Scalable Bit Allocation between Texture and Depth Views for 3D Video Streaming over Heterogeneous Networks

Jinlin XIAO, Miska M. HANNUKSELA, Member, IEEE, Tammam TILLO, Senior Member, IEEE, Moncef GABBOUJ, Fellow, IEEE, Ce ZHU, Senior Member, IEEE, Yao ZHAO, Senior Member, IEEE

Abstract—In the multi-view video plus depth (MVD) coding format, both texture and depth views are jointly compressed to represent the three-dimensional (3D) video content. The MVD format enables synthesis of virtual views through depth-image-based rendering, and hence distortion in the texture and depth views affects the quality of the synthesized virtual views. Bit allocation between texture and depth views has been studied with some promising results. However, to the best of our knowledge, most of the existing bit allocation methods attempt to allocate a fixed amount of total bit rate between texture and depth views, i.e., to select appropriate pair of Quantization Parameters (QPs) for texture and depth views to maximize the synthesized view quality subject to a fixed total bit rate. In this paper, we propose a scalable bit allocation scheme, where a single ordering of texture and depth packets is derived and used to obtain optimal bit allocation between texture and depth views for any total target rates. In the proposed scheme, both texture and depth views are encoded using quality scalable coding method, i.e., Medium Grain Scalable (MGS) coding of the Scalable Video Coding (SVC) extension of the Advanced Video Coding (H.264/AVC) standard. For varying target total bit rates, optimal bit truncation points for both texture and depth views can be obtained using the proposed scheme. Moreover, we propose to order the enhancement layer packets of the H.264/SVC MGS encoded depth view based on their contribution to the reduction of the synthesized view distortion. On one hand, this improves the depth view packet ordering when considered the rate-distortion performance of synthesized views, which is demonstrated by the experimental results. On the other hand, the information obtained in this step is used to facilitate optimal bit allocation between texture and depth views. Experimental results demonstrate the effectiveness of the proposed scalable bit allocation scheme for texture and depth views.

This work was supported in part by the 973 Program under Grant 2011CB302204, in part by the National Natural Science Foundation of China under Grants 60972085, 61025013, 61125106, 61228102, in part by PCSIRT under Grant IRT 201206, and Ph.D. Programs Foundation of Ministry of Education of China under grant 2013018210010. (Corresponding author: C. Zhu)

J. XIAO and M. GABBOUJ are with the Tampere University of Technology, Tampere 33720, Finland (e-mail: xiaojin@1981@gmail.com, moncef.gabbouj@tut.fi).

M. M. HANNUKSELA is with Nokia Research Center, Tampere 33720, Finland (e-mail: miska.hannukela@nokia.com).

T. TILLO is with the Department of Electrical and Electronic Engineering, Xi’an Jiaotong-Liverpool University, 111 Ren Ai Road, Suzhou, P.R. China (e-mail: tammam.tillo@sjtu.edu.cn).

C. ZHU is with the School of Electronic Engineering, University of Electronic Science and Technology of China, Chengdu, China (e-mail: eczhu@uestc.edu.cn).

Y. ZHAO is with Institute of Information Science, Beijing Jiaotong University, Beijing, China (e-mail: yzhaos@bjtu.edu.cn).

Copyright (c) 2014 IEEE. Personal use of this material is permitted. However, permission to use this material for any other purposes must be obtained from the IEEE by sending an email to pubs-permissions@ieee.org.

Index Terms—3D scalability; 3D streaming; 3D; texture view; depth view; bit allocation; quality scalable; synthesized view distortion; medium grain scalable; heterogeneous network

I. INTRODUCTION

Three-dimensional (3D) video has drawn a lot of attention both from industry and academia during the last decade. Stereoscopic video is the basic form of 3D video, and is prevalent in today’s 3D content and services. Autostereoscopic 3D displays [1], [2] enable viewing 3D content from different angles without the use of special headgear or glasses. As many different viewpoints of the same video content are required for autostereoscopic 3D system, compared with the traditional stereo video, the required bit rate increases tremendously.

To further reduce the redundancy between different viewpoints of the 3D video, besides the conventional temporal prediction which is commonly used in video compression, inter-view prediction [3] is also introduced in the Multi-view Video Coding (MVC) [4] extension of the Advanced Video Coding (H.264/AVC) standard [5]. Though MVC has enormously improved the compression performance of multi-view video, it still requires a bit rate that is proportional to the number of views [4]. The Multi-view video plus depth (MVD) format is a promising way to represent 3D video content, and extensions supporting for the MVD format have been finished recently [6], [7]. With the MVD format, only a small number of texture views associated with their depth views are required. At the decoder or display side, depth-image-based rendering (DIBR) [8], [9] is used to synthesize additional viewpoint video.

This paper focuses on adaptive streaming of multi-view video plus depth, where we have two main options [10]. 1) Simulcast encoding: encode each view and/or depth view independently using a scalable or nonscalable monocular video codec, which enables streaming each view over separate channels; and clients can request as many views as their 3D displays require without worrying about inter-view dependencies. 2) Dependent encoding: encode views using MVC to decrease the overall bit rate by exploiting the inter-view redundancies. Simulcast encoding can be regarded as a more flexible approach than dependent encoding. For example, simulcast encoding enables a client to select the viewpoint dynamically out of a large number of views available in the server, whereas inter-view dependencies would either cause...
This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2014.2334011, IEEE Transactions on Circuits and Systems for Video Technology

3D video Server
texture videos + depth maps

Heterogeneous Network

User A, B = 4 Mbps
User B, B = 3 Mbps
User C, B = 2 Mbps

Figure 1. 3D video streaming over heterogeneous networks, different links have different bandwidth.

limitations on viewpoint selection or streaming of a larger number of views than necessary.

Simulcast coding using the H.264 Scalable Video Coding (SVC) [11] standard can produce scalable 3D video, where each view is encoded independently. Here, two approaches can be followed for scalability: either all views can be coded scalably, or some views can be coded scalably using SVC and others can be coded nonscalably using H.264/AVC. On the other hand, encoding views using MVC decreases the overall bit rate for the multi-view video plus depth format. However, MVC offers only temporal and view scalability, but no quality or resolution scalability [10].

In the MVD format, the depth views are always used together with the associated texture views, and both the texture and depth views quality will affect the synthesized view quality. Thus, the allocation of bit rate between the texture and depth views is an important issue. During the last decade, many bit allocation methods have been proposed [12]–[19]. However, these methods work well when the total target bit rate for the texture and depth views is fixed. However, for the 3D video server, which is serving for many users with different link bandwidth in heterogeneous networks, the existing bit allocation methods cannot work well. For example, in the application scenario described in Figure 1, to have proper allocation between texture and depth views, the 3D video server has to allocate bit rate for the target rate allocation between texture and depth views, the 3D video application scenario described in Figure 1, to have proper allocation methods cannot work well. For example, in the link bandwidth in heterogeneous networks, the existing bit rate for the texture and depth views is fixed. However, for the other hand, encoding views using MVC decreases the overall bit rate for the multi-view video plus depth format. However, MVC offers only temporal and view scalability, but no quality or resolution scalability [10].

