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

Overview of the Stereo and Multiview Video Coding Extensions of the H.264/MPEG-4 AVC Standard

Vetro, A.; Wiegand, T.; Sullivan G.J.

TR2011-022 January 2011

Abstract

Significant improvements in video compression capability have been demonstrated with the introduction of the H.264/MPEG-4 Advanced Video Coding (AVC) standard. Since developing this standard, the Joint Video Team of the ITU-T Video Coding Experts Group (VCEG) and the ISO/IEC Moving Picture Experts Group (MPEG) has also standardized an extension of that technology that is referred to as multiview video coding (MVC). MVC provides a compact representation for multiple views of a video scene, such as multiple synchronized video cameras. Stereo-paired video for 3D viewing is an important special case of MVC. The standard enables inter-view prediction to improve compression capability, as well as supporting ordinary temporal and spatial prediction. It also supports backward compatibility with existing legacy systems by structuring the MVC bitstream to include a compatible ”base view”. Each other view is encoded at the same picture resolution as the base view. In recognition of its high quality encoding capability and support for backward compatibility, the Stereo High profile of the MVC extension was selected by the Blu-Ray Disc Association as the coding format for 3D video with high-definition resolution. This paper provides an overview of the algorithmic design used for extending H.264/MPEG-4 AVC towards MVC. The basic approach of MVC for enabling inter-view prediction and view scalability in the context of H.264/MPEG-4 AVC is reviewed. Related supplemental enhancement information (SEI) metadata is also described. Various ”frame compatible” approaches for support of stereo-view video as an alternative to MVC are also discussed. A summary of the coding performance achieved by MVC for both stereo and multiview video is also provided. Future directions and challenges related to 3D video are also briefly discussed.

Proceedings of the IEEE

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.

Copyright © Mitsubishi Electric Research Laboratories, Inc., 2011
201 Broadway, Cambridge, Massachusetts 02139

Overview of the Stereo and Multiview Video Coding Extensions of the H.264/MPEG-4 AVC Standard

ANTHONY VETRO, FELLOW, IEEE, THOMAS WIEGAND, FELLOW, IEEE, AND GARY J. SULLIVAN, FELLOW, IEEE

***Abstract*—Significant improvements in video compression capability have been demonstrated with the introduction of the H.264/MPEG-4 Advanced Video Coding (AVC) standard. Since developing this standard, the Joint Video Team of the ITU-T Video Coding Experts Group (VCEG) and the ISO/IEC Moving Picture Experts Group (MPEG) has also standardized an extension of that technology that is referred to as multiview video coding (MVC). MVC provides a compact representation for multiple views of a video scene, such as multiple synchronized video cameras. Stereo-paired video for 3D viewing is an important special case of MVC. The standard enables inter-view prediction to improve compression capability, as well as supporting ordinary temporal and spatial prediction. It also supports backward compatibility with existing legacy systems by structuring the MVC bitstream to include a compatible "base view". Each other view is encoded at the same picture resolution as the base view. In recognition of its high quality encoding capability and support for backward compatibility, the Stereo High profile of the MVC extension was selected by the Blu-Ray Disc Association as the coding format for 3D video with high-definition resolution. This paper provides an overview of the algorithmic design used for extending H.264/MPEG-4 AVC towards MVC. The basic approach of MVC for enabling inter-view prediction and view scalability in the context of H.264/MPEG-4 AVC is reviewed. Related supplemental enhancement information (SEI) metadata is also described. Various "frame compatible" approaches for support of stereo-view video as an alternative to MVC are also discussed. A summary of the coding performance achieved by MVC for both stereo and multiview video is also provided. Future directions and challenges related to 3D video are also briefly discussed.**

***Index Terms*—MVC, H.264, MPEG-4, AVC, standards, stereo video, multiview video coding, inter-view prediction, 3D video, Blu-ray Disc**

I. INTRODUCTION

3D VIDEO is currently being introduced to the home through various channels, including Blu-ray Disc, cable and satellite transmission, terrestrial broadcast, and streaming and download through the Internet. Today’s 3D video offers a high-quality and immersive multimedia experience, which has only recently become feasible on consumer electronics platforms through advances in display technology, signal processing, transmission technology, and circuit design.

