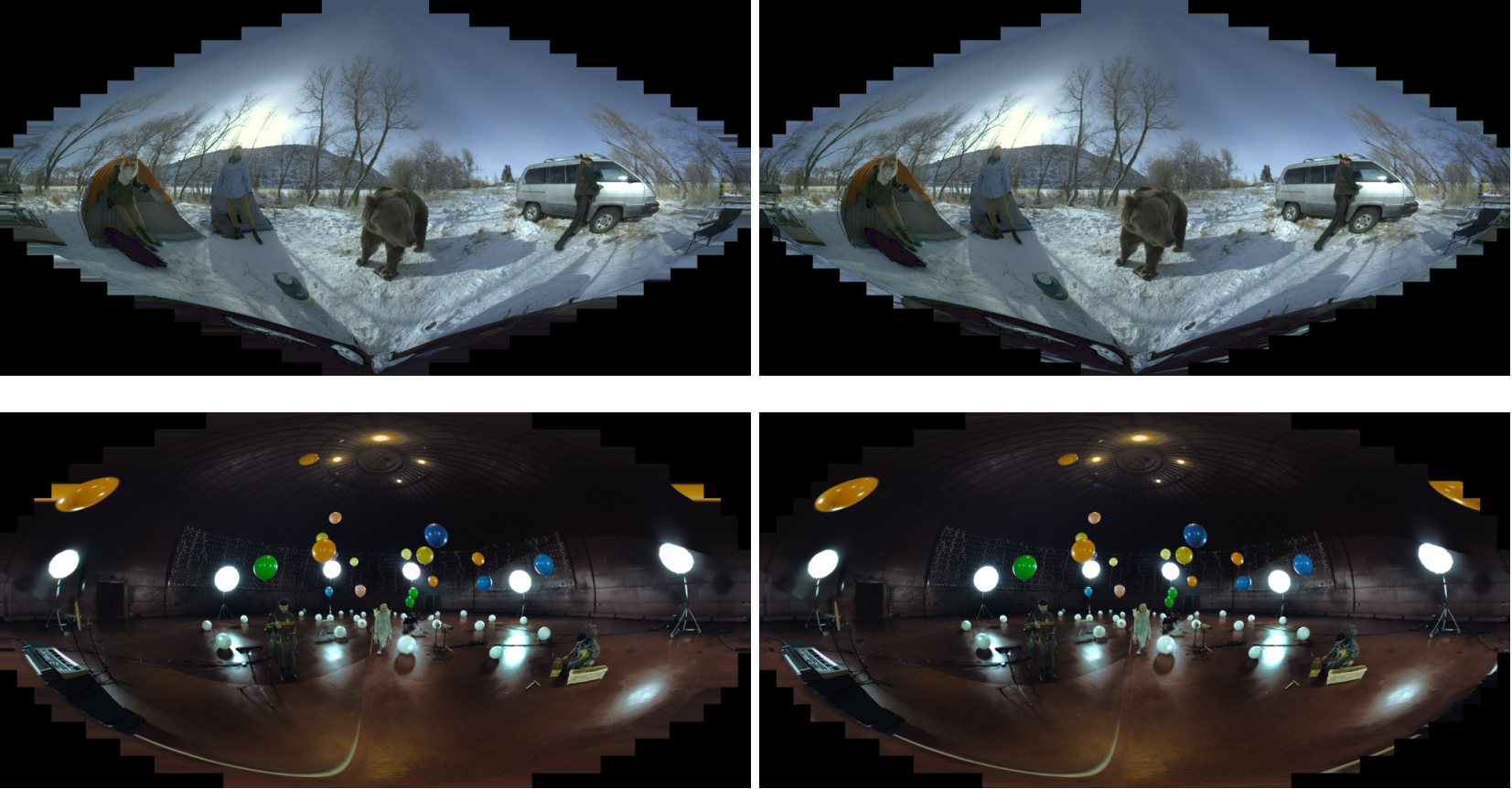
Figure 4.5 *Manipulated intra pictures of Bear Attack and MyShelter Sta with padding and copying plus padding methods*

4.3.2 Inter-Frame Coding

In order to prevent the mentioned problems in section 4.2.2 for inter-frame coding of pseudo-cylindrical panoramas, the advantage of 360-degree characteristics of pseudo-cylindrical panoramic images which the right-most pixel in each row considered to be adjacent to the left-most sample in the same row inside the effective picture area is considered. In order to improve the inter prediction of these 360-degree panoramic videos, in the reference frames, samples are copied from the opposite-side of the effective picture area in the corresponding pixel row, to fill the non-effective picture area in each side of the cropped image.

The row-wise circular copying samples from the opposite-side in the reference frame results for MyShelter stationary camera and Bear Attack sequences are shown in Figure 4.6 respectively. As it can be noticed from the figures, the sample continuity is established in the boundary areas of the manipulated reference frame which enhances the inter prediction of pseudo-cylindrical panoramas.

Expanding the samples from the opposite side to the non-effective picture area helps the prediction of inter frames by filling the prediction blocks in the reference picture with adjacent samples. Two main advantages of this method that improves the coding of inter frames are:

- The non-effective picture area is filled by samples from the opposite side which provides continuity in the boundary areas of the reference picture. This continuity of samples in the boundary areas helps the better prediction from the manipulated reference picture.

- Fractional sample interpolation is improved since the boundary areas do not contain edges anymore, as no overshooting or undershooting pixel values is generated when using motion compensation filter.

Manipulating the reference frame improves the inter prediction, but on the other hand the bitrate increases due to the following reasons.