In the MVD format, the depth views are always used together with the associated texture views, and both the texture and depth views quality will affect the synthesized view quality. Thus, the allocation of bit rate between the texture and depth views is an important issue. During the last decade, many bit allocation methods have been proposed [12]–[19]. However, these methods work well when the total target bit rate for the texture and depth views is fixed. However, for the 3D video server, which is serving for many users with different link bandwidth in heterogeneous networks, the existing bit allocation methods cannot work well. For example, in the application scenario described in Figure 1, to have proper allocation between texture and depth views, the 3D video server has to allocate bit rate for the target rate 2 Mbps, 3 Mbps and 4 Mbps; the resulting 3 different versions of the encoded texture and depth views should be all stored in the server. It is worth noticing that for practical applications, the link bandwidth over heterogeneous networks is varying, and hence more than 3 versions of texture and depth pairs should be stored in the server. This will cause two fundamental problems: one is that the bit allocation algorithm should be carried out many times for varying target rates, which requires huge computational resources; another is that many different encoded versions for varying target rates should be stored in the 3D video server, so the requirement for storage is tremendous.

In this paper, we propose a scalable bit allocation scheme between texture and depth views, which can solve the above two fundamental problems. In this scheme, both texture and depth views are encoded using quality scalable coding method, i.e., H.264/SVC Medium Grain Scalable (MGS) coding. For varying target rates, the optimal bit truncation points for both texture and depth views can be obtained using the proposed scheme. The contribution of this paper is manifold. Firstly, we propose to order the MGS enhancement layer packets (NAL units) based on their contribution to the reduction of the synthesized view distortion. Secondly, the information obtained in the depth view enhancement layer packet ordering step, i.e., the synthesized view distortion reduction of each MGS enhancement layer packet, is used to facilitate optimal bit allocation between texture and depth views. Therefore, the optimal bit allocation can be obtained using one simple formula for varying total target rates, with negligible computational complexity. To the best of authors’ knowledge, this is the first time to propose such a scalable bit allocation scheme between texture and depth views. In addition, we studied how the number of synthesized views affects the bit allocation between texture and depth views by analytical and experimental methods; and it is interesting to find that with the increase of the number of synthesized views, higher ratio of bit should be allocated for the texture views.

The rest of the paper is organized as follows. Related works are reviewed in Section II. A brief description of the synthesized view distortion is provided in Section III. In Section IV the proposed scalable bit allocation scheme is presented in details. In Section V some experimental results validating the proposed approach are given. Finally, some conclusions are drawn in Section VI.

II. RELATED WORK

A. Depth Coding and Bit Allocation

During the last decade, many depth compression techniques have been proposed to improve coding performance by exploiting the depth characteristics. One important technique uses the synthesized view distortion metric in the rate-distortion optimization process. In [20], motivated by the fact that the depth views are not perceived by the viewers but only supplement data for view synthesis, instead of using the distortion of the depth view itself, the authors proposed to use the distortion of the synthesized view in the rate-distortion optimized depth coding mode selection step; later in [21], the synthesized view distortion was modeled at pixel-level and in a more accurate way, which eventually led to better overall rate-distortion performance than that of [20].

Prior to the compression technique of the depth view itself, one more fundamental problem that needs to be addressed is how to allocate bit rate between the texture and depth views. A heuristic approach with fixed ratio (5:1) bit allocation between texture and depth views was used in [9]. Later,
Morvan [12] proposed a full search algorithm to find the optimal quantization parameter (QP) pair for texture and depth views. This algorithm assumes that a real view exists at the synthesized viewpoint, which is not always true; its tremendous computational complexity is another major problem. In [13], Liu proposed a distortion model to estimate the distortion of the synthesized views without the need of comparing the synthesized view with its corresponding real view. A fast bit allocation algorithm was proposed in [14] to reduce the complexity, where the allocation performance is comparable with that of [13]. In recent work [15], a region-based view synthesis distortion estimation approach and a general R-D property estimation model was proposed. The reported results in [15] show that it can provide better R-D performance than [13] with lower computational cost.

Recent advances on texture and depth bit allocation algorithms includes works [16]–[19]. In [16], [17] the concept of rate control was introduced into optimal bit allocation paradigm, whereas in [19] the allocation was carried out at macroblock granularity, which led to better rate-distortion performance than full search algorithm [12].

B. Scalable Coding

The H.264/SVC [11], which is an annex of the H.264/AVC standard [5], provides spatial, temporal, and quality scalability. SVC provides temporal scalability through the usage of hierarchical prediction structures, whereas spatial and quality scalability are supported by multilayer coding. Quality scalability is supported in two modes: coarse-grained scalability (CGS) and medium-grained scalability (MGS). When CGS is used, rate adaptation has to be performed on complete layer basis. However, MGS concept allows any enhancement layer network abstraction layer (NAL) unit to be discarded from a quality scalable bit stream in decreasing quality_id order, providing packet-based scalability. MGS is particularly useful when the server wishes to match the transmitted bit rate with the currently prevailing network throughput for a client in order to minimize the end-to-end delay. Conventionally, when the transmitted bit rate is not accurately matched with the prevailing throughput, clients have to buffer more data to compensate situations where the reception rate is temporarily lower than the media playout rate.

MGS splits a given enhancement layer of a given video frame into up to 16 MGS layers (also referred to as quality layers) [40]. In particular, MGS divides the transform coefficients, obtained through transform coding of a given block, into multiple groups. Each group is assigned to one MGS layer. For example, let us consider a $4 \times 4$ block, and use $w_i$ to denote the number of transform coefficients belonging to MGS layer $i$ within an enhancement layer, with $\sum_{i=1}^{16} w_i = 16$. The number of transform coefficients is also referred to as the weight of MGS layer. A MGS encoding can be represented by giving the weights in the vector form $W = [w_1, w_2, w_3, ..., w_{16}]$, whereby $w_i = 0$ if it is not specified. Figure 2 illustrates the splitting of the transform coefficients of a $4 \times 4$ block into three MGS layers with the weights $W = [3, 3, 10]$, i.e., $w_1 = 3, w_2 = 3$ and $w_3 = 10$, while other weights being 0.

Some pioneer works on adaptive scalable 3D video coding are [22] and [23]. In [22], the authors proposed an optimization framework for joint view and rate scalable coding of multi-view video content represented in the texture plus depth format. However, the view and rate embedded bitstream can only be constructed for a discrete set of transmission rates. In [23], the authors proposed a novel compression strategy for depth views that incorporates geometry information while achieving the goals of scalability and embedded representation. In this work, the texture views were not jointly considered for 3D video scalability.

C. 3D Hole Filling and Quality Assessment

A critical problem in the DIBR system is how to deal with the hole regions after 3D projection. Generally, the holes are generated because the occluded regions in the original view become visible in the virtual view after 3D projection. There are many solutions to address the disocclusion problem. One type of solutions preprocesses the depth views before DIBR, aiming to reduce the depth value difference along the boundary between the foreground and background, so that no disocclusion appears in the virtual view. Solutions of this type include using a symmetric Gaussian filter [24] and an asymmetric filter [25] to smooth the depth view. Another type of solutions is using background information for hole filling. In [26], a background sprite is generated by the original texture and synthesized images from the temporally previous frames for disocclusion filling. In our recent work [27], Gaussian Mixture Model (GMM) is used to generate a stable background for hole filling. One commonly used hole filling solution is view synthesis using the MVD video format. This approach exploits the fact that the invisible background part in the left view may be visible in the right view [28], and then the disoccluded regions in the virtual view warped from the left view can be filled with the background information.
from the right view, and vice versa. In the proposed scheme, MVD video format is used, so most of the disoccluded regions can be filled in complementary way using both the left and right views.

