Depth Intra Coding for 3D Video Based on Geometric Primitives

Philipp Merkle, *Member, IEEE*, Karsten Müller, *Senior Member, IEEE*, Detlev Marpe, *Fellow, IEEE*, and Thomas Wiegand, *Fellow, IEEE*

***Abstract*— This paper presents an advanced depth intra-coding approach for 3D video coding based on the High Efficiency Video Coding (HEVC) standard and the multiview video plus depth (MVD) representation. This paper is motivated by the fact that depth signals have specific characteristics that differ from those of natural signals, i.e., camera-view video. Our approach replaces conventional intra-picture coding for the depth component, targeting a consistent and efficient support of 3D video applications that utilize depth maps or polygon meshes or both, with a high depth coding efficiency in terms of minimal artifacts in rendered views and meshes with a minimal number of triangles for a given bit rate. For this purpose, we introduce intra-picture prediction modes based on geometric primitives along with a residual coding method in the spatial domain, substituting conventional intra-prediction modes and transform coding, respectively. The results show that our solution achieves the same quality of rendered or synthesized views with about the same bit rate as MVD coding with the 3D video extension of HEVC (3D-HEVC) for high-quality depth maps and with about 8% less overall bit rate as with 3D-HEVC without related depth tools. At the same time, the combination of 3D video with 3D computer graphics content is substantially simplified, as the geometry-based depth intra signals can be represented as a surface mesh with about 85% less triangles, generated directly in the decoding process as an alternative decoder output.**

***Index Terms*— 3D video coding, 3D video extension of HEVC (3D-HEVC), depth intra coding, High Efficiency Video Coding (HEVC), inter component prediction, mesh extraction, multiview video plus depth (MVD), wedgelets.**

I. Introduction

In RECENT years, 3D video technology has matured along with intensified research on all stages of the processing chain from 3D video capture to the display technology. This especially includes new and advanced 3D video coding methods for efficient compression and transmission as well as novel applications that combine 3D video and 3D computer graphics elements.

Manuscript received October 22, 2014; revised January 9, 2015 and February 9, 2015; accepted February 24, 2015. Date of publication February 27, 2015; date of current version March 3, 2016. This paper was recommended by Associate Editor J. Lu.

P. Merkle, K. Müller, and D. Marpe are with the Department of Video Coding and Analytics, Fraunhofer Institute for Telecommunications, Heinrich Hertz Institute, Berlin 10587, Germany (e-mail: philipp.merkle@hhi.fraunhofer.de; karsten.mueller@hhi.fraunhofer.de; detlev.marpe@hhi.fraunhofer.de).

T. Wiegand is with the Department of Video Coding and Analytics, Fraunhofer Institute for Telecommunications, Heinrich Hertz Institute, Berlin 10587, Germany, and also with the Department of Electrical Engineering and Computer Science, Technical University of Berlin, Berlin 10623, Germany (e-mail: thomas.wiegand@hhi.fraunhofer.de).

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

Digital Object Identifier 10.1109/TCSVT.2015.2407791

According to [1], 3D video solutions can be categorized by the representation format, namely, those based on stereo and multiview signals and those that additionally use depth maps. Regarding the first category, an important milestone was the multiview video coding (MVC) extension of the H.264/Advanced Video Coding (AVC) standard. Here, the multiview video (MVV) representation format is used, showing the same scene from two or more different perspectives. By adding the concept of disparity-compensated prediction, inter-view dependencies are efficiently exploited with MVC, resulting in a significant coding gain compared with that of simulcast coding [2]. Target applications of MVC were efficient compression and transmission of stereo video, and consequently, it was adopted for the Blu-ray 3D format. For supporting backward compatibility and a direct implementation on top of existing solutions, the design of MVC is restricted to high-level syntax changes of H.264/AVC. This concept is now also applied to the High Efficiency Video Coding (HEVC) standard [3] for defining a simple stereo and multiview video coding extension (MV-HEVC) [4] that benefits from the significantly better coding efficiency of HEVC compared with H.264/AVC.

The second category is based on the multiview video plus depth (MVD) representation. By enhancing MVV with the associated depth signal, a new class of 3D video solutions is enabled: via depth image-based rendering (DIBR), perspectively correct virtual camera views of the scene can be synthesized for arbitrary view positions. Typical applications in this category are the support of autostereoscopic displays with a large number of views, baseline adaptation for different screen sizes, and free-viewpoint video.

One key aspect for the success of such applications is efficient compression and transmission of MVD data. The depth intra-coding solution we present in this paper is a continued development of our previous and ongoing work in this field [5]–[8], but with a new direction, targeting convergence of 3D video and 3D computer graphics content. Our approach is motivated by the fact that the characteristics of depth signals differ from those of natural video signals, featuring sharp edges (representing object borders) and larger areas of nearly constant or only slowly varying sample values (representing object areas). Consequently, hybrid video codecs like HEVC—that are highly optimized for the statistics of natural video signals—are not fully suitable for depth coding. We therefore propose a coding approach that is adapted to the specific characteristics of depth signals, replacing intra-prediction modes as well as transform residual coding of HEVC. Our solution is optimized for both efficient

compression of depth signals and efficient support of triangular meshes for representing the scene surface. This introduces additional constraints on the design of the coding scheme, resulting in the carefully considered combination of adapted existing and newly developed methods we present in this paper.

Specific depth compression methods are a topic of high interest in the area of 3D video coding. Related to our solution, approaches for non-rectangular block segmentation and wedgelets, inter-component prediction, and residual coding for depth have been studied. Morvan *et al.* [9] proposed platelet-based coding of depth maps. By modeling the signal of depth blocks using piecewise-linear functions and omitting the residual signal, they reported a bit rate gain of about 25% compared with that of JPEG2000. For the same method, Merkle *et al.* [6] reported that the bit rate is about 25% higher than that of H.264/AVC intra-only coding, while leading to a better quality of rendered views using the coded depth maps. Liu *et al.* [10] propose a trilateral filter together with a sparse dyadic mode for depth intra coding. The latter combines rectangular and diagonal block partitions with an inter-component predicted contour refinement. The resulting piecewise-constant depth signals are reconstructed from predictively coded constant partition value offsets—a similar approach has later been applied in [8] for residual adaptation of inter-component predicted wedgelet and contour segmentations. As an extension of H.264/MVC, they report about 1.5 dB total gain in rendered view quality. Oh and Ho [11] proposed inter-component prediction for motion vectors using the motion vectors of the corresponding video block also for the depth block. As an extension of H.264/AVC, they report bit rate gains of about 10% for low rates, while a worse performance is achieved for high rates. Winken *et al.* [12] propose inter-component prediction for motion vectors by inheriting motion and block partitioning from video to depth. As an extension of multiview HEVC, they report about 10% reduction in the depth bit rate. Mora *et al.* [13] propose inter-component prediction of quadtree partitions using the video quadtree as a reference for limitation and prediction of the depth quadtree, reporting significant encoder complexity reduction and small depth bit rate gains for 3D video extension of HEVC (3D-HEVC) [4]. Jäger and Naser [14] study low-complexity transform coding for depth maps by replacing the discrete cosine transform (DCT) with Haar or Walsh–Hadamard transforms in 3D-HEVC, reporting less complexity and some bit rate reduction for higher depth qualities. Our previous work on depth intra coding introduced wedgelet block segmentation with residual adaptation in [8] and inter-component prediction of wedgelet and contour segmentations in [7], reporting about 11% and 6% reduction for the depth bit rate, respectively. In contrast to the approaches proposed in this paper, these extend MV-HEVC by additional modes and methods, without integrating and evaluating the effect of the view synthesis optimization (VSO) [15] distortion metric.

The main achievement of our solution relative to related work is the consistent and efficient coding scheme for 3D video applications that utilize depth maps or polygon meshes or both, with all building blocks optimized for a high depth coding efficiency in terms of minimal artifacts in rendered views as well as mesh extraction with a minimal number of triangles for a given bit rate. The important new contributions are the mesh extraction algorithm that smoothly integrates in the decoder, the specially adapted set of modeling functions and corresponding prediction modes, and the enhancement of the constant offset residual concept for fully replacing transform residual coding in all stages of the depth-related intra encoder and decoder. The main principle of our approach is approximation of the signal of a depth block by modeling functions based on geometric primitives that allow representing the scene surface with a minimum number of triangles. In Section II, the two basic types of geometric depth models are introduced, namely, plane fitting for areas with a planar characteristic and wedgelet and contour segmentations for sharp edges. Following this principle, a full intra-coding solution for depth is developed. The details are explained in Section III, including intra prediction and inter-component prediction as well as constant offset residual coding in the spatial domain. Section IV is on the application of our solution, including codec integration based on 3D-HEVC and mesh extraction. Finally, the results are presented in Section V.

