Texture-Distortion-Constrained Joint Source-Channel Coding of Multi-View Video Plus Depth-Based 3D Video

Pan Gao, *Member, IEEE*, Wei Xiang, *Senior Member, IEEE*, and Dong Liang

***Abstract*— A novel joint source and channel coding scheme tailored to 3D video is proposed in this paper to minimize the end-to-end view synthesis distortion within a given total bit rate for both texture and depth as well as a maximum tolerable distortion constraint for texture. First, we formulate a joint texture and depth coding mode selection strategy for error-resilient source coding of multi-view video plus depth-based 3D video through using the Lagrange multiplier method. Then, by considering the effect of residual errors after channel coding, we evolve to a more general formulation that jointly optimizes error-resilient source coding and channel coding in an integrated manner for unequal error protection between texture and depth, for which a theoretic solution using a proposed dual-trellis is derived. Finally, we extend the general formulation by including the texture distortion constraint. We show how to optimize the view synthesis quality while simultaneously catering to the texture quality constraint. Experimental results demonstrate the proposed algorithm has much better performance than existing related work.**

***Index Terms*— Joint source and channel coding, joint texture and depth map coding, texture distortion constraint, 3D video transmission.**

I. Introduction

3D VIDEO transport has become increasingly prevalent in visual communications due to the increased demand for applications of 3D tele-immersion and 3D-video-on-demand [1], [2]. However, the best effort design of the current Internet makes it extremely difficult to provide the quality of service and quality of experience [3]. Further, the time-varying wireless networks will generate additional quality bottlenecks for 3D viewing experience. In addition, compared to 2D video only having one single bit stream, texture plus depth-based 3D video contains two types of bit streams, i.e, one bit stream of texture video and another bit stream of associated depth map. Consequently, depth-based 3D video coder design for error robustness is facing new challenges.

In order to efficiently store and transmit the 3D video data, several standards for 3D video coding have been established, *e.g.*, the 3D extension of AVC (3D-AVC) [4] and 3D extension of HEVC (3D-HEVC) [5]. As in other coding scenarios, 3D video can be compressed by taking advantage of temporal and inter-view redundancies in each texture video and depth. Further, the coding efficiency for 3D video can be improved by exploiting an additional redundancy associated with the similarity between texture and depth, *e.g.*, view synthesis prediction (VSP) [6] and motion vector sharing (MV Sharing) [7]. However, due to extensive use of prediction techniques in video signal dependence removal, numerous kinds of error propagation would occur during transmission of the compressed 3D video. For example, since motion and disparity compensation are typically employed in compression, transmission errors may propagate temporally and spatially to the subsequent frames which depend on the current loss-occurring frame. Further, when inter-component prediction is employed to reduce statistical redundancies, loss of packets in coded texture or depth may cause additional error propagation between the texture and depth. Finally, since the virtual views are synthesized by the loss-distorted texture and depth at the decoder, the accumulated errors in the texture and depth will further propagate to the synthesized views along the warping path. These sophisticated error propagation may combine together, and then lead to substantial quality degradation on both the coded and synthesized views in 3D video. Therefore, it is highly desirable for the 3D video encoder to provide error robustness and correction capability to protect the transmitted video and depth data from channel errors.

To this end, we propose a joint source and channel coding scheme tailored to 3D video transmission. We address two key problems in this paper. Firstly, for a given overall bit rate, how to optimally allocate the source coding rate and channel coding rate over texture and depth of 3D video? Secondly, with additional texture distortion constraint, how the considered joint source and channel coding scheme is performed again for 3D video? Generally, these two problems consist of three basic tasks: finding an optimal bit allocation between source

Manuscript received June 18, 2018; revised September 17, 2018; accepted October 19, 2018. Date of publication October 24, 2018; date of current version October 29, 2019. This work was supported in part by the National Key R&D Program of China under Grant 2017YFB0802300, in part by the Natural Science Foundation of China under Grants 61701227 and 61601223, in part by the Natural Science Foundation of Jiangsu Province of China under Grants BK20170806 and BK20150756, and in part by the Science Foundation Ireland (SFI) under Grant 15/RP/2776. This paper was presented in part at the IEEE International Conference on Multimedia and Expo, Hong Kong, July 2017 [24]. This paper was recommended by Associate Editor H. Schwarz. *(Corresponding author: Pan Gao; Wei Xiang.)*

P. Gao is with the College of Computer Science and Technology, Nanjing University of Aeronautics and Astronautics, Nanjing 211106, China, and also with the School of Computer Science and Statistics, Trinity College Dublin, D02 PN40 Dublin 2, Ireland (e-mail: gaopan.1005@gmail.com).

W. Xiang is with the College of Science and Engineering, James Cook University, Cairns, QLD 4870, Australia (e-mail: wei.xiang@jcu.edu.au).

D. Liang is with the College of Computer Science and Technology, Nanjing University of Aeronautics and Astronautics, Nanjing 211106, China (e-mail: liangdong@nuaa.edu.cn).

Color versions of one or more of the figures in this article are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TCSVT.2018.2877903

and channel coding for given channel loss characteristics; designing a joint bit allocation scheme between texture and depth coding to achieve the target bit rate and error robustness of source coding; designing channel coding for texture and depth under the associated channel coding rate to achieve the required error correction capability.[1]

Toward these problems, we firstly formulate the first problem as a joint source and channel coding of 3D video with the objective that the overall expected view synthesis distortion is minimized. Although the proposed joint source and channel coding framework resembles several other schemes recently proposed for 3D video coding (e.g., [21]), the source coding parameters to be optimized are significantly different. Specifically, we consider a joint selection of texture and depth coding modes for source coding in the overall framework. As the coding modes are generally determined during real encoding, we thus need to perform the proposed joint source and channel coding *online*, which is different from almost all the related work for which the optimal coding parameters (i.e., quantization parameters) are straightforwardly searched over the set of the admissible parameters prior to encoding. Since selectively distributing the coding modes between texture and depth can provide error resilience for 3D video streaming, the proposed framework is able to balance the compression efficiency with error robustness. Further, to allow the proposed framework to balance the redundant bits induced by error-resilient source coding and channel coding, we restrict ourselves to performing joint unequal error-resilient source and unequal channel coding for texture and depth in a single step. Then, we propose an approach that uses operational rate-distortion theory to solve this joint problem, where a dual-trellis model is specifically designed for a tractable solution.

Besides the above differences to the related work, another major contribution of this paper is that we generalize the joint formulation to the coding scenario with additional texture distortion constraint. As will be seen, we re-formulate joint texture and depth map coding as the problem of optimizing the view synthesis quality under the constraints of the total bit rate for both texture and depth as well as a maximum distortion constraint for texture video. We derive an efficient solution that can provide near-optimal view synthesis quality, while providing much better performance for the texture video. To the best of our knowledge, no prior study has explicitly considered the trade-off between the view synthesis quality and texture quality with two constraints.

The rest of this paper is organized as follows. We first review the related work in Section II, and introduce some preliminaries in Section III. Next, in Section IV, joint source and channel coding for 3D video is formulated, and a Lagrangian-based solution is developed. In Section V, we examine the coding scenario with texture distortion constraint. Experimental results are discussed in Section VI, followed by the conclusion remarks in Section VII.

[1]It should be noted that, there exists some cases where the required error correction capability may not be achieved by the available channel coding rate. Nevertheless, our proposed algorithm discussed in the following can always achieve graceful quality degradation due to the consideration of error-resilient source coding.

II. Related Work

A. 3D Image/Video Coding and Streaming

Recently, compression and streaming of 3D image/video has attracted considerable interest. In [8], in order to investigate the best multi-view representation of a scene for a given bit rate budget, the authors proposed a bit allocation algorithm to find the optimal subset of captured views for encoding and assign quantization levels for texture and depth maps of the selected coded views. By examining the scenario of multiple clients with heterogeneous access links and device capabilities, [9] proposed a user-action-driven coding framework to find the best view and rate scalable representation of texture plus depth for a 3D scene. To enable multi-view video compression and streaming, Velisavljevic *et al.* [10] carried out a convexity characterization analysis of the virtual view reconstruction error caused by compression of the captured multi-view content. For the purpose of efficiently transmitting multi-view content, [11] proposed an optimization framework to select the transmission policy for sending the packetized multi-view video data over bandwidth constrained channels. In consideration of the channel quality feedback from the client, [12] designed a system framework for wireless streaming of interactive multi-view video via path diversity and network compression, while, to reduce view switching latency, [13] developed an interactive free viewpoint video streaming scheme by using HTTP adaptive streaming. When view popularity matters, [14] designed a constrained optimization method for sharing the transmission bandwidth of the wireless channel across the visual sensors for the setup of decentralized acquisition of multi-view video. However, all the reviewed works do not consider channel coding at the sender. In [15], a multiple description coding scheme is proposed for multi-path streaming of free-viewpoint video, where the even frames of the left view and the odd frames of the right view is coded as one description on one path, and the remaining frames in the two views are transmitted as the second description over a second path. At the decoder, the lost frame in one description is reconstructed using a patch-based procedure based on the received description. However, as this scheme divides each of the captured views into two descriptions and encodes them independently, the original compression efficiency of multiview video would be greatly degraded.

B. Error-Resilient 3D Video Coding

By considering randomness of depth error caused by packet losses, a theoretical analysis of end-to-end distortion for the synthesized view in 3D video using a graphical model is given in [16]. In this work, the warping competition occurred in the 3D depth-image-based rendering scheme is considered, and the texture and depth probability distribution due to packet losses is calculated at the pixel level through a recursive optimal distribution estimation approach. In [17], a practical robust coding algorithm focused on reference frame selection is presented to protect the depth map bit stream, where, the sensitivity of synthesized view distortion to the reconstruction errors of depth is firstly modeled by a quadratic weighting function through curve fitting, and then the depth reference

block is selected to minimize the expected synthesized view distortion with the bit rate constraint. Afterwards, this method was generalized to the encoding of both texture and depth [18], and was augmented with an adaptive blending error-concealed scheme. In these approaches, inter-view error propagation due to disparity estimation is neglected. Also, these two methods encode the texture and depth map in an independent manner, i.e., the quadratic model is employed for synthesis-oriented depth coding, while the conventional MSE-based distortion metric is used for texture coding. However, since the effect of texture and depth distortions on the overall synthesis distortion is generally intertwined, error resilient coding of 3D video cannot separate these two components. For joint texture and depth map coding in packet loss scenario, a rate-distortion based mode selection scheme was developed, where the total end-to-end distortions of both the synthesized and captured views are used as distortion measure in compression [19]. An iterative optimization scheme is employed to find the optimal mode allocation in the texture and depth. Experiment demonstrate this algorithm achieves better performance than that obtained in [18].