Residual Manipulation

Manipulating the reference frame by copying the samples from the opposite side creates unwanted extra residuals in the prediction error blocks outside of the effective area. These extra residuals should not be coded into bitstream, otherwise it will
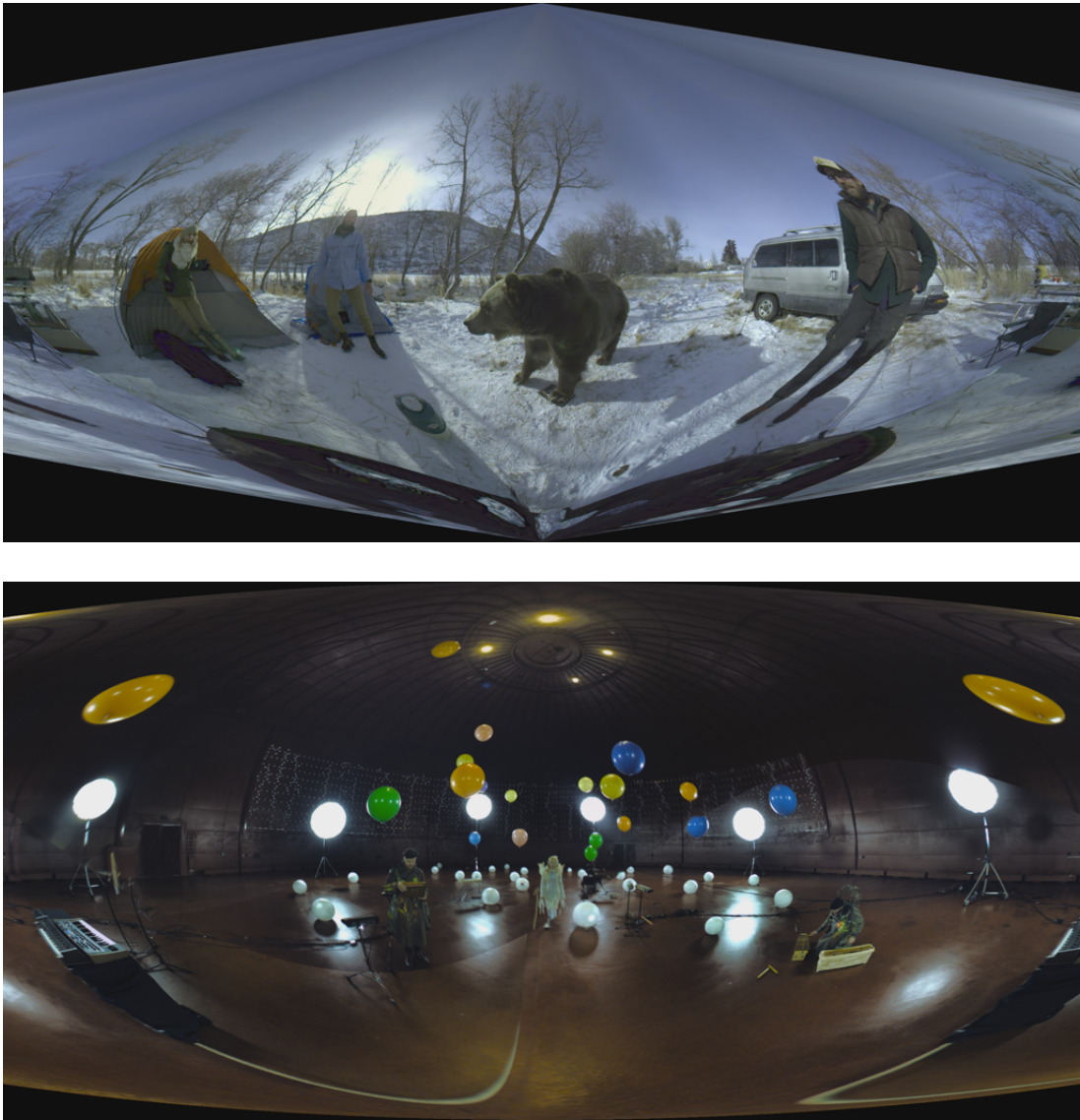
Figure 4.6 *Manipulated reference frame of Bear Attack and MyShelter stationary camera sequences*

increase the bitrate significantly. Hence, in order to avoid such bitrate increment, these motion compensation residuals are replaced by zero values. By replacing the residuals which are located outside of the effective picture area with zero values, the encoder can code these areas with fewer bits compared to non-zero residual values. By replacing these unwanted residuals with zeros we avoid the extra bitrate but on the other hand, the reconstructed image will contain unwanted data in non-effective area which is the result of the copied data in reference frame. The following steps are required for handling this extra data in reconstructed frame:

- The extra data in non-effective area are replaced by zero values before applying the reference frame manipulation step.

- The extra data are removed as a post-processing step after decoding.

Manipulating Distortion Calculation Functions

The extra residuals will affect the distortion cost calculation. During the rate-distortion optimization process at the encoder side, the reconstruction error of the pixels outside of the effective picture area should be excluded from distortion cost (e.g. sum of absolute differences) calculation. Since the residual values outside of the effective picture area of the current frame are replaced by zeros, then the reconstructed picture will contain the copied sample information from the reference frame in the non-effective picture areas. These samples will be omitted in the decoder end.

SAO Modification

After prediction process of current frame, HEVC applies some in-loop filtering techniques (e.g. deblocking filter (DBF) [12]) in order to reduce the coding artifacts in the frames [16]. One of the filtering techniques that applied on reconstructed frames is Sample Adaptive Offset (SAO). SAO applies after deblocking filter and it tries to reduce the mean sample distortion between original and the reconstructed image [13]. Since the reconstructed picture includes the extra samples from the reference frame, the SAO process adds huge offset values to the samples outside the effective area in order to compensate this difference with original picture. The added offset values cause very high bitrate in the encoding process. Hence, to avoid this unnecessary offsets, the SAO is disabled in the encoding side.

Encoding and Decoding Process

The hybrid block diagram of encoder and decoder process in HEVC including the proposed methods for inter prediction in this work is represented in Figure 4.7.a and Figure 4.7.b, respectively.
As Figure 4.7.a illustrates, in the encoding process, the reconstructed frame is passed to Reference Frame Manipulation (RFM) unit prior to filtering (F) and storing in Reference Frame Memory (MEM). The reference frame manipulation is applied in RFM unit and then stored in MEM for inter-frame prediction operations. The process of setting the residual values to zeros outside of the effective picture area is applied in SRZ unit before transform (T) unit, this process is applied only in the encoder side.

Figure 4.7.b demonstrates the decoding process of the proposed algorithm. The similar operations as the encoder side performed in reverse order. The reconstructed picture in pixel prediction operation is passed to RFM unit for reference frame manipulation. The decoded frames later passed to Output Cropping (OC) unit in order to extract the pseudo-cylindrical panorama from the manipulated format. This process includes a predefined mask representing the effective picture area boundaries. The samples outside of the effective picture area will be set to initial background values.
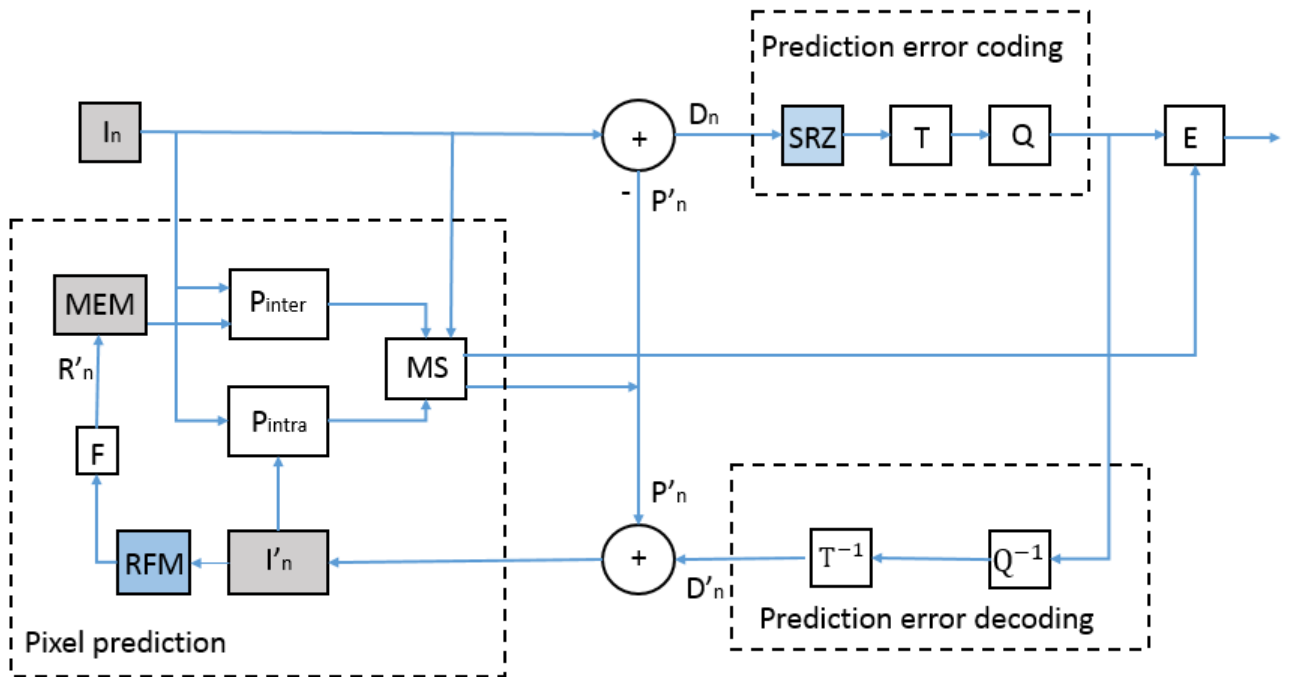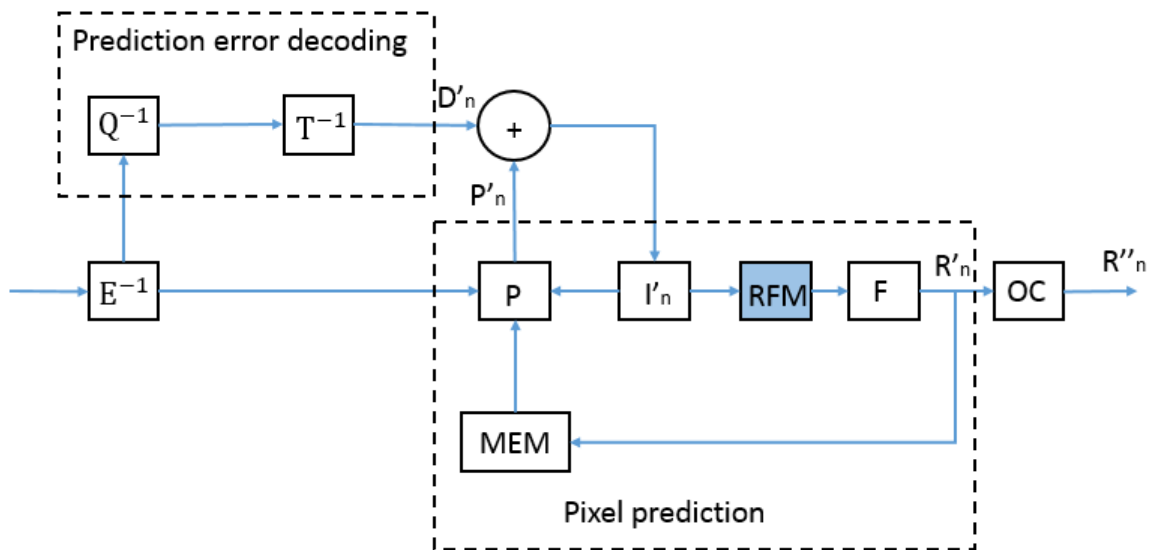
Figure 4.7 *a) Encoder and b) Decoder block diagrams of the proposed inter prediction method*