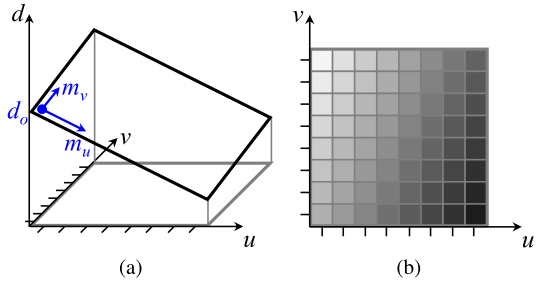

Fig. 1. Plane model of a depth block. (a) Linear function with offset d_o and slopes m_u and m_v. (b) Resulting depth values $d_M(u, v)$.

II. Geometric Depth Modeling

This section introduces the two approaches for approximating the signal of a depth block by geometric modeling functions. Both are adapted to a specific depth signal feature, namely, plane fitting for areas with constant or slowly changing values and non-rectangular block segmentation for sharp edges. Conceptually similar modeling functions have been applied in previous works, for instance, wedgelet and plane models in [9] or wedgelet and contour models in [7].

A. Plane Fitting

The basic principle of this depth signal modeling approach is approximation of the signal of a rectangular block by a linear model that describes a plane. This type of model targets a close approximation of depth blocks with a planar signal characteristic—typically representing flat scene areas or objects.

As shown in Fig. 1, the plane model of a depth block with sample values $d_M(u, v)$ is defined by a linear function as

$$d_M(u, v) = d_o + m_u \cdot u + m_v \cdot v \tag{1}$$

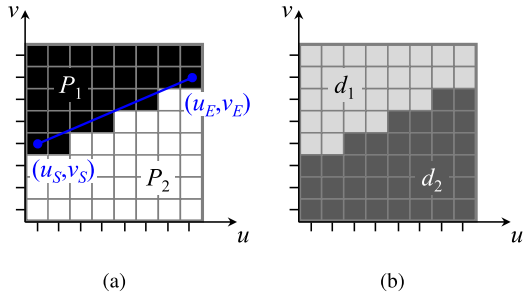

Fig. 2. Wedgelet model of a depth block. (a) Segmentation pattern with partitions P_1 and P_2. (b) Constant segment values d_1 and d_2.

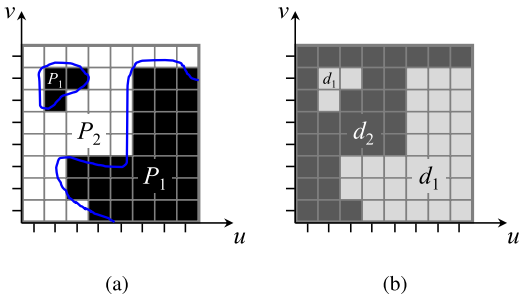

Fig. 3. Contour model of a depth block. (a) Segmentation pattern with partitions P_1 and P_2. (b) Constant segment values d_1 and d_2.

with d_o being the offset at position $(0, 0)$ and $m_{u/v}$ the slope of the plane in both coordinate directions.

Given a depth block with original sample values $d(u, v)$, deriving the best approximation by a plane model means finding the plane with parameters d_o, m_u, and m_v that causes the minimum distortion compared with the original signal. The general approach for deriving the minimum distortion linear model for a given set of sample values is known as linear regression [16]. The most commonly used distortion metric for this is the mean squared error (MSE), such that the least-squares linear regression method derives the linear model with the minimum MSE. For sample values assigned with more than one coordinate, the method is extended to multiple linear regression [16]. In the case of two coordinates, like the (u, v) of our depth block, this is also referred to as least squares regression plane or plane fitting.

B. Non-rectangular Bisegmentation

According to [7], the basic principle of this depth signal modeling approach is approximation of the signal of a rectangular block by a model that segments the area into two non-rectangular regions, where each of the segments is represented by a constant value. The information required for such a model consists of two elements: the segmentation information, specifying the region each sample belongs to, and the region value information, specifying a constant value for each region. For a depth block of size $N \times N$, the signal model is constructed as follows: given the segmentation information in the form of a separation line that determines the two segments P_1 and P_2, a segmentation pattern is constructed by assigning each sample to one segment, depending on its location relative to the line. Such a pattern consists of an array of $N \times N$ binary elements, containing the information whether the corresponding sample belongs to segment P_1 or P_2. Given the region value information in the form of two constant depth values d_1 and d_2, corresponding to segments P_1 and P_2, the signal model with sample values $d_M(u, v)$ is defined with $k \in \{1, 2\}$ as

$$d_M(u, v) = d_k \quad \text{if } (u, v) \in P_k. \tag{2}$$

In the following, we introduce signal modeling with segmentations of type wedgelet and contour in more detail. Both target a close approximation of sharp edges in areas of nearly constant sample values—typically representing the border between foreground and background scene areas or objects.

1) Wedgelet Segmentation: Like in [8], this approach utilizes wedgelets as a special type of non-rectangular block segmentation. According to Fig. 2, the wedgelet segmentation is characterized by the fact that the two partitions are separated by a straight line. Consequently, wedgelet models are very consistent, but restricted to linear approximations of object edges. The separation line is defined by slope m and offset n as

$$v(u) = m \cdot u + n \quad \text{with } m = \frac{v_E - v_S}{u_E - u_S}, \quad n = v_S - m \cdot u_S \tag{3}$$

using start point (u_S, v_S) and end point (u_E, v_E). Fig. 2(a) shows the wedgelet segmentation with partitions P_1 and P_2, including the separation line and resulting pattern, and Fig. 2(b) the wedgelet signal model with the constant segment values d_1 and d_2.

We employ such wedgelet models for approximating the signal of a given depth block of size $N \times N$ with values $d(u, v)$. In the context of image and video processing, the best reconstruction of a signal for a given set of candidate reconstructions is usually defined as the one that causes the minimum distortion compared with the original signal. Translated to our wedgelet approach, this means finding the model that results in the closest approximation of the original depth values $d(u, v)$ by carrying out a minimum distortion search with a set of candidate segmentation patterns. Given a wedgelet pattern with partitions P_1 and P_2, the distortion is derived as follows: first, value d_k for each segment $k \in \{1, 2\}$ is calculated as the mean value of the depth samples covered by the corresponding segment as

$$d_k = \frac{1}{n_k} \cdot \sum_{\forall (u,v) \in P_k} d(u, v) \quad \text{with } n_k = \sum_{\forall (u,v) \in P_k} 1. \tag{4}$$

From the segmentation pattern and the resulting segment values d_1 and d_2, the wedgelet model with sample values $d_M(u, v)$ is constructed using (2). Finally, the distortion is calculated from the difference between $d(u, v)$ and $d_M(u, v)$.

The algorithm for the minimum distortion search consists of calculating the distortion for each candidate segmentation pattern and selecting the one with the minimum value as the best wedgelet model approximation of the depth signal. The set of candidate partitions depends on the particular application

and its constraints. For definitely finding the wedgelet model with the overall minimum distortion, the candidate set has to consist of the wedgelet patterns for all possible combinations of the line start and end positions.

2) Arbitrary Contour Segmentation: Like in [7], this approach utilizes arbitrary contours as a special type of non-rectangular block segmentation. According to Fig. 3, the contour segmentation is characterized by the fact that the two partitions are separated by an arbitrarily shaped line that can even consist of several parts. In contrast to wedgelet models, any complex shaped object edge can be approximated by a contour model, but the segmentation line cannot be described by a simple geometrical function. Fig. 3(a) and (b) shows the contour segmentation with partitions P_1 and P_2 and the contour signal model with segment values d_1 and d_2, respectively.

We employ such contour models for approximating the signal of a given depth block of size $N \times N$ with values $d(u, v)$. Here, the partition is not derived based on a set of candidate segmentations, but by direct image segmentation of the depth block. The most suitable method for this purpose is (adaptive) thresholding [17]. Thresholding is known from digital image processing as a very simple image segmentation approach, extracting two arbitrarily shaped regions from a grayscale image, separating bright from dark areas. Adaptive thresholding means that the image is subdivided into individual subimages and the thresholding is separately applied to each of them.

