MITSUBISHI ELECTRIC RESEARCH LABORATORIES
http://www.merl.com

Depth Based 3D Video Formats and Coding Technology

Vetro, A.; Muller, K.

TR2013-138 April 2013

Abstract

The primary usage scenario for 3D video (3DV) formats is to support depth perception of a visual scene as provided by a 3D display system. There are many types of 3D display systems including classic stereoscopic systems that require special-purpose glasses to more sophisticated multiview autostereoscopic displays that do not require glasses (Konrad and Halle, 2007). While stereoscopic systems only require two views, the multiview displays have much higher data throughput requirements since 3D is achieved by essentially emitting multiple videos in order to form view-dependent pictures. Such displays can be implemented, for example, using conventional high-resolution displays and parallax barriers; other technologies include lenticular overlay sheets and holographic screens. Each viewdependent video sample can be thought of as emitting a small number of light rays in a set of discrete viewing directions–typically between eight and a few dozen for an auto-stereoscopic display. Often these directions are distributed in a horizontal plane, such that parallax effects are limited to the horizontal motion of the observer. A more comprehensive review of 3D display technologies is given in Chapter 15, as well as by Benzie et al (2007). An overview can also be found in Ozaktas and Onural (2007).

Emerging Technologies for 3D Video: Creation, Coding, Transmission and Rendering

8 Depth Based 3D Video Formats and Coding Technology

Anthony Vetro and Karsten Müller

8.1 Introduction

The primary usage scenario for 3D video (3DV) formats is to support depth perception of a visual scene as provided by a 3D display system. There are many types of 3D display systems including classic stereoscopic systems that require special-purpose glasses to more sophisticated multiview auto-stereoscopic displays that do not require glasses (Konrad and Halle, 2007). While stereoscopic systems only require two views, the multiview displays have much higher data throughput requirements since 3D is achieved by essentially emitting multiple videos in order to form view-dependent pictures. Such displays can be implemented, for example, using conventional high-resolution displays and parallax barriers; other technologies include lenticular overlay sheets and holographic screens. Each view-dependent video sample can be thought of as emitting a small number of light rays in a set of discrete viewing directions – typically between eight and a few dozen for an auto-stereoscopic display. Often these directions are distributed in a horizontal plane, such that parallax effects are limited to the horizontal motion of the observer. A more comprehensive review of 3D display technologies is given in Chapter 15, as well as by Benzie et al (2007). An overview can also be found in Ozaktas and Onural (2007).

Other goals of 3D video formats include enabling free-viewpoint video, whereby the viewpoint and view direction can be interactively changed. With such a system, viewers can freely navigate through the different viewpoints of the scene. 3D video can also be used to support immersive teleconference applications. Beyond the advantages provided by 3D displays, it has been reported that a teleconference systems could enable a more realistic communication experience when motion parallax is supported.

Existing stereo and multiview formats are only able to support the above applications and scenarios to a limited extent. As an introduction, these formats are briefly described and the requirements and functionalities that are expected to be fully supported by depth-based formats are discussed. A more comprehensive overview can be found in Chapter 6.

8.1.1 Existing Stereo/Multiview Formats

Currently, there exist two primary classes of multiview formats: frame compatible, and stereo or multiview video, which are briefly reviewed in the following.

Frame compatible formats refer to a class of stereo video formats in which the two stereo views are filtered, sub-sampled and arranged into a single coded frame or sequence of frames, i.e., the left and right views are packed together in the samples of a single video frame (Vetro, Tourapis, Müller and

Chen, 2011). Popular arrangements include the side-by-side and top-bottom formats. Temporal multiplexing is also possible where the left and right views would be interleaved as alternating frames or fields. The primary benefit of frame-compatible formats is that they facilitate the introduction of stereoscopic services through existing infrastructure and equipment. In this way, the stereo video can be compressed with existing encoders, transmitted through existing channels, and decoded by existing receivers. The video-level signaling for these formats is specified by the MPEG-2 Video and MPEG-4 AVC standards (Vetro, Wiegand and Sullivan, 2011).

As an alternative to frame-compatible formats, direct encoding of the stereo and multiview video may also be done using multiview extensions of either MPEG-2 Video or MPEG-4 AVC standards (Vetro, Wiegand and Sullivan, 2011). A key capability is the use of inter-view prediction to improve compression capability, in addition to ordinary intra and inter-prediction modes. Another key aspect of all multiview video coding designs is the inherent support for 2D/backwards compatibility with existing legacy systems. In other words, the compressed multiview stream includes a base view bitstream that is coded independently from all other views in a manner compatible with decoders for single-view profile of the standard.

8.1.2 Requirements for Depth-Based Format

Depth-based representations are another important and emerging class of 3D formats. Such formats are unique in that they enable the generation of virtual views through depth-based image rendering techniques (Kauff et al, 2007), which may be required by auto-stereoscopic or multiview displays (Müller Merkle and Wiegand, 2011). Depth-based 3D formats can also allow for advanced stereoscopic processing, such as adjusting the level of depth perception with stereo displays according to viewing characteristics such as display size, viewing distance or user preference. The depth information itself may be extracted from a stereo pair by solving for stereo correspondences or obtained directly through special range cameras; it may also be an inherent part of the content, such as with computer generated imagery.

ISO/IEC 23002-3 (also referred to as MPEG-C Part 3) specifies the representation of auxiliary video and supplemental information. In particular, it enables signaling for depth map streams to support 3D video applications. The well-known 2D plus depth format (see Fig. 8.1) is supported by this standard. It is noted that this standard does not specify the means by which the depth information is coded, nor does it specify the means by which the 2D video is coded. In this way, backward compatibility to legacy devices can be provided.

Fig. 8.1: 2D plus depth format (images provided courtesy of Microsoft Research).

The main drawback of the 2D plus depth format is that it is only capable of rendering a limited depth range and was not specifically designed to handle occlusions. Also, stereo signals are not easily accessible by this format; i.e., receivers would be required to generate the second view to drive a stereo display, which is not the convention in existing displays.

To overcome the drawbacks of the 2D plus depth format, a multiview video plus depth (MVD) format with a limited number of original input views and associated per pixel depth data can be considered. For instance, with two input views, high quality stereo video is provided and the depth information would enhance 3D rendering capabilities beyond 2D plus depth. However, for high-quality auto-stereoscopic displays, wide-baseline rendering with additional views beyond the stereo range may be required. For example, formats with 3 or 4 views with associated depth map data may be considered.

8.1.3 Chapter Organization

Depth-based 3D formats beyond 2D plus depth are a current topic of study in MPEG. This chapter provides an overview of the current status of research and standardization activity towards defining a new set of depth-based formats that facilitate the generation of intermediate views with a compact binary representation. The next section provides a brief introduction to depth based representation and rendering techniques, which are the basis for the functionality offered by depth-based formats. Next, the different coding architectures are discussed, including those that are compatible to the existing AVC standard as well as those that are compatible to the emerging HEVC standard. Hybrid coding architectures that mix coding formats are also discussed. The following section presents the various compression technologies that have been proposed and are being considered for standardization. Then, results from a large-scale experimental evaluation are presented to indicate the possible improvements over the current-state-of-the-art coding approaches. The chapter concludes with a summary of material that has been presented and discusses future research and standardization.

8.2 Depth Representation & Rendering

For the creation of a generic depth based multiview format for various 3D displays, scene geometry needs to be provided in addition to texture information. A very general scene geometry representation is given by depth maps, which can be provided by different methods as shown in the next subsection. These depth maps can then be used to apply depth image-based rendering (DIBR) in order to generate the required number of intermediate views for any auto-stereoscopic multiview display.

8.2.1 Depth Format and Representation

In computer graphics and computer vision, a 3D scene consists of texture and geometry information. Texture information, like color and luminance, are directly recorded by camera sensors. Geometry or 3D structure information can be obtained in different ways: For synthetic sequences, such as computer-generated scene content and animated films, scene geometry information is directly available, e.g. in the form of wireframe models (ISO/IEC JTC1/SC29/WG11, 1997) or 3D point coordinates (Würmlin, Lamboray and Gross, 2004). For natural scenes, geometry information can be recorded as distance information by special sensors, like time-of-flight cameras (Lee, Jung and Ho, 2010). This distance information is then recorded as a grey value depth image. Usually, these depth images have a lower

resolution than associated texture information from video cameras. Also, time-of-flight sensors currently lack accuracy for larger distances and have to be placed at slightly different positions than the video camera. Accordingly, some additional processing is required, if video and depth data are to be aligned.

For natural scenes that were only recorded by video cameras, depth data can be estimated. For this, intensive investigations of different algorithms were carried out by Scharstein and Szeliski (2002). An overview on disparity estimation techniques can also be found in Chapter 5.The basic principle of most depth estimation algorithms is to match corresponding image blocks or regions in two or more cameras with slightly different positions to obtain the displacement or disparity between corresponding texture pixels. The relation between depth information and disparity is given by projective geometry, where points of a 3D scene are projected onto camera planes with the use of projection matrices (Faugeras, 1993; Hartley and Zisserman, 2000). For parallel cameras, a simplified relation can be derived, as shown in Fig. 8.2.