5. SPHERICAL QUALITY ASSESSMENT FOR VIRTUAL REALITY CONTENT

This section describes the quality/distortion measurement for virtual reality images/video. Section 5.1 describes the conventional quality assessment method in video coding systems. Spherical quality measurement methods are discussed in section 5.2. The section includes the recent spherical PSNR metrics and the proposed USS-PSNR quality metric for VR content.

5.1 Quality Measurement in Video Coding Systems

Delivering video contents with an acceptable level of quality is an important issue in the fast growing video industry. Hence studying new methods in order to measure the quality of videos was conducted by Video Quality Experts Group (VQEG) since 1997 to address the video quality issues [31].

Quality assessment methods can be classified into two categories: subjective and objective methods.

Subjective methods include analysis of decoded videos with human observers in order to rate the quality of the sequences. Although the subjective methods can measure the quality of the videos in a more realistic way, but on the other hand, these methods are usually time consuming and costly. Therefore, using objective quality assessments is necessary. The objective quality metrics can be categorized to three methods: full-reference (FR), reduced reference (RR) and no-reference (NR) methods.

In the full-reference (FR) method, error between perfect reference image and the distorted image is calculated. The calculated error can represent the distortion which is caused by compression algorithms. In the reduced reference (RR) method, the reference signal is not completely available and the quality measurement is done by comparing some features of the distorted signal and the reference signal. In the no-reference (NR) method, the reference signal is not available and the quality assessment is made by measuring the statistics of perfect natural images and video and comparing them with the distorted signal. The NR method is also known as blind quality assessment method [32].

Although the NR and RR methods do not require the original signal for calculating the quality and hence memory requirement is significantly less compared to FR method, but since these methods do not calculate the compression distortion relative to the full reference signal, the results would not represent the true quality of the decoded image/video. Therefore, full-reference methods are studied in this work for analyzing the quality of the VR videos.

There are different full-reference methods for objective quality assessment e.g. peak signal-to-noise ratio (PSNR), structural similarity index (SSIM) [33], moving pictures quality metric (MPQM) [34], etc. Among the objective methods, experiments showed that peak signal-to-noise-ratio (PSNR) metric represents the distortion which is caused by the compression algorithms, much better than the other methods [35]. Equation 5.1 shows the formula for quality measurement using PSNR metric, where the value 255 is the maximum value of the 8-bit luma samples and MSE is the mean square error between original and decoded image. The formula for calculating MSE is shown in equation 5.2. In this equation, N and M represents the resolution of the image and i and j illustrates the corresponding position of the sample in the picture.

$$PSNR[dB] = 10 \log_{10} \frac{255^2}{MSE} \tag{5.1}$$

$$MSE = \frac{\sum_{i=1}^{N-1} \sum_{j=1}^{M-1} (X_{ij} - Y_{ij})^2}{M.N} \tag{5.2}$$

Figure 5.1 demonstrates the block diagram of the encoding and decoding system and the objective quality assessment in a video coding system with the assumption that the transmission network is lossless and does not affect the streamed video quality. As we can see from the Figure 5.1, the PSNR calculation is applied to the decoded video sequences relative to the original uncompressed video.

5.2   Quality Assessment for VR Videos

Virtual reality content are displayed in HMDs [36] and hence it is appropriate to measure the quality of the decoded video in a domain that can represent the display domain properly. The quality measurement domain must comprise real world characteristics of HMDs e.g. uniform sampling and equal distance between samples.
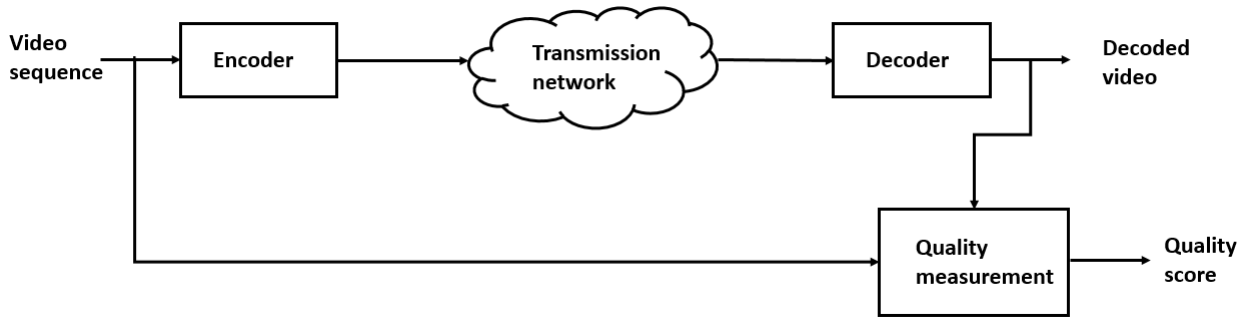
Figure 5.1 *Block diagram of the encoding and decoding system and the objective quality assessment in a video coding system*

5.2.1 Spherical PSNR Calculation

Calculating the quality of the decoded VR content is studied recently in spherical domain by Yu et al [37]. In the proposed method, the decoded video projected on a sphere and the distortion is calculated on the resulted sphere (a.k.a S-PSNR).