We apply adaptive thresholding for deriving the contour segmentation of a depth block. Here, subimages correspond to depth blocks of size $N \times N$ and bright and dark image regions to depth values $d(u, v)$ of foreground and background objects, respectively. The segmentation threshold d_c is set equal to the mean value of the depth block as

$$d_c = \frac{1}{N^2} \cdot \sum_{\forall (u,v)} d(u, v). \tag{5}$$

Finally, each sample is assigned to one of the two partitions, depending on whether its value is above or below the threshold, i.e., all samples with $d(u, v) \geqslant d_c$ are assigned to P_1 and all samples with $d(u, v) < d_c$ to P_2.

III. Depth Coding Based on Geometric Primitives

The geometry-based depth modeling approach introduced in Section II can be used for efficient depth compression by taking advantage of its quality to closely approximate the signal of a depth block for predictive coding. Accordingly, the information or parameters describing the model need to be available for reconstruction at the decoder. In principle, the required information is either derived from the available information of previously decoded pictures and blocks (prediction) or determined at the encoder and transmitted in the bitstream (estimation)—often combined in a way that only the difference between predicted and estimated information is transmitted. At the encoder, the decision on whether and to what extend estimated information is transmitted to the decoder is typically based on a cost function that balances the tradeoff between rate and distortion [18], referred to as rate-distortion optimization (RDO). The following four sections introduce the estimation, prediction, and signaling methods that are required for implementing our depth signal modeling in an MVD coding framework.

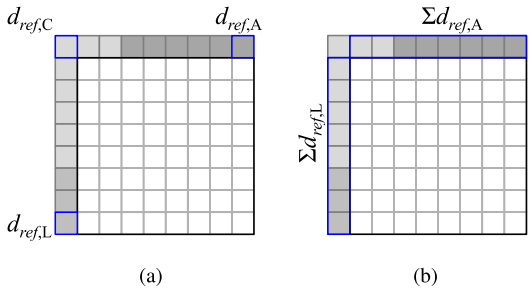

Fig. 4. Prediction of plane geometry. (a) Reference samples for offset and slope. (b) Reference samples for single constant value.

A. Intra-Picture Prediction

According to [19], intra coding of HEVC basically consists of a number of prediction modes together with transform coding of the residual. The prediction modes can be grouped into those for homogeneous regions (Planar and DC mode) and those for directional structures (33 angular modes). Based on the geometric plane and wedgelet models introduced in Section II, we developed the following novel intra-picture prediction methods for depth coding.

1) Intra Prediction of the Plane: The principle of this intra-coding method is predicting a plane model from the information of previously coded blocks in the same picture, i.e., intra prediction. Like for Planar and DC intra modes in HEVC [19], the reference information for prediction consists of the spatially neighboring samples from left and top adjacent blocks. Our plane coding method is purely predictive (encoder estimation and decoder reconstruction processes are identical), such that no additional information has to be signaled in the bitstream.

The plane prediction process is shown in Fig. 4(a). According to (1), the plane model of a depth block is described by the three parameters: offset d_o, horizontal slope m_u and vertical slope m_v. For a depth block of size $N \times N$, these parameters are derived from the three corner reference samples according to Fig. 4(a) as

$$\begin{aligned} m_{u/v} &= \frac{1}{N}(d_{\text{ref},A/L} - d_{\text{ref},C}) \\ d_o &= d_{\text{ref},C} + m_u + m_v. \end{aligned} \tag{6}$$

The predicted signal $d_p(u, v)$ then results from applying (1), describing a perfectly plane surface, which will be referred to as *Plane* mode. In contrast to that the Planar mode in HEVC [19] has no simple geometric description, as it linearly interpolates neighboring samples, which results in a smooth surface with a plane-like characteristic.

2) Intra Prediction of Constant Value: A special case of the general geometric plane model described above is a plane with both slopes defined as zero, i.e., all samples of the depth block have a constant value. This concept is equivalent to the DC intra mode in HEVC [19]. In this case, as shown in Fig. 4(b), the constant value for a depth block of size $N \times N$

is derived as the mean value of left and top reference samples as

$$m_{u/v} \stackrel{\text{def}}{=} 0$$
$$d_o = \tfrac{1}{2N}\left(\sum d_{\text{ref},L} + \sum d_{\text{ref},A}\right). \quad (7)$$

The predicted signal then is $d_p(u,v) = d_o$ for all samples of the block, describing a perfectly flat plane, which will be referred to as DC* mode. In contrast to that the DC mode in HEVC has no simple geometric description, as the predicted signal is subsequently filtered along the left and top edges, which results in a smooth transition between neighboring blocks with a varying signal characteristic and the current block.

3) Estimation of Wedgelet Segmentation: According to [8], the principle of this intra-coding method is estimating the optimum wedgelet segmentation at the encoder by carrying out the minimum distortion search introduced in Section II-B1 and transmitting the segmentation information in the bitstream. In general, the coding performance of this approach benefits from a low distortion using the best wedgelet segmentation for reconstruction at the decoder, but suffers from the additional rate required for transmitting the segmentation information.

To realize an efficient coding mode, two main requirements have to be considered: One is the cost for signaling the segmentation to the decoder and the other the computational complexity of encoder estimation and decoder reconstruction. A solution that meets both requirements is given as wedgelet pattern lookup table: for a given coding block of size $N \times N$, the lookup table is a list of w wedgelet segmentation patterns that result from combining all possible line start and end positions. Duplicate or redundant patterns are excluded from the lookup table, such that $w = 6(N-1)^2 < 6N^2$. Compared with other solutions, these lookup tables have the following advantages.

- The complexity of the estimation process is reduced, as segmentation patterns do not have to be generated every time the minimum distortion search is carried out and redundant segmentations are implicitly excluded from being tested.
- Signaling the segmentation information to the decoder can be realized simple and efficient by transmitting an index in the range $[0, w-1]$ that represents the pattern position in the list, implicitly omitting redundancies.
- The complexity of the reconstruction process is reduced, as the segmentation pattern just needs to be looked up in the list with the transmitted index.

4) Intra Prediction of Wedgelet Segmentation: According to [8], the principle of this intra-coding method is predicting the wedgelet segmentation from the information of previously coded blocks in the same picture. Here, the signal approximation can be improved by additional refinement information, which is estimated at the encoder and signaled in the bitstream. At the decoder, the signal of the block is reconstructed by combining the predicted segmentation with the transmitted refinement information.

The intra prediction of a wedgelet segmentation is shown in Fig. 5. Following the principle of directional intra prediction in HEVC, the wedgelet segmentation of the current block is predicted from the information of the left and top neighboring blocks that are already decoded and reconstructed. In this case, the reference block has to be of type wedgelet, such that the gradient m_{ref} can be calculated from the start and end points of the separation line (S_{ref} and E_{ref}). Given that the line defined by S_{ref} and m_{ref} intersects the current block, S_p and E_p are derived as their intersection points with border samples of the current block. The overall process first considers the top block for prediction and if it is not available or applicable, the left block is used.

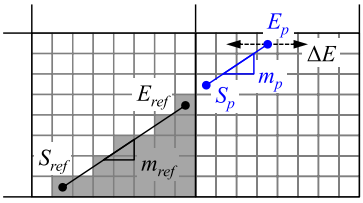

Fig. 5. Intra prediction of wedgelet partition from neighboring wedgelet block.

The coding process for this mode works as follows: At the encoder, the wedgelet segmentation is predicted as described above. For a better approximation, the predicted wedgelet is then refined by testing different values for an offset ΔE to the position of E_p (Fig. 5). Due to complexity and coding efficiency reasons, we restrict the range of ΔE depending on the block size. The ΔE value that leads to the minimum distortion is signaled in the bitstream. For reconstruction at the decoder, the wedgelet segmentation is first predicted as described above and then the separation line end position is adjusted as $E_p + \Delta E$. Regarding coding efficiency, this mode gains from rather low rate and distortion values, given that the edge in a neighboring block continues into the current block.

B. Inter-Component Prediction

According to [7], the principle of this intra-coding method is predicting a non-rectangular bisegmentation of a depth block (Section II-B) from the signal of the co-located block of the associated video picture, i.e., inter-component prediction. Inter-component prediction is based on the assumption that the signals of the video and depth component are correlated, as they only represent different aspects of the same scene. Conceptually, this has some similarities with the layer design of the HEVC extensions for scalable and multiview coding [4]. For depth intra pictures with inter-component prediction, the video slice represents an intra base layer and the depth slice represents an intra enhancement layer. Thus, the information of the video reference picture has to be transmitted before the depth picture.