In addition to advances on the display and receiver side, there has also been a notable increase in the production of 3D content. The number of 3D feature film releases has been growing dramatically each year, and several major studios have announced that all of their future releases will be in 3D. There are major investments being made to upgrade digital cinema theaters with 3D capabilities, several major feature film releases have attracted a majority of their theater revenue in 3D showings (including *Avatar*, the current top grossing feature film of all time[1]), and premium pricing for 3D has become a significant factor in the cinema revenue model. The push from both the production and display sides has played a significant role in fuelling a consumer appetite for 3D video.

There are a number of challenges to overcome in making 3D video for consumer use in the home become fully practical and show sustained market value for the long term. For one, the usability and consumer acceptance of 3D viewing technology will be critical. In particular, mass consumer acceptance of the special eyewear needed to view 3D in the home with current display technology is still relatively unknown. In general, content creators, service providers and display manufacturers need to ensure that the consumer has a high quality experience and is not burdened with high transition costs or turned off by viewing discomfort or fatigue. The availability of premium 3D content in the home is another major factor to be considered. These are broader issues that will significantly influence the rate of 3D adoption and market size, but are beyond the scope of this paper.

With regard to the delivery of 3D video, it is essential to determine an appropriate data format, taking into consideration the constraints imposed by each delivery channel – including bit rate and compatibility requirements. Needless to say, interoperability through the delivery chain and among various devices will be essential. The 3D representation, compression formats, and signaling protocols will largely define the interoperability of the system.

For purposes of this paper, 3D video is considered to refer to either a general *n*-view multiview video representation or its

Manuscript received April 1, 2010.
Revised version submitted MM DD, 2010.

A. Vetro is with Mitsubishi Electric Research Labs, Cambridge, MA, 02139 USA (email: avetro@merl.com).

T. Wiegand is jointly affiliated with the Berlin Institute of technology and the Fraunhofer Institute for Telecommunications – Heinrich Hertz Institute (HHI), Einsteinufer 37, 10587 Berlin, Germany (email: wiegand@hhi.de).

G. J. Sullivan is with Microsoft Corporation, Redmond, WA, 98052 USA (email: garys@ieee.org).

[1] Based on total revenue without inflation adjustments.

important stereo-view special case. Efficient compression of such data is the primary subject of this paper. The paper also discusses stereo representation formats that could be coded using existing 2D video coding methods – such approaches often being referred to as *frame-compatible* encoding schemes.

Multiview video coding (MVC) is the process by which stereo and multiview video signals are efficiently coded. The basic approach of most MVC schemes is to exploit not only the redundancies that exist temporally between the frames within a given view, but also the similarities between frames of neighboring views. By doing so, a reduction in bit rate relative to independent coding of the views can be achieved without sacrificing the reconstructed video quality. In this paper, the term MVC is used interchangeably for either the general concept of coding multiview video or for the particular design that has been standardized as a recent extension of the H.264/MPEG-4 AVC standard [1].

The topic of multiview video coding has been an active research area for more than 20 years, with early work on disparity-compensated prediction by Lukacs first appearing in 1986 [2], followed by other coding schemes in the late 1980's and early 1990's [3][4]. In 1996, the international video coding standard H.262/MPEG-2 Video [5] was amended to support the coding of multiview video by means of design features originally intended for temporal scalability [6][7]. However, the multiview extension of H.262/MPEG-2 Video was never deployed in actual products. It was not the right time to introduce 3D video into the market since the more fundamental transition from standard-definition analog to high-definition digital video services was a large challenge in itself. Adequate display technology and hardware processing capabilities were also lacking at the time. In addition to this, the H.262/MPEG-2 Video solution did not offer a very compelling compression improvement due to limitations in the coding tools enabled for inter-view prediction in that design [8]-[10].