The method considers the reason that in the display domain, equator areas on sphere are of great interest to the viewer and hence quality in the polar areas have less importance. Hence, by assigning weights to the spherical coordinates based on users' access frequency measured the compression distortion (a.k.a L-PSNR). Higher weights are dedicated to the equator areas and lower weights to the areas near the poles. Figure 5.2 shows the the assigned weights to the spherical points on sphere in L-PSNR metric.

Although the proposed method is considering the display domain characteristics for PSNR calculation, it suffers from the following problems:

- The selected sampling positions are not uniform in the spherical domain. Hence, the sample positions of the PSNR derivation do not represent an equal contribution in the true display format used in display devices like head mounted displays (HMDs).

- The number of samples used on sphere is very limited (655362 samples), which does not represent the resolution of the original omnidirectional video.

- Assigning different weights to the points is not always reliable, considering the fact that the viewing direction in display domain is content dependent and the user might choose any viewing direction e.g. in the contents with fast and global motion, the user is more likely to view various directions of the scene.
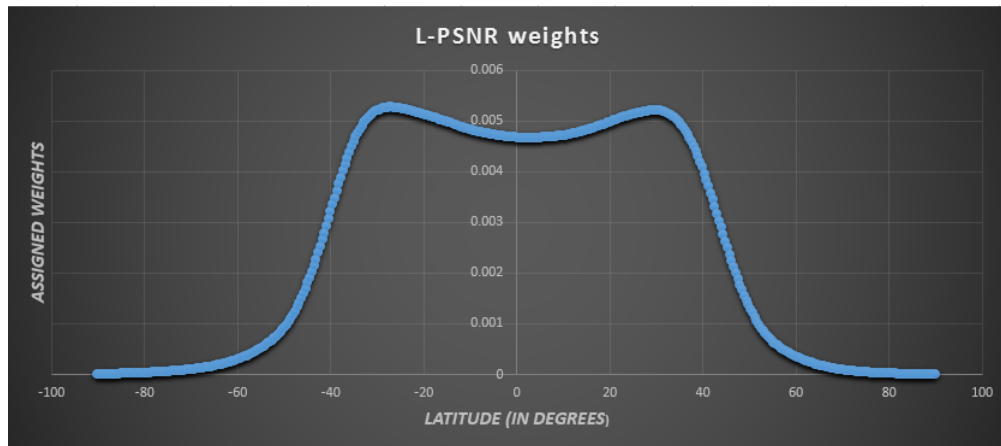
Figure 5.2 *L-PSNR assigned weights based on users' access frequency*

5.2.2 Uniformly Sampled Spherical PSNR (USS-PSNR)

In order to evaluate the quality of the decoded video in VR applications, uniformly sampled spherical PSNR (USS-PSNR) metric is proposed in this work which can measure the distortion in a more realistic way. The method consists of a true distribution of samples on sphere based on latitude and longitude, hence measures the quality of decoded video as close as possible to the displayed content. In other words, the quality measurement domain should correspond to uniform sampling in the display domain.

In this work, the projection from the equirectangular panorama which is the most used projection format in VR applications in compression domain to a uniformly sampled sphere is studied.

Figure 5.3.a and Figure 5.3.b show the equirectangular panorama and its corresponding spherical projection, respectively. Each row of samples in the equirectangular panorama corresponds to a line of samples on the surface of sphere having the equivalent latitude coordinate. The number of samples over each line of a sphere is equal to the circumference of the circular slice with radius of R from the center of that circle, as it illustrated in Figure 5.3.b. Hence, the distribution of samples over sphere changes based on the circumference of each slice and it results a uniform sampling on the sphere. As it can be observed, the sample density in the areas near the equator of the sphere is close to the sample density on equirectangular panorama.

Projection from equirectangular to spherical coordinates is computed as follows:

- Each point (P) in the spherical coordinates can be represented with three elements: radial distance (r), and two polar angles (α, β) as shown in Figure 5.4.
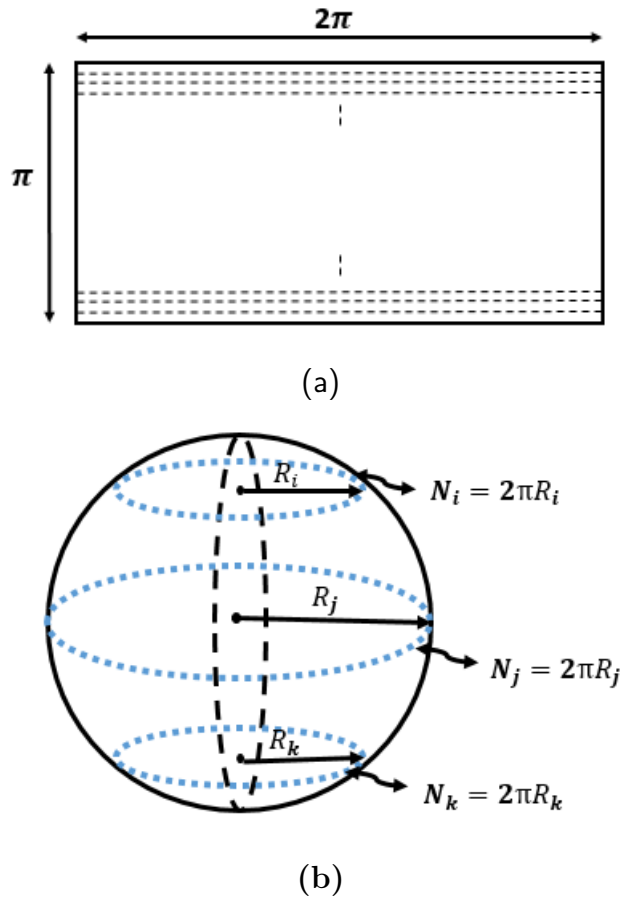
(a)

(b)

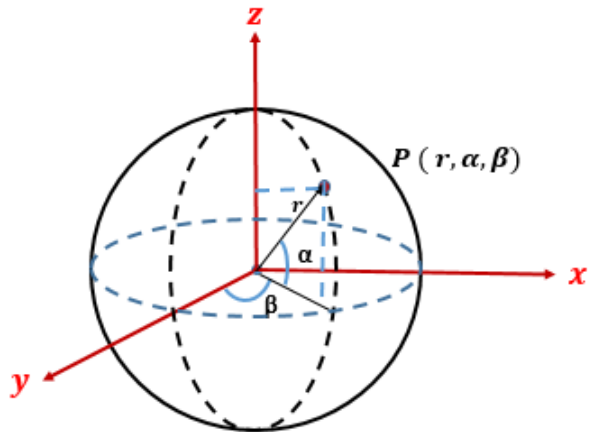**Figure 5.3** *Equirectangular grid and corresponding spherical grid*

Figure 5.4 *Arbitrary point P on sphere*

The corresponding Cartesian coordinates can be calculated as below:

$$X = r \times \cos(\alpha) \times \cos(\beta)$$
$$Y = r \times \cos(\alpha) \times \sin(\beta) \qquad (5.3)$$
$$Z = r \times \sin(\alpha)$$

- For each pixel line on equirectangular which corresponds to a circular slice on sphere with the same latitude coordinate, derive alpha (α):

$$\alpha = \pi \times (\frac{h}{H} - 0.5) \qquad (5.4)$$

  Where, h and H show the height of each pixel line and the original height of the equirectangular panorama, respectively.