C. Joint Source and Channel Coding of 3D Video

While these algorithms can produce significant improvement in error resilience performance, they are inadequate for the larger problem of 3D video communications. This is because, practically, Shannon's separation theorem, which allows for separate design of the source and channel coding schemes, is no longer hold due to the complexity limitations in the source coder and finite block length restrictions in the channel coder. Therefore, a key research would be the investigation of a joint design of source and channel coder. Further, since texture errors directly change the pixel intensity of the synthesized pixel, while the associated depth errors caused by packet losses induce geometry errors in the synthesized pixel, the importance of these two different bit streams may not equal [20]. As a result, it is possible to apply different amounts of protection to the texture and depth bit streams.

In [21], a joint source and channel coding scheme was studied for single video plus single depth based 3D video transmission over a wireless channel. In this scheme, full resolution and downsampled depth were investigated. The texture and depth are assumed to be independently encoded by an H.264/AVC encoder and then protected by FEC using UEP at the packet level. Using the branch and bound method, the proposed scheme can select the optimum color and depth quantization parameters as well as the channel coding rate. However, error-resilient source coding is not considered in this paper, and the view synthesis distortion due to the compound effect of texture error and depth error is not analytically characterized. A similar joint source and channel coding approach is employed in [22] to provide reliability for transmission of pure multiview video signals, where, to allow for different views having different quality constraints, the total number of bits is introduced in the objective function to be minimized with the distortion of each view being fixed to a predetermined threshold. In [23], a popularity-aware joint

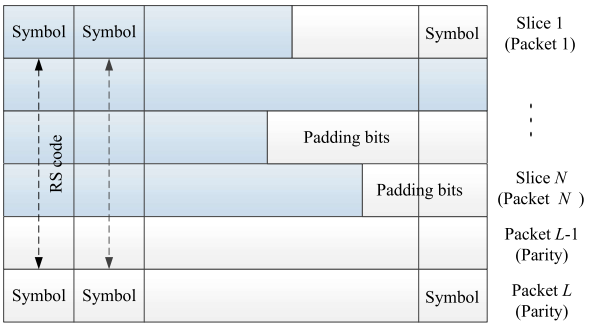


Fig. 1. Packetization scheme used for texture and depth frames in 3D video. Each slice corresponds to one packet, and RS coding is performed vertically for the packets.

source and channel coding optimization framework is proposed for view and rate scalable multi-view video multicast, where shape-adaptive wavelets is used for source coding, while Expanding-Window Random Linear Codes are employed for error protection. The purpose of this work is to maximize the aggregate video quality across the client population for the given channel characteristics and the view selection preference of the clients. However, as in [21] and [22], this algorithm still only addressed the trade-off between non-loss-aware source coding and channel coding, and does not include any error resiliency tools for prevention of potential error propagation. A preliminary study on joint error-resilient source coding and channel coding is presented in [24]. However, the computational complexity is not analyzed and trimmed, and the trade-off between the view synthesis quality and texture quality is also not handled.

In this paper, we consider the new 3D video coding scenario of employing a more advanced motion-compensated prediction structure. Further, we integrate the channel coding with error-resilient source coding, which can effectively mitigate error propagation caused by residual errors due to the nonergodic channel behavior. Error-resilient source coding usually sacrifices compression efficiency to enhance error robustness, and thus demands more source bits to obtain the same video quality in the absence of any transmission errors. The redundant bits incurred by error-resilient source coding may affect the number of redundant bits that would be allocated to channel coding. Therefore, both the complex prediction structure and error-resilient source coding will make the optimal bit allocation problem between source and channel coding more challenging as opposed to the related work discussed above.

III. Preliminaries

A. Video Packetization and Error Concealment

Fig. 1 depicts the packetization scheme utilized in this work. As in a 2D video streaming system, a texture or depth frame is first partitioned into N slices, each of which is encapsulated into one transport packet. Then, a block code is applied to the N packets to generate an L-packet block, where $L > N$. Here, we use Reed-Solomon (RS) code to perform channel coding, with its representation being as RS(L, N), where N is the length of the source symbols, and $L - N$ is the length of the parity symbols. Since the channel errors discussed here are in

the form of packet erasure, N is thus equal to the number of slices and $L-N$ is the number of parity packets. It should be noted that the proposed framework can be applied with other codes. With the protection of systematic RS codes, a packet is considered as lost after error recovery only when the packet is lost and the block containing the lost packet cannot be recovered [25]. Consequently, the probability of packet loss p after error recovery is defined as

$$p = \varepsilon\left(1 - \sum_{i=0}^{L-1-N} \binom{L-1}{i} \varepsilon^i (1-\varepsilon)^{L-1-i}\right) \tag{1}$$

where ε is the transport packet loss probability before packet error recovery. Since the packet sizes (one slice per packet) are different, the maximum packet size of a block is first determined, and then some padding is added for equalization of the size of the packets. The stuffing bits are discarded after generation of the parity packets. It should be emphasized that the texture and depth frames can generate different numbers of slices in this scheme. However, to better show the unequal number of parity packets for protection of texture and depth frames, it is assumed that both texture and depth frames produce the identical number of source video packets. Further, one horizontal row of macro-blocks (MBs) is assumed to form a slice.

For decoder error concealment, we choose the error concealment scheme proposed in [26]. When a packet is lost, the missing motion vector is concealed by using the median of the three motion vectors of its neighboring MBs in the previous packet (i.e., top-left, top, and top-right MBs). Then, the MB in the lost packet is replaced with the MB in the previous frame pointed to by the concealed motion vector. In case that the previous packet is also lost, the concealed motion vector is set to zero, i.e, the MB in the same location in the previously reconstructed frame is resorted to inpaint the current lost packet.

B. Overall Expected View Synthesis Distortion Model

To optimize the view synthesis quality, the overall expected synthetic distortion induced by texture and depth distortions should be theoretically modeled from the encoder. Let $D_{T,t}^{x_i,y_i}$ denote the end-to-end distortion of texture pixel (x_i, y_i), with $D_{D,t}^{x_i,y_i}$ denoting the associated depth distortion. Considering texture errors and depth errors from both views, we can write the combined expected distortion $D_{V,t}^{u,v}$ at the synthesized pixel (u, v) of frame t at a certain intermediate virtual view position as [27]

$$D_{V,t}^{u,v} = \sum_{i\in\{l,r\}} \left(w_i^2 D_{T,t}^{x_i,y_i} + w_i^2 \psi_i f^2 L_i^2 C^2 \cdot D_{D,t}^{x_i,y_i} + w_i^2 \psi_i E(\varsigma^2)\right) \tag{2}$$

where L_i denotes the baseline between the virtual view and the reference view i, and w_i is the resulting weighting factor from this captured view. f is the focal length, ς is the rounding error, and $C = 1/255(1/Z_{\text{near}} - 1/Z_{\text{far}})$. Z_{near} and Z_{far} are the values of the nearest and farthest depth of the scene, respectively. ψ_i is the motion sensitivity of the reference view, which can be calculated using the energy density of the texture video based upon discrete Fourier transform as shown in [27]. It should be noted that, in modeling the expected view synthesis distortion, we assume that the disparity occurs in the horizontal axis. However, our proposed joint source and channel coding scheme in the following is also applicable when disparity occurs in both horizontal and vertical axes.

As illustrated in (2), the synthesized view distortion is expressed as a linear combination of the distortions of texture and depth. Due to random channel losses, we use the expected end-to-end distortion to evaluate the video quality of texture and depth. In general, the expected distortion model is a recursive model, relating the distortion of the current frame to that of other frames [28]. It has the generic form of

$$D_t^{x_i,y_i} = (1-p)\left(D_t^{x_i,y_i}\right)^S + (1-p)\left(D_t^{x_i,y_i}\right)^{CR} + p\left(D_t^{x_i,y_i}\right)^{CL} \tag{3}$$

where $\left(D_t^{x_i,y_i}\right)^S$ is the source coding distortion caused by quantization in lossy compression, $\left(D_t^{x_i,y_i}\right)^{CR}$ is the expected error-propagated distortion from the reference frame when the pixel (x_i, y_i) is correctly received, $\left(D_t^{x_i,y_i}\right)^{CL}$ is the expected channel-induced distortion when the pixel is lost. p is the loss probability as calculated in (1). Note that the subscripts T and D are dropped in (3) due to the same form of distortion model used for texture and depth. $\left(D_t^{x_i,y_i}\right)^{CR}$ describes how much distortion in the associated pixel in the reference frame propagate into the current frame. In other words, it depends on the accumulated distortion of the predictor pixel in the reference frame. In 3D video coding, the frames used for differential encoding can be the temporal, inter-view, or synthesized view frames, which correspond to the inter (or MV sharing), inter-view, VSP coding modes, respectively. $p\left(D_t^{x_i,y_i}\right)^{CL}$ depends on the adopted error concealed strategy at the decoder [29]. For traditional coding modes, it is typically estimated as the error-propagated distortion of the concealed pixel plus the distortion between the concealed pixel and the current pixel. However, for the MV sharing mode, as the motion vector of the depth map is inherited from the associated texture video, the resulting residue of depth map and the inferred motion vector from texture are very likely to be separately transmitted in two different packets [30]. Therefore, the concealment distortion for this mode should be analyzed depending on whether the residue and MV have been lost or correctly received. As in [31], $\left(D_t^{x_i,y_i}\right)^{CL}$ can be further expanded as $p\left(D_t^{x_i,y_i}\right)^{CL\_mv} + p\left(D_t^{x_i,y_i}\right)^{CL\_res} + p^2\left(D_t^{x_i,y_i}\right)^{CL\_both}$, where $\left(D_t^{x_i,y_i}\right)^{CL\_mv}$ and $\left(D_t^{x_i,y_i}\right)^{CL\_res}$ are the expected distortions when the MV packet and the residual packet are lost, respectively. $\left(D_t^{x_i,y_i}\right)^{CL\_both}$ denotes the expected concealment distortion when both are lost.

IV. Problem Formulation and Solution of Joint Source Channel Coding for 3D Video Coding

We begin with the mathematical formulation of joint texture and depth mode selection based error-control 3D video coding. Then, we design an integrated formulation that optimizes error-resilient source and channel coding for texture and depth

simultaneously. Finally, we give the details of the complexity of the proposed approach, and develop a trellis state pruning algorithm to reduce the time complexity of searching the optimal solution.