Based on [7], inter-component prediction for the wedgelet and contour segmentations introduced in Section II-B are considered. For both of them, estimation at the encoder and reconstruction at the decoder are identical and only consist of the inter-component prediction process, such that no additional information has to be signaled in the bitstream. The reference for predicting the segmentation is the luminance signal

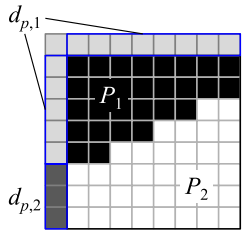

Fig. 6. Partition value prediction from reference samples for wedgelet segmentation.

of the co-located block in the decoded video picture. Regarding coding efficiency, these two modes gain from a very low rate, while the distortion depends on how consistent the object edges in video and depth are.

1) Inter-Component Prediction of Wedgelet Segmentation: The wedgelet segmentation of the current depth block is predicted from the video reference block by carrying out the minimum distortion search for the best matching wedgelet segmentation, as described in Section II-B1.

2) Inter-Component Prediction of Arbitrary Contour Segmentation: The contour segmentation of the current depth block is predicted from the video reference block by carrying out the thresholding segmentation, as described in Section II-B2.

C. Partition Value Prediction for Bisegmentation

According to Section II-B, the bisegmentation model of a depth block consists of the segmentation information and the region value information. Sections III-A3, III-A4, III-B1, and III-B2 introduced four methods for deriving a wedgelet or contour bisegmentation of a depth block, which will be referred to as bisegmentation model (BSM) modes. Given the segmentation information, the process for predicting the constant value for each of the two partitions is identical. Like for the dc* mode, reference information for prediction consists of the left and top neighboring samples.

As shown in Fig. 6, the predicted constant value $d_{p,k}$ for partition P_k with $k \in \{1, 2\}$ is derived as the mean value of the reference samples d_{ref} adjacent to the samples of partition P_k according to the segmentation pattern. In cases where one of the two partitions does not touch the left and top borders at all, $d_{p,k}$ would be undefined and is therefore set to the default value 2^{n-1} for sample bit depth n. The predicted signal $d_p(u, v)$ then results from applying (2) with values $d_{p,k}$.

D. Residual Coding

The residual signal d_r of a depth block is defined as the difference between original and predicted signals with $d_r(u, v) = d(u, v) - d_p(u, v)$. In conventional video coding such as HEVC, residual coding basically consists of applying a 2D separable DCT to the residual signal of a block and a subsequent quantization of the transform coefficients. This set of values is transmitted in the bitstream. At the decoder, the dequantization and inverse 2D DCT are applied. This method will be referred to as transform quantization residual (TQR) coding. In general, the distortion introduced by lossy video coding originates from the quantization error. Consequently, the reconstructed residual signal at the decoder $\hat{d}_r$ differs from the original residual signal d_r, resulting in the reconstructed signal $\hat{d}(u, v) = d_p(u, v) + \hat{d}_r(u, v)$ with distortion D as

$$D(u, v) = \hat{d}(u, v) - d(u, v) = \hat{d}_r(u, v) - d_r(u, v). \quad (8)$$

Given the specific signal and distortion characteristics of depth, we propose an alternative residual coding approach for depth intra that accompanies our concept of prediction signals based on geometric models, as introduced in Sections III-A and III-B. Instead of per-sample differences, the transmitted residual of our method only consists of one constant offset value per partition in the spatial domain. This method will be referred to as constant offset residual (COR) coding. Here, the residual signal consists of two COR values for BSM and one for Plane and DC* mode blocks. For the BSM mode blocks, the transmitted residual is derived with $k \in \{1, 2\}$ as

$$\hat{d}_{r,k} = d_k - d_{p,k} \quad \text{if } (u, v) \in P_k \quad (9)$$

with d_k according to (4). For Plane and DC* modes, the transmitted residual is derived as

$$\hat{d}_r = \frac{1}{N^2} \cdot \sum_{\forall (u,v)} d(u, v) - d_p(u, v) \quad (10)$$

assuming a block size of $N \times N$. For a given geometric model (plane, wedgelet, or contour), these COR values together with the predicted signal lead to the closest approximation of the original signal by the respective model, i.e., minimum distortion D.

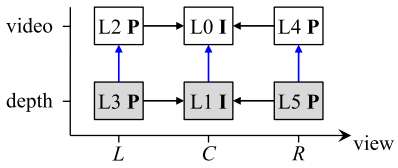

Fig. 7. Layers and dependencies for three-view MVD coding with inter-view (black arrows) and inter-component (blue arrows) prediction.

IV. Application

This section is about the application of our depth intra-coding approach based on geometric primitives, with the first part addressing codec integration and the second mesh extraction.

A. Codec Integration

For integrating the proposed depth intra-coding methods, we select 3D-HEVC as a basic framework for MVD compression. 3D-HEVC builds upon MV-HEVC, which supports efficient compression of MVV with inter-view prediction. Due to the layer design, MV-HEVC can be extended by additional depth layers for supporting MVD in a straightforward way [4]. Fig. 7 shows the layers and dependencies of our basic 3D-HEVC framework with three original camera views.

3D-HEVC contains several additional and modified block-level tools for improving the coding performance. These can be categorized by those specific to video or depth and those for intra or inter coding [4]. The depth coding performance is further improved by disabling deblocking filters and including some special encoder optimizations in 3D-HEVC. The most important one is VSO [15], a distortion metric for multiview depth coding that relates distortions in the depth directly to the overall synthesized view distortion. By changing the distortion metric for depth, VSO has an significant effect on RDO and the related decisions at the encoder.

The reference framework for the integration of our depth intra-coding approach is the above-described 3D-HEVC codec, but without the two depth intra-coding tools referred to as DMM and SDC [4], which methodically overlap with parts of our BSM and COR approaches and were originally included in 3D-HEVC as part of our 3D video standardization proposal [20].

Given this reference, the integration of the methods described in Section III requires the following changes for depth intra blocks. As the first main module, the regular intra-prediction modes are fully replaced by our six geometry model modes, namely, Plane, DC*, and the four BSMs, including the support of inter-component prediction for depth according to Fig. 7. As the second main module, the regular TQR coding is replaced by our COR coding method. The integration of these main modules includes several minor changes and optimizations for depth intra coding, such as signaling and binarization of related syntax elements or disabling reference sample smoothing (to preserve sharp edges for prediction). Furthermore, we use the fast search strategy for minimum distortion wedgelet derivation according to [8], which leads to a substantial complexity reduction, especially in combination with VSO. Regarding the encoder estimation of COR values, VSO has the effect that values derived according to (9) and (10) not necessarily result in the minimum distortion. Thus, with VSO the optimal COR values are derived by a minimum distortion search. Altogether, the resulting codec will be referred to as geometry-based depth intra (GDI) extension.

B. Mesh Extraction

Interoperability between 3D video based on the MVD representation and applications in the 3D computer graphics domain can be achieved by converting the depth signal to a polygon mesh representation. Merkle *et al.* [6], Farin *et al.* [21], and Sarkis *et al.* [22] presented approaches for extracting the surface mesh from depth or disparity images for different applications. Such surface meshes typically consist of triangular faces, described by the position of so-called vertices in 3D scene space and their connectivity. Assuming a set of n vertices that represents the scene surface, the number of triangles T is

$$T = 2 \cdot (n - 1) - k \tag{11}$$

with k being the number of vertices on the convex hull of the surface. Deriving the connectivity for a given set of vertices is also referred to as triangulation and one very popular algorithm is the Delaunay triangulation [23].

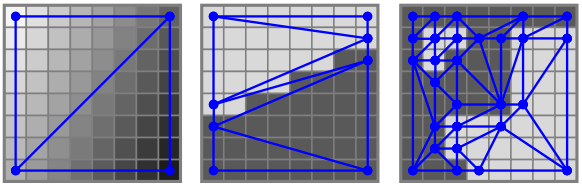

Fig. 8. Mesh extraction for GDI blocks. (a) Plane/DC*. (b) Wedgelet. (c) Contour.