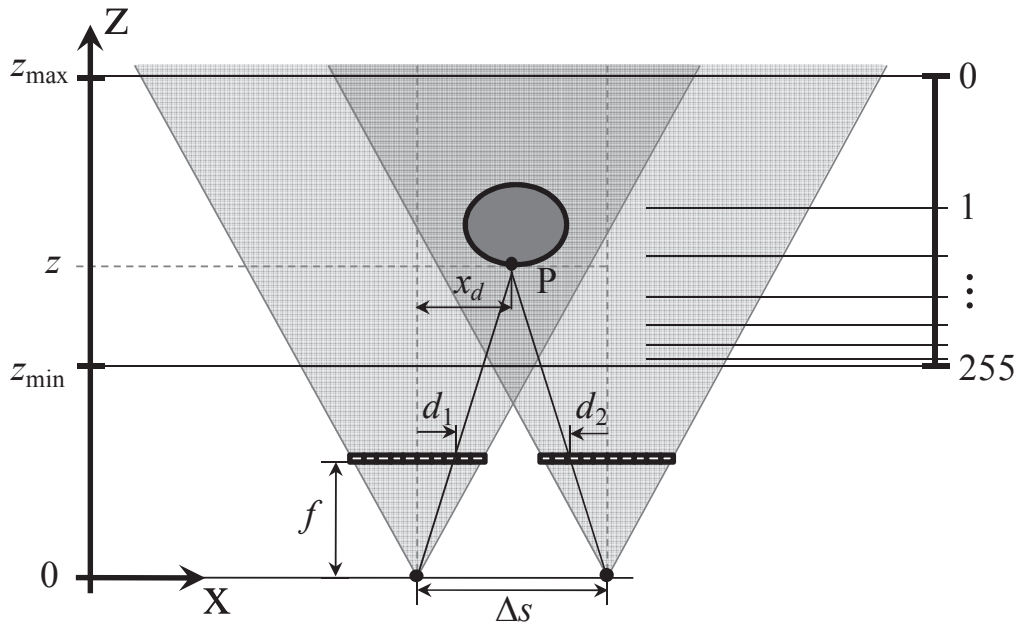

Fig. 8.2: Relation between depth values *z* and disparity *d*.

Here, a point **P** of a 3D scene is recorded by two cameras and projected onto their image sensors in the camera planes. The depth distance of **P** from the camera centers shall be z. Let the distance between both cameras be Δs and assume identical focal lengths f. Then, **P** is projected onto both sensors with an offset from the centre of d_1 in the left and d_2 in the right camera plane. This gives the following relations:

$$\frac{d_1}{x_d} = \frac{f}{z} \text{ for the left camera and } \frac{d_2}{\Delta s - x_d} = \frac{f}{z} \text{ for the right camera.} \quad (8.1)$$

Adding both equations, the following inverse relationship between depth value z and disparity value d between both projections of **P** can be found:

$$d = d_1 + d_2 = \frac{f \cdot x_d}{z} + \frac{f \cdot (\Delta s - x_d)}{z}$$
$$\Rightarrow d = \frac{f \cdot \Delta s}{z} \tag{8.2}$$

The disparity value d for each pixel is obtained from matching left and right image content. Depending on the desired accuracy, disparity values are usually linearly quantized, e.g. into 256 discreet values to operate with 8bit resolution data as done for many practical applications. According to equation (8.2), this refers to an inverse quantization of depth values, as indicated in the depth scale on the right side in Fig. 8.2. For optimal usage of the depth range, the nearest and farthest depth values, z_{min} and z_{max} are determined and inverse depth quantization is carried out in the range $[z_{min}, z_{max}]$. Therefore, the stored inverse depth values $I_d(z)$ are calculated as:

$$I_d(z) = \text{round}\left[255 \cdot \left(\frac{1}{z} - \frac{1}{z_{max}}\right) \Big/ \left(\frac{1}{z_{min}} - \frac{1}{z_{max}}\right)\right] \tag{8.3}$$

In practical applications, disparity estimation algorithms use a matching function (Szeliski et al, 2006) with different area support and size for corresponding image blocks in left and right view (Bleyer and Gelautz, 2005). Furthermore, they apply a matching criterion, e.g. the sum of absolute differences or cross-correlation. The estimation process is optimized by different means, including graph cuts (Kolmogorov and Zabih, 2002), belief propagation (Felzenszwalb and Huttenlocher, 2006), plane sweeping (Cigla, Zabulis and Alatan, 2007), or combined methods (Atzpadin, Kauff and Schreer, 2004; Kolmogorov, 2006). Depth estimation has also been studied with special emphasis for multiview video content and temporal consistency in order to provide depth data for multiview video sequences (Lee and Ho, 2010; Min, Yea and Vetro, 2010; Tanimoto, Fujii and Suzuki, 2008).

Although depth estimation algorithms have been improved considerably in recent years, they can still be erroneous in some cases due to mismatches, especially for partially occluded image and video content that is only visible in one view.

8.2.2 Depth Image Based Rendering

If a generic format with texture and depth components of a few views is used in 3D or free viewpoint video applications, additional views have to be generated, e.g. for 3D displays with different number of views. For this view generation process, depth image-based rendering (DIBR) is used (Kauff et al, 2007; Redert et al, 2002). In this process, the texture information is projected or warped to a new viewing position, at which an intermediate view is to be synthesized. The warping distance of each texture sample is determined from the associated per-sample depth information. Current video solutions use a rectification process, in which a strictly parallel camera scenario is enforced, as shown in the stereo scheme in Fig. 8.2. In such settings, the general DIBR process can be simplified to a horizontal sample shift from original to newly rendered views. For calculating the shift values, the disparity values between original views are obtained first by combining equations (8.2) and (8.3):

$$d = f \cdot \Delta s \cdot \frac{I_d(z)}{255} \cdot \left(\frac{1}{z_{\min}} - \frac{1}{z_{\max}} \right) + \frac{1}{z_{\max}} \tag{8.4}$$

In addition to the inverse depth values $I_d(z)$, focal length f, camera baseline Δs, as well as the nearest and farthest depth values $z_{\min}$ and $z_{\max}$ have to be known. Here, d gives the disparity or shift value between corresponding texture samples in the two original views, separated by Δs. For calculating the required horizontal shift value for a new view to be synthesized, the view position has to be known and the disparity scaled accordingly. An example is given in Fig. 8.3, where two original camera positions, *Cam* 1 and *Cam* 2 with texture samples c_1 and c_2 at positions (u_1,v_1) and (u_2,v_2) are given.

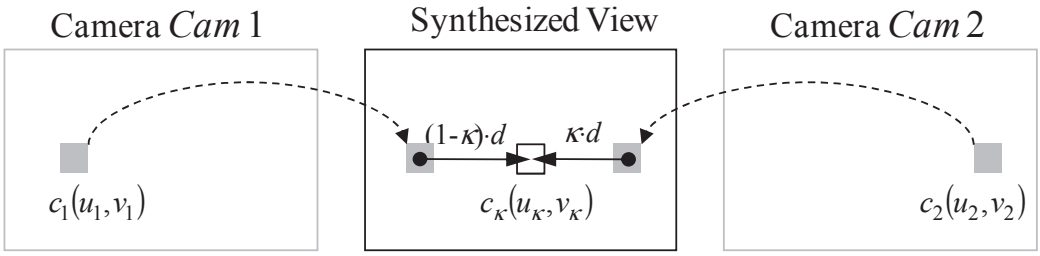

Fig. 8.3: View synthesis principle with horizontal disparity based shift from original data (*Cam* 1 and *Cam* 2) to new position in synthesized view.

Furthermore, a new view shall be synthesized between both original cameras with texture sample c_κ at position (u_κ,v_κ). Here, $\kappa \in [0\ldots1]$ represents the intermediate position parameter, which specifies the location between both original cameras. Accordingly, the relation between texture sample $c_1(u_1,v_1)$ in *Cam* 1 and its shifted version in the intermediate view is given by $c_{\kappa 1}(u_\kappa,v_\kappa) = c_1(u_1+(1-\kappa)\cdot d,v_1)$. Similarly, $c_{\kappa 2}(u_\kappa,v_\kappa) = c_2(u_2+\kappa\cdot d,v_2)$ for *Cam* 2. In addition to shifting both original texture samples to the new position, color blending can be applied, if color variances due to different illumination in both original views occur (Müller et al, 2008). This results in the final synthesized sample $c_\kappa(u_\kappa,v_\kappa)$:

$$\begin{aligned} c_\kappa(u_\kappa,v_\kappa) &= (1-\kappa)\cdot c_{\kappa 1}(u_\kappa,v_\kappa) + \kappa \cdot c_{\kappa 2}(u_\kappa,v_\kappa) \\ &= (1-\kappa)\cdot c_1(u_1+(1-\kappa)\cdot d, v_1) + \kappa \cdot c_2(u_2-\kappa\cdot d, v_2) \end{aligned} \tag{8.5}$$

In cases, where only one original texture sample is available, e.g. if scene content is occluded in one view, a synthesized texture sample $c_\kappa(u_\kappa,v_\kappa)$ is obtained by shifting the visible original sample without color blending. For an improved visual impression, additional processing steps, such as hole filling, filtering and texture synthesis of disoccluded areas are usually applied after sample-wise shifting and texture blending (Müller et al, 2008).

Besides synthesizing new views between two original cameras, additional views can also be extrapolated (synthesized outside the viewing range of original cameras) by setting $\kappa < 0$ or $\kappa > 1$.

8.3 Coding Architectures

8.3.1 AVC-based

Significant improvements in video compression capability have been demonstrated with the introduction of the H.264/MPEG-4 Advanced Video Coding (AVC) standard (ITU-T and ISO/IEC, 2010; Wiegand, Sullivan, Bjøntegaard and Luthra, 2003), which has been extensively deployed for a wide range of video products and services. An extension of this standard for Multiview Video Coding (MVC) was first finalized in July 2009 and later amended in July 2010. The MVC format was selected by the Blu-Ray Disc Association as the coding format for stereo video with high-definition resolution (BDA, 2009), and has recently been standardized for stereo broadcast as well (DVB, 2012).