A. Problem Formulation and Trellis-Based Solution

We formulate the joint coding mode selection problem in an integrated manner at the bock level. Suppose that $M^T_{k,i} \in I_T$ is the coding mode selected for the ith MB in the packet k, where I_T is the texture coding mode set, *i.e.*, {Intra, Inter, Inter − view, VSP}. Let $\mathbf{M^T_k} = (M^T_{k,1}, \cdots, M^T_{k,M})$ denote the texture coding mode vector for the kth packet, where M is the number of available MBs in one packet. Similarly, denote $M^D_{k,i}$ as the depth mode selected for the ith MB in the packet k, which is chosen from the set I_D = {Intra, Inter, Inter − view, MV sharing}, and let $\mathbf{M^D_k} = (M^D_{k,1}, \cdots, M^D_{k,M})$ denote the depth coding mode vector to be determined for packet k. It should be noted that the size of the I_T or I_D can be variable depending on the actual number of coding modes used in the codec. Then, for a given frame in a reference view,[2] the texture and depth modes assigned to all the packets are given by two $N-$length tuples, $\mathbf{M^T} = (\mathbf{M^T_1}, \cdots, \mathbf{M^T_N})$ and $\mathbf{M^D} = (\mathbf{M^D_1}, \cdots, \mathbf{M^D_N})$, where N is the number of packets in a frame. Correspondingly, the texture and depth coding mode pair for the opposing view is denoted by $(\mathbf{M^T_o}, \mathbf{M^D_o})$. Given the overall coding rate R_c and $(\mathbf{M^T_o}, \mathbf{M^D_o})$, the problem is to optimally select the texture and depth modes for each MB in current reference view such that the expected view synthesis distortion is minimized, which is,

$$\min_{\mathbf{M^T},\mathbf{M^D}} \; E\left[D_V(\mathbf{M^T}, \mathbf{M^D}, \mathbf{M^T_o}, \mathbf{M^D_o})\right] \quad \text{s.t. } R(\mathbf{M^T}, \mathbf{M^D}, \mathbf{M^T_o}, \mathbf{M^D_o}) \le R_c \tag{4}$$

where the terms $E\left[D_V(\mathbf{M^T}, \mathbf{M^D}, \mathbf{M^T_o}, \mathbf{M^D_o})\right]$ and $R(\mathbf{M^T}, \mathbf{M^D}, \mathbf{M^T_o}, \mathbf{M^D_o})$ represent the expected overall view synthesis distortion and total bit rate of texture and depth, respectively, resulting from a vector choice of the combined texture and depth modes for the current frame. $E\left[D_V(\mathbf{M^T}, \mathbf{M^D}, \mathbf{M^T_o}, \mathbf{M^D_o})\right]$ can be calculated using (2) by summing up the expected distortions of all pixels within the frame, while $R(\mathbf{M^T}, \mathbf{M^D}, \mathbf{M^T_o}, \mathbf{M^D_o})$ can be directly obtained after actual entropy coding. When operating on the non-inter-view predicted left view, both the Inter-view and VSP modes are set to *NULL* in the texture mode set, and in the depth coding mode set, the Inter-view mode is set to *NULL*.

Generally, the constrained problem in (4) can be equivalently transformed into an unconstrained Lagrangian problem using the Lagrange multiplier λ_V, which is shown as follows

$$\min_{\mathbf{M^T},\mathbf{M^D}} \sum_{k=1}^{N} J_k(\mathbf{M^T}, \mathbf{M^D}, \mathbf{M^T_o}, \mathbf{M^D_o}) = \min \sum_{k=1}^{N} \left\{ E\left[D_k(\mathbf{M^T}, \mathbf{M^D})\right] + \lambda_V R_k(\mathbf{M^T}, \mathbf{M^D}) \right\} \tag{5}$$

[2]Here, for convenience, the frame in the reference view refers to the integrated frame, which is composed of the texture frame and the corresponding depth frame in the reference view.

where $E[D_k]$ and R_k represent the expected view synthesis distortion and bit rate, for packet k, respectively. With an appropriate $\lambda_V \ge 0$, (4) can be solved within a convex hull approximation by solving (5) with an operational rate-distortion curve. Due to the monotonic relationship between λ_V and bit rate, λ_V can be determined by using the bisection iterative search [32]. It should be noted that, since we aim to find the optimal texture and depth mode pair for current view for a given $(\mathbf{M^T_o}, \mathbf{M^D_o})$, we omit the $(\mathbf{M^T_o}, \mathbf{M^D_o})$ in the right hand side of the above equation and in the following derivation for conciseness. Once we obtain the optimal $(\mathbf{M^T}, \mathbf{M^D})$ for the current view, we then search the optimal $(\mathbf{M^T_o}, \mathbf{M^D_o})$ using the similar algorithm for the opposing view. We alternate these two steps until the variables converge, *i.e.*, minimum combined view synthesis distortion. Thus, in the following, we mainly describe how to optimally search $(\mathbf{M^T}, \mathbf{M^D})$.

With the formulation of joint texture and depth mode switching in source coding in (5), we now take into account the effect of residual errors after channel coding. As stated above, the importances of the texture and depth bit stream relative to the synthetic quality are not equal. Therefore, they should be protected differently by assigning an unequal amount of FEC to each bit stream. Consider a set of FEC parameters given by $C = \{(L_1, N), \cdots (L_q, N)\}$, where q is the number of available code options. Let C_T and C_D be the channel coding parameters assigned for the texture and depth frames, respectively, both of which take on the values from the common set C. It should be emphasized that we apply UEP to the frames between the texture and depth components in 3D video, i.e., no UEP is applied within the texture or depth. Define a particular collection of these two channel coding parameters by $C_{TD} = \{C_T, C_D\}$, with $C_{TD} \in \mathbf{C}$, where $\mathbf{C} = C \times C$ is the set of all possible combinations of FEC parameters of the texture and depth. As such, the optimization problem of searching for the best coding modes and optimal UEP rates between texture and depth can be formulated as

$$\begin{aligned} \min_{\mathbf{C}} \left\{ \min_{\mathbf{M^T},\mathbf{M^D}} \sum_{k=1}^{N} J_k(\mathbf{M^T}, \mathbf{M^D}, C_{TD}) \right\} \\ = \min \left\{ \sum_{k=1}^{N} E\left[D_k(\mathbf{M^T}, \mathbf{M^D}, C_{TD})\right] \right. \\ \left. + \lambda_V \left\{ \sum_{k=1}^{N} R_k(\mathbf{M^T}, \mathbf{M^D}) + R(C_T) + R(C_D) \right\} \right\} \end{aligned} \tag{6}$$

where $R(C_T)$ and $R(C_D)$ refer to the channel coding bit rates for the whole frames of texture and depth map, respectively. As can be observed from (6), the coding mode vectors and the channel coding parameters of texture and depth are jointly determined in a single step, which means that we consider joint source and channel coding in an integrated fashion. To the best of our knowledge, no prior study on joint source and channel coding formulation has explicitly considered the inter-component dependency for error-resilient texture and depth map coding, while simultaneously considering the trade-off between the resulting intrinsic error resilience and the error

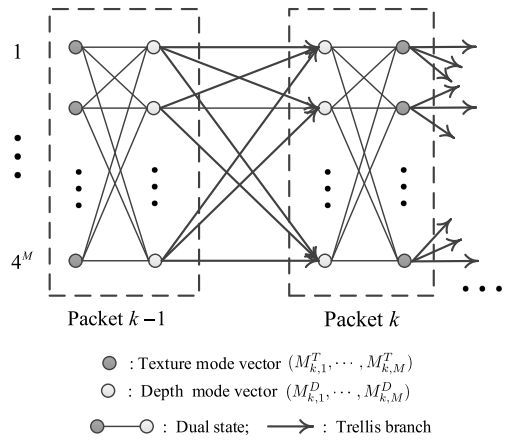


Fig. 2. Illustration of designed trellis representation for joint selection of texture and depth coding modes with adjacent packet dependency in 3D video, where each packet represents a stage in the trellis denoted by a dashed box, and each line connecting the texture mode vector and depth mode vector within the packet represents a dual trellis state.

correction capability of channel coding during *real encoding* of 3D video.

For a given λ_V, there are two minimizations to be solved in (6). The inner minimization is the joint texture and depth mode selection for each packet representing loss-resilient source coding. The outer minimization corresponds to joint source channel coding, *i.e.*, unequal error protection, where different amounts of FEC coding rates are to be allocated for texture and depth at the frame level. Due to finite FEC options, the outer minimization can be easily solved using the exhaustive search. Next, we examine how to solve the inner Lagrangian optimization, which remains unwieldy as the view synthesis distortion and rate associated with a particular packet is coupled to the chosen modes for every other packet in the texture and depth frame. However, since the error concealment scheme introduces the dependencies only between two neighboring packets, the inner optimization can be done by using dynamic programming (DP) as follows.

When the above mentioned error concealment method using the motion vectors of neighboring packets is used, $E[D_k]$ depends on the prediction modes of texture and depth selected for the packet $k-1$. Due to the introduced adjacent packet dependency, for a given pair of texture and depth channel coding parameters, the Lagrangian cost of joint source texture and depth coding can be re-written as

$$J_k(\mathbf{M^T},\mathbf{M^D},C_{TD}) = J_k(\mathbf{M^T_{k-1}},\mathbf{M^D_{k-1}},\mathbf{M^T_k},\mathbf{M^D_k},C_{TD}), \quad \text{for} \quad k=2,\cdots,N. \tag{7}$$

In order to solve the constrained optimization, we employ a DP solution based upon the Viterbi Algorithm (VA) [33]. Prior to establishing a forward DP, an associated trellis has to be constructed for a given frame in the reference view. The corresponding trellis is shown in Fig. 2, where 2 stages corresponding to packets $k-1$ and k are shown. The nodes in the trellis are given by the combined texture and depth coding mode vectors for a given packet in the reference view. At each stage, since the cardinalities of the coding mode sets of texture and depth are both four, there are $4^M \times 4^M$ trellis states, each representing a particular combination of texture and depth modes (i.e., $\{\mathbf{M^T_k},\mathbf{M^D_k}\}$) for the kth packet. Since the control parameter set which influences the overall rate and distortion for a node is a dual combination of prediction mode vectors of two different coding signals, the admissible states of the nodes are also dubbed the dual states. The transitional costs from node $\{\mathbf{M^T_{k-1}},\mathbf{M^D_{k-1}}\}$ to $\{\mathbf{M^T_k},\mathbf{M^D_k}\}$ are given by Lagrangian cost terms defined in (7). Once the trellis is formed, the VA is applied to find the shortest path, which is defined as the path that has the minimal overall rate-distortion cost.