The main step of converting a depth map to a surface mesh thus consists of extracting the set of vertices: basically every sample of a depth picture in MVD data represents a vertex, with position (x, y, z) in 3D scene space determined by the sample position (u, v) and the depth value d via projective geometry. Assuming a depth map with $M \times N$ samples, the number of triangles would be $T = 2 \cdot (M - 1)(N - 1)$, e.g., more than four million triangles for high-definition (HD) resolution. Rendering such an extremely large number of triangles for every frame causes complexity problems in 3D computer graphics applications; however, by removing redundant information, the number of vertices can be reduced without introducing additional distortions. In the case of depth maps, adjacent samples with an identical depth value lead to collinear and coplanar vertices, such that regions of constant depth require a smaller number of vertices: all vertices inside the region are redundant and only those along the boundary are relevant.

In contrast to this general approach, GDI is explicitly designed for supporting mesh extraction with a very small number of triangles. Due to the geometry-based models in combination with COR and disabling smoothing/deblocking filtering of reconstructed samples, Plane and DC* mode blocks can be represented by four and wedgelet blocks by six to eight vertices—independent of the block size. Contour blocks require more vertices depending on the shape of the segmentation, however, delimited by the characteristic that all vertices of each segment are coplanar. Fig. 8 shows mesh extraction for the different GDI types, with vertices as blue dots and connectivity as blue lines.

For evaluating the impact of our depth coding approach on mesh extraction, we apply the following method: in the case of GDI, the set of vertices is compiled from the geometry model parameters within the decoding process as described above. For contour blocks as well as for all other blocks (e.g., inter), the set of vertices is compiled from the reconstructed depth signal of the block. The initial set consists of the four corner vertices. Now, all vertices that represent horizontal and vertical pairs of samples with a different depth value are added to the set, except for the redundant case that both adjacent pairs have the same depth values as the current pair. From the resulting set of vertices for all blocks of a picture, the number of triangles T can be calculated with (11). Here, the convex hull is formed by the k vertices that correspond to samples on the picture border. The described method is integrated in our

TABLE I
AVERAGE RELATIVE BD-RATE VALUES FOR ALL SEQUENCES (TOP) AND FOR GROUPS WITH HIGH AND LOWER DEPTH QUALITY (BOTTOM)

		random access		all-intra	
		VSO on	VSO off	VSO on	VSO off
GDI – 3DR		-3.1%	-4.7%	-7.9%	-8.7%
GDI – 3DH		1.0%	0.8%	1.6%	-0.2%
GDI – 3DH	high	0.5%	-1.4%	0.7%	-4.6%
	lower	1.4%	2.1%	2.1%	2.5%

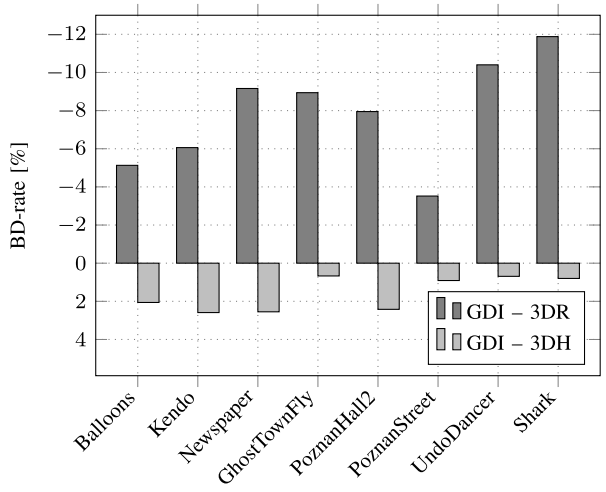

Fig. 9. Average relative BD-rate values per sequence.

3D-HEVC framework as an alternative decoder output, with aligned new processes on all levels of the decoder from blocks to layers.

V. RESULTS

In this section, the experimental results are reported and analyzed. The experiments were conducted following the common test conditions for experiments based on 3D-HEVC developed by the 3D video standardization group [24] as far as possible: this includes eight test data sets, a camera setup with three original and six synthesized views, layer coding order and inter-layer dependencies according to Fig. 7, random access coding structure, video and depth quantization parameter (QP) settings for four rate points, renderer software and configuration, and the evaluation methodology for the resulting objective quality. As usually done for intra-coding tools, we additionally test an all-intra coding structure, where neither temporal prediction nor inter-view prediction is enabled. Regarding the test data sets, two groups of sequences must be distinguished for depth coding, namely, those with high quality depth maps (synthetic sequences *GhostTownFly*, *UndoDancer*, and *Shark*) and those with lower quality depth maps (five other, natural sequences with estimated depth). Due to the large impact on the encoder RDO decisions and thus on the depth coding performance, some additional experiments are conducted with VSO disabled. Unless explicitly specified differently, all results in this section refer to the configuration with all-intra coding structure and VSO enabled.

For implementing the reference framework and our GDI extension according to Section IV-A, we use version 11 of

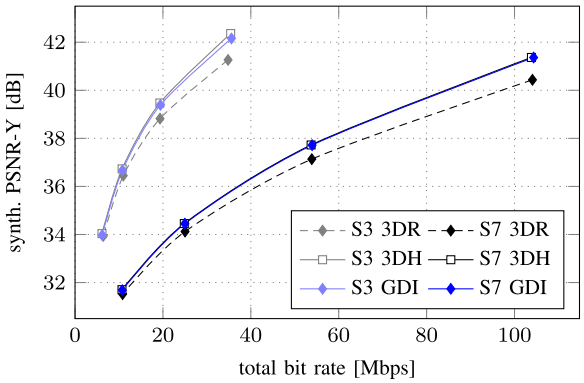

Fig. 10. R-D performances of *Newspaper* (S3) and *UndoDancer* (S7) sequences (total bit rate versus average PSNR-Y of the synthesized views).

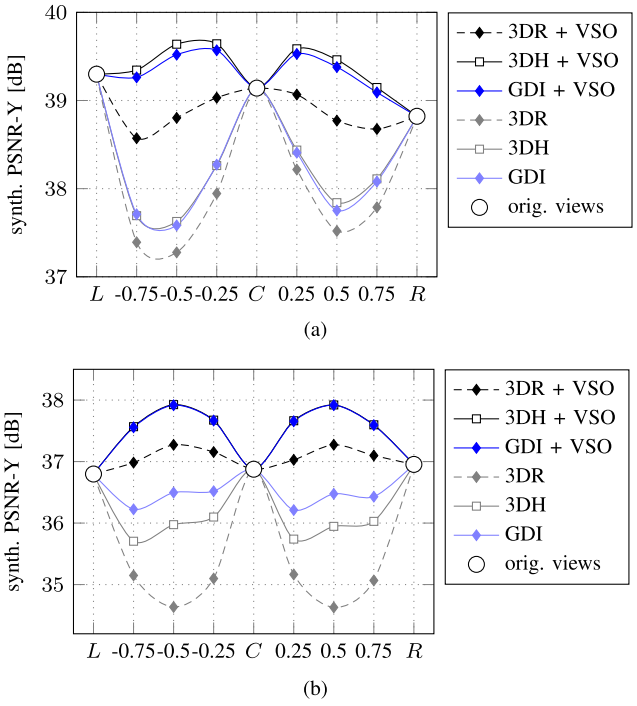

Fig. 11. Distortion of synthesized and original views (view position versus average PSNR-Y) for QP30. (a) *Newspaper* sequence. (b) *UndoDancer* sequence.

the 3D-HEVC reference software (HTM-11.0) [24], which is backward compatible with version 1 of the HEVC standard [3]. Moreover, the decoder of our coding framework is extended by the mesh extraction method introduced in Section IV-B.

To identify and analyze all effects of GDI, two reference methods are selected: one is full 3D-HEVC with all tools (3DH) and the other is the 3D-HEVC implementation reference (3DR), namely, 3D-HEVC without two overlapping depth coding tools (Section IV-A). This second configuration allows one to clearly evaluate and classify the impact of GDI on top of the reference.