2D image/video quality metrics have been widely researched [29], and many quality assessment metrics, such as peak signal-to-noise ratio (PSNR), Structural SIMilarity (SSIM) [30], visual information fidelity (VIF) [31], have been proposed. Most of the 3D video quality assessment work in the literature is based on applying 2D video quality measures on the depth view as well as the stereoscopic views and then finding the combination of these measures that best correlates with the subjective scores [33]–[35]. In [36], 3VQM was proposed for objectively evaluating the quality of stereoscopic 3D videos generated by DIBR. In this method, the authors tried to derive an ideal depth estimation. Three measures, namely temporal outliers (TO), temporal inconsistencies (TI), and spatial outliers (SO) are used to constitute a vision-based quality measure for 3D DIBR-based videos. Subjective quality assessment and objective quality measurement of 2D video are mature fields. However, subjective assessment of 3D video quality is still facing many problems to solve before the performance of 3D video models can be properly evaluated in order to capture the essential QoE involved by such media [32]. Moreover, to the best of our knowledge, none of the 3D quality measures is commonly recognized to be superior from others. Thus, JCT-3V common test conditions [37] still use PSNR as the only metric to evaluate the synthesized view quality in the 3D video coding standardization process. Hence, based on this, in this paper we will use PSNR for our measures.

III. DISTORTION MODEL FOR SYNTHESIZED VIEW

In the proposed scheme, for both depth coding and scalable bit allocation between the texture and depth views, the distortion model for the synthesized view is required. The synthesized view distortion will be estimated without comparing the virtual view with its corresponding real view, because in practical applications the existence of the real view is not guaranteed. The synthesized view distortion model presented in [21], which will be used in this paper, is reviewed in Subsection.III-A. In this model, the distortion is modeled at pixel level, and it mimics the view synthesizing process with sub-pixel interpolation. It is worth noticing that the model in [21] has been adopted in JCT-3V. However, this model is based on the assumption that the virtual view is generated using one reference view. So in Subsection.III-B, this model is extended for MVD video format, where the virtual view is merged from wrapped views of both left and right reference views, and more than one virtual view is generated. Throughout the paper, subscript \( t \) and \( d \) indicate texture and depth information, respectively; \( l \) and \( r \) represent the left and right view, respectively.

A. Synthesized View Distortion Using One Reference View

In this subsection, our analysis will be based on the assumption that the virtual view is generated using one reference view. The distortion of the synthesized view will be the sum of squared distance (SSD) between two versions of the synthesized view. The first version, denoted by \( V_{x', y'} \), is synthesized from the original texture and depth views, whereas the other one is generated from the compressed version of the decoded texture and depth views, denoted by \( \tilde{V}_{x', y'} \). The SSD in this case is:

\[
SSD = \sum_{(x',y')} \left| V_{x',y'} - \tilde{V}_{x',y'} \right|^2
\]

\[
= \sum_{(x,y)} \left| f_w(C, D_{x,y}) - f_w(\tilde{C}, \tilde{D}_{x,y}) \right|^2
\]

where \( C \) and \( D \) indicate the original texture view and depth view, respectively; \( \tilde{C} \) and \( \tilde{D} \) denote the decoded texture and depth view, respectively; \( (x', y') \) is warped pixel position for the synthesized view \( V \) corresponding to \( (x, y) \) in \( C \) and \( D \) by the predefined warping function, \( f_w \), and \( (x, y) \) is the pixel inside the current non-synthesized block. As in [21], (1) can be further simplified as

\[
SSD = E_t + E_d(s)
\]

with

\[
E_t = \sum_{(x,y)} \left| f_w(C, D_{x,y}) - f_w(\tilde{C}, \tilde{D}_{x,y}) \right|^2
\]

\[
E_d(s) = \sum_{(x,y)} \left| f_w(C, D_{x,y}) - f_w(\tilde{C}, \tilde{D}_{x,y}) \right|^2
\]

where \( E_t \) denotes the distortion caused by the compression of the texture view, and \( E_d(s) \) denotes the distortion caused by the compression of the depth view. The parameter \( s \) in \( E_d(s) \) indicates the distance between the current and the rendered view which will affect the value of \( E_d(s) \).

In the 1-D parallel camera setting configuration, the 3D configuration used in this paper, the synthesized view distortion caused by the depth view \( E_d(s) \) can be further approximated as [21]:

\[
E_d(s) \approx \sum_{(x,y)} \left| \tilde{C}_{x, y} - \tilde{C}_{x - \Delta p(x, y, s), y} \right|^2
\]

where \( \Delta p \) denotes the translational horizontal rendering position error. It is already proven that it is proportional to depth view error:

\[
\Delta p(x, y, s) = \alpha(s) \cdot (D_{x,y} - \tilde{D}_{x,y})
\]

where \( \alpha(s) \) is a proportional coefficient determined by the following equation:

\[
\alpha(s) = \frac{f \cdot s}{255} \left( \frac{1}{Z_{near}} - \frac{1}{Z_{far}} \right)
\]

with \( f \) being the focal length, \( Z_{near} \) and \( Z_{far} \) being the values of the nearest and farthest depth of the scene, respectively. The value of \( E_d(s) \) can be approximated as (6). Finally, (6) is modified to use the original texture when the reconstructed texture is unavailable as (7). This is because in MVD video coding process, some of the depth views may be encoded before the associated texture views [38]. During encoding the depth views, the information of the reconstructed texture values may not be available. In (7), the value of

\[
\sum_{(x,y)} \left| (D_{x,y} - \tilde{D}_{x,y}) \left( |C_{x,y} - C_{x-1,y}| + |C_{x,y} - C_{x+1,y}| \right) \right|^2
\]
This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2014.2334011, IEEE Transactions on Circuits and Systems for Video Technology

\[ E_d(s) = \sum_{(x,y)} \left[ \frac{\Delta p(x, y, s)}{2} \right]^2 \left( \left| \tilde{C}_{x,y} - \tilde{C}_{x-1,y} \right| + \left| \tilde{C}_{x,y} - \tilde{C}_{x+1,y} \right| \right)^2 \]

\[ = \frac{s^2 f^2}{255^2} \left( \frac{1}{Z_{learn}} - \frac{1}{Z_{far}} \right)^2 \sum_{(x,y)} \left[ (D_{x,y} - \tilde{D}_{x,y}) \left( \left| \tilde{C}_{x,y} - \tilde{C}_{x-1,y} \right| + \left| \tilde{C}_{x,y} - \tilde{C}_{x+1,y} \right| \right) \right]^2 \]

\[ E_d(s) \approx \frac{s^2 f^2}{255^2} \left( \frac{1}{Z_{learn}} - \frac{1}{Z_{far}} \right)^2 \sum_{(x,y)} \left[ (D_{x,y} - \tilde{D}_{x,y}) \left( \left| C_{x,y} - C_{x-1,y} \right| + \left| C_{x,y} - C_{x+1,y} \right| \right) \right]^2 \]

where \( SSD^l \) and \( SSD^r \) are the distortion of the wrapped views from the left and right reference views, respectively; and they can be evaluated using (2).