It should be worth noting that, if we consider inter-packet dependencies over the entire group of packets in a pair of texture an depth frame, a diverging trellis may be generated. In this case, the size of the trellis-based tree grows exponentially with the tree depth, and only if the number of combined coding modes is relatively small can the optimal solution be feasibly found. In this work, to efficiently find the optimal solution to (6) while fully considering the characteristics of 3D video coding, we firstly integrate texture and depth modes as a whole, and then consider the block-to-block dependency within an integrated frame. This block-level texture mode and depth mode integration can avoid the occurrence of the dependence between the coding mode vector of the whole texture frame and the coding mode vector of the whole depth frame. Further, to facilitate a tractable solution, we consider one dimensional packet dependency, and finally propose a non-diverging dual trellis as shown in Fig. 2.

B. Complexity Consideration

1) Dual-Trellis State Reduction: Since we deal with a finite number of admissible texture and depth modes, and channel coding parameters, the above optimization problem can be solved by an exhaustive search. The time complexity for such an exhaustive search is $O\left(q^2\cdot\left[|I_T|^M\cdot|I_D|^M\right]^N\right)$, where $|I_T|$ and $|I_D|$ are the cardinality of the I_T and I_D, respectively. In this paper, both $|I_T|$ and $|I_D|$ equal 4. While using the proposed DP-based optimization algorithm, the time complexity becomes $O\left(q^2\cdot\left[(|I_T|^M\cdot|I_D|^M)\cdot(|I_T|^M\cdot|I_D|^M)N\right]\right)$, which is significantly smaller than the complexity for the exhaustive approach. Recall that q is the number of FEC code options. However, even though the complexity is reduced, due to the large number of possible states involved in each stage, the proposed algorithm still poses prohibitively high computational and memory requirements. To further reduce the computational burden, we propose a trellis state reduction algorithm based on the block-to-block dependency within the packet. Specifically, as shown in the complexity of the proposed DP algorithm, there are $|I_T|^M\cdot|I_D|^M$ states at each stage. This means the rate-distortion cost of each MB depends not only on its own modes but also on the mode decisions of all the other MBs in the same packet.

However, in practice, due to the differential coding techniques (e.g., motion vector prediction), the rate term for a given MB is dependent only on the current mode and the mode of the adjacent MB [5]. Further, since the disparity errors caused by depth errors are generally small, the warping errors

also lead to that the view synthesis distortion of a particular rendered block is only related to the modes from the current MB and neighboring MB [34]. Consequently, the influence of the mode decisions from other texture and depth MBs on the current texture and depth MBs is typically limited to that from the immediately preceding texture and depth MBs, respectively. Under this assumption, we can employ the DP to find the optimal texture and depth mode vectors within a packet as well. In this case, the number of all possible combinations of texture and depth coding modes for a block is $(|I_T| \cdot |I_D|)$, and the number of the corresponding possible combinations of all pairs of texture and depth modes in a packet is $((|I_T| \cdot |I_D|)(|I_T| \cdot |I_D|) \cdot M)$. Thus, the dual-trellis states of each stage can be reduced to $(|I_T|^2 \cdot |I_D|^2 \cdot M)$ with adjacent block dependency consideration. Finally, the complexity of the proposed algorithm after state pruning will be $O\left(q^2 \cdot \left[(|I_T|^2 \cdot |I_D|^2 \cdot M) \cdot (|I_T|^2 \cdot |I_D|^2 \cdot M)N\right]\right)$. It should be noted that since some trellis states of each stage have been pruned for the purpose of reducing the complexity to search the solution, the rate-distortion performance of the proposed joint source and channel coding scheme may be degraded. However, through the experiments over various target total bit rates, we found that the performance loss induced by the proposed state reduction within a packet is negligible. It should also be noted that when we use the term "time complexity", we refer to the number of comparison necessary to find the optimal solution. This does not include the time complexity consumed to evaluate the operational rate distortion functions.

2) Complexity Analysis for Distortion Estimation: We also provide the complexity analysis for the distortion computation for each pixel and packet. As illustrated in (2), the view synthesis distortion encompasses the texture, depth, and rounding distortions. Since the rounding distortion can be predetermined using the uniform distribution [27], we focus on the complexity analysis of estimating the expected texture and depth errors. Due to different prediction tools used, the time consumptions for error estimation in different coding modes may be slightly different. For ease of analysis, we use the general form (3) to provide an approximate indication of the complexity. As can be observed, the expected distortion involves determining the source distortion, the error propagation distortion, and the concealment distortion. Since source distortion can be obtained after the reconstruction at the encoder, no additional complexity is imposed. For error propagation distortion in (3), as shown in [28], it can be further decomposed into three terms, *i.e.*, the error propagation distortion of the reference pixel, and the error-propagated distortion of the concealed pixel, and the error concealment distortion. Based on the definitions of these distortion terms, we need four additions and three multiplications to calculate the second term in (3). Similarly, the last term in (3) needs two additions and two multiplications. Therefore, the recursive distortion model requires eight additions and six multiplications for the calculation of texture or depth errors, so, in (2), a total of eighteen additions and twenty multiplications is required for the synthesis distortion estimation of each warped pixel from one reference view. Consequently, the estimation of expected synthesis distortion for a packet introduces a total $18\times16\times16\times M$ additions and $20\times16\times16\times M$ multiplications.

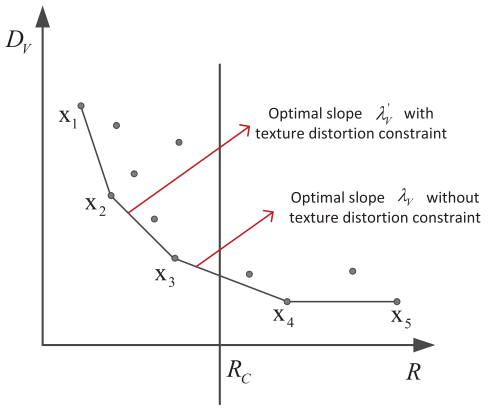


Fig. 3. Illustration of operational rate-distortion function for the synthesized view. The convex hull of the points provides the optimal rate-distortion points. The negative slope of the straight line connecting the two operating points in the convex hull is defined as the singular slope, which varies monotonically with respect to the total bit rate.

V. Joint Texture and Depth Coding With Texture Distortion Constraint

In the previous section, we deal with UEP for texture and depth coding, which mainly aims at optimizing the view synthesis quality only. Needless to say, this proposed algorithm can effectively minimize the expected distortion of the synthesized views, but it may result in unacceptable texture video quality of the coded views. However, as the originally coded views are usually combined with the synthesized views to form the stereo pair for human viewing, the coded view cannot be sacrificed too much. Therefore, in this section, to guarantee the quality of the texture video, we re-formulate the above joint optimization problem by throwing an additional constraint on the texture video. Specifically, the problem we consider here is to provide the best view synthesis quality for a given total bit rate constraint and a maximum tolerable coded texture distortion constraint D_T', i.e.,

$$\min_{\mathbf{C}} \left\{ \min_{\mathbf{M^T},\mathbf{M^D}} E\left[D_V(\mathbf{M^T},\mathbf{M^D}, C_{TD})\right] \right\}$$
$$\text{s.t. } R\left(\mathbf{M^T}, C_T\right) + R\left(\mathbf{M^D}, C_D\right) \le R_C \tag{8}$$

and

$$D_T(\mathbf{M^T}, C_T) \le D_T' \tag{9}$$

To solve the problem of the combined (8) and (9), we firstly derive the solution to only (8) using Lagrangian multiplier method. In solving (8), a rate-distortion curve for the synthesized view is generated by acquiring the total bit rate and the synthesized distortion pairs. We then select a set of feasible points that define the vertices of the convex hull. Fig. 3 gives an example of the operational rate-distortion function of the synthesized view, in which the convex hull of the points provides the optimal rate-distortion points, and the negative value of the slope of the straight line connecting the two

operating points in the convex hull is marked as the singular slope λ_V. Once the rate-distortion curve of the synthesized view is determined, the problem of (8) can be found by searching through different λ_V and finding the operating point that leads to the bit rate equal to R_C. Due to the monotonic relationship between the bit rate and λ_V, the optimal λ_V for (8) can be found by using the bisection method [25]. Afterwards, for a given optimal λ_V, we may not find the optimal operating point whose slope value (i.e., the slope of the line tangent to the operating point) is exactly equal to the given slope value due to the finitely many points on the convex hull. However, we can find two neighbouring operating points such that the given slope value is between the slope values at these two points. Thus, the optimal operating point is the one whose slope is the minimum slope that satisfies the condition of being larger or equal to the optimal singular slope. As can be seen in Fig. 3, for a given bit rate budget R_C, it is found that the optimal singular slope and rate-distortion point in the solution to only (8) are λ_V and x_3, respectively.

After having obtained the solution to (8), we then check whether the obtained solution to (8) meets the additional constraint of (9). Obviously, if the optimal solution of (8) coincidentally meets the texture distortion constraint of (9), then the problem with two constraints is done. However, if the optimal solution of (8) does not, we have the following lemma for the solution to the combined (8) and (9).

Lemma 1: If the optimal solution of (8) does not meet the texture distortion constraint of (9), i.e., the optimal texture bit rate obtained by (8) makes the texture distortion larger than the distortion constraint D'_T, then the optimal solution to the combined (8) and (9) can be obtained by just setting the texture distortion equal to the texture distortion constraint, i.e., $D^*_T(\mathbf{M^T}, C_T) = D'_T$.

Proof: As, in this case, $D_T(\mathbf{M^T}, C_T) > D'_T$, the solution generated by λ_V in (8) is no longer feasible. Therefore, the problem now becomes solving (8) in the case of the texture distortion over D'_T truncated. To decrease the $D_T(\mathbf{M^T})$, we should increase the bit allocation $R(\mathbf{M^T}, C_T)$ for the texture video. This means that the opposed $R(\mathbf{M^D}, C_D)$ for the depth should be reduced for the fixed bit budget R_C. When the depth bit rate is decreased, we must have a new singular Lagrange multiplier $\lambda'_V > \lambda_V$ for depth map due to the bit rate being a nonincreasing function of λ_V. Assume now, the texture distortion is decreased to the highest available distortion D'_T, it corresponds to λ'_V and x_2 in the rate-distortion function of the synthesized views in Fig. 3, which are the possible solution to the combined (8) and (9) with two constraints.[3] If the texture distortion is further decreased under D'_T, we can get a larger λ'_V, but it will result in D_V that is greater than that of x_2. On the other hand, in the original rate-distortion curve of the texture, the increase of λ_V will also increase the texture distortion. However, since all the points with $D_T(\mathbf{M^T}, C_T) > D'_T$ have been removed, we have $D_T(\mathbf{M^T}, C_T) = D'_T$. Therefore, $D^*_T(\mathbf{M^T}, C_T) = D'_T$ is indeed the optimal solution to (8) and (9). Proof is completed. □

It should be noted here that, in the operational rate-distortion curve of the synthesized view in Fig. 3, the overall view synthesis distortion of point x_2 is larger than that of x_3. However, this does not indicate that the texture or depth distortion of point x_2 is necessarily larger than that of x_3. This is because, with the texture distortion constraint consideration, the optimal solution to the minimization of view synthesis cost does not necessarily correspond to the optimal operating points on the respective rate-distortion curves of the texture and depth.