- Calculate the radius (R) of each circular slice:

$$R = r \times \cos(\alpha) \qquad (5.5)$$

- Calculate the number of samples in each latitude-wise slice on sphere:

$$N = round(2 \times \pi \times R) \qquad (5.6)$$

- Based on the number of samples for each slice, resample the pixels from the corresponding sample row of the equirectangular panorama using a polyphase filter. The filter window is wrapped around in the beginning and the end of the pixel line in order to use circular resampling.

- Calculate PSNR based on the Cartesian coordinates (total number of samples) and corresponding interpolated pixel values for both projected original and decoded video on a sphere.

The resultant projection will be a spherical 3D image which has uniform sampling over the whole sphere. Figure 5.5 demonstrates the equirectangular panoramic of Kremlin sequence and its two-dimensional (2D) representation of projected image on sphere.

Figure 5.6 demonstrates the projection of Kremlin panorama on a sphere using uniformly sampled sphere (USS) method from different viewing directions from outside of the sphere.

Figure 5.7 shows different viewing directions of the projected Kremlin sequence on sphere from the inside of the sphere.

Figure 5.5 *2D representation of projected Kremlin sequence on sphere*

Figure **5.6** *Different views of projected Kremlin sequence on sphere from outside of the sphere*
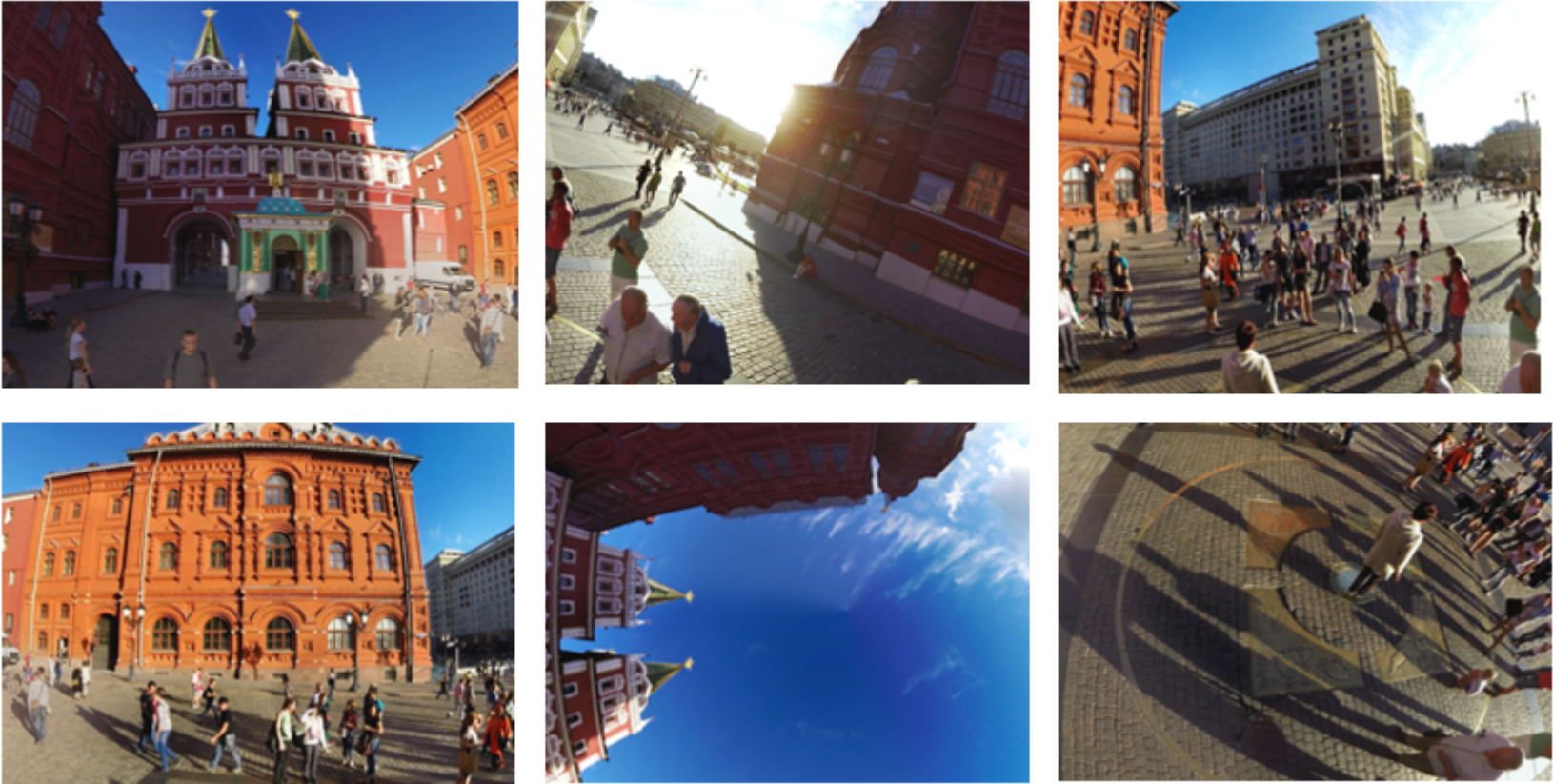
Figure 5.7 *Different views of projected Kremlin sequence on sphere from inside of the sphere*

6. EXPERIMENTAL RESULTS

This chapter presents the experimental results of implemented algorithms in chapters 3 and 4. The quality measurement is done with the proposed USS-PSNR metric in chapter 5. Section 6.2 includes the results of coding equirectangular panoramas with persistent RDS and temporal RDS methods. The results of coding pseudo-cylindrical panoramas presented in section 6.3.

All the proposed methods implemented in the HM version 16.6, the HEVC reference software [38] with the JCT-VC common test condition [39] using four quantization parameters (QP). The performances were evaluated in terms of bitrate reduction and the decoded picture quality using the well-known Bjøntegaard delta bitrate (BDBR) metric [40] [41].

6.1 Video Sequences

In the simulations 9 video sequence are used to analyze the performance of the proposed methods.

Sequences Bear Attack, Daisy, VRC Concert, MyShelter Moving Camera and MyShelter Stationary Camera are captured by Nokia's virtual reality camera OZO [2]. Ghost Town Sheriff is an animated video sequence created by UNDO [42]. Kremlin is a time-laps sequence, Moscow and Lisboa sequences are a camera captured videos which are provided by Airpano [43]. The sequences contain various characteristics including fast and slow object motion, camera motion, etc. All the sequences are converted from RGB to YUV raw format with FFMPEG tool [44] for encoding.