Within the AVC-based framework, one target for standardization is an MVC compatible extension including depth, where the main target is to enable 3D enhancements while maintaining MVC stereo compatibility for the texture videos. The approach would invoke an independent second stream for the representation of stereo depth maps as if they were monochrome video data, as well as high-level syntax signaling of the necessary information to express the interpretation of the depth data and its association with the video data. An illustration of the compatibility supported by this architecture is shown in Fig. 8.4 (left). Macroblock-level changes to the AVC or MVC syntax, semantics and decoding processes are not considered in this configuration in order to maintain compatibility. The standardization of this approach is currently underway and is expected to be completed by early 2013.

Considering that certain systems only maintain compatibility with monoscopic AVC, a second architecture that is being considered is an AVC-compatible extension that includes depth. In this approach, further coding efficiency gains could be obtained by improving the compression efficiency of non-base texture views and the depth data itself. However, in contrast to the MVC-compatible approach, this method requires changes to the syntax and decoding process for non-base texture views and depth information at the block level. An illustration of the compatibility supported by this architecture is shown in Fig. 8.4 (right). Clearly, a notable coding efficiency benefit relative to the MVC-compatible approach would be required to justify the standardization of this approach. This is a current topic of study within the standardization committees.

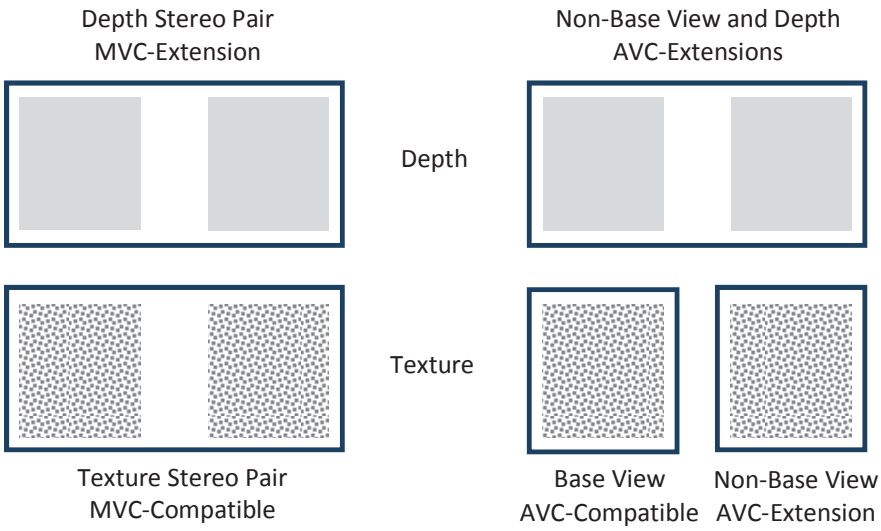

Fig. 8.4: Illustration of MVC-compatible (left) and AVC-compatible (right) architecture for depth-based 3D video coding.

8.3.2 HEVC-based

High-efficiency video coding (HEVC) is a new video coding standard that is designed for monoscopic video to provide the same subjective quality at half the bit rate compared to AVC High Profile. A primary usage of HEVC is to support the delivery of ultra-high definition (UHD) video. It is believed that many UHD displays will also be capable of decoding stereo video as well. The first version of the standard will be approved by January 2013.

For the highest compression efficiency, 3D video coding extensions based on the HEVC are being developed, where multiview video data with associated depth maps are coded (see Fig. 8.5). In this architecture, the base view is fully compatible with HEVC in order to extract monoscopic video, while the coding of dependent views and depth maps would utilize additional tools as described in section 8.4. It is also anticipated that there would be profiles of the standard in which stereo video can be easily extracted to support existing stereoscopic displays; in such cases, the dependency between the video data and depth data may be limited.

A subset of this 3D video coding extension would include a simple multiview extension of HEVC, utilizing the same design principles of MVC in the MPEG-4/H.264 AVC framework (i.e., providing backwards compatibility for monoscopic decoding). It is expected that this extension of HEVC be available as final standard by early 2014. Additionally, it is planned to develop a suite of tools for scalable coding, where both view scalability and spatial scalability would allow for backward-compatible extensions for more views. These extensions would also accommodate the depth data that would be associated with each view and/or provide for a way to enhance the resolution of views. Ideally, all of this would be achieved in such a way that decoding by legacy devices is still possible.

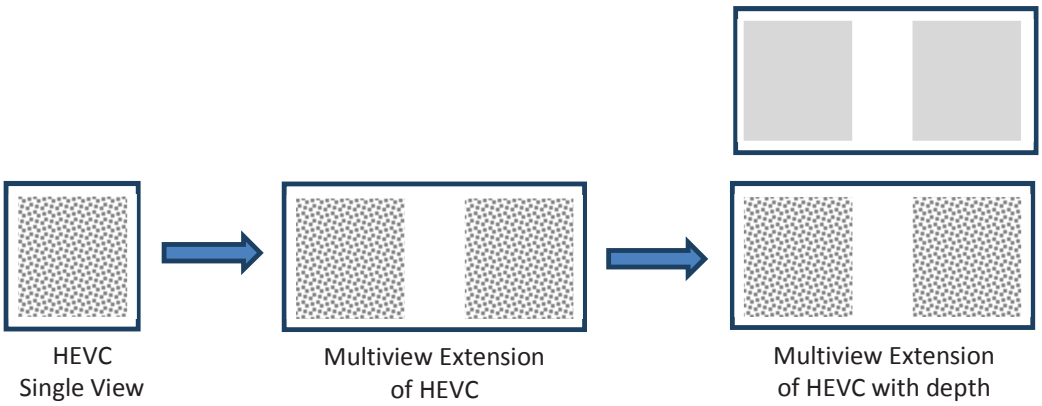

Fig. 8.5: Extensions of HEVC coding standard for support of 3DV architectures with multiple views and depth.

8.3.3 Hybrid

From a pure compression efficiency point of view, it is always best to use the most advanced codec. However, when introducing new services, providers must also consider capabilities of existing receivers and an appropriate transition plan. Considering that most terrestrial broadcast systems are based on MPEG-2 or AVC, it may not be easy to simply switch codecs in the near-term.

One solution to this problem is to transmit the 2D program in the legacy format (e.g., MPEG-2), while transmitting an additional view to support stereo services in an advanced coding format (e.g. AVC). The obvious advantage is that backward compatibility with the existing system is provided with significant bandwidth savings relative to simulcast in the legacy format. One drawback of this approach is that there is a strong dependency between the 3D program and the 2D program. Such a system would not support independent programming of 2D and 3D content programs, which may be desirable for production reasons. Also, this approach requires legacy and advanced codecs to operate synchronously, which may pose implementation challenges for certain receiver designs. Nevertheless, broadcasting trials of hybrid MPEG-2 and AVC based systems are underway in Korea, and there are plans to standardize the transmission of such a hybrid format in ATSC.

In the context of depth-based 3D formats, there are clearly many variations that could be considered. In an AVC-compatible framework, the base view would be coded with AVC, while additional texture views and supplemental depth videos could be encoded with HEVC. A slight variation on this would be for the stereo pair of the texture to be coded with MVC, and only the depth videos are coded with HEVC. A simple block diagram illustrating the hybrid video coding architecture for the right and left views of a stereoscopic video program is given in Fig. 8.6.

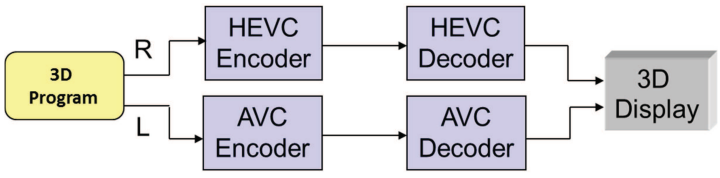

Fig. 8.6: Illustration of hybrid architecture with monoscopic AVC base and HEVC enhancement program for stereo support.

One open issue with hybrid architectures that requires further study is the degree of inter-component dependency that would be permitted across different components and different coding standards. For instance, the benefits of using decoded pictures from the AVC base view to predict pictures in the second view coded with HEVC needs to be weighed against the implementation challenges. Also, it needs to be considered whether lower-level dependencies could be supported, e.g., sharing of motion or mode data.

8.4 Compression Technology

For the efficient compression of 3D video data with multiple video and depth components, a number of coding tools are used to exploit the different dependencies among the components. First, one video component is independently coded by a conventional block-based 2D video coding method, such as AVC or HEVC without additional tools in order to provide compatibility with existing 2D video services. For each additional 3D video component, i.e. the video component of the dependent views as well as the depth maps, additional coding tools are added on top of a 2D coding structure. Thus, a 3DV encoder can select the best coding method for each block from a set of conventional 2D coding tools and additional new coding tools, some of which are described in the following subsections.

8.4.1 Inter-View Prediction

The basic concept of inter-view prediction, which is employed in all standardized designs for efficient multiview video coding, is to exploit both inter-view and temporal redundancy for compression. Since the cameras of a multiview scenario typically capture the same scene from nearby viewpoints, substantial inter-view redundancy is present. This holds for both texture views and the corresponding depth map images associated with each view, thus inter-prediction can be applied to both types of data independently.

A sample prediction structure is shown in Fig. 8.7. In modern video coding standards such as AVC and HEVC, inter-view prediction is enabled through the flexible reference picture management capabilities of those standards. Essentially, the decoded pictures from other views are made available in the reference picture lists for use by the inter-picture prediction processing. As a result, the reference picture lists include the temporal reference pictures that may be used to predict the current picture along with the inter-view reference pictures from neighboring views. With this design, block-level decoding modules remain unchanged and only small changes to the high-level syntax are required, e.g., indication of the prediction dependency across views and corresponding view identifiers. The prediction is adaptive, so the best predictor among temporal and inter-view references can be selected on a block basis in terms of rate-distortion cost.