This paper focuses on the MVC extension of the H.264/MPEG-4 AVC standard. Relevant supplemental enhancement information (SEI) metadata and alternative approaches to enabling multiview services are also discussed. The paper is organized as follows. Section II explains the various multiview video applications of MVC as well as their implications in terms of requirements. Section III gives the history of MVC, including prior standardization action. Section IV briefly reviews basic design concepts of H.264/MPEG-4 AVC. The MVC design is summarized in Section V, including profile definitions and a summary of coding performance. Alternative stereo representation formats and their signaling in the H.264/MPEG-4 AVC standard are described in Section VI. Concluding remarks are given in Section VII. For more detailed information about MVC and stereo support in the H.264/MPEG-4 AVC standard, the reader is referred to the most recent edition of the standard itself [1], the amendment completed in July 2008 that added the MVC extension to it [11], and the additional amendment completed one year later that added the Stereo High profile and frame packing arrangement SEI message [12].

II. MULTIVIEW SCENARIOS, APPLICATIONS, AND REQUIREMENTS

The prediction structures and coding schemes presented in this paper have been developed and investigated in the context of the MPEG, and later JVT, standardization project for MVC. Therefore, most of the scenarios for multiview coding, applications and their requirements are specified by the MVC project [13] as presented in the next sections.

A. *Multiview Scenarios and Applications*

The primary usage scenario for multiview video is to support 3D video applications, where 3D depth perception of a visual scene is provided by a 3D display system. There are many types of 3D display systems [14] including classic stereo systems that require special-purpose glasses to more sophisticated multiview auto-stereoscopic displays that do not require glasses [15]. The stereo systems only require two views, where a left-eye view is presented to the viewer's left eye, and a right-eye view is presented to the viewer's right eye. The 3D display technology and glasses ensure that the appropriate signals are viewed by the correct eye. This is accomplished with either passive polarization or active shutter techniques. The multiview displays have much greater data throughput requirements relative to conventional stereo displays in order to support a given picture resolution, since 3D is achieved by essentially emitting multiple complete video sample arrays 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 autostereoscopic 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 covered by other articles in this special issue.

Another goal of multiview video is to enable free-viewpoint video [16][17]. In this scenario, the viewpoint and view direction can be interactively changed. Each output view can either be one of the input views or a virtual view that was generated from a smaller set of multiview inputs and other data that assists in the view generation process. With such a system, viewers can freely navigate through the different viewpoints of the scene – within a range covered by the acquisition cameras. Such an application of multiview video could be implemented with conventional 2D displays. However, more advanced versions of the free-viewpoint system that work with 3D displays could also be considered. We have already seen the use of this functionality in broadcast production environments, e.g., to change the viewpoint of a sports scene to show a better angle of a play. Such functionality may also be of interest in surveillance, education, gaming, and sightseeing applications. Finally, we may also imagine providing this interactive capability directly to the home viewer, e.g., for special events such as concerts.

Another important application of multiview video is 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. Motion parallax is caused by the change in the appearance of a scene when the viewer shifts their viewing position, e.g., shifting the viewing position to reveal occluded scene content. In an interactive system design, it can be possible for the transmission system to adaptively shift its encoded viewing position to achieve a dynamic perspective change [18][19][20]. Perspective changes can be controlled explicitly by user intervention through a user interface control component or by a system that senses the observer's viewing position and adjusts the displayed scene accordingly.

Other interesting applications of multiview video have been demonstrated by Wilburn, et al. [21]. In this work, a high spatial sampling of a scene through a large multiview video camera array was used for advanced imaging. Among the capabilities shown was an effective increase of bit depth and frame rate, as well as synthetic aperture photography effects. Since then, there have also been other exciting developments in the area of computational imaging that rely on the acquisition of multiview video [22].

For all of the above applications and scenarios, the storage and transmission capacity requirements of the system are significantly increased. Consequently, there is a strong need for efficient multiview video compression techniques. Specific requirements are discussed in the next subsection.