At this stage, let us assume \( N \) total virtual views are generated between the left and right views. The \( N \) virtual views are evenly distributed, which means the distance between two neighboring virtual views is \( \frac{1}{N+1} \). From left to right the index of the virtual views are 1, 2, 3, ..., \( N \). For the \( i \)th virtual view, \( \zeta = \frac{N+1}{N+1} \); the left view caused distortion, \( SSD^l(i) \), can be evaluated using (2) as (11). Hence, for the \( N \) virtual views, the total distortion caused by the left view, \( SSD^l_{TV} \), can be evaluated as (12). Among the distortion \( SSD^l_{TV} \), the left texture-view-caused distortion in \( N \) virtual views, \( E^l_i \), could be evaluated as:

\[ E^l_i = \sum_{i=1}^{N} \left( \frac{N+1-i}{N+1} \right)^2 E_t \]

The left depth view caused distortion in \( N \) virtual views, \( E^l_d \), could be evaluated as:

\[ E^l_d = \sum_{i=1}^{N} \left( \frac{N+1-i}{N+1} \right)^2 \left( \frac{iL}{N+1} \right)^2 \frac{f^2}{255^2} \left( \frac{1}{Z_{learn}} - \frac{1}{Z_{far}} \right)^2 \psi \]

Let us assume \( \mu = \frac{\sum_{i=1}^{N} \left( \frac{N+1-i}{N+1} \right)^2}{\sum_{i=1}^{N} \left( \frac{N+1-i}{N+1} \right)^2 (\frac{iL}{N+1})^2} \). It is interesting to note that for \( N \in \{1, 3, 7\} \), \( \mu \) will be \{4, 6.58, 8.20\}. This means that with the increase of the number of synthesized views, the ratio of texture-view-caused distortion over depth-caused distortion is increasing. This observation also indicates that with the increase of the number of synthesized views, more bit should be allocated for the texture views. This interesting observation is also supported in the experimental section.

Similarly, for the \( N \) virtual views, the total distortion caused by the right view, \( SSD^r_{TV} \), the right texture (depth) view caused distortion, \( E^r_i (E^r_d) \) can be evaluated with the same methods. If the left and right views are encoded using the same coding parameters, which is the coding scheme used in
This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2014.2334011, IEEE Transactions on Circuits and Systems for Video Technology

\[ SSD_{TV}^i(\zeta) = \left( \zeta \tilde{V}_{x,y}^t + (1 - \zeta) \tilde{V}_{x,y}^r - (\zeta V_{x,y}^t + (1 - \zeta) V_{x,y}^r) \right)^2 \]

\[ = \zeta^2 \left( \tilde{V}_{x,y}^t - V_{x,y}^t \right)^2 + (1 - \zeta)^2 \left( \tilde{V}_{x,y}^r - V_{x,y}^r \right)^2 + 2\zeta(1 - \zeta) \left( \tilde{V}_{x,y}^t - V_{x,y}^t \right) \left( \tilde{V}_{x,y}^r - V_{x,y}^r \right) \] (9)

\[ SSD^i_T = \frac{N + 1 - i}{N + 1} SSD^i_V \]

\[ = \left( \frac{N + 1 - i}{N + 1} \right)^2 E_t + \left( \frac{N + 1 - i}{N + 1} \right)^2 \frac{f^2}{255^2} \left( \frac{1}{Z_{near}} - \frac{1}{Z_{far}} \right)^2 \psi \] (11)

\[ SSD^i_{TV} = \sum_{i=1}^{N} SSD^i_T \] (12)

\[ = \sum_{i=1}^{N} \left( \frac{N + 1 - i}{N + 1} \right)^2 E_t + \sum_{i=1}^{N} \left( \frac{N + 1 - i}{N + 1} \right)^2 \left( \frac{iL}{N + 1} \right)^2 \frac{f^2}{255^2} \left( \frac{1}{Z_{near}} - \frac{1}{Z_{far}} \right)^2 \psi \]

IV. PROPOSED SCALABLE BIT ALLOCATION SCHEME

In the proposed scalable bit allocation scheme, both texture and depth views are encoded using quality scalable coding method, H.264/SVC MGS [11] coding. In the current system, for simplicity, two viewpoints both including texture and depth views are used to represent 3D video content. However, it is straightforward to extend the system to more than two views. The H.264/SVC MGS encoded texture and depth views are stored in the 3D video server, including all the MGS base layer and enhancement layer packets, as depicted in Figure 1. A user in the heterogeneous network can start 3D video streaming service by informing the server its own link bandwidth, \( B \). Meanwhile, some auxiliary information is also provided to the server, such as the number of virtual views that are going to be synthesized at the receiver side and the position of each virtual view. These auxiliary information will affect the bit rate allocation process. Upon receiving the video streaming request, the 3D video server will decide the optimal MGS enhancement layer truncation points for both the texture and depth views, so that the total bit rate is within the user link bandwidth, \( B \).

The procedure of the proposed scalable bit allocation scheme works as follows.

1) Both texture and depth views are encoded using H.264/SVC MGS with two layers, i.e., MGS base layer and enhancement layer. The base layer and enhancement layer QP pairs are \( \{QP_b^t, QP_e^t\} \), \( \{QP_b^d, QP_e^d\} \) for texture and depth views, respectively. Here subscript \( b \) and \( e \) denote base and enhancement layer, respectively.

To improve the depth coding performance, in the rate-distortion optimized coding mode selection step, the synthesized view distortion metric (14) is used for both the MGS base layer and enhancement layer coding.

2) For both the texture and depth views, the MGS enhancement layer packets (NAL units) are ordered based on their contribution to the synthesized view distortion reduction to the whole GOP due to the drift distortion [41], [42]. For the texture and depth view, each enhancement layer packet’s contribution to the synthesized view distortion reduction to the current frame needs to be evaluated using (13) and (14), respectively. The synthesized view distortion reduction to the whole GOP is evaluated using the distortion drift model implemented in JSVM [41], [42]. According to (13), the synthesized view distortion is proportional to the texture view distortion. Thus, the enhancement layer packets of the texture view are ordered using existing method in H.264/SVC reference software, Joint Scalable video Model (JSVM) [43]. While for the depth view, the synthesized view distortion is not proportional to the depth view distortion. Thus, the enhancement layer packets of depth view should be ordered based on their contribution to the reduction of the synthesized view distortion to the GOP. The enhancement packet ordering information can be conveyed in the NAL unit header, through the syntax element priority_id, or using an optional Supplemental Enhancement Information (SEI) message. Meanwhile, each texture (depth) view enhancement packet’s size and its contribution to the synthesized view distortion reduction to the whole GOP is recorded as \( \{R_t^i, D_t^i\} \) \( \{R_d^i, D_d^i\} \) for the \( i \)th enhancement packet, where the enhancement packet

1051-8215 (c) 2013 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.
index $i$ is obtained after the MGS enhancement layer packet ordering. The use of $\{R^e_i, D^e_i\}$ and $\{R^d_i, D^d_i\}$ will be described in the following steps.

3) The 3D video server will decide the optimal MGS enhancement layer truncation points for both the texture and depth views.

4) The truncated texture and depth views, which suits for the user’s link bandwidth, will be sent to the user. The user equipment can synthesize virtual views.

A. Optimal Enhancement Layer Truncation Point Selection

It is important to select proper MGS enhancement layer truncation points for both texture and depth views, which affects the bit allocation performance. The detailed algorithm of selecting optimal truncation points is described as follows.