6.2 Results for P-RDS and T-RDS Methods

This section presents the regionally down-sampled (RDS) methods for coding equirectangular panoramas which are discussed in chapter 3. In order to maintain the codec as simple as possible, only three stripe divisions (top, middle and bottom)

Table   6.1 *Video Sequences used in the experiments*

Sequence	Number of Frames	Resolution
Kremlin	100	4096x2048
Lisboa	100	4096x2048
Moscow	100	4096x2048
Ghost Town Sheriff	100	4096x2048
Bear Attack	100	3584x1792
Daisy	100	3584x1792
VRC Concert	100	3584x1792
MyShelter Moving Camera	100	2048x1024
MyShelter Stationary Camera	100	2048x1024

Table   6.2 *BD-rate results for T-RDS and P-RDS methods using USS-PSNR metric*

Sequence	T-RDS	P-RDS
Kremlin	-2.29	-0.83
Lisboa	-5.18	-5.43
Moscow	-2.12	-1.81
Bear Attack	-6.53	-5.73
Daisy	-5.22	-3.04
VRC Concert	-5.09	-6.15
MyShelter Moving Camera	-5.19	-5.75
MyShelter Stationary Camera	-5.77	-5.71

considered in equirectangular video. The height of top and bottom stripes are one fourth of the original height of the video. The down-sampling process is only applied to top and bottom stripes in horizontal direction with the ratio of 2. The middle stripe was coded in the full resolution.

The quality of decoded videos are assessed using the proposed USS-PSNR metric. Moreover, in order to investigate the other spherical quality measurement methods, the S-PSNR and L-PSNR methods also used for analyzing the P-RDS and T-RDS methods.

All the simulations are done with Random Access (RA) configuration using QP values 22, 27, 32 and 37. The intra period used for the simulations is 32 (4 intra frames and 96 inter frames).

Table 6.2 shows the results for T-RDS and P-RDS method using USS-PSNR metric (section 5.2.2). As it can be observed, sequence-based mixed results achieved between T-RDS and P-RDs methods, although T-RDS method slightly outperforms P-RDS method.

The simulation results using S-PSNR quality metric are shown in Table 6.3. Similar performance differences between T-RDS and P-RDS as in Table 6.2 achieved using S-PSNR method, where equal weight sampling over the sphere is used.

Table ***6.3*** *BD-rate results for T-RDS and P-RDS methods using S-PSNR*

Sequence	T-RDS	P-RDS
Kremlin	-2.26	-0.73
Lisboa	-4.84	-4.45
Moscow	-1.94	-1.60
Bear Attack	-6.26	-5.35
Daisy	-1.81	0.84
VRC Concert	-3.93	-4.94
MyShelter Moving Camera	-5.79	-6.26
MyShelter Stationary Camera	-6.12	-6.18

Table ***6.4*** *BD-rate results for T-RDS and P-RDS methods using L-PSNR*

Sequence	T-RDS	P-RDS
Kremlin	-5.99	-7.81
Lisboa	-10.96	-13.88
Moscow	-4.80	-5.51
Bear Attack	-10.57	-13.24
Daisy	-8.50	-11.42
VRC Concert	-14.46	-17.33
MyShelter Moving Camera	-9.90	-12.61
MyShelter Stationary Camera	-8.31	-12.05

The BD-rate results in Table 6.4 using L-PSNR demonstrate higher compression gain compared to USS-PSNR and S-PSNR metrics. Also, for all video sequences down-sampling all frames (P-RDS) method outperforms T-RDS method. However, as is has been discussed in section 5.2.1, using weighted PSNR (e.g. L-PSNR) does not represent the quality of the video fairly.

6.3   Results for Pseudo-Cylindrical Panoramas

This section presents the results of implemented methods for coding pseudo-cylindrical panoramas.

6.3.1   Intra-frame Coding Results

In order to evaluate the performance of proposed intra-frame coding method which is proposed in section 4.3.1, all the sequences have been coded using all intra configuration using QP values of 23, 28, 33 and 38. Table 6.5 and Table 6.6 present

Table 6.5 *Bjøntegaard results for padding method*

Sequence	BD-rate	BD-PSNR
Kremlin	-2.41	0.14
Lisboa	-12.77	0.58
Moscow	-4.20	0.25
Ghost Town Sheriff	-5.41	0.14
Bear Attack	-5.30	0.23
Daisy	-16.52	0.72
VRC Concert	-5.80	0.29
MyShelter Moving Camera	-5.21	0.26
MyShelter Stationary Camera	-7.10	0.31

Table 6.6 *Bjøntegaard results for copying plus padding method*

Sequence	BD-rate	BD-PSNR
Kremlin	3.29	-0.20
Lisboa	-10.76	0.45
Moscow	-0.01	0.002
Ghost Town Sheriff	1.61	-0.05
Bear Attack	-0.80	0.04
Daisy	-6.43	0.28
VRC Concert	5.70	-0.26
MyShelter Moving Camera	1.63	-0.07
MyShelter Stationary Camera	-4.81	0.21

the BD-rate and BD-PSNR results for *padding* and *copying plus padding* methods for intra-frame coding, respectively.

The results show that both *padding* and *copying plus padding* methods improve the compression performance of pseudo-cylindrical panoramas. However, *padding* method gives significant bitrate reduction compared to *copying plus padding* method for all sequences mainly because of the uniform texture in the boundary areas resulted by padding the boundary samples. For the case of *copying plus padding* method, better performance achieved for the sequences which have uniform texture in the boundaries of the effective picture area and hence, the encoder is able to compress these areas more efficiently.

6.3.2   Overall Performance of Intra and Inter Coding Methods

For evaluating the overall performance of the methods, the combination of both coding methods is used. Since the *padding* method for intra-frame coding gives better

Table   6.7 *Bjøntegaard results for both intra and inter coding methods*

Sequence	BD-rate	BD-PSNR
Kremlin	-1.01	0.05
Lisboa	-7.22	0.20
Moscow	-1.13	0.05
Ghost Town Sheriff	-3.30	0.11
Bear Attack	-4.20	0.11
Daisy	-8.80	0.30
VRC Concert	-2.71	0.10
MyShelter Moving Camera	-6.23	0.16
MyShelter Stationary Camera	-7.70	0.21

compression performance compared to *copying plus padding*, this method used for coding the intra frames.

The analysis is based on 4 QP values as before using Random Access (RA) configuration in which the intra period is 32 (4 intra-frames and 96 inter-frames).

Table 6.7 presents the achieved results using both intra and inter coding methods compared to coding the pseudo-cylindrical panoramas with unchanged HEVC codec.

As the results illustrates in Table 6.7, the combination of both methods for intra and inter coding of pseudo-cylindrical panoramas can boost the performance. The highest bitrate reduction was achieved for the sequences containing either or both uniform texture in the boundaries (improves intra prediction) and higher global motion (improves inter prediction). For example, Lisboa and MyShelter Moving Camera sequences contain both uniform boundary texture and global motion but Daisy has only uniform texture in the boundary areas.