B. Standardization Requirements

The central requirement for most video coding designs is high compression efficiency. In the specific case of MVC this means a significant gain compared to independent compression of each view. Compression efficiency measures the tradeoff between cost (in terms of bit rate) and benefit (in terms of video quality) – i.e. the quality at a certain bit rate or the bit rate at a certain quality. However, compression efficiency is not the only factor under consideration for a video coding standard. Some requirements may even be somewhat conflicting, such as desiring both good compression efficiency and low delay. In such cases, a good trade-off needs to be found. General requirements for video coding capabilities, such as minimum resource consumption (memory, processing power), low delay, error robustness, and support of a range of picture resolutions, color sampling structures, and bit depth precisions, tend to be applicable to nearly any video coding design.

Some requirements are specific to MVC – as highlighted in the following. Temporal random access is a requirement for virtually any video coding design. For MVC, view-switching random access also becomes important. Both together ensure that any image can be accessed, decoded, and displayed by starting the decoder at a random access point and decoding a relatively small quantity of data on which that image may depend. Random access can be provided by insertion of pictures that are intra-picture coded (i.e., pictures that are coded without any use of prediction from other pictures). Scalability is also a desirable feature for video coding designs. Here, we refer to the ability of a decoder to access only a portion of a bitstream while still being able to generate effective video output – although reduced in quality to a degree commensurate with the quantity of data in the subset used for the decoding process. This reduction in quality may involve reduced temporal or spatial resolution, or a reduced quality of representation at the same temporal and spatial resolution. For MVC, additionally, *view scalability* is desirable. In this case, a portion of the bitstream can be accessed in order to output a subset of the encoded views. Also, *backward compatibility* was required for the MVC standard. This means that a subset of the MVC bitstream corresponding to one "base view" needs to be decodable by an ordinary (non-MVC) H.264/MPEG-4 AVC decoder, and the other data representing other views should be encoded in way that will not affect that base view decoding capability. Achieving a desired degree quality consistency among views is also addressed – i.e., it should be possible to control the encoding quality of the various views – for instance to provide approximately constant quality over all views or to select a preferential quality for encoding some views versus others. The ability of an encoder or decoder to use parallel processing was required to enable practical implementation and to manage processing resources effectively. It should also be possible to convey camera parameters (extrinsic and intrinsic) along with the bitstream in order to support intermediate view interpolation at the decoder and to enable other decoding-side enhanced capabilities such as multi-view feature detection and classification, e.g., determining the pose of a face within a scene, which would typically require solving a correspondence problem based on the scene geometry.

Moreover, for ease of implementation, it was highly desirable for the MVC design to have as many design elements in common with an ordinary H.264/MPEG-4 AVC system as possible. Such a commonality of design components can enable an MVC system to be constructed rapidly from elements of existing H.264/MPEG-4 AVC products and to be tested more easily.

III. HISTORY OF MVC

One of the earliest studies on coding of multiview images was done by Lukacs [2]; in this work, the concept of disparity-compensated inter-view prediction was introduced. In later work by Dinstein, et al. [3], the predictive coding approach was compared to 3D block transform coding for stereo image compression. In [4], Perkins presented a transform-domain technique for disparity-compensated prediction, as well as a mixed-resolution coding scheme.

The first support for multiview video coding in an international standard was in a 1996 amendment to the H.262/MPEG-2 video coding standard [6]. It supported the coding of two views only. In that design, the left view was referred to as the "base view" and its encoding was compatible with that for ordinary single-view decoders. The right view was encoded as an *enhancement view* that used the pictures of the left view as reference pictures for inter-view prediction.

The coding tool features that were used for this scheme were actually the same as what had previously been designed for providing temporal scalability (i.e., frame rate enhancement) [7]-[10]. For the encoding of the enhancement view, the same basic coding tools were used as in ordinary H.262/MPEG-2 video coding, but the selection of the pictures used as references was altered, so that a reference picture could either be a picture from within the enhancement view or a picture from the base view. An example of a prediction structure that can be used in the H.262/MPEG-2 multiview profile is shown in Fig. 1. Arrows in the figure indicate the use of a reference picture for the predictive encoding of another picture. A significant benefit of this approach, relative to *simulcast* coding of each view independently, was the ability to use inter-view prediction for the encoding of the first enhancement-view picture in each random-accessible encoded video segment. However, the ability to predict in the reverse-temporal direction, which was enabled for the base view, was not enabled for the enhancement view. This helped to minimize the memory storage capacity requirements for the scheme, but may have reduced the compression capability of the design.