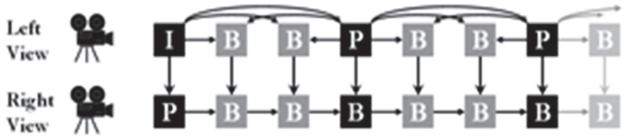

Fig. 8.7: Illustration of inter-view prediction.

Relative to simulcast, which does not utilize inter-view prediction, it has been shown through experiments that inter-view prediction is responsible for the majority of the coding efficiency gains. This leads to a simplified design for efficient multiview video coding (both texture and depth) with good coding efficiency capability. For additional information on the design, syntax and coding efficiency of inter-view prediction, readers are referred to Merkle, Smolic, Müller and Wiegand (2007) as well as Vetro, Wiegand and Sullivan (2011).

In the following subsections, coding tools that go beyond picture-based inter-view prediction are described. Many of these require changes to lower-levels of the decoding syntax and process, with the benefit of additional gains in coding efficiency.

8.4.2 View Synthesis Prediction

In addition to being used for generation of intermediate views, the depth-image based rendering techniques described in section 8.2.2 could also be used as a unique form of inter-view prediction that is referred to as view synthesis prediction. In contrast to the inter-view prediction technique presented in the previous subsection, which essentially predict a block of pixels in one view by means of a linear disparity vector, view synthesis prediction exploits the geometry of the 3D scene by warping the pixels from a reference view to the predicted view as illustrated in Fig. 8.3 and Fig. 8.8.

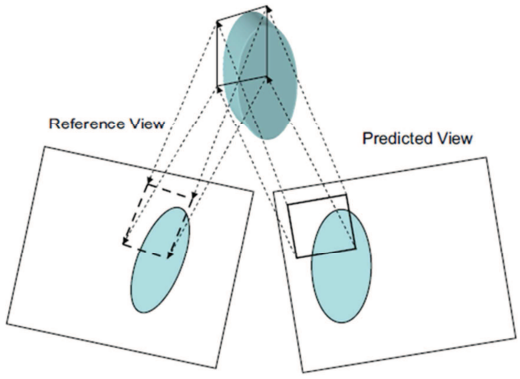

Fig. 8.8: Illustration of view synthesis prediction.

During the development of the multiview extensions of MPEG-4/H.264 AVC, depth maps were not assumed to be available or an integral part of the data format. To enable view synthesis prediction in this framework, the depth for each block would need to be estimated and explicitly coded as side information, so that the decoder could generate the view synthesis data used for prediction. Such a scheme was first described by Martinian, Behrens, Xin and Vetro (2006) and more fully elaborated on by Yea and Vetro (2009). Although coding efficiency gains were reported, the benefit of this type of prediction was limited by the overhead incurred by the additional block-based depth that was required to be sent.

In the 3DV framework, depth is available as an integral part of the data format, therefore the generation of a synthesized view can be done without any additional side information. In this way, a synthesized

view could be generated and added to the reference picture list and thus used for prediction as any other reference picture. The only requirement is to signal the appropriate reference picture index so that the decoder knows that the prediction for a particular block is done with respect to a view synthesis reference picture.

8.4.3 Depth Resampling and Filtering

Reducing the resolution of the depth maps image could provide substantial rate reductions. However, filtering and reconstruction techniques need to be carefully designed to maximize quality. Specifically, the quality of the depth map will have a direct impact on the quality of the synthesized views.

There have been several past studies on up-sampling and reconstruction techniques of reduced resolution depth. For instance, a non-linear reconstruction filter that refines edges based on the object contours in the video data was proposed by Oh, Yea, Vetro and Ho (2009). A key advantage of this method was that the edge information was preserved. It was shown that bit rate reductions greater than 60% relative to full-resolution coding of the depth videos could be achieved. Furthermore, improved rendering quality around the object boundaries compared to conventional filtering techniques was demonstrated.

Due to the unique characteristics of edges within the depth image, further work in this area has verified that non-linear approaches are generally more favorable than linear filtering techniques, e.g., as reported by Beni, Rusanovskyy and Hannuksela (2012) and Lee, Lee, Wey and Lee (2012). As shown in Fig. 8.9 (left), linear filters will tend to blur edges and introduce artifacts in the rendered edge, while non-linear filtering techniques (e.g., median or dilation filters) are able to better preserve the true edge characteristics and render a synthetic image with fewer artifacts as shown in Fig. 8.9 (right).

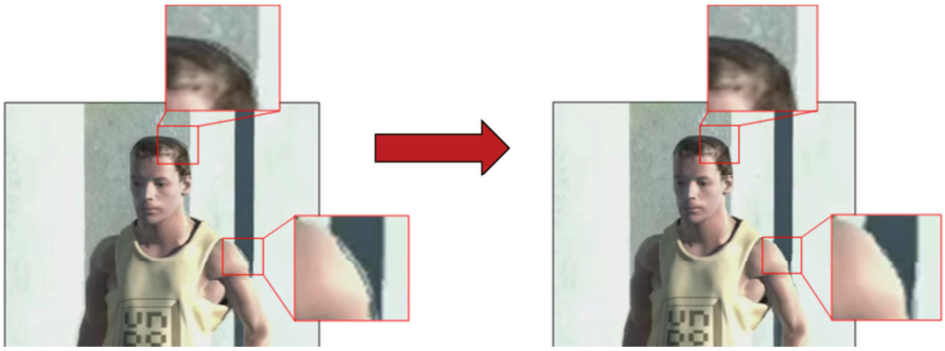

Fig. 8.9: Comparison of linear interpolation for depth up-sampling (left) versus a non-linear filtering approach (right) (original images provided courtesy of Nokia).

The filtering techniques can be applied either within the decoder loop or outside the decoder loop as a post-process. While there is ongoing work in this area, preliminary results suggest that most of the subjective visual benefit is achieved with post-processing techniques (Rusert, 2012). However, in-loop processing certainly has the potential to improve coding efficiency, especially for tools that rely on accurate depth values around edges to make predictions (e.g., view synthesis prediction). Additional techniques on depth video coding can also be found in Chapter 7.

8.4.4 Inter-Component Parameter Prediction

For the joint coding of video and depth data in an MVD format, dependencies between both components are identified and exploited. As each video component has an associated depth map at identical time instance and view point, a similar scene characteristic exists. This includes the collocation of scene objects with their texture and distance information in the video and depth component respectively. Furthermore, the object motion in both components is identical. Therefore, an additional coding mode can be used for depth maps, where the block partitioning into sub-blocks, as well as associated motion parameters are inferred from the co-located block in the associated video picture (Winken, Schwarz and Wiegand, 2012). Accordingly, it is adaptively decided for each depth block, whether partitioning and motion information are inherited from the co-located region of the video picture, or new motion data transmitted. If such information is inherited, no additional bits for partitioning and motion information are required. Note however, that the real object motion and the motion vector, estimated by the encoder are not necessarily identical: The estimated motion has to be coded together with other information, such as residual data. Therefore, a motion vector can be selected for a block that results in the best encoder decision, but differs significantly from the true object motion. Thus, co-located blocks in the video and depth component may have different estimated motion vectors, such that the inter-component parameter prediction mode is not selected.

The common structure information in video and depth component is further used for specific depth modeling modes in a 3DV encoder, as described in the next subsection 8.4.5.

8.4.5 Depth Modeling

For the coding of depth maps, the special characteristic and purpose of this information has to be considered. As depth maps represent the 3D geometry information of recorded or generated 3D content in the form of distance information, they are mainly characterized by unstructured constant or slowly changing areas for scene objects. In addition, abrupt value changes can occur at object boundaries between foreground and background areas. Experiments with state-of-the-art compression technology have shown that such depth maps can be compressed very efficiently. In addition, sub-sampling to a lower resolution prior to encoding and decoder-side up-sampling similar to chrominance sub-sampling has also been studied with good results by Oh, Yea, Vetro and Ho (2009), as discussed in subsection 8.4.3. Since the purpose of depth maps is to provide scaled disparity information for texture data for view synthesis, coding methods have to be adapted accordingly. Especially sharp depth edges between foreground and background areas should be preserved during coding. A smoothing of such edges, as done by classical block-based coding methods, may lead to visible artifacts in intermediate views (Müller, Merkle and Wiegand, 2011). Furthermore, depth coding has to be optimized with respect to the quality of synthesized views, as the quality of the reconstructed depth data is irrelevant.

For a better preservation of edge information in depth maps, wedgelet (Merkle et al, 2009) and contour-based coding modes have been introduced. During encoding, each depth block is analyzed for significant edges. If such an edge is present, a block is subdivided into two non-rectangular partitions $\mathbf{P}_1$ and $\mathbf{P}_2$ as shown in Fig. 8.10.