In the previous steps, we have got each enhancement packet’s rate and contribution to the reduction of the synthesized view distortion to the GOP: $\{R^e_i, D^e_i\}$ for the texture view and $\{R^d_i, D^d_i\}$ for the depth view. For the texture view, let us use $R_t(i) = \sum_{n=1}^{i} R^e_n$ and $D_t(i) = \sum_{n=1}^{i} D^e_n$ to denote the accumulated MGS enhancement rate and synthesized view distortion reduction, respectively. Similarly, for the depth views, let us use $R_d(i) = \sum_{n=1}^{i} R^d_n$ and $D_d(i) = \sum_{n=1}^{i} D^d_n$, to denote the accumulated MGS enhancement rate and synthesized view distortion reduction, respectively. To have a better understanding of the values, in Figure 3.a and Figure 3.c, we have reported $D_t(i)$ versus $R_t(i)$, and $D_d(i)$ versus $R_d(i)$ for the Balloons and Poznan_Street sequences. It is very interesting to note that, for the texture view, how $D_t(i)$ changes with $R_t(i)$ could be closely approximated by the following exponential function:

$$D_t(R_t) = a_t \cdot \left(1 - e^{-\frac{R_t}{b_t}}\right) + c_t,$$

(15)

with $a_t > 0$ and $b_t > 0$; here $D_t$ and $R_t$ are the short form of $D_t(i)$ and $R_t(i)$, respectively. Similarly, for the depth views, how $D_d(i)$ changes with $R_d(i)$ could be approximated by the following exponential function:

$$D_d(R_d) = a_d \cdot \left(1 - e^{-\frac{R_d}{b_d}}\right) + c_d,$$

(16)

with $a_d > 0$ and $b_d > 0$; $D_d$ and $R_d$ are the short form of $D_d(i)$ and $R_d(i)$, respectively. The parameters $\{a_e, b_e, c_e\}$ and $\{a_d, b_d, c_d\}$ could be obtained with least square curve fitting [44]. It is worth noticing that only the curve fitting results for Balloons and Poznan_Street sequences are reported herein. Nevertheless, for other sequences, e.g., Newspaper, Lovebird1, BookArrival, Poznan_Hall2, similar results have been obtained. It is important to note that one synthesized virtual view is generated in Figure 3. Nevertheless, the number of synthesized views, $N$, will not affect this finding, because according to (13) and (14) $N$ will only change the scaling factor. To the best of our knowledge, this is the first time in literature to study the relationship between the reduction of the synthesized view distortion and the accumulated MGS enhancement rate of the texture and depth views.

At this stage, let us assume the user link bandwidth is $B$, and there is two views (left and right view) both including texture and depth views. It is reasonable to allocate the same amount of rate ($B/2$) to both the left and right views, this is because the two views typically share similar characteristics. For simplicity, the following analysis will be based on one view. Let us use $R_{00}$ and $R_{0d}$ to denote the base layer rate of one texture and depth view, respectively; so the maximum allocated rate for the MGS enhancement layer of one view should be $B_E = B/2 - R_{00} - R_{0d}$. In fact, based on (15) and (16) and the fact that the synthesized view distortion caused by texture and depth views is additive [21], the synthesized view distortion (2) could be rewritten as follows:

$$SSD(R_t, R_d) = E_0 - D_t(R_t) - D_d(R_d) = E_0 - \left( a_t \cdot \left(1 - e^{-\frac{R_t}{b_t}}\right) + c_t \right) - \left( a_d \cdot \left(1 - e^{-\frac{R_d}{b_d}}\right) + c_d \right),$$

(17)

where $E_0$ is the synthesized view distortion when no MGS enhancement layer packet is used for both texture and depth views (only base layer packets used). Then the problem becomes to find the optimal $\{R_t, R_d\}$ that could minimize $SSD(R_t, R_d)$, which could be mathematically written as:

$$\min_{R_t, R_d} SSD(R_t, R_d) \quad \text{subject to} \quad R_t + R_d \leq B_E$$

(18)

It is worth mentioning, in (18) the mean squared error (MSE) is used as the quality metric. For more sophisticated quality metric, 3VQM [36] might be a good choice for the synthesized views quality assessment, which will be left for the future research work.

![Figure 3](image-url)
For the texture view based on the fact that \( a_t > 0, b_t > 0 \), we could have \( \frac{\partial D_t(R_t)}{\partial R_t} = \frac{a_t}{b_t} e^{-\frac{n_t}{\sigma_t}} > 0 \) and \( \frac{\partial^2 D_t(R_t)}{\partial R_t^2} = -\frac{a_t}{\sigma_t^2} e^{-\frac{n_t}{\sigma_t}} < 0 \). Similarly, for the depth view we could have \( \frac{\partial D_d(R_d)}{\partial R_d} > 0, \frac{\partial^2 D_d(R_d)}{\partial R_d^2} < 0 \). At this stage, we can get the following properties for \( SSD(R_t, R_d) \):

\[
\frac{\partial SSD(R_t, R_d)}{\partial R_t} = -\frac{\partial D_t(R_t)}{\partial R_t} < 0 \tag{19}
\]

\[
\frac{\partial^2 SSD(R_t, R_d)}{\partial R_t^2} = -\frac{\partial^2 D_t(R_t)}{\partial R_t^2} > 0 \tag{20}
\]

\[
\frac{\partial SSD(R_t, R_d)}{\partial R_d} = -\frac{\partial D_d(R_d)}{\partial R_d} < 0 \tag{21}
\]

\[
\frac{\partial^2 SSD(R_t, R_d)}{\partial R_d^2} = -\frac{\partial^2 D_d(R_d)}{\partial R_d^2} > 0 \tag{22}
\]

Based on (19)-(22), \( SSD(R_t, R_d) \) is a concave function of both \( R_t \) and \( R_d \), so the constrained optimization problem (18) can be solved by means of the standard Lagrangian optimization by minimizing the cost function:

\[
J = SSD(R_t, R_d) + \lambda (R_t + R_d) \tag{23}
\]

where \( \lambda \) is the Lagrangian multiplier. So by imposing \( \nabla J = 0 \) we get:

\[
\frac{\partial J}{\partial R_t} = -\frac{a_t}{b_t} e^{-\frac{n_t}{\sigma_t}} + \lambda = 0 \tag{24}
\]

\[
\frac{\partial J}{\partial R_d} = -\frac{a_d}{b_d} e^{-\frac{n_d}{\sigma_d}} + \lambda = 0 \tag{25}
\]

From (24) and (25), we can conclude that in order to minimize \( J \), the following condition must be satisfied:

\[
\frac{a_t}{b_t} e^{-\frac{n_t}{\sigma_t}} = \frac{a_d}{b_d} e^{-\frac{n_d}{\sigma_d}} = \lambda \tag{26}
\]

Hence, by jointly solving equations (26) and \( R_t + R_d = B_E \), the optimal rate for the MGS enhancement layer of texture and depth views should be as follows:

\[
R_t = \frac{1}{b_t + b_d} \left( b_t B_E + b_d b_d \ln \frac{a_t b_d}{b_t a_d} \right) \tag{27}
\]

\[
R_d = \frac{1}{b_t + b_d} \left( b_d B_E - b_t b_t \ln \frac{a_t b_d}{b_t a_d} \right) \tag{28}
\]

By taking the fact that the enhancement layer rate should be in the range of \([0, B_E]\), the final allocated rate for the enhancement layer of texture view should be:

\[
R_t' = \begin{cases} 
R_t, & \text{if } 0 \leq R_t \leq B_E \\
0, & \text{if } R_t < 0 \\
B_E, & \text{if } R_t > B_E 
\end{cases} \tag{29}
\]

Similarly, the final allocated rate for the enhancement layer of depth view should be:

\[
R_d' = \begin{cases} 
R_d, & \text{if } 0 \leq R_d \leq B_E \\
0, & \text{if } R_d < 0 \\
B_E, & \text{if } R_d > B_E 
\end{cases} \tag{30}
\]

It is worth noticing that for (29) and (30), \( R_t' + R_d' = B_E \) holds for all the cases.

