

## **HEVC-Compatible Extensions for Advanced Coding of 3D and Multiview Video**

Vetro, A.; Chen, Y.; Mueller, K.

TR2015-046 April 2015

### **Abstract**

This article provides an overview of standardized extensions of HEVC for the advanced coding of 3D and multiview video. In those extensions, new coding tools that better exploit the inter-view redundancy of the multiview texture videos have been developed. Additionally, dedicated tools for the improved coding of depth have been extensively studied and incorporated into the standard. In this paper, the performance of these extensions is assessed, and experimental results demonstrate notable gains in coding efficiency.

*Data Compression Conference (DCC)*

This work may not be copied or reproduced in whole or in part for any commercial purpose. Permission to copy in whole or in part without payment of fee is granted for nonprofit educational and research purposes provided that all such whole or partial copies include the following: a notice that such copying is by permission of Mitsubishi Electric Research Laboratories, Inc.; an acknowledgment of the authors and individual contributions to the work; and all applicable portions of the copyright notice. Copying, reproduction, or republishing for any other purpose shall require a license with payment of fee to Mitsubishi Electric Research Laboratories, Inc. All rights reserved.



# HEVC-Compatible Extensions for Advanced Coding of 3D and Multiview Video

Anthony Vetro<sup>\*</sup>, Ying Chen<sup>+</sup> and Karsten Mueller<sup>#</sup>

<sup>\*</sup>*Mitsubishi Electric Research Labs  
201 Broadway, 8th Floor  
Cambridge, MA 02139, USA  
avetro@merl.com*

<sup>+</sup>*Qualcomm, Inc  
5775 Morehouse Drive  
San Diego, CA 92121, USA  
cheny@qti.qualcomm.com*

<sup>#</sup>*Fraunhofer HHI  
Einsteinufer 37  
10587 Berlin, Germany  
karsten.mueller@hhi.fraunhofer.de*

**Abstract:** This article provides an overview of standardized extensions of HEVC for the advanced coding of 3D and multiview video. In those extensions, new coding tools that better exploit the inter-view redundancy of the multiview texture videos have been developed. Additionally, dedicated tools for the improved coding of depth have been extensively studied and incorporated into the standard. In this paper, the performance of these extensions is assessed, and experimental results demonstrate notable gains in coding efficiency.

## 1. Introduction

3D displays enable a visual scene to be viewed with enhanced perception of depth. One prevalent type of 3D display is a stereoscopic display, which requires special glasses to display the left and right views to each eye of the viewer. Another important category of 3D displays are auto-stereoscopic displays, which generally emit view-dependent pixels and do not require glasses for viewing. Such displays often employ depth-based image rendering techniques, where it is desirable to use high-quality depth maps as part of the coded representation.

The multiview video plus depth format has emerged as the primary 3D format to support advanced stereoscopic processing, such as adjusting the level of depth perception in conventional stereo display, which is especially required when heterogeneous devices are involved, and to drive the rendering requirements in autostereoscopic displays that require a large number of views for different perspectives. Recognizing the need to efficiently store and transmit this format, HEVC extensions for the efficient compression of multiview video with depth have been developed and standardized by JCT-3V, a joint team on 3D Video Coding Extension Development from ISO/IEC and ITU-T.

The rest of this paper is organized as follows. In Section 2, a baseline architecture for the coding of multiview videos and depth in the context of HEVC is described. Then, advanced coding tools for texture and depth are summarized in Sections 3 and 4, respectively. An analysis of coding performance is provided in Section 5 and concluding remarks are given in Section 6.

## **2. Baseline Architecture: MV-HEVC**

A multiview extension of HEVC, referred to as MV-HEVC, was finalized in July 2014 [1]. This specification follows the same design principles of the prior MVC extension in the AVC framework [2][3]. As with the MVC design, this scheme enables inter-view prediction so that pictures from other views at the same time instant can be used for prediction of a picture in the current view. As such, a block-based disparity shift between the reference view and the current views is determined and is used to perform a disparity-compensated prediction. This is very similar to the motion-compensated prediction used in conventional video coding, but is based on pictures with different viewpoints rather than pictures at different time instances. A key feature of this design is that it provides compatibility with single-view coding of HEVC [4].