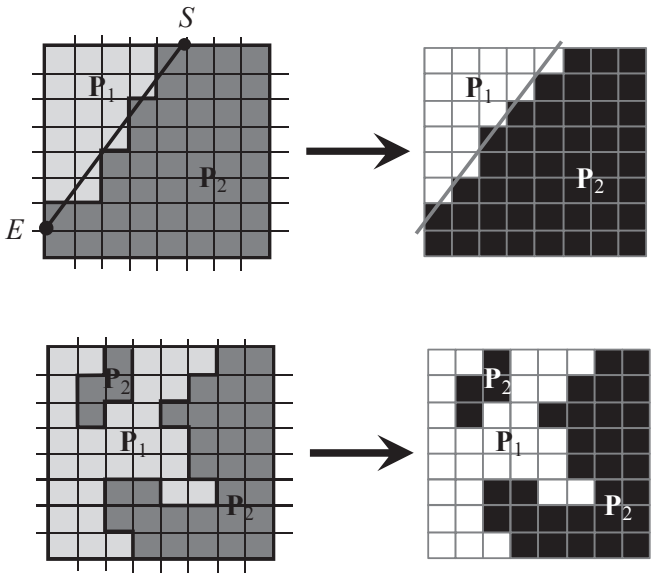

Fig. 8.10: Illustration of wedgelet partition (top) and contour partition (bottom) of a depth block: Original sample assignment to partitions P_1 and P_2 (left) and partition pattern (right). For the wedgelet partition, this pattern slightly differs from the original sample assignment due to the wedgelet approximation of the original contour.

The partitions can be separated by a straight line as an approximation of a rather regular depth edge in this block (see Fig. 8.10 top). Both partitions are then represented by a constant value. In addition to these values, the position of the separation line needs to be encoded. This is done in different ways: First, an explicit signaling is carried out, using a look-up table. This table contains all possible separation lines within a block in terms of position and orientation, and provides an index for them. Second, a separation line can also be derived from neighboring blocks, e.g. if an already coded neighboring block contains significant edge information which end at the common block boundary. Then, a continuation of this edge into the current block can be assumed. Accordingly, one or both end points of the separation line (*S* end *E* in Fig. 8.10, top left) can be derived from the already coded upper and left neighboring block, and thus don't need to be signaled. Third, the position of a separation line can be derived from the corresponding texture block. If a depth block contains a more complex separation between both partitions, as shown in Fig. 8.10 bottom, its contour can also be derived from the corresponding texture block. In the depth encoding process, either one of the described depth modeling modes with signaling of separation information and partition values, or a conventional intra prediction mode is selected (Merkle et al, 2012). Additional techniques on depth video coding can also be found in Chapter 7.

In addition to a good approximation of sharp depth edges by the new depth modeling modes, the rate-distortion optimization for the depth coding is adapted. As decoded depth data is used for view synthesis of the associated texture information, the distortion measure for the depth coding is obtained by comparing uncoded synthesized and reconstructed synthesized views. In this view synthesis optimization method, a block-wise processing aligned with depth data coding was introduced in order to provide a fast encoder operation (Tech, Schwarz, Müller and Wiegand, 2012). For an encoding optimization for one synthesized view, the method operates as follows: first, the uncoded synthesized

view is rendered from uncoded texture and depth data once per frame prior to encoding. Then, the synthesized reconstructed view is rendered separately for each block from decoded texture and decoded depth data. Furthermore, only those neighboring blocks are considered in the block-wise synthesis that influence the current depth block under encoding. Examples are neighboring blocks of foreground objects in original views that occlude the current block in the synthesized view. If more synthesized views are considered, the distortion measures of each single view are averaged. More information and test results of the view synthesis optimization method with different number of views have been investigated by Tech, Schwarz, Müller and Wiegand (2012).

8.4.6 Bit Allocation

When coding 3D video data in the MVD format, the video component of the base view is usually coded by classical methods, such as AVC or HEVC, for providing compatibility with existing 2D video coding standards. On top of that, new coding tools are used for the video component of dependent views and for the depth data, as described in the previous subsections. These tools provide additional coding efficiency such that the dependent views and depth maps require a much smaller portion of the overall bit rate, than the base view video data. An example for the bit rate distribution in percent for the individual components of an MVD format with two views *Cam* 1 and *Cam* 2 is shown in Fig. 8.11.

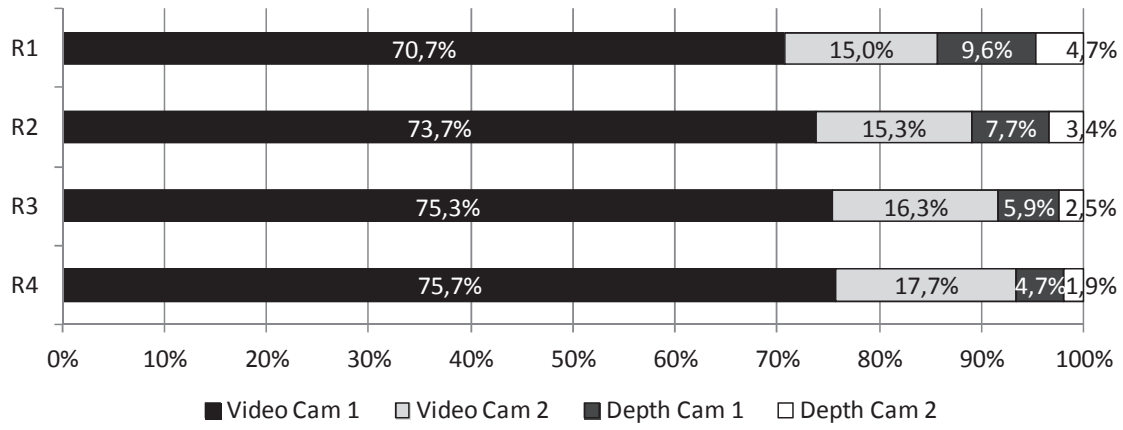

Fig. 8.11: Example for average bit rate distribution in percent of total rate over all test sets for the video and depth components in 2-view 3D-HEVC-based coding with views *Cam* 1 and *Cam* 2 for four different rate points R1-R4. Here, *Cam* 1 is the independent base view and receives the largest bit rate portion.

In this example, a 3D video codec was used that is based on HEVC 2D video coding as developed by Schwarz et al (2012). The results in Fig. 8.11 show bitrate distributions in percent at four rate points R1-R4, according to Table 8.2. For each rate point, the individual bit rate distributions of eight different sequences from the MPEG 3DV test set were averaged.

Fig. 8.11 first shows, that most of the bit rate is distributed to the video component of the independent base view of *Cam* 1. In particular, the base view video component receives 71% of the bit rate at R1 and 76% at R4. Accordingly, all other components only require 29% at R4 and 24% at R1. Therefore, efficient 3D video transmission of MVD data with two views and depth maps can be achieved at approximately

1.3-times the bitrate of a 2D video transmission. Furthermore, Fig. 8.11 shows that most of the bit rate is distributed to the video data. Here, the video/depth rate distribution varies from 86%/14% at the lowest rate point R1 to 93%/7% at the highest rate point R4 on average. Thus, depth data can be coded very efficiently. A comparison between MVD and pure stereo video coding showed that the perceived video quality in both cases is almost identical, even though MVD additionally provides depth data at the decoder for high quality view synthesis at the 3D display.

8.5 Experimental Evaluation

The main objective of a 3D video coding technology is to provide high compression efficiency for a generic format that can be used to synthesize any number of views for a variety of stereoscopic and auto-stereoscopic multiview displays. For the evaluation of suitable compression methods, the quality of these synthesized views need to be assessed. However, no original reference views exist for newly synthesized positions, such that classical objective comparison methods e.g. mean squared error (MSE) between decoded synthesized views and original reference cannot be applied directly. In this section, the evaluation framework that was used to assess the compression efficiency for depth-based formats is described, followed by a summary of results from a large-scale quality assessment process.

8.5.1 Evaluation Framework

As shown in section 8.4.5 for the encoding of depth maps, a pseudo-reference can be created by synthesizing intermediate views from uncoded data and thus also applying error measures like MSE. This method however neglects synthesis errors that can occur due to erroneous depth maps and thus a high quality pseudo-reference has to be assumed. Even in cases where original views at intermediate positions might be available, measures like the pixel-wise MSE and derived classical PSNR are unsuitable: Consider a perfectly synthesized view that is shifted by one pixel in any direction. Then, the PSNR value for this view would be very low and thus doesn't relate to the high subjective quality.

Therefore, a large-scale subjective evaluation has to be carried out for assessing 3D video coding methods, where participants judge the quality of coding methods subjectively in test sessions by viewing the reconstructed views on different 3D displays. For the quality evaluation, the mean opinion score (MOS) is used, which provides a quality scale from 0 (very bad) to 10 (very good). The individual MOS values from many test participants are then averaged for each tested sequence. This method was applied in the ISO-MPEG Call for Proposals (CfP) for 3D video technology (ISO/IEC JTC1/SC29/WG11, 2011a) in 2011. For this call, a number of test parameters were considered. First, two test categories were specified: AVC-compatible and HEVC-compatible/unconstrained for testing the different 3D video coding proposals, based on the respective 2D video coding technology for the base view. Next, the 3D video test material was created from 8 sequences with multiview video and depth components. Four of these sequences had a progressive HD resolution of 1920 x 1088 at 25 fps (1920x1088p@25fps) and four had a progressive resolution of 1024 x 768 at 30 fps (1024x768p@30fps), as listed in Table 8.1 and Table 8.2. For all test sequences, four rate points were specified, as explained in subsection 8.5.2 for AVC-based technologies and in subsection 8.5.3 for HEVC-based/unconstrained technologies.

All 8 sequences were evaluated in two test scenarios: In the 2-view scenario, video and depth components of 2 views {V0,V1} were coded and a stereo pair with one original and one intermediate viewing position reconstructed and synthesized as shown in Fig. 8.12. This stereo pair was evaluated on a stereoscopic display. In the 3-view scenario, video and depth components of 3 views {V0,V1,V2} were coded and different types of video data extracted. Then, a dense range of 28 views was synthesized and viewed on an auto-stereoscopic 28-view display as shown in Fig. 8.13 and used for ISO/IEC JTC1/SC29/WG11 (2011a). For additional assessment, a central stereo pair in the middle of the 3-view range and a random stereo pair within the viewing range were synthesized for viewing on a stereoscopic display. All results were evaluated in large-scale subjective tests with 13 international test laboratories involved (ISO/IEC JTC1/SC29/WG11, 2011b).