The rate-distortion (RD) curves of intra coding method and overall (intra + inter) performances are shown in Figure 6.1.
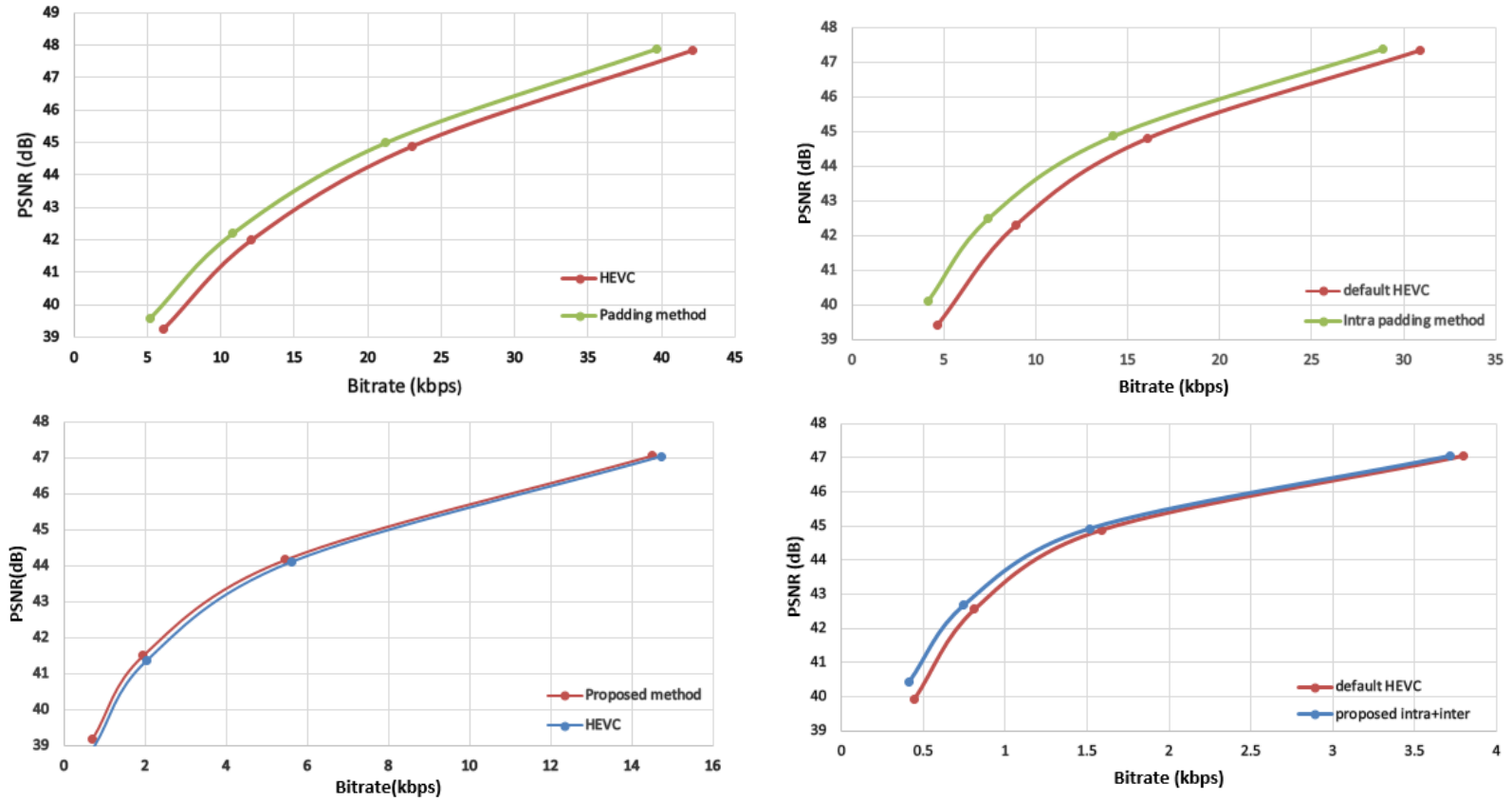
Figure 6.1 *Rate-Distortion curves*

7. CONCLUSION AND FUTURE WORK

This thesis work attempted to develop methods for efficient coding of equirectangular and pseudo-cylindrical panoramas for virtual reality applications. Moreover, the quality measurement metrics for VR video have been investigated for the proposed coding methods.

Regional down-sampling (RDS) methods were investigated for compressing equirectangular panoramas. Two approaches were analyzed for RDS method: persistent regional down-sampling (P-RDS) and temporal regional down-sampling (T-RDS). In RDS method, the equirectangular panorama was divided to multiple stripes/tiles and each stripe/tile is down-sampled horizontally to decrease the redundant information caused by stretching. The stripes in polar areas include higher down-sampling ratios since the stretching effect is more severe in those areas. In the P-RDS method, down-sampling has been applied to all frames of the video sequence. However, in the T-RDS approach, the down-sampling was used only in temporal (inter) frames and the intra frames were encoded in conventional full-resolution format. The simulation results illustrated that both persistent RDS and temporal RDS methods improved the rate-distortion (RD) performance compared to the conventional coding of equirectangular panoramas consistently in all test sequences. The temporal RDS method had less sequence-wise RD performance variation and slightly better RD performance when compared to the persistent RDS technique. However, since the performance of the methods is content dependent, as a future work we will study the content-based analyzing process to select the optimum resampling technique for each video sequence.

Another contribution in this thesis was using pseudo-cylindrical panoramas in VR applications. Pseudo-cylindrical projections map the 3D spherical scene to a 2D image plane in a more realistic sample distribution by avoiding the over stretching polar areas. Hence, the scene is represented by fewer pixels compared to cylindrical projections and can be more efficient for compression. However, the projected image is not in rectangular format, accordingly it is not suitable for the current video coding standards. In this work, simple yet efficient methods were proposed to improve the compression performance of these panoramic formats. The proposed approach reduces the bitrate of these contents and improves the quality of decoded video, par-

ticularly in the boundaries of effective picture area which have undesirable quality because of sharp edges. The experimental results showed that both intra-frame and inter-frame coding performance have been improved by using the investigated techniques. However, the coding performance has not been compared to other projection formats (e.g. cylindrical projections). As a future work in this topic, the compression performance of pseudo-cylindrical panoramas will be compared to other well-known projection formats such as equirectangular and cube map projections.

The last contribution was investigating spherical quality assessment for VR videos. In this work, uniformly sampled spherical PSNR (USS-PSNR) metric was proposed in order to evaluate the performance of different coding methods in omnidirectional content. The USS-PSNR metric uses uniform and equal weight sampling of the decoded video on sphere which represents the true display characteristics of head mounted display (HMD) devices and hence, provides a fair quality measurement approach for VR video coding systems.

BIBLIOGRAPHY

[1] A. B. Craig, W. R. Sherman, and J. D. Will, *Developing virtual reality applications: Foundations of effective design.* Morgan Kaufmann, 2009.

[2] Nokia's Virtual Reality Camera OZO. https://ozo.nokia.com/. [Online, accessed August 2016].

[3] T. Wiegand, G. J. Sullivan, G. Bjontegaard, and A. Luthra, "Overview of the H. 264/AVC video coding standard," *IEEE Transactions on circuits and systems for video technology*, vol. 13, no. 7, pp. 560–576, 2003.

[4] G. J. Sullivan, J. R. Ohm, W. J. Han, and T. Wiegand, "Overview of the high efficiency video coding (HEVC) standard," *IEEE Transactions on circuits and systems for video technology*, vol. 22, no. 12, pp. 1649–1668, 2012.

[5] Samsung Gear VR. http://www.samsung.com/global/galaxy/gear-vr/. [Online, accessed August 2016].

[6] Joint Video Team (JVT), ITU-T website. http://www.itu.int/en/ITU-T/studygroups/com16/video/Pages/jvt.aspx. [Online, accessed August 2016].

[7] J. R. Ohm, G. J. Sullivan, H. Schwarz, T. K. Tan, and T. Wiegand, "Comparison of the coding efficiency of video coding standards including high efficiency video coding (HEVC)," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 22, no. 12, pp. 1669–1684, 2012.

[8] I. K. Kim, J. Min, T. Lee, W. J. Han, and J. H. Park, "Block partitioning structure in the HEVC standard," *IEEE transactions on circuits and systems for video technology*, vol. 22, no. 12, pp. 1697–1706, 2012.