The approach is realized in a straightforward manner by (i) extending the high-level syntax so that the appropriate signaling of view identifiers and their references is supported, (ii) defining a process by which decoded pictures of other views can be used for prediction of a current picture in another view, and (iii) reusing the motion vector prediction and coding scheme for disparity vectors, therefore no hardware changes in the block-level design are needed. Such a simple inter-view prediction in MV-HEVC is called inter-view sample prediction in this paper.

In the MV-HEVC framework, support for depth maps is enabled through auxiliary picture high-level syntax. The auxiliary picture decoding process would be the same as that for video or multiview video, and the required decoding capabilities could be specified as part of the bit stream. In this way, applications that would like to make use of depth in the MV-HEVC framework could still do so, while a profile that requires the capability to decode both multiview and depth could be specified at a later stage.

The approach of adding support for depth through auxiliary pictures is quite different from that taken in the 3D extension of HEVC, referred to as 3D-HEVC, which adds normative support for decoding of depth pictures as part a dedicated profile, along with tools for the decoding of multiview video plus depth [5]. The 3D-HEVC extension is scheduled to be finalized in mid-2015. A preliminary description of 3D-HEVC coding tools and its performance was published in [6]. In the following sections, updated information based on the nearly completed standard is provided.

## **3. Advanced Texture Coding Tools in 3D-HEVC**

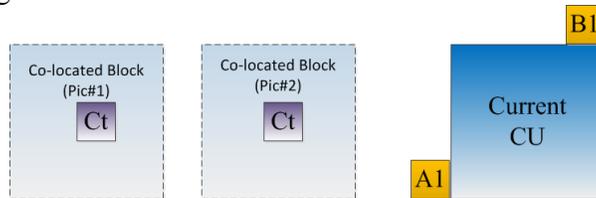
To achieve higher coding efficiency, advanced coding tools that better exploit the inter-view redundancy have been studied and evaluated. In contrast to the MV-HEVC standard discussed in the previous section, block-level changes to the syntax and decoding process are considered to maximize the possible coding gain.

### **3.1. Neighboring block-based disparity vector derivation (NBDV)**

Due to the disparity-compensated prediction, disparity motion vectors are populated in a motion field together with normal motion vectors. As the name implies, the concept of NBDV derives a disparity vector for a current block using an available disparity motion vector from spatial and temporal neighboring blocks [7]. The spatial neighbors are the same as those used in HEVC for motion prediction, while the temporally neighboring

blocks include those covering the center pixel of the co-located block in two reference pictures [8]. The neighboring blocks are checked in a pre-specified order and once a disparity motion vector in a neighboring block is identified from the motion field, the NBDV process terminates and the derived disparity vector is set equal to the identified disparity motion vector. The main benefit of this technique is that disparity vectors to be used for inter-view prediction can be directly derived without additional bits, and independent of an associated depth picture. Almost all coding tools (as to be described below) can be efficiently built on top of the NBDV technique without further accessing depth information.

Various simplifications have been introduced for NBDV. For instance, 3D-HEVC tools are applied at the Coding Unit (CU) level and a single derived disparity vector is used for each Prediction Unit (PU) of the CU [8]. Additionally, as further illustrated in Figure 1, temporal neighbor blocks (denoted as “Ct”) in two reference pictures are checked followed by the left and above neighboring blocks of the current CU. The temporal blocks are identified in two pictures. The first one is the picture used in temporal motion vector prediction in HEVC and the second is the random access picture, or if unavailable, the picture with the lowest temporal level in the reference picture lists. It is noted that the second picture is chosen so that it is more probable to find disparity motion vectors in the temporal neighboring blocks.



**Figure 1:** Neighboring blocks in NBDV.

### 3.2. Inter-view motion prediction

The motion information between views exhibit a high degree of correlation, and inferring it from one view to another view leads to notable gains in coding efficiency since good predictions generally reduce the required bit rate to send such information. To achieve this, the disparity, such as that derived by the NBDV process, is used to establish a correspondence between the blocks in each view. An illustration of motion prediction between views is shown in Figure 2, where the motion vector of view 1 is inferred from the motion vector of view 0 from corresponding blocks at time 1 based on the disparity between those blocks as derived by the NBDV process.