In summary, when the optimal solution to only (8) does not satisfy the constraint of (9), the optimal solution to the combined (8) and (9), is to select appropriate $R(\mathbf{M^T}, C_T)$ for texture coding to make $D^*_T(\mathbf{M^T}, C_T) = D'_T$, and then allocate the remaining bit rate $R_C - R(\mathbf{M^T}, C_T)$ for depth map coding.

Finally, given D'_T and λ'_V (or λ_V), one can solve for all possible $\mathbf{M^T}$ and C_T, sum the bit rate of them up to get $R(\mathbf{M^T}, C_T)$ and $D_T(\mathbf{M^T}, C_T)$, and then find the optimal coding parameters which minimize the Lagrangian cost $J(\mathbf{M^T}, C_T)$, that is, $D_T(\mathbf{M^T}, C_T) + \lambda'_V R(\mathbf{M^T}, C_T)$. If $J^*(\mathbf{M^T}, C_T)$ is the minimal one, the desired solution in texture coding has been achieved. Similarly, the desired solution of joint source and channel coding in depth map coding can be obtained with the remaining bit rate.

The complexity of solving (8) and (9) mainly comprises two parts. The first part is from the iterative search of the λ'_V in the framework of joint source and channel coding with texture distortion constraint, and the other one is from finding the corresponding optimal source and channel coding parameters for a given multiplier. As can be seen from the derivations, the search of λ'_V can be achieved by setting $D^*_T(\mathbf{M^T}, C_T) = D'_T$, which means we need to trace out the operating points in the operational rate-distortion curves of the texture video and depth in addition to those of the synthesized view. Due to the monotonic relationship between the bit rate and Lagrange multiplier, the optimal λ'_V in the respective operational rate-distortion curves of the texture video and depth can also be found by using the bisection method. In the second part, the costs of the operating points in the respective rate-distortion curves of the texture and depth need to be calculated for a given Lagrange multiplier. Similar to the complexity analysis for finding the source and channel coding parameters for texture and depth for overall view synthesis cost in Section IV-B, the complexity for finding the source and channel coding parameters for texture only using the proposed state-pruning algorithm can be derived as $O\left(q^2 \cdot \left[(|I_T|^2 \cdot M) \cdot (|I_T|^2 \cdot M)N\right]\right)$, while the complexity for finding coding parameters for depth with the remaining bit rate is $O\left(q^2 \cdot \left[(|I_D|^2 \cdot M) \cdot (|I_D|^2 \cdot M)N\right]\right)$, which represents a modest complexity increase compared to the optimization scheme introduced in the previous section.

VI. Experimental Results

In this section, we first elaborate on the experimental configuration. Then, we evaluate the performance of the

[3]Due to the monotonicity of the view synthesis distortion with respect to the texture error and depth error and the approximate independence between the texture errors and depth errors, at optimality, the texture, the depth and the synthesized view should be operating at the same Lagrange multiplier on their operational rate-distortion curves [18]. This is also in accordance with the selection of Lagrange multiplier in 3D video coding standards [35].

TABLE I
3D TEST SEQUENCE ATTRIBUTES AND SOURCE CODING PARAMETERS

Test sequence	Captured views	Virtual view	Temporal resolution	Spatial resolution	Source coding parameters
Lovebird1	6, 8	7	16.7	1024 × 768	Depth map resolution: Full VSP and MV Sharing tools: Enabled In-loop filter: Enabled Quantization parameters for texture and depth: 22, 27, 32, 37, 42
Newspaper	4, 6	5	30		
BookArrival	8, 10	9	30		
Ballet	1-8	3 views*	15		
Breakdancer	1-8	3 views*	15		
Undo_Dancer	2, 5	3	25	1920 × 1088	
GT_Fly	5, 9	6	25		

* 3 views means that there are 3 virtual views between each pair of captured views, which are assumed to be evenly distributed and indexed as non-integers.

proposed 3D video coding framework using joint source and channel coding. We also compare the proposed framework to the state-of-the-art unequal error protection schemes. Finally, we evaluate the performance of the proposed framework in the presence of texture distortion constraint.

A. Simulation Configuration

For source coding, we choose the 3D-AVC reference software 3D-ATM v6.0 [36] to compress multi-view video sequences and depth maps, and the View Synthesis Reference Software (VSRS) 3.5 [37] for rendering the virtual views at the decoder. Based on texture motion intensity and depth fidelity, seven standard sequences are chosen. For each test sequence, the size of group of picture of each view is set to 30, where the first frame in the left view is compressed as an I-frame, and the remaining frames are encoded as P and B frames. In 3D video compression, the order $T_0D_0T_1D_1$ that indicates texture coding prior to depth coding is employed, where T_i and D_i are the texture and depth components of the i^{th} view, respectively, corresponding to the captured left or right views. In motion estimation and mode decision, we fixed the block size to 16×16, although the quadtree structure is allowed in emerging 3D video compression. The reason for that is, if we enable quadtree structure partition during coding, another dependency between the quadtree structure and texture and depth modes will be generated in addition to block and packet dependencies, making the optimization problem intractable. We leave the joint optimization of quadtree structure, texture modes, and depth modes as a future work. The virtual views are generated with half pixel precision rendering and symmetric rounding. Backward VSP and MV sharing are enabled for texture and depth coding, respectively. The descriptions of the used 3D test sequences and their coding parameters are listed in Table I. It should also be pointed out that the other experimental setup follows the Common Test Condition of the Joint Collaborative Team for 3DV [38].

For channel coding, the admissible set of channel coding rates is $C = \{(68, 48), (64, 48), (60, 48), (56, 48), (52, 48)\}$ for the sequences with the resolution of 1024 × 768, and $C = \{(88, 68), (84, 68), (80, 68), (76, 68), (72, 68)\}$ for the remaining sequences. It should be noted that the set of channel coding rates can provide a wide range of channel coding bits used for error protection. Unless otherwise stated, we employ the random packet loss pattern for simulating video packet losses, and the losses of two different packets are independent of each other [39]. Various transport packet loss rates of 3%, 5%, 10% and 20% are tested on both the compressed texture video and depth stream, and, to simulate the error-prone channel, for each packet loss rate, 150 packet loss patterns are produced by randomness. For 3D video objective quality assessment, we collect the total bit rate of the texture and depth along with the average peak signal-to-noise ratio (PSNR) of the synthesized views, in which the PSNR is measured by comparing the virtual view image synthesized by the original texture and depth images and the counterpart synthesized by the decoded texture and depth images.

B. Performance of the Proposed Robust 3D Video Coding Framework

In this subsection, the proposed general 3D video coding framework optimizing the view synthesis quality for only one constraint, *i.e.*, a given total bit rate, is tested. Two related frameworks are compared. The first one is a pure error-resilient 3D video source coding framework as in (5), which does not consider any channel coding. This kind of framework resembles the existing work recently proposed in [18] and [19], and is referred to as ERSC for simplicity. The other one is an error-resilient source coding framework combined with equal error protection (ERSC_EEP) for 3D video. More specifically, in this framework, we still utilize (6) to guide joint selection of source and channel coding parameters for 3D video. However, when doing the outer optimization in (6), we assume the same channel coding rate is used for texture and depth, i.e., no channel rate allocation is further performed between texture and depth. To distinguish the proposed framework from these two competing frameworks, the proposed framework in Section IV is denoted by "ERSC_UEP".

In Fig. 4, we illustrate the performances of these three algorithms at a designated total bit rate of 1.6 Mbps. It can be seen that ERSC_UEP outperforms ERSC_EEP and ERSC across all the considered packet loss rates, with the maximum PSNR gains of 0.9 dB and 2.1 dB, respectively. The gains in ERSC_UEP compared to ERSC primarily stem from joint consideration of error resilient source coding and channel coding. In pure ERSC as done in [18] and [19], since the texture and depth modes are already rate-distortion optimized for error-resilient 3D video transmission, ERSC can effectively stop potential error propagation caused by various predictive coding techniques. However, it does not improve the quality of the images for which packet losses occur. The gains of

TABLE II

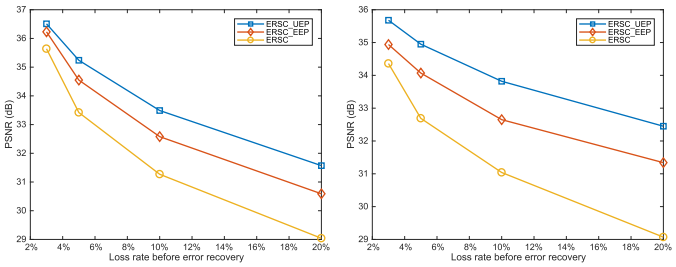
SOURCE AND CHANNEL CODING BIT RATE ALLOCATION (kbps) FOR TEXTURE AND DEPTH IN "BOOKARRIVAL" SEQUENCE. THE SYMBOL "s" IN THE PARENTHESIS REPRESENTS THE SOURCE CODING BIT RATE, WHILE "c" INDICATES THE CHANNEL CODING BIT RATE

Loss rate (ε)	ERSC_UEP				ERSC_EEP			
	Texture (s)	Depth (s)	Texture (c)	Depth (c)	Texture (s)	Depth (s)	Texture (c)	Depth (c)
3%	1137	321	112	34	1137	321	73	73
5%	1097	304	137	66	1097	304	102	102
10%	1008	287	212	97	1008	287	155	155
20%	989	269	242	104	989	269	173	173

TABLE III

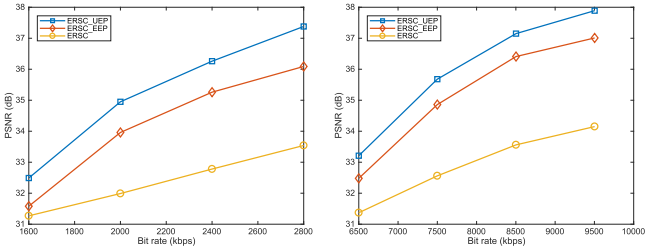
SOURCE AND CHANNEL BIT RATE ALLOCATION (kbps) FOR TEXTURE AND DEPTH IN "BOOKARRIVAL" SEQUENCE IN THE TRANSPORT PACKET LOSS RATE OF 10%. THE SYMBOL "s" IN THE PARENTHESIS REPRESENTS THE SOURCE CODING BIT RATE, WHILE "c" INDICATES THE CHANNEL CODING BIT RATE

Total bit rate	ERSC_UEP				ERSC_EEP			
	Texture (s)	Depth (s)	Texture (c)	Depth (c)	Texture (s)	Depth (s)	Texture (c)	Depth (c)
1600	1008	287	212	97	1008	287	154	154
2000	1133	365	341	161	1133	365	251	251
2400	1288	476	405	231	1288	476	318	318
2800	1405	567	526	302	1405	567	414	414



(a) (b)

Fig. 4. PSNR of the synthesized views versus transport packet loss probability ε at the total bit rate of 1.6 Mbps. (a) BookArrival. (b) Newspaper.