V. EXPERIMENTAL RESULTS

In the experiments, 6 typical 3D video sequences are used: BookArrival [45], Newspaper [46], Lovebird1 [46], Balloons [47], Poznan Street [48] and Poznan Hall2 [48]. The general experimental setting is listed in Table I unless otherwise noted. The proposed algorithm is implemented based on H.264/SVC reference software JSVM 9.19.15. A hierarchical prediction structure is used, with the GOP size being 8. The left view and right view are independently encoded using H.264/SVC MGS coding tool, and the virtual view is synthesized using View Synthesis Reference Software (VSRS 3.5) [49]. The default hole filling algorithm implemented in VSRS is used for the occluded regions. The transform coefficients of a macroblock are split into six MGS layers with the weights \( W = [1, 2, 2, 3, 4, 4] \). We selected this weight vector because it was reported that these MGS weights led to competitive rate-distortion performance [40]. The baseline profile is used for the MGS base layer, while the scalable baseline profile is used for the enhancement layer, with \( 8 \times 8 \) transform disabled at both layers. The enhancement layer is used in motion estimation and motion prediction for non-key pictures in MGS layers.

A. Performance of Proposed Depth Coding and MGS Packet Ordering

In the first experiment, we tested the effects of the MGS base layer and enhancement layer QP pair \( \{QP_b^p, QP_e^p\} \) and \( \{QP_b^d, QP_e^d\} \) on the coding performance. The depth views of the Newspaper sequence are encoded using different QP pairs: \{35, 25\}, \{35, 27\} and \{35, 30\}, with QP difference between the two layers being 10, 8 and 5, respectively. In this experiment, the texture view is not compressed. View 2 and 4 are used to synthesize virtual view 3. The synthesized view quality versus depth bit rate is reported in Figure 4. It is noted that having large QP difference between the base layer and enhancement layer, i.e., QP pair \{35, 25\}, leads to better depth coding performance. Meanwhile, with large QP difference, the MGS bit rate range is large, which can provide more flexibility for rate adaptation. It is interesting to note that, when the MGS bitstream is truncated close to the base layer point, i.e., rate lower than 1000 kbps, QP pair \{35, 30\} leads to the best performance, with QP pair \{35, 25\} being the worst case. This is because the mismatch error is large when the QP difference is large, and the effects of mismatch error is more obvious when the truncation point is close to the base layer. Nevertheless, based on the whole bit rate range performance, generally, QP pair \{35, 25\} outperforms other two cases. Similar results are obtained for the texture view. Meanwhile, we noticed that by using QP pair \{35, 25\} for both the texture and depth views, \( 0 < R_t' < B_E \) and \( 0 < R_d' < B_E \) holds for various values of \( B_E \), which also indicates the rationality of using this QP pair for the texture and depth views. Hence, in the following experiments, MGS base layer and enhancement layer QP pair \{35, 25\} will be used for both the texture and depth views if not otherwise noted.
views are not compressed. The rate-distortion performance
"SVDC"
used, whereas for the case when
\{ QP \text{ pair} \}
are compared. In the first approach, during the depth coding
process, the synthesized view distortion metric is not used
for both rate-distortion optimized coding mode selection and
MGS enhancement layer packet ordering. It is worth mentioning that
the fixed ratio (1:5) bit allocation between depth and texture
is used at either stage. For the curve "Full Search"
which is the upper bound for bit rate allocation performance.

In the second set of experiments, the advantage of the proposed
depth coding and MGS enhancement packet ordering
methods are demonstrated. To do this, three different depth
coding and bit extraction approaches using H.264/SVC MGS
are compared. In the first approach, during the depth coding
process, the synthesized view distortion metric is not used
in the rate-distortion optimized coding mode selection and
MGS enhancement layer packet ordering steps; whereas
the traditional MSE is used [11], so this approach is
named “SVDC OFF+SVDO OFF”. Here "SVDC" stands for
synthesized view distortion metric based rate-distortion
optimized coding mode selection, whereas “SVDO” stands for
synthesized view distortion metric based MGS enhancement
layer packet ordering. In the second approach, the synthesized
view distortion metric is applied in the rate-distortion
optimized coding mode selection step, whereas the newly
proposed synthesized view distortion based MGS enhancement
layer packet ordering is not used, so we call this approach as
“SVDC ON+SVDO OFF”. These two approaches are
used as the benchmarks for the proposed approach, “SVDC
ON+SVDO ON”, where the synthesized view distortion metric
is used for both coding mode selection and MGS enhancement
layer packet ordering. MGS base layer and enhancement layer
QP pair \{35,25\} is used for depth coding when “SVDC” is
used, whereas for the case when “SVDC” is not used, other
QP pairs that lead to similar bit rate are adopted. Texture
views are not compressed. The rate-distortion performance
comparison for the 6 video sequences are reported in Figure
5. The proposed approach outperforms other two approaches
for all the six video sequences. Comparing with the “SVDC
OFF+SVDO OFF”, the PSNR gain of the proposed approach is
around 1-7 dB. The BD-Rate and BD-PSNR [50] results
compared with “SVDC ON+SVDO OFF ” are listed in Table II,
the average BD-PSNR gain is 0.31 dB, and the BD-
Rate is −8.95%. All the results demonstrate the importance
of using the synthesized view distortion metric for both
rate-distortion optimized coding mode selection and MGS
enhancement layer packet ordering. It is worth mentioning that
for the start and end points in Figure 5, the performance of
the proposed approach and “SVDC ON+SVDO OFF” is the
same, because no MGS enhancement packets and all MGS
enhancement packets, respectively, are used, so MGS packet
ordering algorithm has no effect for these cases.

B. Performance of Proposed Scalable Bit Allocation

In Figure 6, the performance of the proposed scalable bit
allocation scheme is reported where we compared it with
the fixed ratio (1:5) bit allocation between depth and texture
views. In order to visualize the gain of each step, three
reference curves are reported: “SVDC ON+SVDO ON (1:5)”
means that for both the MGS depth coding and enhancement
packet ordering, the synthesized view distortion metric is used;
“SVDC ON+SVDO OFF (1:5)” means that the SVD metric
is used for depth coding but not for packet ordering; “SVDC
OFF+SVDO OFF (1:5)” means that the SVD metric is not
used at either stage. For the curve “Full Search”, the allocated
depth rates obtained using (30) are multiplied with a scaling
factor of \{0.7, 0.8, 0.9, 1.0, 1.1, 1.2, 1.3\}, and this curve is
generated by selecting the best points in terms of synthesized
view PSNR among all the scaled depth view rates. This curve
is used to determine an approximated full-search result [12],
which is the upper bound for bit rate allocation performance.