In 3D-HEVC, the above inter-view motion prediction is realized in a manner that is compatible to the motion prediction scheme in HEVC, and extends far beyond that. In HEVC, a merge candidate list can be formed, where each candidate could contain both reference index and motion vector of both prediction directions corresponding to the reference picture lists, List 0 and List 1 [1]. At the decoder, an index is used to specify which candidate is used so that the motion information of the current PU can be derived without further information transmitted. The main idea of inter-view motion prediction in 3D-HEVC is to derive additional candidates and put them in the merge candidate list (with limited pruning) together with the existing merge candidates as derived in HEVC.

Based on the disparity vector ( $DV$ ), the following three candidates can be generated and inserted into the merge candidate list, which can contain up to six entries in 3D-HEVC [9].

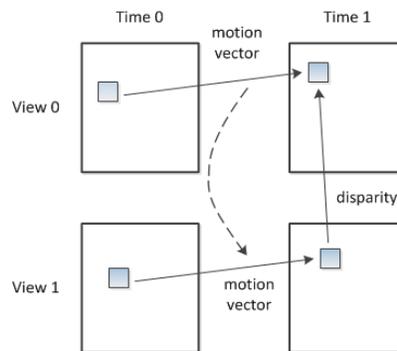
(1) Inter-view motion candidate: The concept is as illustrated in Figure 2. Given a disparity vector (e.g.,  $DV$  or any other vector) a corresponding block of current PU in a reference view of the same access unit is identified. If the corresponding block is not intra-coded and not inter-view predicted and its reference picture has a picture order count (POC) value equal to that of one entry in the list of current PU, its motion information is derived to be the inter-view motion candidate.

(2) Disparity motion vector candidate: The motion vector for this candidate is generated by converting an input disparity vector into a disparity motion vector, and the reference index is set to the reference index of the inter-view reference picture associated with the disparity vector.

(3) Shifted candidate: This candidate is generated with the same process as defined above with a slightly shifted disparity vector so that to compensate the case that the disparity vector is not accurate enough. Firstly, an additional inter-view motion candidate is generated with an input disparity vector equal to the  $DV$  with a shifting vector having a horizontal component and vertical component equal to half width and half height of the current PU, respectively. If the additional inter-view motion candidate is available, the shift candidate is derived; otherwise, the shift candidate is set to a disparity motion vector candidate with an input disparity vector equal to  $DV$  with the horizontal component shifted by 4.

It is noted that the first and the second additional candidates are the inter-view motion candidate and disparity motion vector candidate derived with input disparity vector being  $DV$ .

In 3D-HEVC, the coding efficiency of the first additional candidate has been further improved by the sub-PU based inter-view motion prediction, wherein smaller blocks (e.g.,  $8 \times 8$ ) of each PU can derive its own motion information and thus a more detailed motion field of a PU can be represented without additional signaling [10]. Each sub-PU has its individual motion information derived in the same way as the inter-view motion candidate describe above and motion compensation may be performed separately for it.



**Figure 2:** Illustration of motion prediction between views.

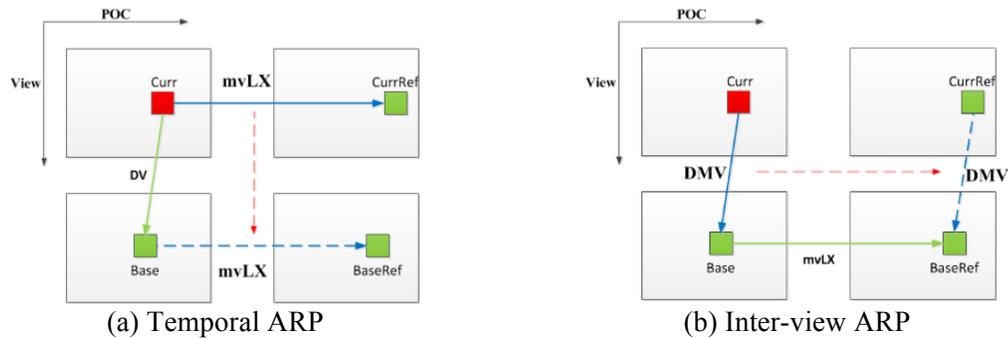
### 3.3. Inter-view residual prediction

Also known as advanced residual prediction (ARP) and only supported in 3D-HEVC, this prediction mode increases the accuracy of the residual predictor by calculating the residual block on-the-fly. In ARP, the motion vector is aligned for the current block and the reference block, thus the similarity between the residual predictor and the residual signal the current block is much higher and the remaining energy after ARP is significantly reduced [11].