A. Objective Results

The results for the overall coding performance are shown in Table I (top) as the average Bjøntegaard delta (BD)

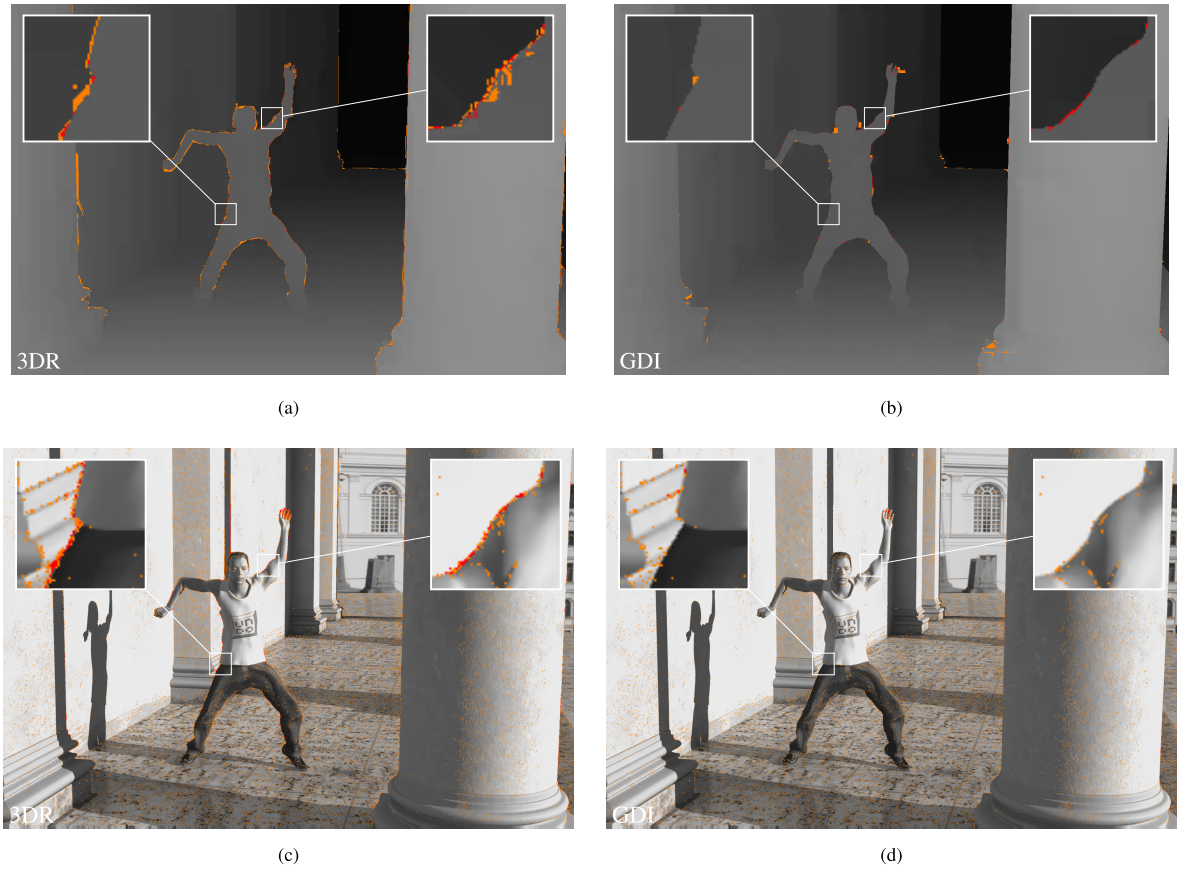

Fig. 12. Subjective examples of *UndoDancer* sequence (frame 53) for QP30, highlighting large distortions (orange: PSNR-Y < 30 dB and red: PSNR-Y < 20 dB): depth view is L (top) and synthesized view is −0.5 (bottom).

rate [25] for all sequences in terms of total bit rate versus luminance peak signal-to-noise ratio (PSNR-Y) of the synthesized views. GDI achieves a significant coding gain for all configurations compared with 3DR—note that a negative BD-rate means a gain. For assessing the magnitude of the gains, it has to be taken into account that only a very small portion of the total rate can be influenced by GDI (about 5% for all-intra configuration), while the video coding parts of reference methods and GDI are identical. It is noted that enabling VSO leads to an average coding gain of about 19% compared with the VSO off case for all configurations, but also to a huge increase in the encoder complexity.

The results further indicate that the overall performance of GDI is slightly worse than that of 3DH. However, as highlighted by the sequence-specific results in Fig. 9, the coding efficiency differs for individual sequences, depending on factors like scene complexity and quality of depth signals. Splitting the results for the two groups introduced above in Table I (bottom) discloses that the performance of GDI relative to 3DH strongly depends on the depth quality: For the group with high-quality depth, noticeable gains are achieved with VSO off and only marginal losses with VSO on. For the other group, COR and inter-component prediction suffer from the worse depth quality, resulting in some losses. Fig. 10 shows the corresponding rate-distortion (R-D) curves for one representative sequence of each group.

Fig. 11 shows the distortion of synthesized and original views relative to the view position. It is important to note that distortion values of synthesized and original views cannot be directly compared, because they are calculated against a different type of reference. These results highlight that both VSO and GDI lead to a significant improvement in the quality of synthesized views. Without VSO, the characteristic of the curves is the same as observed in [6] with decreasing synthesized view quality for increasing distance from the original views. With VSO, the characteristic of the curves changes, resulting in a rather constant synthesized view quality. For GDI, the diagrams confirm the great positive effect of our approach on the objective quality of synthesized views relative to 3DR.

B. Subjective Results

Fig. 12 shows examples of the subjective quality, comparing the amount and type of artifacts of 3DR and GDI by highlighting large distortions. These examples clearly illustrate that especially distortions around edges between foreground and background objects are significantly reduced by GDI. These types of depth distortions have the largest impact on the synthesized view quality, leading to subjectively annoying inconsistencies in the depth perception.

TABLE II
AVERAGE NUMBER OF TRIANGLES PER 32 × 32 BLOCK FOR ALL SEQUENCES

	random access		all-intra	
	VSO on	VSO off	VSO on	VSO off
3DR	216.2	208.4	195.0	173.9
3DH	180.9	179.2	141.9	138.4
GDI	196.8	197.2	23.2	22.0

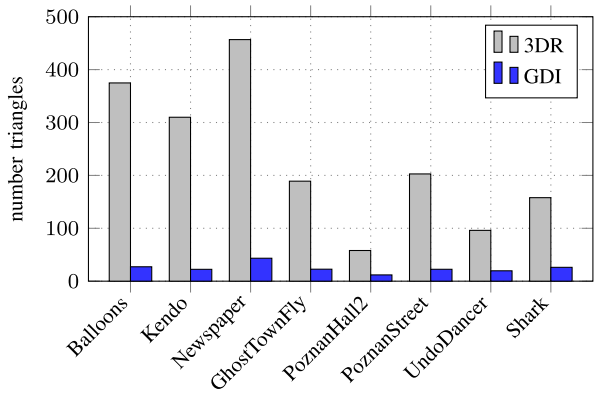

Fig. 13. Average number of triangles per 32 × 32 block per sequence.

C. Triangulation Results

The overall results for mesh extraction according to Section IV-B are shown in Table II for all sequences as the average number of triangles per 32×32 depth block (the largest coding unit in our experiments). For a classification of the absolute values, it should be noted that the number of triangles for a 32 × 32 block, in principle, ranges from 2 to 2048.

The values for all-intra configuration clearly indicate the major advantage of our GDI coding method as the number of triangles required for representing the scene surface by a mesh is significantly smaller than for both reference methods, corresponding to a reduction of about 88% and 84% relative to 3DR and 3DH, respectively. Comparing 3DH and 3DR shows that the additional geometry-based depth coding tools in 3DH only lead to a relatively small reduction. Regarding the random access configuration, the values are overall higher, as the reconstructed signal of inter-coded blocks and pictures has a somewhat different characteristic. Here, motion- and disparity-compensated prediction together with TQR results in rather smoothly changing signals and, consequently, a higher number of triangles for the surface mesh.

The sequence-specific results in Fig. 13 highlight that the number of triangles differs for individual sequences, depending on factors like scene complexity and quality of depth signals. The factor of reduction by GDI ranges from about 5 up to 14 for the different sequences. Another advantage of GDI is the considerably smaller relative deviation between individual sequences, with an average of 24% compared with 49% for 3DR.

D. Statistical Analysis

For better understanding the impact of our depth intra-coding approach, this section presents additional statistical results. First, the complexity in terms of encoder and decoder runtimes is analyzed. Table III shows the relative runtimes of our experiments. Although these runtime values should be considered as rough estimates, GDI obviously leads to an increased computational complexity relative to 3DR, but a smaller encoder complexity than 3DH. The main reason for the increase in decoder complexity is the minimum distortion search of optimum wedgelet segmentations for inter-component prediction, even with the fast search strategy we apply. Note that the level of optimization regarding implementation complexity is of course much higher for the 3D-HEVC reference codec than for our implementation.