(a) (b)

Fig. 5. PSNR of the synthesized views versus designated total bit rate at the packet loss rate of 10%. (a) BookArrival. (b) GT_Fly.

ERSC_UEP over ERSC_EEP can be attributed to the consideration of the unequal importance of texture and depth to the overall view synthesis quality. In other words, the proposed framework has the flexibility to vary the channel coding rates adapting to the varying video contents of texture and depth. The resultant bit rate allocation between source and channel coding in texture and depth for sequence "BookArrival" at various packet loss rates is illustrated in Table II. As can be seen from the results of ERSC_UEP, the texture and depth indeed receive different amounts of protection from a channel coder. This confirms what we claimed earlier in this paper, i.e., the texture and depth bit stream are not of the same importance. Further, we also found that the texture usually requires more channel coding bits than the depth at the same transport packet loss rate, which means that the view synthesis distortion is more sensitive to texture errors in transmission.

In the next, we explore the performances of these three frameworks with different total bit rate budgets being used. As can be observed from Fig. 5, with the increase of the bit budget, the performance gap between ERSC_UEP and ERSC also increases. The reason for this is as follows. In the 3D video system with low bit budget, since the majority of the bits are allocated to the source coding, the correction ability of the channel coding is somewhat restricted. However, when the designated total bit rate gets larger, ERSC_UEP becomes more flexible in allocating the bits to channel coding, thus improving the overall error resilience performance. In addition, we can also see that ERSC_UEP consistently outperforms ERSC_EEP at various total bit rates. The resulting bit rate allocation between source and channel coding in 3D video impacted by the total bit budget is given in Table III. Finally, we also test the proposed joint source and channel coding framework on top of HEVC-based 3D video coding reference software 3D-HTM [40], which yields very similar results as those shown in Figs. 4−5 and Tables II−III.

C. Performance With Error Concealment Mismatch

As mentioned earlier, solving the problem formulation by dynamic programming is based on the assumption of adjacent packet dependency, which relies on the prerequisite that only

TABLE IV
PSNR PERFORMANCE COMPARISON WITH ERROR CONCEALMENT MISMATCH AT THE TARGET BIT RATE OF 2 Mbps

Sequence	PSNR performance (dB)	
	ERSC_UEP (matched)	ERSC_UEP (mismatched)
Lovebird1	36.3	35.7
BookArrival	35.6	34.9
Newspaper	34.5	35.1
GT_Fly	33.7	33.2

the motion vectors of the neighbouring packet are employed to conceal the corrupted packet at the decoder. Therefore, it is very interesting to investigate how the proposed joint source and channel coding scheme performs when using other error concealment schemes at the decoder. To test the performance of the proposed algorithm with other error concealment methods, at the sender, we use the assumed error concealment method (i.e., the simple error concealment method in [26]) for distortion estimation and dynamic programming at the encoder. However, at the decoder, we actually utilize two other sophisticated error concealment algorithms for the left and right view, respectively. For the left view, the motion-copy error concealment scheme for video transmission introduced in [41] is adopted. In this scheme, the motion vectors from co-located MBs in the previous frame are employed to conceal the MBs of the packet of the current frame via motion compensation. While, for the right view, the adaptive temporal and view synthesis error concealment scheme designed for 3D video proposed in [42] is adopted, where the view synthesis error concealment approach that directly uses the synthesized pixels to conceal the corrupted MBs is adaptively utilized in conjunction with the temporal error concealment approach.

The results on the performance of the proposed scheme is listed in the Table IV. For benchmark comparison, we also include the matched results where both the encoder and decoder employ the error concealment introduced in [26]. Based on the results in the table, we found, the mismatched error concealment causes performance degradation of less than 0.6 dB for the proposed algorithm on average. For the particular sequence of "Newspaper", it is surprising that the proposed algorithm with mismatched error concealment even outperforms that with matched error concealment. The substantial reason behind can be explained as follows. Compared to the simple concealment algorithm in [26], the more complicated algorithms in [41] and [42] mainly exploit the temporal correlation and inter-view correlation to conceal the lost packet, which thus introduces temporal and inter-view dependencies between packets. Generally speaking, these kinds of dependencies between packets in different frames are very difficult to be characterized in the optimization as the concealed motion vectors can have any direction and further a diverging trellis may be generated if considered. Our proposed solution only considers the packet dependencies within a frame, which thus brings performance degradation in the case of other complicated concealment methods practically used. However, by utilizing more available information in 3D video to conceal the lost packets, the mismatched error concealment algorithms themselves can actually improve the reconstructed 3D video quality compared to the matched error concealment approach. As a result, the overall performance of the proposed algorithm may be enhanced to some extent by using the mismatched sophisticated error concealment algorithms.

D. Comparison With the Existing UEP Method

In this section, we compare the proposed "ERSC_UEP" algorithm to the existing UEP method developed for 3D video transmission. The most closely related work that is proposed in [21] is chosen as the comparative approach here. In [21], a joint source-channel coding and UEP scheme is designed for 3D stereo video transmission in video plus depth format, which is referred to as "SC_UEP" for brevity. In this algorithm, the objective function is defined as the maximization of the average of the qualities of the left and the right views subject to the total bit rate. During the measurement of 3D video quality, the expected average score of the texture packets is taken as the quality of the left view, while the quality of the right view is modeled as the average score of texture and depth packets of the left view. The branch and bound method is then employed to find the optimum source coding parameters and UEP code rates for texture and depth at the packet level. Since "SC_UEP" is focused on joint source and channel optimization for stereo video in format of single video plus single depth, for fair comparison, we also implement "ERSC_UEP" in the special scenario where the right view is assumed not to exist and synthesized from the left view. Similar to "SC_UEP", the qualities of the left view and the synthesized right view are employed as the optimization criterion. In addition, in this test, both algorithms use RS code for channel coding, and only the full resolution depth map is considered.

We first compare "ERSC_UEP" to "SC_UEP" in terms of FEC protection. We compute the average code rate $\bar{R}$ as follows.

$$\bar{R} = \frac{R_{ts} + R_{ds}}{R_t} \tag{10}$$

where R_{ts} and R_{ds} represent the texture source bits and the depth source bits, respectively. R_t refers to the total bits including source and channel bits for texture and depth. Fig. 6 (a) shows the average code rate versus bit rate constraint for the sequence of "Undo_Dancer". As can be observed, when the bit rate constraint increases, $\bar{R}$ decreases, which means that more protection is offered by both algorithms for 3D video transmission when increasing the bit rate. We also see that the average code rate of "ERSC_UEP" is lower than that of "SC_UEP" for all the bit rate constraints, which means that a stronger protection is consistently carried out by our proposed algorithm. The reason for the relatively lower $\bar{R}$ in "SC_UEP" can be explained as follows. Firstly, when modeling the expected quality of the synthesized right view, "SC_UEP" assumes that either the texture packet or the depth packet is lost, which explicitly ignores the case of the simultaneous loss of them. Secondly, "SC_UEP" only considers error propagation occurred in the event that the texture or depth packet is lost, and assumes error-free decoded

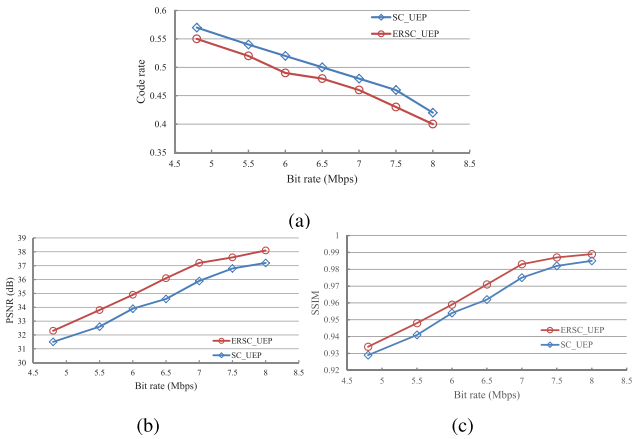


Fig. 6. Comparison between SC_UEP and ERSC_UEP in terms of code rate and objective video quality for "Undo_Dancer" sequence at the transport packet loss rate of 10%. (a) Code rate versus bit rate. (b) PSNR versus bit rate. (c) SSIM versus bit rate.

synthesized view can be generated in the case of the texture or depth packet being not lost. However, in practice, even though the current packet is received correctly, there still exists channel distortion propagated from the predictor packet in the reference frame. These two simplified assumptions will lead to underestimation of the overall expected distortion, and then less channel coding bit rate is allocated for error protection in joint source and channel optimization in "SC_UEP". Lastly, we compare the performance of "ERSC_UEP" to that of "SC_UEP". The corresponding result in terms of PSNR is shown in Fig. 6 (b). It is evident that "ERSC_UEP" achieve substantial performance gain over "SC_UEP", with up to 1.5 dB at the particular target bit rate of 6.5 Mbps. This is expected since "ERSC_UEP" allocates more bits for channel coding than "SC_UEP". In addition to that, "ERSC_UEP", which utilizes mode switching as a mean of error-resilient source coding, can effectively eliminates error propagation caused by the residual channel errors that cannot be corrected by channel coding. Fig. 6 (c) illustrates the performance comparison by using SSIM metric [43]. It is also demonstrated that ERSC_UEP outperforms SC_UEP. We also verify the performance of our algorithm compared to SC_UEP under other packet loss rates, and the results are similar to those of Fig. 6.