As illustrated in Fig. 2, there exist two types of ARP designs: temporal ARP and inter-view ARP. In temporal ARP, the residual predictor is calculated as a difference between the two blocks in the reference view, the base block (Base) in the inter-view reference picture and its reference block (BaseRef), wherein the former is identified purely by the  $DV$ , while the latter is identified jointly by the  $DV$  and the current motion vector. This way, the motion information used to derive the residue at the reference view is aligned with that of the current block [11].

With inter-view ARP, an inter-view residual is calculated as a difference between the two blocks in a different time instant, the temporal reference block (CurrRef) in a temporal reference picture and its reference block in the reference view (BaseRef). In case of inter-view ARP, the disparity motion vector (DMV) is signaled and used to identify an inter-view reference block in the reference view (Base), within which a motion vector (mvLX) may be derived and used to further identify the two blocks in the different time instance: CurrRef and BaseRef.

In ARP, since the residual block needs to be calculated on-the-fly, simply performing an additional motion compensation introduces high computational complexity and more memory access. To resolve these issues, a low-complexity design of the ARP has been adopted into 3D-HEVC with optimizations in various aspects [12]: for example, bi-linear interpolation is used for the motion compensation of the current block as well as generation of the residual block.



**Figure 2:** Illustration of temporal ARP and inter-view ARP.

### 3.4. Illumination compensation

Inter-view prediction may fail when cameras capturing the same scene are not calibrated in color transfer or by lighting effects. Therefore, illumination compensation was introduced to improve the coding efficiency for blocks predicted from inter-view

reference pictures [13]. This mode only applies to blocks that are predicted by an inter-view reference picture.

For the current PU, its neighboring samples in the top neighboring row and left neighboring column, together with the corresponding neighboring samples of the reference block are the input parameters for a linear model, to derive a scaling factor  $a$  and an offset  $b$  by a least squares solution.

After disparity motion compensation from an inter-view reference for the current PU, the gain/offset model is applied to each value predicted for the PU, being scaled by  $a$ , and then the offset  $b$  is added.

### **3.5. View synthesis prediction (VSP)**

This method uses the depth information to warp texture data from a reference view to the current view such that a predictor for the current view can be generated. Although depth is often available with pixel-level precision, a block-based VSP scheme has been specified in 3D-HEVC in order to align this type of prediction with existing modules for motion compensation. To perform VSP, the depth information of the current block is used to determine the corresponding pixels in the inter-view reference picture. Since texture is typically coded prior to depth, the depth of the current block can be estimated using the NBDV process described earlier. In 3D-HEVC, the VSP design is realized by extensions of the merge mode, whereby the disparity and inter-view reference picture corresponding to the VSP operation is added to the merge candidate list [14].

## **4. Advanced Depth Coding Tools in 3D-HEVC**

While the video or texture information in a multi-view video plus depth representation is directly displayed to the user after decoding, depth information is used to generate additional intermediate views at new viewpoints. This special usage of depth maps, combined with the fact that depth maps have characteristic that are different than video data (e.g., large homogeneous areas and sharp edges), requires new coding methods to be considered. An overview of the depth-based coding tools that have been adopted in 3D-HEVC is provided in this section.

It is noted that since a reconstructed depth map is not directly viewed, but rather used for synthesizing additional texture views, the rate-distortion optimization for encoding depth maps is modified from traditional formulations. Instead of directly optimizing according to the depth distortion, a new view synthesis optimization (VSO) is used that considers the distortion of intermediate video views [15]. As such, depth maps are encoded in a manner that provides the best quality for intermediate views at a given minimum rate.