Fig. 14 shows the depth intra mode distribution: for 3DR about two-thirds of the samples are covered by angular mode blocks, followed by Planar and DC modes. In contrast to that with GDI, about two-thirds of the samples are covered by Plane mode blocks, followed by DC* and BSM. The reason for the relatively small portion of BSM coded blocks is that this kind of bisegmentation primarily comes into consideration

TABLE III
AVERAGE RUNTIMES OF GDI RELATIVE TO THE REFERENCE METHODS FOR ALL SEQUENCES AND RATE POINTS

		random access		all-intra	
		VSO on	VSO off	VSO on	VSO off
GDI – 3DR	enc.	111%	102%	192%	134%
	dec.	105%	105%	131%	144%
GDI – 3DH	enc.	98%	99%	75%	86%
	dec.	104%	102%	138%	147%

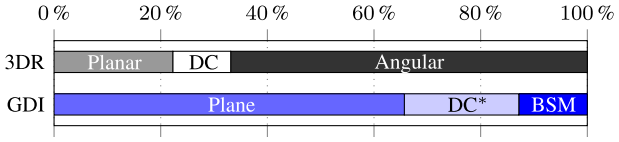

Fig. 14. Average portion of samples coded with specified mode type as part of all intra-coded depth samples for all sequences and rate points.

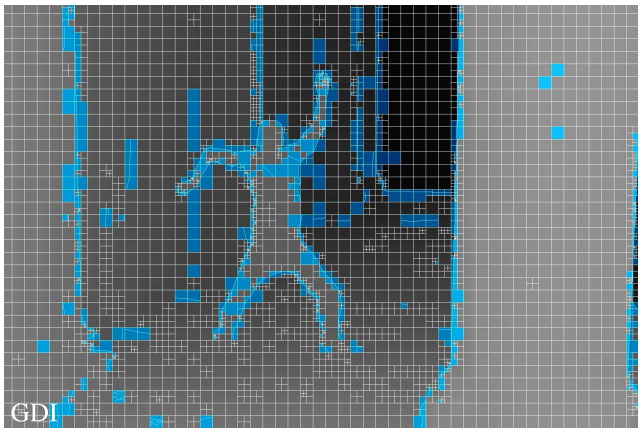

Fig. 15. Example of *UndoDancer* sequence (depth view *L* and frame 53) for QP30, highlighting BSM coded blocks (blue).

TABLE IV
AVERAGE PORTION OF INTRA-CODED DEPTH SAMPLES WITH SPECIFIED RESIDUAL FOR ALL SEQUENCES AND RATE POINTS

	random access		all-intra	
	VSO on	VSO off	VSO on	VSO off
3DR (TQR)	17.7%	19.8%	17.7%	18.3%
GDI (COR)	19.8%	15.4%	20.2%	23.5%

TABLE V
AVERAGE BD-RATE VALUES OF GDI WITH SPECIFIED RESIDUAL RELATIVE TO 3DR FOR ALL SEQUENCES

	random access		all-intra	
	VSO on	VSO off	VSO on	VSO off
GDI (COR)	-3.1%	-4.7%	-7.9%	-8.6%
GDI (TQR)	-0.2%	-3.4%	0.2%	-5.6%

for certain areas of a depth picture, i.e., along the edges between objects, as highlighted by the example in Fig. 15. Within the BSM coded blocks, about 85% are of type wedgelet and about 30%–35% are inter-component predicted for all configurations.

Regarding the residual coding, Table IV shows the portion of samples covered by depth intra blocks with a non-zero residual signal, i.e., with TQR and COR values transmitted in the bitstream, respectively. The amount of blocks with an additional residual neither increases nor decreases significantly for GDI. A detailed analysis including the additional results for GDI with TQR instead of COR in Table V shows that COR is very efficient for depth coding in terms of RDO, especially in combination with BSM and VSO.

VI. CONCLUSION

We presented a depth intra-coding approach for 3D video based on geometric primitives. Our method is optimized for the specific characteristics of the depth signals in MVD data. Based on the 3D-HEVC extension, our approach fully substitutes the intra-prediction modes as well as the residual coding method of HEVC for depth intra pictures and intra blocks.

The main objective for the design of these two functional units was increasing the depth coding efficiency and at the same time improving the mesh extraction capabilities for bridging the gap between 3D video and 3D computer graphics applications. We achieve this by combining prediction modes based on geometric primitives with COR residual coding, ensuring that all reconstructed sample values in a segment or block are either constant or coplanar.

Regarding the coding efficiency, GDI has the advantage of being optimized for the specific depth signal characteristics of sharp object edges and larger areas of nearly constant or slowly varying sample values. This means that GDI achieves a closer approximation of such depth signals at a lower bit rate than HEVC intra coding, with the BSM modes well adapted to sharp object edges and Plane/DC* modes well adapted to constant and slowly varying areas with a planar characteristic. The results show that GDI achieves about the same coding performance as 3D-HEVC for high quality depth maps, with some losses for lower-quality depths. Relative to 3D-HEVC without overlapping tools, GDI leads to significant coding gains in synthesized views.

Regarding the mesh extraction, GDI has the advantage of being optimized for representing the scene geometry of a block by a very limited number of vertices and triangles. This means that the vertices of the surface mesh can be directly derived within the decoding process from the geometry-based signal model parameters and constant offset residual values. The results show that GDI leads to surface meshes with significantly less triangles than generated from an ordinary decoder output of the reference codecs.

In summary, we showed that GDI is superior to HEVC intra coding for depth, resulting in a considerably higher quality of synthesized views for 3D video applications and, at the same time, in a significantly less complex surface mesh for 3D computer graphics applications. One of the remaining issues for further research on this topic is the extension of our approach to inter-picture coding. Here, additional challenges are foreseen with regard to motion- and disparity-compensated prediction as well as the temporal- and inter-view consistency of surface meshes.

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] P. Merkle, A. Smolic, K. Müller, and T. Wiegand, "Efficient prediction structures for multiview video coding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 17, no. 11, pp. 1461–1473, Nov. 2007.

[3] *High Efficiency Video Coding*, document Rec. ITU-T H.265, International Telecommunications Union Standard, April 2013.

[4] G. J. Sullivan, J. M. Boyce, Y. Chen, J.-R. Ohm, C. A. Segall, and A. Vetro, "Standardized extensions of High Efficiency Video Coding (HEVC)," *IEEE J. Sel. Topics Signal Process.*, vol. 7, no. 6, pp. 1001–1016, Dec. 2013.

[5] P. Merkle, A. Smolic, K. Müller, and T. Wiegand, "Multi-view video plus depth representation and coding," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, vol. 1. Sep. 2007, pp. I-201–I-204.

[6] P. Merkle *et al.*, "The effects of multiview depth video compression on multiview rendering," *Signal Process., Image Commun.*, vol. 24, nos. 1–2, pp. 73–88, Jan. 2009.

[7] P. Merkle, C. Bartnik, K. Müller, D. Marpe, and T. Wiegand, "3D video: Depth coding based on inter-component prediction of block partitions," in *Proc. Picture Coding Symp. (PCS)*, May 2012, pp. 149–152.

[8] P. Merkle, K. Müller, and T. Wiegand, "Coding of depth signals for 3D video using wedgelet block segmentation with residual adaptation," in *Proc. IEEE Int. Conf. Multimedia Expo (ICME)*, Jul. 2013, pp. 1–6.

[9] Y. Morvan, P. H. N. de With, and D. Farin, "Platelet-based coding of depth maps for the transmission of multiview images," *Proc. SPIE*, vol. 6055, pp. 60550K-1–60550K-12, Jan. 2006.

[10] S. Liu, P. Lai, D. Tian, and C. W. Chen, "New depth coding techniques with utilization of corresponding video," *IEEE Trans. Broadcast.*, vol. 57, no. 2, pp. 551–561, Jun. 2011.

[11] H. Oh and Y.-S. Ho, "H.264-based depth map sequence coding using motion information of corresponding texture video," in *Advances in Image and Video Technology* (Lecture Notes in Computer Science), vol. 4319, L.-W. Chang and W.-N. Lie, Eds. Berlin, Germany: Springer-Verlag, 2006, pp. 898–907.

[12] M. Winken, H. Schwarz, and T. Wiegand, "Motion vector inheritance for high efficiency 3D video plus depth coding," in *Proc. Picture Coding Symp. (PCS)*, May 2012, pp. 53–56.