QP pair \{35,25\} is used for both texture and depth coding

![Figure 4. Effects of having different QP pairs (MGS base layer and enhancement layer) for depth coding. Newspaper sequence is used; texture view is not compressed.](image-url)
Figure 5. Synthesized view PSNR versus bit rate of depth views for three different approaches; (a) Balloons, (b) BookArrival, (c) Lovebird1, (d) Newspaper, (e) Poznan_Street, (f) Poznan_Hall2.

Figure 6. Synthesized view PSNR versus total bit rate of texture and depth views; (a) Balloons, (b) BookArrival, (c) Lovebird1, (d) Newspaper, (e) Poznan_Street, (f) Poznan_Hall2.
This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCVT.2014.2334011, IEEE Transactions on Circuits and Systems for Video Technology

The allocated bit rate ratio between depth view and texture view, and BD-Rate and BD-PSNR results by comparing the proposed scalable bit allocation scheme with “SVDC ON+SVDO ON (1:5)”.

<table>
<thead>
<tr>
<th>Sequence</th>
<th>Ratio</th>
<th>BD-Rate</th>
<th>BD-PSNR</th>
</tr>
</thead>
<tbody>
<tr>
<td>Balloons</td>
<td>0.249</td>
<td>−0.75%</td>
<td>0.05 dB</td>
</tr>
<tr>
<td>BookArrival</td>
<td>0.380</td>
<td>−4.55%</td>
<td>0.12 dB</td>
</tr>
<tr>
<td>Lovebird1</td>
<td>0.187</td>
<td>−2.27%</td>
<td>0.11 dB</td>
</tr>
<tr>
<td>Newspaper</td>
<td>0.314</td>
<td>−0.74%</td>
<td>0.06 dB</td>
</tr>
<tr>
<td>Poznan Street</td>
<td>0.209</td>
<td>0.60%</td>
<td>−0.02 dB</td>
</tr>
<tr>
<td>Poznan Hall2</td>
<td>0.292</td>
<td>−2.25%</td>
<td>0.07 dB</td>
</tr>
<tr>
<td>Average</td>
<td>0.205</td>
<td>−1.65%</td>
<td>0.07 dB</td>
</tr>
</tbody>
</table>

Table IV

BD-Rate and BD-PSNR results by comparing the proposed scheme with “Full Search”.

<table>
<thead>
<tr>
<th>Sequence</th>
<th>BD-Rate</th>
<th>BD-PSNR</th>
</tr>
</thead>
<tbody>
<tr>
<td>Balloons</td>
<td>0.99%</td>
<td>−0.03 dB</td>
</tr>
<tr>
<td>BookArrival</td>
<td>0.04%</td>
<td>−0.00 dB</td>
</tr>
<tr>
<td>Lovebird1</td>
<td>0.06%</td>
<td>0.00 dB</td>
</tr>
<tr>
<td>Newspaper</td>
<td>0.00%</td>
<td>0.00 dB</td>
</tr>
<tr>
<td>Poznan Street</td>
<td>0.79%</td>
<td>−0.02 dB</td>
</tr>
<tr>
<td>Poznan Hall2</td>
<td>0.03%</td>
<td>0.00 dB</td>
</tr>
<tr>
<td>Average</td>
<td>0.32%</td>
<td>−0.01 dB</td>
</tr>
</tbody>
</table>

When “SVDC” is used, whereas for the case when “SVDC” is not used, the QP pairs that lead to similar bit rate is adopted for depth views. From Figure 6, we can notice that the proposed scheme outperforms the three reference schemes using fix ratio allocation. “SVDC ON+SVDO ON (1:5)” performs best among the three fix ratio allocation schemes. It is also very interesting to note that, the curve of the proposed scheme is almost overlapping with that of the “Full Search”. The allocated bit rate ratio between depth view and texture view, and the BD-Rate and BD-PSNR gains for the proposed scheme over “SVDC ON+SVDO ON (1:5)” are listed in Table III. The average bit rate reduction of the proposed scheme over “SVDC ON+SVDO ON (1:5)” is 1.65%. It is noted that for some video sequences, e.g., Lovebird1, the allocated depth bit rate is less than 20% of the texture view; whereas for some video sequences, the allocated depth bit rate is more than that 20% of the texture. This observation serves to demonstrate the robustness and effectiveness of the proposed scalable bit allocation scheme. The BD-Rate and BD-PSNR for the proposed scheme over “Full Search” are listed in Table IV, where the average BD-Rate of the proposed scheme over “Full Search” is only 0.32%. The precision of the proposed scalable bit allocation scheme is demonstrated by the fact that performance of the proposed bit allocation scheme is very close to the upper bound curve “Full Search”.

In Figure 6, the reported results are for the case that only one synthesized view is generated. In order to demonstrate the effectiveness of the proposed bit allocation method for more than one synthesized view, in Figure 7 the bit allocation performance is reported when 3 and 7 synthesized views are generated. The generated synthesized views are evenly distributed between the left and right views (i.e., view 6 and view 8) for the Lovebird1 sequence. It is seen that with 3 and 7 synthesized views, the proposed scalable bit allocation performance is still quite close to that of “Full Search”, and the BD-Rate over “SVDC ON+SVDO ON (1:5)” are −3.08% and −3.44% for 3 and 7 synthesized views, respectively, which are larger than the gain of one synthesized view (−2.22%). All these results indicate the accuracy of the proposed bit allocation algorithm. It is also observed that the average bit rate ratios between depth view and texture view are {0.157, 0.137, 0.129} for generating {1, 3, 7} synthesized views. These results confirm our argument that with the increase of the number of synthesized views, more bit should be allocated for the texture views.

C. Examples of Subjective Quality

Besides the objective results, some examples of subjective quality are also reported herein. Firstly, the results of the proposed depth view coding and MGS enhancement packet ordering are reported in Figure 8. In this testing, texture views are not compressed, whereas depth view coding and bit extraction approaches include “SVDC OFF+SVDO OFF”, “SVDC ON+SVDO OFF” and “SVDC ON+SVDO ON”. The depth view bit rate is the same for the three approaches (1421 kbps). To better visualize the difference of the three frames, some regions, i.e., the edges of the balloons, are zoomed in for better display. It is clear that the subjective quality of “SVDC ON+SVDO ON” is the best, with the balloon edges well protected. Secondly, the subjective results of the proposed bit allocation scheme are reported in Figure 9. In this comparison, synthesized frames of BookArrival sequence are reported for “SVDC OFF+SVDO OFF(1:5)”, “SVDC ON+SVDO ON(1:5)” and “Proposed method”. To have fair comparison, for all the three approaches, the total bit rate is 6758 kbps. It is noted that the proposed bit allocation scheme leads to the best subjective quality, especially for the zoomed in regions.

D. Computational Complexity Analysis

The computational complexity of the proposed scalable bit allocation scheme is moderate. Firstly, for synthesized view distortion based depth MGS enhancement packet ordering, the only difference with normal MGS enhancement packet ordering is using (14) to evaluate distortion instead of using mean squared error (MSE) of depth view. Thus, the computational complexity of this step should be equivalent with MSE based MGS packet ordering process. Regarding
the computational complexity of normal MGS enhancement packet ordering, please refer to [41], [42]. Secondly, for the optimal bit allocation step, least square fitting is used to get parameter value \( \{a_t, b_t, c_t\} \) and \( \{a_d, b_d, c_d\} \), which needs a computational complexity of \( O(NM^2) \), where \( N \) is the number of samples, and \( M \) is the number of parameters for fitting. In our case, \( N \) equals to the number of frames in the video sequence multiply the MGS layer number (6 in our case); \( M \) equals to 3. The curve fitting although has complexity \( O(NM^2) \), it is negligible in comparison with the video coding complexity, because the complexity of coding is proportional to the pixel number in each frame in addition to the frame number.