### **4.1. Depth motion and disparity prediction**

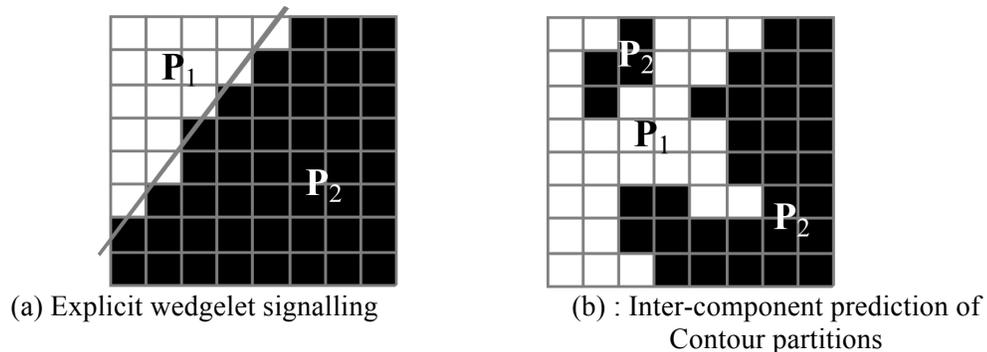
Although depth maps exhibit different characteristics for describing the same scene content, they also share some common features, like similar motion vectors in temporal direction of each view, as well as similar disparity vectors across views. In general, a video and its associated depth maps have identical motion and disparity vectors, as the respective pixels describe the same scene object. However, in video coding, estimated motion and disparity vectors are sought, which minimize the overall rate for coded

motion/disparity data as well as residual information. Accordingly, the motion/disparity vectors used for coding the video and depth component can thus deviate from the true scene motion as well as also differ between both components.

Therefore, the general concept of motion and disparity compensated prediction (MCP and DCP respectively) are also applied for depth maps, however with some changes for specific candidates. For similar motion in video and depth, motion parameter inheritance (MPI) is used, which introduces a texture candidate for the merge mode in depth coding that allows the inheritance of motion parameters from the texture signal. Similar to motion prediction in texture coding, depth motion prediction is achieved by adding new candidates into the merge candidate list. The additional candidates include an inter-view merge candidate, and a sub-PU motion parameter inheritance (MPI) candidate. The inter-view merge candidate is similar to that derived for texture coding, with a disparity vector being converted from a default depth value. The MPI candidate, however, derives the motion information of the depth from the co-located block from the already coded texture picture of the same view. This derivation process can also be applied at the sub-PU level, wherein each sub-PU of the depth PU identifies its co-located region from the coded texture picture to grab the motion vectors.

## 4.2. Partition-based depth intra coding

To better represent the particular characteristics of depth maps, each depth block may be geometrically partitioned and more efficiently represented. In 3D-HEVC, these non-rectangular partitions are collectively referred to as depth modeling modes (DMM) [16]. Here, two methods for separating a depth block into two different non-rectangular partitions are applied, each using its own type of partitioning pattern: First, a wedgelet or straight line as shown in Figure 3(a) is used together with direct wedgelet signaling. Here, the best-matching wedgelet partition is sought, which yields the minimum distortion between the original signal and the wedgelet approximation. This partition information is not predicted from neighboring blocks, but rather taken from a list of possible patterns. Second, a contour partition as shown in Figure 3(b) is used as partitioning pattern. The contour partition is predicted from the co-located texture block and can have any irregular partition.



**Figure 3:** Different partitioning patterns of depth blocks: (a) straight line and (b) derivation of an arbitrary contour from the co-located texture block.

Besides the separation information, also the values of each partition  $P_1$  and  $P_2$  (see Figure 3) are coded through constant partition value coding, where the mean value of each partition is predicted from adjacent samples of the previously coded left and top neighboring block.

### **4.3. Segment-wise DC Coding (SDC)**

This coding mode provides an alternative residual coding method. Here, the transform and quantization process are skipped so that depth prediction residual data is directly coded. This residual data can contain one constant value or two constant values from a regular depth mode or two constant values or from a DMM mode. It also supports a depth look-up table (DLT) to convert the depth values to a reduced dynamic range, giving coding gains for sequences with reduced depth value ranges [17]. SDC can be applied to both intra and inter prediction, including all depth intra prediction, the DMM modes, as well as all Intra prediction modes as in HEVC [18]. Thus, this coding tool also reflects the two main depth map characteristics of blocks with homogenous regions vs. blocks with sharp edges. When the SDC mode is applied, only one DC predictor is derived for each partition, and based on that, only one DC difference value is coded as the residual for the whole partition. For calculating the DC difference value, a predicted depth value is calculated from the corner points of the current depth block. The DC difference value may be set as the difference between mean value of the original depth block and the predicted depth value, or chosen from multiple values with the least rate-distortion cost as measured by VSO [19].