[9] P. Helle, S. Oudin, B. Bross, D. Marpe, M. O. Bici, K. Ugur, J. Jung, G. Clare, and T. Wiegand, "Block merging for quadtree-based partitioning in HEVC," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 22, no. 12, pp. 1720–1731, 2012.

[10] J. Lainema, F. Bossen, W. J. Han, J. Min, and K. Ugur, "Intra coding of the HEVC standard," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 22, no. 12, pp. 1792–1801, 2012.

[11] V. Sze and M. Budagavi, "High throughput CABAC entropy coding in HEVC," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 22, no. 12, pp. 1778–1791, 2012.

[12] A. Norkin, G. Bjøntegaard, A. Fuldseth, M. Narroschke, M. Ikeda, K. Andersson, M. Zhou, and G. Van der Auwera, "HEVC deblocking filter," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 22, no. 12, pp. 1746–1754, 2012.

[13] C. M. Fu, E. Alshina, A. Alshin, Y. W. Huang, C. Y. Chen, C. Y. Tsai, C. W. Hsu, S. M. Lei, J. H. Park, and W. J. Han, "Sample adaptive offset in the HEVC standard," *IEEE Transactions on Circuits and Systems for Video technology*, vol. 22, no. 12, pp. 1755–1764, 2012.

[14] I. Bauermann, M. Mielke, and E. Steinbach, "H. 264 based coding of omnidirectional video," in *Computer Vision and Graphics*, pp. 209–215, Springer, 2006.

[15] C. W. Fu, L. Wan, T. T. Wong, and C. S. Leung, "The rhombic dodecahedron map: An efficient scheme for encoding panoramic video," *IEEE Transactions on Multimedia*, vol. 11, no. 4, pp. 634–644, 2009.

[16] H. Kimata, S. Shimizu, Y. Kunita, M. Isogai, and Y. Ohtani, "Panorama video coding for user-driven interactive video application," in *2009 IEEE 13th International Symposium on Consumer Electronics*, pp. 112–114, IEEE, 2009.

[17] K. T. Ng, S. C. Chan, and H. Y. Shum, "Data compression and transmission aspects of panoramic videos," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 15, no. 1, pp. 82–95, 2005.

[18] I. Tosic and P. Frossard, "Low bit-rate compression of omnidirectional images," in *Proceedings of the 27th conference on Picture Coding Symposium*, pp. 53–56, IEEE Press, 2009.

[19] M. Yu, H. Lakshman, and B. Girod, "Content adaptive representations of omnidirectional videos for cinematic virtual reality," in *Proceedings of the 3rd International Workshop on Immersive Media Experiences*, pp. 1–6, ACM, 2015.

[20] Z. Wen, J. Li, S. Li, and J. Wen, "Requirements on cinematic virtual reality mapping schemes," in *MPEG M38110*, February 2016.

[21] G. I. Evenden, *Cartographic projection procedures for the UNIX environment: A user's manual.* Citeseer, 1990.

[22] L. M. Bugayevskiy and J. Snyder, *Map projections: A reference manual.* CRC Press, 1995.

[23] H. G. Debarba, S. Perrin, B. Herbelin, and R. Boulic, "Embodied interaction using non-planar projections in immersive virtual reality," in *Proceedings of the 21st ACM Symposium on Virtual Reality Software and Technology*, pp. 125–128, ACM, 2015.

[24] J. Ardouin, A. Lécuyer, M. Marchal, and E. Marchand, "Navigating in virtual environments with 360 omnidirectional rendering," in *3D User Interfaces (3DUI), 2013 IEEE Symposium on*, pp. 95–98, IEEE, 2013.

[25] J. Ardouin, A. Lécuyer, M. Marchal, and E. Marchand, "Stereoscopic rendering of virtual environments with wide field-of-views up to 360°," in *2014 IEEE Virtual Reality (VR)*, pp. 3–8, IEEE, 2014.

[26] T. Ebrahimi and C. Horne, "MPEG-4 natural video coding–an overview," *Signal Processing: Image Communication*, vol. 15, no. 4, pp. 365–385, 2000.

[27] A. Kaup, "Object-based texture coding of moving video in MPEG-4," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 9, no. 1, pp. 5–15, 1999.

[28] L. Long, W. Zhanhui, S. Qijun, and Z. Haiyan, "New padding algorithm for object-based coding," in *Image and Signal Processing (CISP), 2011 4th International Congress on*, vol. 1, pp. 556–559, IEEE, 2011.

[29] J.-H. Moon, J.-H. Kweon, and H.-K. Kim, "Boundary block-merging (BBM) technique for efficient texture coding of arbitrarily shaped object," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 9, no. 1, pp. 35–43, 1999.

[30] J.-L. Shih, C.-H. Lee, and Y.-Y. Chiang, "An MPEG-4 texture coding approach using refined boundary block merging and adaptive SA-DCT," in *IMECS*, pp. 578–583, 2006.

[31] The Video Quality Experts Group (VQEG). http://www.its.bldrdoc.gov/vqeg/vqeg-home.aspx. [Online, accessed August 2016].

[32] Z. Wang, H. Sheikh, and A. Bovik, "The handbook of video databases: Design and applications (b. furht and o. marques, eds), chapter objective video quality assessment, pages pp. 1041–1078," 2003.

[33] W. Zhou, L. Liang, and A. Bovik, "Video quality assessment using structural distortion measurement," in *Proc. 2002 Int. Conf. on Image Processing, Rochester, NY, USA*, vol. 3, 2002.

[34] C. J. Van den Branden Lambrecht and O. Verscheure, "Perceptual quality measure using a spatiotemporal model of the human visual system," in *Electronic Imaging: Science & Technology*, pp. 450–461, International Society for Optics and Photonics, 1996.

[35] Q. Huynh-Thu and M. Ghanbari, "Scope of validity of PSNR in image/video quality assessment," *Electronics letters*, vol. 44, no. 13, pp. 800–801, 2008.

[36] J. Rolland and H. Hua, "Head-mounted display systems," *Encyclopedia of optical engineering*, pp. 1–13, 2005.

[37] M. Yu, H. Lakshman, and B. Girod, "A framework to evaluate omnidirectional video coding schemes," in *Mixed and Augmented Reality (ISMAR), 2015 IEEE International Symposium on*, pp. 31–36, IEEE, 2015.

[38] High Efficiency Video Coding (HEVC) reference software HM. https://hevc.hhi.fraunhofer.de/. [Online, accessed August 2016].

[39] D. Flynn and C. Rosewarne, "Common test conditions and software reference configurations for HEVC range extensions," in *Proceedings of the 14th Meeting of Joint Collaborative Team on Video Coding (JCT-VC) of ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11*, 2013.

[40] G. Bjøntegaard, "Calcuation of average PSNR differences between RD-curves," *Doc. VCEG-M33 ITU-T Q6/16, Austin, TX, USA, 2-4 April 2001*, 2001.

[41] G. Bjøntegaard, "Improvements of the BD-PSNR model," *ITU-T SG16 Q*, vol. 6, p. 35, 2008.

[42] UNDO's website. http://undo.fi/. [Online, accessed August 2016].

[43] Airpano's website. www.airpano.com. [Online, accessed August 2016].

[44] FFMPEG Software. https://www.ffmpeg.org/. [Online, accessed August 2016].