In the experiments discussed above, all the simulations run over a random packet loss pattern based packet-switched network (i.e., Internet). In order to verify the performance of the proposed joint source and channel coding algorithm in the context of wireless channel, in this test, we conduct simulations of 3D video over a flat Rayleigh fading channel with binary phase-shift keying (BPSK) modulation/demodulation. Due to the time-varying characteristic of the wireless channel, the packet loss probability is no longer constant as in (1), but depends on the source packet in bits, the code rate allocated to that packet, single to noise ratio (SNR), and the coherence time [21]. The coherence time of a fading channel here represents the number of symbols affected by the same fade level, and supposing a block-fading channel, each fade is

TABLE V

PERFORMANCE COMPARISON OF "UNDO_DANCER" SEQUENCE BETWEEN SC_UEP AND ERSC_UEP FOR A FLAT RAYLEIGH FADING CHANNEL WITH THE SNR OF 8 dB AND THE COHERENCE TIME OF 4000

Total bit rate (Mbps)	Performance comparison			
	PSNR		SSIM	
	SC_UEP	ERSC_UEP	SC_UEP	ERSC_UEP
5.5	30.2	31.6	0.942	0.948
6.5	32.4	33.8	0.963	0.972
7.5	33.8	35.7	0.981	0.987
8.5	36.4	37.9	0.984	0.989

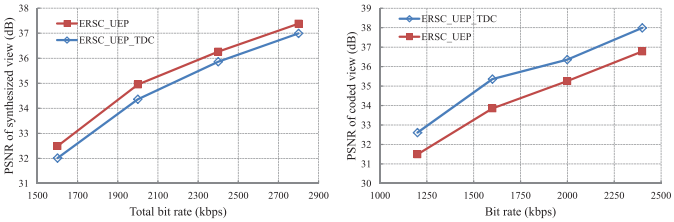


Fig. 7. Quality comparison between ERSC_UEP_TDC and ERSC_UEP for "BookArrival" sequence at the transport packet loss rate of 5%. (a) PSNR versus total bit rate for the synthesized view. (b) PSNR versus bit rate for the texture video.

considered to be independent of the others. Since there is no closed-form expression that can be used to calculate the packet loss probabilities p, we obtain the probabilities for various packet sizes experimentally as done in [21]. The obtained packet loss probabilities are then used for distortion estimation. The performance comparison results between SC_UEP and ERSC_UEP for a flat Rayleigh fading channel are shown in Table V, where the SNR is 8 dB and the coherence time is 4000. As can be observed, ERSC_UEP also consistently outperforms SC_UEP in a variety of target bit rate for the wireless channels.

E. Performance of the Proposed Framework With Texture Distortion Constraint

In this section, the proposed method described above solving the joint source and channel coding of 3D video with texture distortion constraint is denoted by "ERSC_UEP_TDC". Fig. 7 shows a comparison of the results of ERSC_UEP_TDC versus ERSC_UEP for the sequence of "Undo_Dancer". Both schemes use the same bit budget. As can be observed, at the transport packet loss probability of 5%, ERSC_UEP beats ERSC_UEP_TDC by less than 0.5 dB in the view synthesis quality. However, in terms of the texture quality, ERSC_UEP_TDC outperforms ERSC_UEP by around 1.4 dB. This demonstrates the benefits of the consideration of the additional constraint of texture distortion when optimizing the overall video quality of 3D video for a given total bit rate. It should be noted that, although Fig. 7 only shows the performance improvement relative to ERSC_UEP for packet loss rate of 5%, our test over other packet loss rates also yields similar results.

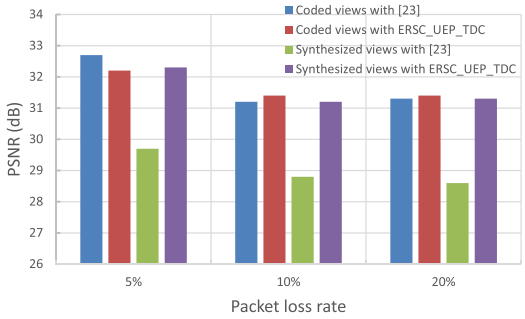


Fig. 8. Average PSNR of the coded and synthesized views versus the packet loss rate for the "Ballet" sequence.

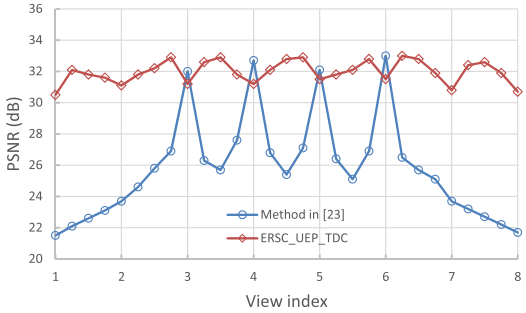


Fig. 9. PSNR per view for the proposed algorithm and the method introduced in [23].

In order to further demonstrate the effectiveness of the proposed algorithm, we make a comparison with the state-of-the-art [23], where a popularity-aware joint source and channel coding optimization framework for multi-view video is proposed. In this method, the scenario is considered, where multiview video content is streamed to multiple heterogeneous clients, and each of them has varying access link characteristics. Thus, the objective of this method is to optimally allocate source and channel coding rates to the captured content to maximize the aggregate video quality across the client population. In contrast to [23], the objective of this paper is to minimize the view synthesis distortion given a total bit rate for both texture and depth of the captured content with a maximum tolerable distortion constraint of the coded view for only one target client class. Therefore, for fair comparison, we consider the delivery of multi-view video to one client class, where the client access link is characterized by a packet erasure channel. Further, in the comparison, the Ballet sequence with 8 captured views is compressed, whereas three virtual views between each pair of camera viewpoints are synthesized. The clients' view popularity distribution is characterized by a smooth Gaussian function with a peak at the view 4.5 and variance of 1.5 as in [23]. The transmission rate is set to 4.45 MB/sec.

Fig. 8 compares the average video qualities of the captured and synthesized views, versus the packet loss rates, and Fig. 9 shows the associated video quality per view. It can be observed from Fig. 8 that the proposed algorithm significantly outperforms the scheme in [23] for the synthesized views on average, while the quality of the coded views of the proposed algorithm is only slightly inferior to that of [23] at the packet loss rate of 5%. Fig. 9 also demonstrates that the proposed algorithm leads to smoother video quality transition across all reconstruction viewpoints. The reason for the performance gains and the characteristic of much less quality variation between views of the proposed algorithm as opposed to [23] can be explained as follows. Firstly, in [23], the video qualities of the captured and synthesized views are optimized in an integrated manner based on view popularity distribution. Although this kind of optimization framework would lead to a higher reconstruction quality of the popular views (i.e., captured views) by placing more weights on these views, it also generally leads to a poor reconstruction quality for the remaining views (i.e., synthesized views), resulting in a large variation in reconstruction quality across views. However, in the designed objective function of our work, we only optimize the reconstruction quality of the intermediate virtual views, which can improve the quality of neighbouring synthetic views as much as possible. In order to guarantee the quality of the captured views, we impose an additional distortion constraint on the captured view during the optimization process as illustrated in Section V, which can thus better balance the view synthesis quality and the captured view quality. Secondly, compared to [23] that selects the optimal source and channel coding rates at the frame level, we allocate the source and channel coding rates for texture and depth at the block level for finer granularity, which is more flexible in varying the channel rates in response to the image content. We also develop a trellis state pruning algorithm to facilitate the search of the optimal solution for each texture and depth blocks. Finally, we consider various error propagation distortions (including the temporal, intra, inter-view, inter-component propagated distortions) incurred in predictive coding of texture and depth map in modeling the expected view synthesis distortion, and design an efficient error-resilient source coding in the form of joint prediction mode switching of texture and depth, which can effectively mitigate the potential error propagation induced by residual channel errors that cannot be corrected by channel coding as demonstrated in Section VI-B. It should be noted that, the results presented here for the proposed scheme are obtained with the channel coding scheme RS. It is expected that the proposed method can further improve the reconstruction quality for both the virtual and captured views, providing that more sophisticated channel coding methods (e.g., expanding window rateless codes) are employed.

VII. Conclusion

In this paper, we develop a joint source and channel coding scheme for depth-based 3D video transmission. Specifically, we propose a general framework that jointly considers error-resilient source coding and channel coding for texture and depth. In particular, error-resilient source coding is achieved by coding mode selection in both texture and depth with the objective to minimize the overall end-to-end synthetic distortion. The proposed framework can automatically allocate the available bit rate between texture and depth, within texture or depth, between source and channel coding. Further, we also tackle the optimization problem with additional texture distortion constraint, for which we develop a

solution to trade off the synthesized view and texture qualities. Finally, we have compared the performance of the proposed framework to those of the pure error-resilient source coding, the error-resilient source coding with equal channel coding protection, and the existing channel coding schemes without error resiliency consideration. Experimental results reveal that the proposed framework yields superior performance in both general bit rate constrained and additional texture distortion constrained cases.

REFERENCES

[1] K. Müller, P. Merkle, and T. Wiegand, "3-D video representation using depth maps," *Proc. IEEE*, vol. 99, no. 4, pp. 643–656, Apr. 2011.

[2] C. G. Gurler, B. Gorkemli, G. Saygili, and A. M. Tekalp, "Flexible transport of 3-D video over networks," *Proc. IEEE*, vol. 99, no. 4, pp. 694–707, Apr. 2011.

[3] P.-C. Huang, J.-R. Lin, G.-L. Li, K.-H. Tai, and M.-J. Chen, "Improved depth-assisted error concealment algorithm for 3D video transmission," *IEEE Trans. Multimedia*, vol. 19, no. 11, pp. 2625–2632, Nov. 2017.

[4] M. M. Hannuksela *et al.*, "Multiview-video-plus-depth coding based on the advanced video coding standard," *IEEE Trans. Image Process.*, vol. 22, no. 9, pp. 3449–3458, Sep. 2013.

[5] G. Tech, Y. Chen, K. Müller, J.-R. Ohm, A. Vetro, and Y.-K. Wang, "Overview of the multiview and 3D extensions of high efficiency video coding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 26, no. 1, pp. 35–49, Jan. 2016.

[6] A. I. Purica, E. G. Mora, B. Pesquet-Popescu, M. Cagnazzo, and B. Ionescu, "Multiview plus depth video coding with temporal prediction view synthesis," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 26, no. 2, pp. 360–374, Feb. 2016.

[7] M. Winken, H. Schwarz, and T. Wiegand, "Motion vector inheritance for high efficiency 3D video plus depth coding," in *Proc. Picture Coding Symp.*, May 2012, pp. 45–48.

[8] G. Cheung, V. Velisavljevic, and A. Ortega, "On dependent bit allocation for multiview image coding with depth-image-based rendering," *IEEE Trans. Image Process.*, vol. 20, no. 11, pp. 3179–3194, Nov. 2011.

[9] J. Chakareski, V. Velisavljević, and V. Stanković, "User-action-driven view and rate scalable multiview video coding," *IEEE Trans. Image Process.*, vol. 22, no. 9, pp. 3473–3484, Sep. 2013.