## **5. Performance Evaluation**

To evaluate the compression efficiency of the different architectures and coding techniques, simulations were conducted using the reference software and experimental evaluation methodology that has been developed and is being used by the standardization community [20]. In the experimental framework, multiview video and corresponding depth are provided as input, while the decoded views and additional views synthesized at selected positions are generated as output. As defined in the common test conditions, the base view is coded as the center view of each input test sequence and two non-base (dependent) views positioned to the left and right of the center view are also coded. Two results are generated for each comparison: one that assesses the quality of the decoded video as a function of the video bit rate (without depth) and another that assesses the synthesized video quality as a function of the total bit rate including texture and depth. In both cases, the respective PSNR values and bit rates are averaged for all three views.

Table 1 provides a comparison of different codec configurations, including simulcast, MV-HEVC and 3D-HEVC. The results indicate that MV-HEVC provides an average bit rate savings of 40% for the decoded video and 30% for the synthesized video relative to simulcast, which demonstrates the effectiveness of the inter-view sample prediction of texture. A more detailed analysis of the rate savings for each view shows that more than a 60% reduction in bit rate could be achieved for each dependent view using MV-HEVC.

The performance of 3D-HEVC is compared to both simulcast and MV-HEVC. Relative to MV-HEVC, a modest bit rate savings of 10% could be achieved. However, when

considering the depth and the synthesized video quality, a coding efficiency gain of 21% could be achieved.

**Table 1:** Evaluation of coding efficiency performance of MV-HEVC and 3D-HEVC.

	MV-HEVC vs Simulcast		3D-HEVC vs Simulcast		3D-HEVC vs MV-HEVC	
	video PSNR / video bitrate	synth PSNR / total bitrate	video PSNR / video bitrate	synth PSNR / total bitrate	video PSNR / video bitrate	synth PSNR / total bitrate
Balloons	-31.5%	-24.3%	-42.3%	-38.9%	-15.9%	-19.7%
Kendo	-30.4%	-21.4%	-40.9%	-38.5%	-15.2%	-22.1%
Newspaper_CC	-33.2%	-25.7%	-39.3%	-39.3%	-9.2%	-18.7%
GT_Fly	-52.4%	-40.0%	-57.2%	-54.4%	-10.0%	-24.1%
Poznan_Hall2	-32.8%	-24.2%	-40.4%	-38.4%	-11.2%	-19.2%
Poznan_Street	-41.4%	-32.6%	-45.4%	-41.1%	-6.9%	-13.3%
Undo_Dancer	-47.3%	-36.3%	-51.8%	-49.8%	-8.5%	-21.5%
Shark	-54.1%	-39.9%	-58.3%	-57.6%	-9.0%	-29.6%
<b>Average</b>	<b>-40.4%</b>	<b>-30.6%</b>	<b>-46.9%</b>	<b>-44.8%</b>	<b>-10.7%</b>	<b>-21.0%</b>

## 6. Concluding Remarks

This article reviewed recent extensions to the HEVC coding standard, which provide state-of-the-art compression of multiview video, and for the first time, a compression format that includes depth video to support improved compression and advanced functionalities. Both the MV-HEVC and 3D-HEVC extensions provide backward compatibility with HEVC for a single view. Moreover, 3D-HEVC adopts several new coding tools for more accurate derivation and better usage of disparity information to improve the compression efficiency of dependent texture views and also specifies novel coding tools for efficient representation of the depth videos. It is expected that these formats will be considered for next-generation broadcast services and other emerging 3D video applications.

## References

- [1] High Efficiency Video Coding, Rec. ITU-T Rec. H.265 and ISO/IEC 23008-2, 2014.
- [2] Y. Chen, Y. -K. Wang, K. Ugur, M. M. Hannuksela, J. Lainema, and M. Gabbouj, "The Emerging MVC Standard for 3D Video Services," *EURASIP Journal on Advances in Signal Processing*, Volume 2009, Article ID 786015.
- [3] A. Vetro, T. Wiegand, and G. J. Sullivan, "Overview of the Stereo and Multiview Video Coding Extensions of the H.264/AVC Standard," *Proc. of the IEEE*, vol. 99, no. 4, pp. 626–642, Apr. 2011.
- [4] G. Tech, K. Wegner, Y. Chen, M. M. Hannuksela, and J. Boyce, "MV-HEVC Draft Text 9", Joint Collaborative Team on 3D Video Coding Extension Development (JCT-3V) Doc. JCT3V-I1002, Sapporo, JP, July 2014.
- [5] G. Tech, K. Wegner, Y. Chen, and S. Yea, "3D-HEVC Draft Text 6," Joint Collaborative Team on 3D Video Coding Extension Development (JCT-3V) Doc. JCT3V-J1001, Strasbourg, FR, Oct. 2014.