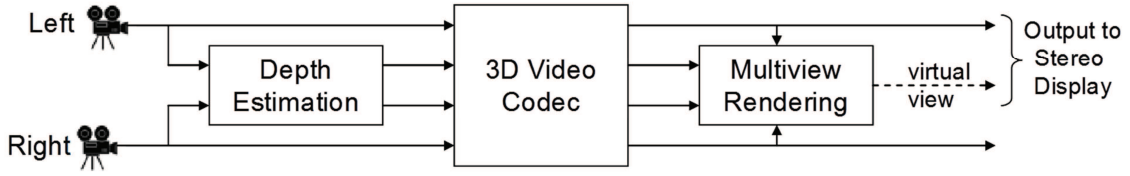

Fig. 8.12: Advanced stereoscopic processing with 2-view configuration.

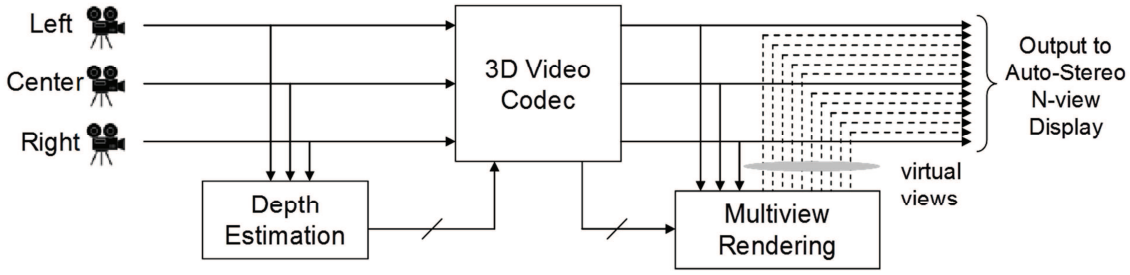

Fig. 8.13: Auto-stereoscopic output with 3-view configuration.

8.5.2 AVC-based 3D video coding results

For 3D video coding technology, that uses the AVC standard for coding the base view video, four different rate points R1-R4 (from low to high bit rate) were specified for the 2-view and 3-view scenario and each test sequence in an iterative process. For finding these rate points, each sequence with video and depth component was coded at four different initial rate points. For this, the video components where jointly coded, using the MVC reference software JMVC 8.3.1. The same coding was also applied to the depth components. Then, stereo pairs, as well as the set of 28 views for both display types were generated (ISO/IEC JTC1/SC29/WG11, 2011a). The results were subjectively assessed with the target to obtain noticeable differences between the four rate points as well as in comparison to the uncoded synthesized data. If different rate points could not be distinguished by subjective viewing, the associated bit rates for the sequence were adapted. Finally, the bit rates for each sequence and rate point were fixed to the values, shown in Table 8.1.

Test Sequence (Resolution, Frame Rate)	2-view test scenario				3-view test scenario			
	AVC bit rates (kbps)				AVC bit rates (kbps)			
	R1	R2	R3	R4	R1	R2	R3	R4
S01: Poznan_Hall2 (1920x1088p@25fps)	500	700	1000	1500	750	900	1300	2300
S02: Poznan_Street (1920x1088p@25fps)	500	700	1000	1250	750	1100	1800	4000
S03: Undo_Dancer (1920x1088p@25fps)	1000	1300	1700	2200	1380	1750	2300	2900
S04: GT_Fly (1920x1088p@25fps)	1200	1700	2100	2900	2000	2380	2900	4000
S05: Kendo (1024x768p@30fps)	400	500	800	1300	800	1000	1300	1900
S06: Balloons (1024x768p@30fps)	320	430	600	940	500	600	800	1250
S07: Lovebird1 (1024x768p@30fps)	375	500	750	1250	500	800	1250	2000
S08: Newspaper (1024x768p@30fps)	400	525	800	1300	500	700	1000	1350

Table 8.1: AVC-based rate points R1-R4 for 2-View and 3-View Test Scenario for AVC-based 3D Video Technology.

For the AVC-based category, 12 proposals were submitted to the CfP and subjectively tested (ISO/IEC JTC1/SC29/WG11, 2011b). Proponents had to encoded the 2-view and 3-view MVD format with video and depth data at the four different bit rates for each sequence according to Table 8.1 and generate the required stereo pairs and the set of 28 views for both display types respectively. Since all proposals as well as the coded anchors had the same bit rate, an MOS-based subjective quality comparison at each bit rate could be performed. A summary of the achieved quality improvement of the best performing proposal in comparison to the anchor coding is shown in Fig. 8.14.

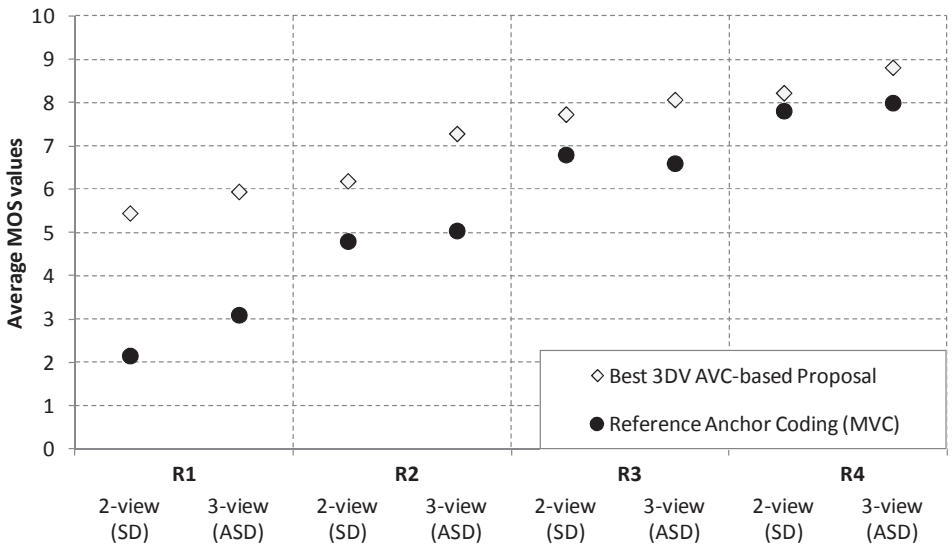

Fig. 8.14: 3D-AVC-based subjective results: Averaged MOS scores over all eight test sequences at four different bit rates R1-R4 according to Table 8.1: Evaluation of 2-view scenario on stereoscopic display (SD), evaluation of 3-view scenario on auto-stereoscopic 28-view display (ASD).

Here, the viewing results of the 2-view scenario on a stereoscopic display with polarized glasses (SD) as well as results of the 3-view scenario on an auto-stereoscopic 28-view display (ASD) are given for each rate point. Note, that each point in Fig. 8.14 represents the average MOS value over all 8 sequences at that rate point and display type. The best 3DV AVC-based proposal outperforms the anchor coding

results; especially at the low rate point R1. In addition, the best proposal achieves a similar or better MOS value at the next lower bit rate in comparison to the anchors, e.g. the MOS for the best proposal at R1 is better than the anchor coding MOS at R2 for both display types. Comparing the associated individual bit rates in Table 8.1 of each sequence, an overall bit rate saving of 30% could be obtained in comparison to the anchor coding.

Comparing the 2- and 3-view scenario in Fig. 8.14, the subjective MOS results for each rate point are in the same quality range and thus consistent. Since two very different 3D displays were used for the viewing, the results also prove the suitability of the proposed codec for a targeted display-independent reconstruction quality based on view synthesis from a decoded generic 3D video format.

8.5.3 HEVC-based 3D video coding results

For 3D video coding technology, that uses the HEVC standard for coding the base view video, also four different rate points were specified for the 2-view and 3-view scenario and for each test sequence, using the same iterative procedure as described in the previous subsection. For the HEVC-based/ unconstrained category, the anchors were produced by coding each video and depth component separately with the HEVC test Model HM2.0. Then, stereo pairs, as well as the set of 28 views for both display types were generated similarly to the AVC-based category (ISO/IEC JTC1/SC29/WG11, 2011a). In Table 8.2, the final bit rates for the compressed MVD data at rate points R1-R4 for all sequences are shown. Note, that the HEVC-based bit rates in Table 8.2 are considerable lower than the AVC-based bit rates in Table 8.1 due to the higher compression efficiency of HEVC.

Test Sequence (Resolution, Frame Rate)	2-view test scenario				3-view test scenario			
	HEVC bit rates (kbps)				HEVC bit rates (kbps)			
	R1	R2	R3	R4	R1	R2	R3	R4
S01: Poznan_Hall2 (1920x1088p@25fps)	140	210	320	520	210	310	480	770
S02: Poznan_Street(1920x1088p@25fps)	280	480	800	1310	410	710	1180	1950
S03: Undo_Dancer (1920x1088p@25fps)	290	430	710	1000	430	780	1200	2010
S04: GT_Fly (1920x1088p@25fps)	230	400	730	1100	340	600	1080	1600
S05: Kendo (1024x768p@30fps)	230	360	480	690	280	430	670	1040
S06: Balloons (1024x768p@30fps)	250	350	520	800	300	480	770	1200
S07: Lovebird1 (1024x768p@30fps)	220	300	480	830	260	420	730	1270
S08: Newspaper (1024x768p@30fps)	230	360	480	720	340	450	680	900

Table 8.2: HEVC-based rate points R1-R4 for 2-View and 3-View Test Scenario for HEVC-based 3D Video Technology.