VI. CONCLUSIONS AND FUTURE WORK

In this paper, a scalable bit allocation scheme for texture and depth views has been proposed. In this scheme, both the texture and depth views are encoded using the Medium Grain Scalable (MGS) tool of H.264/SVC. The optimal truncation points for the texture and depth views can be found using this scheme. This kind of scalable bit allocation is very important for the 3D video server, which provides 3D video streaming service for users with different link bandwidth in the heterogeneous networks. Another merit of this scheme is that the information generated in the MGS enhancement packet ordering process is exploited during the bit allocation stage, so the optimal truncation points for the texture and depth views can be obtained using one simple formula for varying total target rates. Experimental results have demonstrated the effectiveness of the proposed scalable bit allocation scheme.

In the current work, in order to use the H.264/SVC MGS tool, different views are compressed independently. This is because the current multi-view video coding standard, i.e., Multi-view Video Coding, does not support quality scalable coding. In the future, we are going to investigate the proposed framework in a quality-scalable multi-view coding system where inter-view prediction is supported.

ACKNOWLEDGMENT

The authors would like to thank Associate Editor Prof. Shahram Shirani and the anonymous reviewers for their valuable comments and suggestions.

REFERENCES

Miska M. Hannuksele (M03) received his Master of Science degree in engineering and Doctor of Science degree in technology from Tampere University of Technology, Finland, in 1997 and 2010, respectively. He has been with Nokia since 1996 in different roles including research manager/leader positions in the areas of video and image compression, end-to-end multimedia systems, as well as sensor signal processing and context extraction. Currently he works as Distinguished Scientist in Multimedia Technologies in Nokia Research Center, Tampere, Finland. He has published more than 100 conference papers, more than 25 journal papers, and hundreds of standardization contributions in JCT-VC, JCT-3V, JVT, MPEG, 3GPP, and DVB. He has granted patents from more than 70 patent families. His research interests include video compression and multimedia communication systems. Dr. Hannuksele received an award of the best doctoral thesis of Tampere University of Technology in 2009 and Scientific Achievement Award nominated by Tampere University of Technology in 2010. He is an author of the paper that received the best paper award of 2012 Visual Communications and Image Processing conference. He has been an Associate Editor in IEEE Transactions on Circuits and Systems of Video Technology since 2010.

Tammam Tillo (M05-SM12) received the Engineer Diploma in electrical engineering from the University of Damascus, Syria, in 1994, and the Ph.D. degree in electronics and communication engineering from Politecnico di Torino, Italy, in 2005. In 2004, he served as a Visiting Researcher at Ecole Polytechnique Federale de Lausanne (EPFL), Switzerland, and from 2005 to 2008 he worked as a Postdoctoral Researcher at the Image Processing Lab of Politecnico di Torino. For few months he was Invited Research Professor at the Digital Media Lab, SungKyunKwan University, Republic of Korea, before joining Xi’an Jiaotong-Liverpool University (XJTLU), China, in 2008. He was promoted to full professor in 2012. From 2010 to 2013 he was the Acting Head of the Department of Electrical and Electronic Engineering at XJTLU University, and the was the Acting Head of the Department of Computer Science and Software Engineering from 2012 to 2013. He serves as an expert evaluator for several national-level research programs. His research interests are in the areas of robust transmission of multimedia data, image and video compression, and hyperspectral image compression.

Moncef Gabbouj received his BS degree in electrical engineering in 1985 from Oklahoma State University, Stillwater, and his MS and PhD degrees in electrical engineering from Purdue University, West Lafayette, Indiana, in 1986 and 1989, respectively. Dr. Gabbouj is an Academy Professor with the Academy of Finland. He held several visiting professorships at different universities. He holds a permanent position of Professor of Signal Processing at the Department of Signal Processing, Tampere University of Technology. His research interests include multimedia content-based analysis, indexing and retrieval, machine learning, nonlinear signal and image processing and analysis, voice conversion, and video processing and coding. Dr. Gabbouj is a Fellow of the IEEE and member of the Finnish Academy of Science and Letters. He is currently the Chairman of the IEEE CAS TC on DSP and committee member of the IEEE Fourier Award for Signal Processing. He served as Distinguished Lecturer for the IEEE CASS. He served as associate editor and guest editor of many IEEE, and international journals. Dr. Gabbouj was the recipient of the 2012 Nokia Foundation Visiting Professor Award, the 2005 Nokia Foundation Recognition Award, and several Best Paper Awards. He published over 550 publications and supervised 38 doctoral theses.

Ce Zhu received the B.S. degree from Sichuan University, Chengdu, China, and the M.Eng and Ph.D. degrees from Southeast University, Nanjing, China, in 1989, 1992, and 1994, respectively, all in electronic and information engineering. Dr. Zhu is currently with the School of Electronic Engineering, University of Electronic Science and Technology of China, China. He had been with Nanyang Technological University, Singapore, from 1998 to 2012, where he was promoted to Associate Professor in 2005. Before that, he pursued postdoctoral research at Chinese University of Hong Kong in 1995, City University of Hong Kong, and University of Melbourne, Australia, from 1996 to 1998. He has held visiting positions at Queen Mary, University of London (UK), and Nagoya University (Japan). His research interests include image/video coding, streaming and processing, 3D video, joint source-channel coding, multimedia systems and applications. Dr. Zhu serves on the editorial boards of seven international journals, including as an Associate Editor of IEEE Transactions on Circuits and Systems for Video Technology, IEEE Transactions on Broadcasting, IEEE Signal Processing Letters, Editor of IEEE Communications Surveys and Tutorials, and Area Editor of Signal Processing: Image Communication (Elsevier). He has served on technical/program committees, organizing committees and as track/area/session chairs for about 60 international conferences. He received 2010 Special Service Award from IEEE Broadcast Technology Society, and is an IEEE BTS Distinguished Lecturer (2012-2014).

Yao Zhao (M’06-SM’12) received the B.S. degree from Fuzhou University, Fuzhou, China, in 1989, and the M.E. degree from Southeast University, Nanjing, China, in 1992, both from the Radio Engineering Department, and the Ph.D. degree from the Institute of Information Science, Beijing Jiaotong University (BJTU), Beijing, China, in 1996. He became an Associate Professor at BJTU in 1998 and became a Professor in 2001. From 2001 to 2002, he was a Senior Research Fellow with the Information and Communication Theory Group, Faculty of Information Technology and Systems, Delft University of Technology, Delft, The Netherlands. He is currently the Director of the Institute of Information Science, BJTU. His current research interests include image/video coding, digital watermarking and forensics, and video analysis and understanding. He is currently leading several national research projects from the 973 Program, 863 Program, and the National Science Foundation of China. He serves on the editorial boards of several international journals, including as an Associate Editor of the IEEE Transactions on Cybernetics, Associate Editor of the IEEE Signal Processing Letters, Area Editor of Signal Processing: Image Communication (Elsevier), and Associate Editor of Circuits, System, and Signal Processing (Springer). He was named a Distinguished Young Scholar by the National Science Foundation of China in 2010, and was elected as a Chang Jiang Scholar of Ministry of Education of China in 2013.