- [6] G.J. Sullivan, J.M. Boyce, Y. Chen, J-R. Ohm, C.A. Segall, A. Vetro, "Standardized Extensions of High Efficiency Video Coding (HEVC)", *IEEE Journal of Selected Topics in Signal Processing*, Dec. 2013.
- [7] L. Zhang, Y. Chen, and M. Karczewicz, "Disparity Vector based Advanced Inter-view Prediction in 3D-HEVC," *IEEE International Symposium on Circuits and Systems (ISCAS)*, Beijing, May 2013, pp. 1632–1635.
- [8] J. Kang, Y. Chen, L. Zhang, and M. Karczewicz, "Low Complexity Neighboring Block Based Disparity Vector Derivation in 3D-HEVC," *IEEE International Symposium on Circuits and Systems (ISCAS)*, Melbourne, June 2014, pp. 1921–1924.
- [9] L. Zhang, Y. Chen, J.-L. Lin, Y.-W. Chen, J. An, S. Lei, "Inter-view motion prediction in 3D-HEVC," in *Proc. of IEEE International Symposium on Circuits and Systems (ISCAS)*, Melbourne, June 2014, pp. 17–20.
- [10] J. An, K. Zhang, J.-L. Lin, S. Lei "3D-CE3: Sub-PU level inter-view motion prediction," Joint Collaborative Team on 3D Video Coding Extension Development (JCT-3V) Doc. JCT3V-F0110, Geneva, CH, Oct/Nov. 2013.
- [11] X. Li, L. Zhang and Y. Chen, "Advanced residual prediction in 3D-HEVC," *IEEE International Conference on Image Processing (ICIP)*, Melbourne, Australia, Sept. 2013, pp. 1747–1751.
- [12] L. Zhang, Y. Chen, X. Li and S. Xue, "Low-complexity advanced residual prediction design in 3D-HEVC," in *Proc. of IEEE International Symposium on Circuits and Systems (ISCAS)*, Melbourne, June 2014, pp 13–16.
- [13] H. Liu, J. Jung, J. Sung, J. Jia, and S. Yea, "3D-CE2.h: Results of Illumination Compensation for Inter-View Prediction," Joint Collaborative Team on 3D Video Coding Extension Development (JCT-3V) Doc. JCT3V-B0045, Shanghai, Oct. 2012.
- [14] F. Zou, D. Tian, A. Vetro, H. Sun, O. C. Au, and S. Shimizu, "View Synthesis Prediction in the 3-D Video Coding Extensions of AVC and HEVC," *IEEE Trans. on Circuits and Syst. for Video Technol.*, vol. 24, no. 10, pp. 1696-1708, Oct. 2014.
- [15] G., Tech, K. Müller and T. Wiegand, "3D video coding using the synthesized view distortion change," *Picture Coding Symposium (PCS)*, May 2012.
- [16] K. Müller, P. Merkle, G. Tech, and T. Wiegand, "3D Video Coding with Depth Modeling Modes and View Synthesis Optimization," *Proceedings of the Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*, Oct. 2012.
- [17] F. Jager, "Simplified depth map intra coding with an optional depth lookup table," *International Conference on 3D Imaging (IC3D)*, Dec. 2012.
- [18] H. Liu and Y. Chen, "Generic Segment-Wise DC for 3D-HEVC Depth Intra Coding," *IEEE International Conference on Image Processing (ICIP)*, Oct. 2014.
- [19] J. Y. Lee, M. W. Park, C. Kim, "3D-CE5: Segment-wise depth inter mode coding," Joint Collaborative Team on 3D Video Coding Extension Development (JCT-3V) Doc. JCT3V-G0101, San Jose, CA, Jan. 2014.
- [20] K. Müller and A. Vetro, "Common test conditions of 3DV core experiments," Joint Collaborative Team on 3D Video Coding Extension Development (JCT-3V) Doc. JCT3V-G1100, San Jose, CA, Jan. 2014.