[10] V. Velisavljevic, C. Dorea, J. Chakareski, and R. de Queiroz, "Convexity characterization of virtual view reconstruction error in multi-view imaging," in *Proc. IEEE Workshop Multimedia Signal Process. (MMSP)*, Luton, U.K., Oct. 2017, pp. 1–6.

[11] J. Chakareski, "Transmission policy selection for multi-view content delivery over bandwidth constrained channels," *IEEE Trans. Image Process.*, vol. 23, no. 2, pp. 931–942, Feb. 2014.

[12] J. Chakareski, "Wireless streaming of interactive multi-view video via network compression and path diversity," *IEEE Trans. Commun.*, vol. 62, no. 4, pp. 1350–1357, Apr. 2014.

[13] A. Hamza and M. Hefeeda, "Adaptive streaming of interactive free viewpoint videos to heterogeneous clients," in *Proc. ACM Multimedia Syst. (MMSys) Conf.*, Klagenfurt, Austria, May 2016, Art. no. 10.

[14] J. Chakareski, "Uplink scheduling of visual sensors: When view popularity matters," *IEEE Trans. Commun.*, vol. 63, no. 2, pp. 510–519, Feb. 2015.

[15] Z. Liu, G. Cheung, J. Chakareski, and Y. Ji, "Multiple description coding and recovery of free viewpoint video for wireless multi-path streaming," *IEEE J. Sel. Topics Signal Process.*, vol. 9, no. 1, pp. 151–164, Feb. 2015.

[16] D. Zhang and J. Liang, "View synthesis distortion estimation with a graphical model and recursive calculation of probability distribution," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 25, no. 5, pp. 827–840, May 2015.

[17] B. Macchiavello, C. Dorea, E. M. Hung, G. Cheung, and W.-T. Tan, "Reference frame selection for loss-resilient depth map coding in multiview video conferencing," *Proc. SPIE*, vol. 8305, pp. 83050C-1–83050C-11, Feb. 2012.

[18] B. Macchiavello, C. Dorea, E. M. Hung, G. Cheung, and W.-T. Tan, "Loss-resilient coding of texture and depth for free-viewpoint video conferencing," *IEEE Trans. Multimedia*, vol. 16, no. 3, pp. 711–725, Apr. 2014.

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

[20] B. Kamolrat, W. A. C. Fernando, M. Mrak, and A. Kondoz, "Joint source and channel coding for 3D video with depth image—Based rendering," *IEEE Trans. Consum. Electron.*, vol. 54, no. 2, pp. 887–894, May 2008.

[21] A. Vosoughi, P. C. Cosman, and L. B. Milstein, "Joint source-channel coding and unequal error protection for video plus depth," *IEEE Signal Process. Lett.*, vol. 22, no. 1, pp. 31–34, Jan. 2015.

[22] A. Vosoughi, V. Testoni, P. Cosman, and L. B. Milstein, "Joint source-channel coding of 3D video using multiview coding," in *Proc. Int. Conf. Acoust., Speech, Signal Process. (ICASSP)*, Apr. 2013, pp. 2050–2054.

[23] J. Chakareski, V. Velisavljević, and V. Stanković, "View-popularity-driven joint source and channel coding of view and rate scalable multi-view video," *IEEE J. Sel. Topics Signal Process.*, vol. 9, no. 3, pp. 474–486, Apr. 2015.

[24] P. Gao, W. Xiang, and B. Wang, "Optimal mode selection and channel coding for 3-D video streaming over the Internet," in *Proc. IEEE Int. Conf. Multimedia Expo Workshops (ICMEW)*, Jul. 2017, pp. 31–36.

[25] M. Gallant and F. Kossentini, "Rate-distortion optimized layered coding with unequal error protection for robust Internet video," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 11, no. 3, pp. 357–372, Mar. 2001.

[26] R. Zhang, S. L. Regunathan, and K. Rose, "Video coding with optimal inter/intra-mode switching for packet loss resilience," *IEEE J. Sel. Areas Commun.*, vol. 18, no. 6, pp. 966–976, Jun. 2000.

[27] P. Gao, Q. Peng, and W. Xiang, "Analysis of packet-loss-induced distortion in view synthesis prediction-based 3D video coding," *IEEE Trans. Image Process.*, vol. 26, no. 6, pp. 2781–2796, Jun. 2017.

[28] Y. Zhang, W. Gao, Y. Lu, Q. Huang, and D. Zhao, "Joint source-channel rate-distortion optimization for H.264 video coding over error-prone networks," *IEEE Trans. Multimedia*, vol. 9, no. 3, pp. 445–454, Apr. 2007.

[29] Z. He and H. Xiong, "Transmission distortion analysis for real-time video encoding and streaming over wireless networks," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 16, no. 9, pp. 1051–1062, Sep. 2006.

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

[31] Y.-L. Chang, Y. Zhang, and P. C. Cosman, "Joint source-channel rate-distortion optimization with motion information sharing for H.264/AVC video-plus-depth coding," in *Proc. 48th Asilomar Conf. Signals, Syst. Comput.*, Nov. 2014, pp. 678–682.

[32] K. Ramchandran and M. Vetterli, "Best wavelet packet bases in a rate-distortion sense," *IEEE Trans. Image Process.*, vol. 2, no. 2, pp. 160–175, Apr. 1993.

[33] D. P. Bertsekar, *Nonlinear Programming*, 2nd ed. New York, NY, USA: Athena Scientific, 2003.

[34] G. Tech, K. Müller, H. Schwarz, and T. Wiegand, "Partial depth image based re-rendering for synthesized view distortion computation," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 28, no. 6, pp. 1273–1287, Jun. 2018.

[35] H. Yuan, S. Kwong, J. Liu, and J. Sun, "A novel distortion model and lagrangian multiplier for depth maps coding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 24, no. 3, pp. 443–451, Mar. 2014.

[36] *3D-ATM Reference Software Version 6.0*. Accessed: Nov. 2012. [Online]. Available: http://mpeg3dv.nokiaresearch.com/svn/mpeg3dv/tags/3DV-ATMv6.0/

[37] *ISO/IEC JTC1/SC29/WG11, 3DV/FTV EE2: Report on VSRS Extrapolation*, document M18356, Moving Picture Experts Group, Guangzhou, China, 2010.

[38] *Common Test Conditions of 3DV Core Experiments*, document JCT3V-G1100, ITU-T SG 16 WP 3, ISO/IEC JTC 1/SC 29/WG 11, Joint Collaborative Team on 3D Video Coding Extension Development, San Jose, CA, USA, 2014.

[39] S. Wenger, "Proposed error patterns for Internet experiments," document Q 15-I-16r1, ITU-T VCEG, Oct. 1999.

[40] *3D-HTM Reference Software Version 16.0*. Accessed: Mar. 2016. [Online]. Available: https://hevc.hhi.fraunhofer.de/svn/svn_3DVCSoftware/tags/HTM-16.0/

[41] M.-C. Hong, H. Schwab, L. P. Kondi, and A. K. Katsaggelos, "Error concealment algorithms for compressed video," *Signal Process., Image Commun.*, vol. 14, nos. 6–8, pp. 473–492, 1999.

[42] V.-H. Doan, V.-A. Nguyen, and M. N. Do, "Efficient view synthesis based error concealment method for multiview video plus depth," in *Proc. IEEE Int. Symp. Circuits Syst. (ISCAS)*, May 2013, pp. 2900–2903.

[43] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: From error visibility to structural similarity," *IEEE Trans. Image Process.*, vol. 13, no. 4, pp. 600–612, Apr. 2004.

Pan Gao (S'14–M'16) received the B.Eng. degree in computer science and technology from Sichuan Normal University, Chengdu, China, in 2009, and the Ph.D. degree in electronic engineering from the University of Southern Queensland (USQ), Toowoomba, Australia, in 2017.

Since 2016, he has been an Assistant Professor with the College of Computer Science and Technology, Nanjing University of Aeronautics and Astronautics, Nanjing, China. He is currently a Post-Doctoral Research Fellow at the School of Computer Science and Statistics, Trinity College Dublin, Dublin, Ireland, working on the V-SENSE project funded by the Science Foundation Ireland. His research interests include Multiview/3D video coding, volumetric video processing and compression, and graph signal processing. He received the Publication Excellence Award from USQ in 2015.

Wei Xiang (S'00–M'04–SM'10) received the B.Eng. and M.Eng. degrees in electronic engineering from the University of Electronic Science and Technology of China, Chengdu, China, in 1997 and 2000, respectively, and the Ph.D. degree in telecommunications engineering from the University of South Australia, Adelaide, Australia, in 2004. From 2004 to 2015, he was with the School of Mechanical and Electrical Engineering, University of Southern Queensland, Toowoomba, Australia.

He is currently the Founding Professor and the Head of the Discipline of Internet of Things Engineering, James Cook University, Cairns, Australia. He has published over 250 peer-reviewed papers with over 130 journal articles. His research interests fall under the broad areas of communications and information theory, particularly the Internet of Things, and coding and signal processing for multimedia communications systems. Due to his instrumental leadership in establishing Australia's first Internet of Things Engineering Degree Program, he was elected into Pearcy Foundation's Hall of Fame in 2018. He is an Elected Fellow of IET, U.K., and Engineers Australia. He received the TNQ Innovation Award in 2016, the Pearcey Entrepreneurship Award (Highly Commended) in 2017, and the Engineers Australia Cairns Engineer of the Year in 2017. He was a co-recipient of three Best Paper Awards at 2009 ICWMC, 2011 IEEE WCNC, and 2015 WCSP. He has been awarded several prestigious fellowship titles. He was named a Queensland International Fellow (2010–2011) by the Queensland Government of Australia, an Endeavour Research Fellow (2012–2013) by the Commonwealth Government of Australia, a Smart Futures Fellow (2012–2015) by the Queensland Government of Australia, and a JSPS Invitational Fellow jointly by the Australian Academy of Science and Japanese Society for Promotion of Science (2014–2015). He is a Vice Chair of the IEEE Northern Australia Section. He has served a large number of international conferences in the capacity of a general co-chair, TPC co-chair, symposium chair, and so on. He was an Editor of the IEEE COMMUNICATIONS LETTERS (2015–2017), and he is an Associate Editor for *Telecommunications Systems* (Springer).

Dong Liang received the B.S. and M.S. degrees from the School of Information Science and Engineering, Lanzhou University, China, in 2008 and 2011, respectively, and the Ph.D. degree from the Graduate School of Information Science and Technology, Hokkaido University, Japan, in 2015. He is currently an Assistant Professor with the College of Computer Science and Technology, Nanjing University of Aeronautics and Astronautics, Nanjing, China.

His research interests include machine vision, pattern recognition, and computational photography.