For the HEVC-based/unconstrained category, 11 proposals were submitted to the CfP and subjectively tested (ISO/IEC JTC1/SC29/WG11, 2011b). All proposals used HEVC for coding the base view. Proponents had to encode the 2-view and 3-view MVD format with video and depth data at the four different bit rates for each sequence according to Table 8.2 and generate the required stereo pairs and the set of 28 views for both display types respectively. Similar to the AVC-based category described in the previous subsection, all proposals as well as the coded anchors had the same bit rate, such that an MOS-based subjective quality comparison at each bit rate could be performed as well. A summary of the achieved quality improvement of the best performing proposal in comparison to the anchor coding is shown in Fig. 8.15.

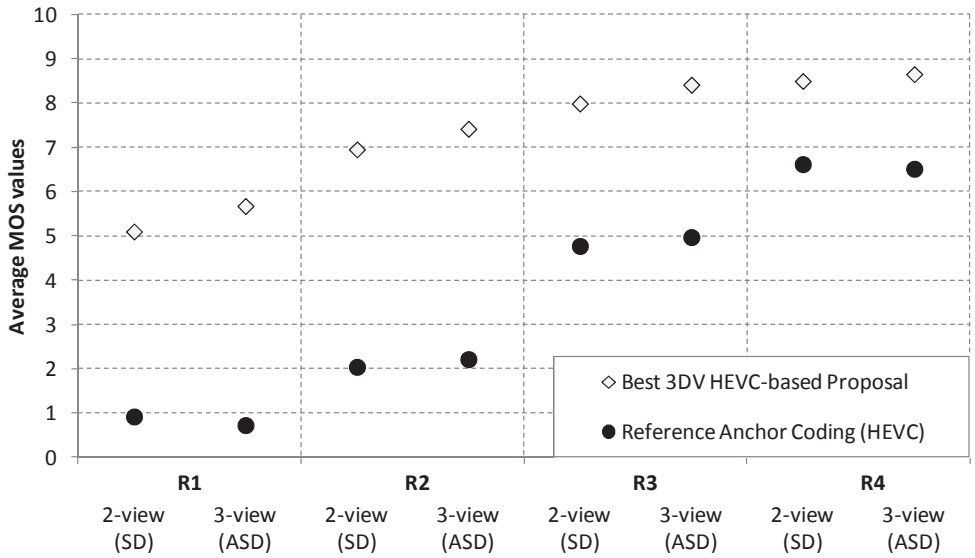

Fig. 8.15: 3D-HEVC-based subjective results: Averaged MOS scores over all eight test sequences at four different bit rates R1-R4 according to Table 8.2: Evaluation of 2-view scenario on stereoscopic display (SD), evaluation of 3-view scenario on auto-stereoscopic 28-view display (ASD).

Again, the viewing results of the 2-view scenario on a stereoscopic display with polarized glasses (SD) as well as the results of the 3-view scenario on an auto-stereoscopic 28-view display (ASD) are given for each rate point as the average MOS value over all 8 sequences. The best 3D-HEVC-based proposal significantly outperforms the anchor coding in both settings, such that e.g. a similar subjective quality of 5 for the anchor coding at R3 is already achieved by the best proposal at R1. Comparing the associated individual bit rates of each sequence at R3 and R1, as given in Table 8.2, average bit rate savings of 57% and 61% for the 2- and 3-view scenario respectively are achieved. These are significantly higher than the bit rate savings for the 3D-AVC-based technology for two reasons: First, the HEVC anchors were produced with single view coding and thus no inter-view prediction was applied in contrast to the MVC-based anchor coding. Second, a smaller random access period with a group of pictures of 8 (GOP8) was used for the HEVC anchors, while the 3D-HEVC-based proposals used a larger period of GOP12 and GOP15. Accordingly, the overall bit rate savings in the HEVC-based category also include the savings for the different random access periods as well as for the inter-view prediction. A more detailed analysis of individual results by Schwarz et al (2012) showed bit rate savings of 38% and 50% for the 2- and 3-view scenario in comparison to HEVC simulcast with equal random access periods. Furthermore, bit rate savings of 20% and 23% for the 2- and 3-view scenario in comparison to multiview HEVC were found.

Comparing the 2-view and 3-view scenario in Fig. 8.15, the subjective MOS results for each rate point are in the same quality range. Thus, also the 3D-HEVC-based coding technology provides consistent results across very different 3D displays and thus a display-independent high-quality view synthesis from a generic 3D video format.

8.5.4 General observations

The subjective evaluation of new 3D video coding technology showed significant improvements over the anchor coding methods in both AVC-based and HEVC-based categories. In particular, a bit rate saving of 30% at the same subjective quality could be achieved for the 3D-AVC-based technology. For 3D-HEVC coding, bit rate savings of more than 57% were achieved in comparison to the simulcast HEVC-anchors and still more than 20% vs. a multiview HEVC-based anchor. This latter anchor has the same properties, as the MVC-anchor in terms of random access period and usage of inter-view prediction.

The obtained results showed that a higher coding efficiency was achieved in both categories by optimizing existing coding tools and adding new methods, as described in section 8.4. Especially an improved inter-view prediction, new methods of inter-component parameter prediction, special depth coding modes, and an encoder optimization for depth data coding towards the synthesized views were applied for optimally encoding 3DV data and synthesizing multiview video data for different 3D displays from the decoded bit stream.

8.6 Concluding Remarks

The video coding standardization committees are moving quickly to define a new set of 3D video formats. A major feature of these new formats will be the inclusion of depth data to facilitate the generation of novel viewpoints of a 3D scene. This chapter introduced the depth based representation and rendering techniques, and described a number of different coding architectures that are being considered. The leading compression technologies to support the efficient representation and rendering of the depth-based 3D formats have also been described, and the results from a large-scale experimental evaluation have been summarized.

While the current 3D services are based only on stereoscopic video, the market is expected to evolve and auto-stereoscopic displays will soon mature and become cost-effective. The depth-based formats and associated compression techniques described in this chapter will support such future displays and services. It is believed that a low-complexity process to generate multiple views at the receiver with high rendering quality will be an essential feature for auto-stereoscopic displays and related equipment.

Although the standardization of these formats is well underway, there are still a number of research challenges to overcome. For instance, the accurate estimation or acquisition of depth information is not considered to be fully mature at this stage, especially for real-time applications and outdoor scenes in which the depth range is large, and illumination (or other factors) could have a significant impact on the quality of the depth image. Also, while current standards will be based on the MVD format and somewhat conventional coding architectures, further research on alternative representations continues to be of interest among researchers working in this area. For instance, there have been investigations on using the transition between views as a representation of the 3D scene by Kim, Ortega, Lee and Wey (2010), as well as work on dictionary-based representations of a 3D scene by Palaz, Tosic and Frossard (2011). With these new representations formats and evaluation frameworks that consider the quality of the rendered view, there is also scope for further work on modeling and optimizing the quality of the system, e.g., as discussed by Kim et al (2010); and Tech, Schwarz, Müller and Wiegand (2012a; 2012b).

References:

Atzpadin, N., Kauff, P. and Schreer, O. (2004) 'Stereo Analysis by Hybrid Recursive Matching for Real-Time Immersive Video Conferencing', *IEEE Transactions on Circuits and Systems for Video Technology, Special Issue on Immersive Telecommunications*, vol. 14, no. 3, March, pp. 321-334.

Beni, P. A., Rusanovskyy, D. and Hannuksela M. M. (2012), 'Non-linear Depth Map Resampling for 3DV-ATM Coding', *ISO/IEC JTC1/SC29/WG11*, Doc. m23721, February, San Jose, CA, USA.

Benzie, P., Watson, J., Surman, P., Rakkolainen, I., Hopf, K., Urey, H., Sainov, V. and Kopylow, C. v. (2007), 'A Survey of 3DTV Displays: Techniques and Technologies', *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 17, no. 11, November, pp. 1647-1658.

Bleyer, M. and Gelautz, M. (2005), 'A layered stereo matching algorithm using image segmentation and global visibility constraints', *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 59, no. 3, pp. 128–150.

Blu-ray Disc Association (2009), *Blu-ray Disc Association Announces Final 3D Specification*, <http://www.blu-raydisc.com/assets/Downloadablefile/BDA-3D-Specifcation-Press-Release---Proposed-Final12-14-08version-clean-16840.pdf>

Cigla, C., Zabulis, X. and Alatan, A. A. (2007), 'Region-based dense depth extraction from multiview video', In: *Proceedings of the IEEE International Conference on Image Processing (ICIP'07),* San Antonio, TX, USA, September, pp. 213–216.

Digital Video Broadcast (2012), *3D Moves Forward: DVB Steering Board Approves Phase 2a Of 3DTV Specification*, <http://www.dvb.org/news_events/press_releases/press_releases/DVB_pr227-Steering-Board-Approves-Phase-2a-3D-Specification.pdf>

Faugeras, O. (1993) *Three-dimensional computer vision: A geometric viewpoint*, Cambridge, Massachusetts, MIT Press.

Felzenszwalb, P. F. and Huttenlocher, D. P. (2006), 'Efficient Belief Propagation for Early Vision', *International Journal of Computer Vision*, vol. 70, no. 1, October, pp. 41-54.

Hartley, R. and Zisserman, A. (2000) *Multiple View Geometry in Computer Vision*, Cambrigde Universitity Press.

ITU-T and ISO/IEC (2010), 'Advanced video coding for generic audiovisual services', *ITU-T Rec. H.264 and ISO/IEC 14496-10 (MPEG-4 AVC), Version 10*, March 2010.