[13] E. G. Mora, J. Jung, M. Cagnazzo, and B. Pesquet-Popescu, "Initialization, limitation, and predictive coding of the depth and texture quadtree in 3D-HEVC," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 24, no. 9, pp. 1554–1565, Sep. 2014.

[14] F. Jäger and K. Naser, "Low complexity transform coding for depth maps in 3D video," in *Proc. Vis. Commun. Image Process. (VCIP)*, Nov. 2013, pp. 1–6.

[15] G. Tech, H. Schwarz, K. Müller, and T. Wiegand, "3D video coding using the synthesized view distortion change," in *Proc. Picture Coding Symp. (PCS)*, May 2012, pp. 25–28.

[16] S. Weisberg, *Applied Linear Regression* (Probability and Statistics). New York, NY, USA: Wiley, 2005.

[17] R. C. Gonzalez and R. E. Woods, *Digital Image Processing*, 3rd ed. Hoboken, NJ, USA: Pearson, 2008.

[18] G. J. Sullivan and T. Wiegand, "Rate-distortion optimization for video compression," *IEEE Signal Process. Mag.*, vol. 15, no. 6, pp. 74–90, Nov. 1998.

[19] J. Lainema, F. Bossen, W.-J. Han, J. Min, and K. Ugur, "Intra coding of the HEVC standard," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 22, no. 12, pp. 1792–1801, Dec. 2012.

[20] K. Müller *et al.*, "3D high-efficiency video coding for multi-view video and depth data," *IEEE Trans. Image Process.*, vol. 22, no. 9, pp. 3366–3378, Sep. 2013.

[21] D. Farin, R. Peerlings, and P. H. N. de With, "Depth-image representation employing meshes for intermediate-view rendering and coding," in *Proc. 3DTV Conf.*, May 2007, pp. 1–4.

[22] M. Sarkis, W. Zia, and K. Diepold, "Fast depth map compression and meshing with compressed tritree," in *Computer Vision*. Berlin, Germany: Springer-Verlag, 2010, pp. 44–55.

[23] M. de Berg, O. Cheong, M. van Kreveld, and M. Overmars, *Computational Geometry: Algorithms and Applications*. Berlin, Germany: Springer-Verlag, 2008.

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

[25] G. Bjontegard, *Calculation of Average PSNR Differences Between RD Curves*, document Rec. VCEG-M33, ITU-T Q.6/SG16, Apr. 2001.

Philipp Merkle (S'06–M'12) received the Dipl.-Ing. degree in electrical engineering from Technical University of Berlin, Berlin, Germany, in 2006.

He joined Fraunhofer Institute for Telecommunications—Heinrich Hertz Institute, Berlin, in 2003 and has been a Research Associate since 2006. Since 2005, he has been an active participant in standardization activities of ISO/IEC Moving Picture Experts Group (MPEG) and ITU-T Video Coding Experts Group (VCEG) with successful contributions to the Multiview Video Coding (MVC) amendment of the H.264/Advanced Video Coding (AVC) standard, and the multiview and 3D video extensions of the H.265/High Efficiency Video Coding (HEVC) standard. His research interests include 3D video and television, MVC, representation and compression of multiview video plus depth scenes, free viewpoint video, 2D and 3D video-based rendering, and 3D scene reconstruction.

Mr. Merkle received the Rudolf-Urtel Prize of the German Society for Technology in Television and Cinema for his journal paper on MVC in IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY in 2006. He received the EURASIP Best Paper Award for his journal paper on multiview depth compression in *Signal Processing: Image Communication* (Elsevier) in 2014.

Karsten Müller (M'98–SM'07) received the Dipl.-Ing. and Dr.-Ing. degrees in electrical engineering from Technical University of Berlin, Berlin, Germany, in 1997 and 2006, respectively.

He has been with Fraunhofer Institute for Telecommunications—Heinrich Hertz Institute, Berlin, since 1997, where he is currently the Head of the Image and Video Understanding Group with the Department for Video Coding and Analytics. He has been involved in international standardization activities, successfully contributing to the ISO/IEC Moving Picture Experts Group for work items on visual media content description, multitexture graphics representation, multiview, and 3D video coding. His research interests include semantic visual content description, coding and reconstruction of 3D scenes in free-viewpoint video scenarios, multiview applications, and combined 2D/3D similarity analysis.

Dr. Müller has served as the Chair and an Editor of the IEEE conferences, and is an Associated Editor of IEEE TRANSACTIONS ON IMAGE PROCESSING. He was the Co-Chair of the Ad-hoc Group on 3D Video Coding from 2003 to 2012.

Detlev Marpe (M'00–SM'08–F'15) received the Dipl.-Math. (Hons.) degree from Technical University of Berlin, Berlin, Germany, in 1990 and the Dr.-Ing. degree from University of Rostock, Rostock, Germany, in 2004.

He joined Fraunhofer Institute for Telecommunications—Heinrich Hertz Institute, Berlin, in 1999, where he is currently the Head of the Video Coding & Analytics Department and Head of the Image and Video Coding Research Group. He was a Major Technical Contributor for the development of both the H.264/Advanced Video Coding (AVC) standard and the H.265/High Efficiency Video Coding (HEVC) standard, including several generations of major enhancement extensions. In addition to the CABAC contributions for both standards, he also contributed to the Fidelity Range Extensions (which include the High Profile that received the Emmy Award in 2008) and the Scalable Video Coding Extensions of H.264/AVC. During the recent development of its successor H.265/HEVC, he successfully contributed to the first model of the corresponding standardization project and further refinements. In addition, he also made successful proposals to the standardization of its range extensions and 3D extensions. His academic work includes more than 200 publications in the areas of image and video coding. He holds over 250 internationally issued patents and numerous patent applications in this field. His current research interests include still image and video coding, signal processing for communications, and computer vision and information theory.

Dr. Marpe is Fellow of the IEEE and member of the Informationstechnische Gesellschaft of the Verband der Elektrotechnik Elektronik Informationstechnik e.V. He was a co-recipient of two Technical Emmy Awards as a Key Contributor and Co-Editor of the H.264/MPEG-4 AVC standard in 2008 and 2009, respectively. He received the IEEE Best Paper Award at the 2013 IEEE International Conference on Consumer Electronics-—Berlin in 2013 and the SMPTE Journal Certificate of Merit in 2014. He was nominated for the German Future Prize in 2012, and received the Karl Heinz Beckurts Award in 2011, the best paper award of the IEEE Circuits and Systems Society in 2009, the Joseph-von-Fraunhofer Prize in 2004, and the best paper award of the German Society for Information Technology in 2004. As the Co-Founder of the Berlin-based daviko GmbH, he was the winner of the Prime Prize of the Multimedia Start-Up Competition of the German Federal Ministry of Economics and Technology in 2001. Since 2014, he has been an Associate Editor of IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY.

Thomas Wiegand (M'05–SM'08–F'11) is a professor in the department of Electrical Engineering and Computer Science at the Technical University of Berlin and is jointly heading the Fraunhofer Heinrich Hertz Institute, Berlin, Germany. He received the Dipl.-Ing. degree in Electrical Engineering from the Technical University of Hamburg-Harburg, Germany, in 1995 and the Dr.-Ing. degree from the University of Erlangen-Nuremberg, Germany, in 2000.

As a student, he was a Visiting Researcher at Kobe University, Japan, the University of California at Santa Barbara and Stanford University, USA, where he also returned as a visiting professor. He was a consultant to Skyfire, Inc., Mountain View, CA, and is currently a consultant to Vidyo, Inc., Hackensack, NJ, USA.

Since 1995, he has been an active participant in standardization for multimedia with many successful submissions to ITU-T and ISO/IEC. In 2000, he was appointed as the Associated Rapporteur of ITU-T VCEG and from 2005-2009, he was Co-Chair of ISO/IEC MPEG Video.

The projects that he co-chaired for the development of the H.264/MPEG-AVC standard have been recognized by an ATAS Primetime Emmy Engineering Award and a pair of NATAS Technology & Engineering Emmy Awards. For his research in video coding and transmission, he received numerous awards including the Vodafone Innovations Award, the EURASIP Group Technical Achievement Award, the Eduard Rhein Technology Award, the Karl Heinz Beckurts Award, the IEEE Masaru Ibuka Technical Field Award, and the IMTC Leadership Award. He received multiple best paper awards for his publications. Thomson Reuters named him in their list of "The World's Most Influential Scientific Minds 2014" as one of the most cited researchers in his field. In 2015, he received the ITU150 award.