ISO/IEC JTC1/SC29/WG11 (2011a), 'Call for Proposals on 3D Video Coding Technology', *Doc. N12036, Geneva, CH*, March 2011.

ISO/IEC JTC1/SC29/WG11 (2011b), 'Report of Subjective Test Results from the Call for Proposals on 3D Video Coding', *Doc. N12347, Geneva, CH*, Nov. 2011.

ISO/IEC JTC1/SC29/WG11 (1997), 'The Virtual Reality Modeling Language', *DIS 14772-1,* April 1997.

Kauff, P., Atzpadin, N., Fehn, C., Müller, M., Schreer, O., Smolic, A. and Tanger, R. (2007), 'Depth Map Creation and Image Based Rendering for Advanced 3DTV Services Providing Interoperability and Scalability', *Signal Processing: Image Communication. Special Issue on 3DTV*, vol. 22, no. 2, February, pp. 217-234.

Kim, W.-S., Ortega, A., Lai, P., Tian, D. and Gomila, C. (2010a), 'Depth map coding with distortion estimation of rendered view', *Visual Information Processing and Communication*, *Proceedings of the SPIE*, vol. 7543, pp. 75430B-75430B-10.

Kim, W.-S., Ortega, A., Lee, J. and Wey, H. (2010b), '3-D video coding using depth transition data', In: *Proceedings of the Picture Coding Symposium (PCS'2010)*, Nagoya, Japan, pp. 178-181.

Kolmogorov, V. (2006), 'Convergent Tree-reweighted Message Passing for Energy Minimization', *IEEE*

Transactions on Pattern Analysis and Machine Intelligence, vol. 28, no. 10, October, pp. 1568 - 1583.

Kolmogorov, V. and Zabih, R. (2002), 'Multi-Camera scene Reconstruction via Graph Cuts', In: *Proceedings of the European Conference on Computer Vision (ECCV'2002) Part III*, May, pp. 82-96.

Konrad, J. and Halle, M. (2007), '3-D Displays and Signal Processing', *IEEE Signal Processing Magazine*, vol. 24, no. 6, November, pp. 97-111.

Lee, E.-K., Jung, Y.-K. and Ho, Y.-S. (2010), '3-D Video Generation Using Foreground Separation and Disocclusion Detection', In: *Proceedings of the IEEE 3DTV Conference (3DTV CON'2010)*, June, Tampere, Finland.

Lee, S.-B. and Ho, Y.-S. (2010), 'View Consistent Multiview Depth Estimation for Three-dimensional Video Generation', In: *Proceedings of the IEEE 3DTV Conference (3DTV CON'2010)*, June, Tampere, Finland.

Lee, S., Lee, S., Wey, H. and Lee, J. (2012), '3D-AVC-CE6 related results on Samsung's in-loop depth resampling', *ISO/IEC JTC1/SC29/WG11*, Doc. m23661, February, San Jose, CA, USA.

Martinian, E., Behrens, A., Xin, J. and Vetro, A. (2006), 'View synthesis for multiview video compression', In: *Proceedings of the Picture Coding Symposium (PCS'2006)*, April, Beijing, China.

Merkle, P., Bartnik, C., Müller, K., Marpe, D. and Wiegand, T. (2012), '3D Video: Depth Coding Based on Inter-component Prediction of Block Partitions', In: *Proceedings of the Picture Coding Symposium (PCS'2012)*, May, Krakow, Poland.

Merkle, P., Morvan, Y., Smolic, A., Farin, D., Müller, K., de With, P.H.N. and Wiegand, T. (2009), 'The Effects of Multiview Depth Video Compression on Multiview Rendering', *Signal Processing: Image Communication*, vol. 24, no. 1+2, January, pp. 73-88.

Merkle, P., Smolic, A., Müller, K. and Wiegand, T. (2007), 'Efficient Prediction Structures for Multiview Video Coding', invited paper, *IEEE Trans. on Circuits and Systems for Video Technology*, vol. 17, no. 11, November, pp. 1461-1473.

Min, D., Yea, S. and Vetro, A. (2010), 'Temporally Consistent Stereo Matching Using Coherence Function', In: *Proceedings of the IEEE 3DTV Conference (3DTV CON'2010)*, June, Tampere, Finland.

Müller, K., Merkle, P. and Wiegand, T. (2011), '3D Video Representation Using Depth Maps', *Proceedings of the IEEE, Special Issue on 3D Media and Displays*, vol. 99, no. 4, April, pp. 643 - 656.

Müller, K., Smolic, A., Dix, K., Merkle, P., Kauff, P. and Wiegand, T. (2008), 'View Synthesis for Advanced 3D Video Systems', *EURASIP Journal on Image and Video Processing, Special Issue on 3D Image and Video Processing*, vol. 2008, pp. 1-11, Article ID 438148, doi:10.1155/2008/438148.

Oh, K.-J., Yea, S., Vetro, A. and Ho Y.-S. (2009), "Depth Reconstruction Filter and Down/Up Sampling for Depth Coding in 3-D Video", *IEEE Signal Processing Letters*, vol. 16, no. 9, September, pp. 747-750.

Ozaktas, H. M. and Onural L. (Eds.) (2007) *Three-Dimensional Television: Capture, Transmission, Display*, Heidelberg, Springer.

Palaz, D., Tosic, I. and Frossard, P. (2011), 'Sparse stereo image coding with learned dictionaries', In: *Proceedings of the IEEE International Conference on Image Processing (ICIP'2011),* September, Brussels, Belgium, pp. 133-136.

Redert, A., de Beeck, M.O., Fehn, C., Ijsselsteijn, W., Pollefeys, M., Van Gool, L., Ofek, E., Sexton, I. and Surman, P. (2002), 'ATTEST–advanced three-dimensional television system techniques', IN: *Proceedings of the International Symposium on 3D Data Processing, Visualization and Transmission (3DPVT'2002)*, Padova, Italy, June, pp. 313-319.

Rusert, T. (2012), '3D-CE3 summary report: in-loop depth resampling', *ISO/IEC JTC1/SC29/WG11*, Doc.

m24823, May, Geneva, Switzerland.
Scharstein, D. and Szeliski, R. (2002), 'A taxonomy and evaluation of dense two-frame stereo correspondence algorithms', *International Journal of Computer Vision*, vol. 47, no. 1, May, pp. 7-42.
Schwarz, H., Bartnik, C., Bosse, S., Brust, H., Hinz, T., Lakshman, H., Marpe, D., Merkle, P., Müller, K., Rhee, H., Tech, G., Winken, M. and Wiegand, T., '3D Video Coding Using Advanced Prediction, Depth Modeling, and Encoder Control Methods', In: *Proceedings of the Picture Coding Symposium (PCS'2012)*, May, Krakow, Poland.
Szeliski, R., Zabih, R., Scharstein, D., Veksler, O., Kolmogorov, V., Agarwala, A., Tappen, M. and Rother, C. (2006), 'A Comparative Study of Energy Minimization Methods for Markov Random Fields', In: *Proceedings of the European Conference on Computer Vision (ECCV 2006)*, vol. 2, May, Graz, Austria, pp. 16-29.
Tanimoto, M., Fujii, T. and Suzuki, K. (2008), "Improvement of depth map estimation and view synthesis', *ISO/IEC JTC1/SC29/WG11*, Doc. m15090, January, Antalya, Turkey.
Tech, G., Schwarz, H., Müller, K. and Wiegand, T (2012a), '3D Video Coding using the Synthesized View Distortion Change', In: *Proceedings of the Picture Coding Symposium (PCS'2012)*, May, Krakow, Poland.
Tech, G., Schwarz, H., Müller, K. and Wiegand, T (2012b), 'Effects of synthesized View Distortion based 3D Video Coding on the Quality of interpolated and extrapolated Views', In: *Proceedings of the IEEE International Conference on Multimedia and Exposition (ICME'2012)*, July, Melbourne, Australia.
Vetro, A., Tourapis, A., Müller, K. and Chen, T. (2011), '3D-TV Content Storage and Transmission', *IEEE Transactions on Broadcasting, Special Issue on 3D-TV Horizon: Contents, Systems and Visual Perception*, vol. 57, no. 2, June, pp. 384-394.
Vetro, A., Wiegand, T. and Sullivan, G. J. (2011), 'Overview of the Stereo and Multiview Video Coding Extensions of the H.264/AVC Standard', *Proceedings of the IEEE, Special Issue on 3D Media and Displays*, vol. 99, no. 4, April, pp. 626 - 642.
Wiegand, T., Sullivan, G. J., Bjøntegaard, G. and Luthra, A. (2003), 'Overview of the H.264/AVC video coding standard', *IEEE Trans. on Circuits and Systems for Video Technology*, vol. 13, no. 7, July, pp. 560-576.
Winken, M., Schwarz, H. and Wiegand, T. (2012), 'Motion Vector Inheritance for High Efficiency 3D Video plus Depth Coding', Proc. PCS 2012, In: *Proceedings of the Picture Coding Symposium (PCS'2012)*, May, Krakow, Poland.
Würmlin, S., Lamboray, E. and Gross, M. (2004), '3D Video Fragments: Dynamic Point Samples for Real-Time Free-Viewpoint Video', *Computers and Graphics, Special Issue on Coding, Compression and Streaming Techniques for 3D and Multimedia Data*, Elsevier, pp. 3–14.
Yea, S. and Vetro, A. (2009), 'View Synthesis Prediction for Multiview Video Coding', *Signal Processing: Image Communication*, vol. 24, no. 1+2, January, pp. 89-100.