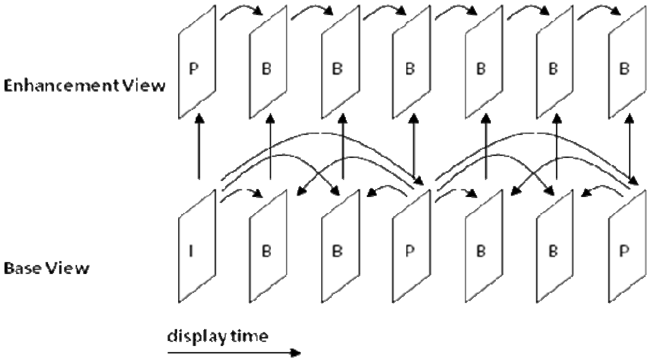


Fig. 1. Illustration of inter-view prediction in H.262/MPEG-2.

Considering recent advancements in video compression technology and the anticipated needs for state-of-the-art coding of multiview video, MPEG issued a Call for Proposals (CfP) for efficient multiview video coding technology in October of 2005. Although not an explicit requirement at the time, all proposal responses were based on H.264/MPEG-4 AVC and included some form of inter-view prediction [23]. As reported in [24], significant gains in visual quality were observed from the formal subjective tests that were conducted in comparison to independent simulcast coding of views based on H.264/MPEG-4 AVC. Specifically, when comparing visual quality at the same bit rate, MVC solutions achieved up to 3 MOS points (mean opinion score points on a 0-10 scale) better visual quality than simulcast H.264/MPEG-4 AVC for low and medium bit rate coding, and about 1 MOS point better quality for high bit rate coding. When comparing bit rates for several of the test sequences, some proposed MVC solutions required only about half the bit rate to achieve equivalent or better visual quality than the H.264/MPEG-4 AVC coded anchors[2]. The proposal described in [25] was found to provide the best visual quality over the wide range of test sequences and rate points. A key feature of that proposal was that it did not introduce any change to the lower levels of the syntax and decoding process used by H.264/MPEG-4 AVC, without any apparent sacrifice of compression capability. This intentional design feature allows for the implementation of MVC decoders to require only a rather simple and straightforward change to existing H.264/MPEG-4 AVC decoding chipsets. As a result of these advantages, this proposal was selected as the starting point of the MVC project – forming what was called the joint multiview model (JMVM) version 1.0.

In the six-month period that followed the responses to CfP, a thorough evaluation of the coding scheme described in [25] was made. This proposal made use of hierarchical prediction in both time and view dimensions to achieve high compression performance. However, views were encoded in an interleaved manner on a group-of-picture (GOP) basis, which resulted in a significant delay and did not allow for simultaneous decoding and output of views at a given time instant. A number of contributions were made to propose a different approach for reference picture management and a time-first coding scheme to reduce encoding/decoding delay and enable parallel input and output of views [26]-[29]. These proposals were adopted into the design at the stage referred to as joint multiview model (JMVM) version 2.0 [30], which was an early draft of the standard that established the basic principles of the eventual MVC standard.

During the development of MVC, a number of macroblock-level coding tools were also explored, including the following:

- Illumination compensation: The objective of this tool is to compensate for illumination differences as part of the inter-view prediction process [31][32].
- Adaptive reference filtering: It was observed by Lai, et al. [33][34] that there are other types of mismatches present in multiview video in addition to illumination differences, which led to the development of an adaptive reference filtering scheme to compensate for focus mismatches between different views.
- Motion skip mode: Noting the correlation between motion vectors in different views, this method infers motion vectors from inter-view reference pictures [35][36].
- View synthesis prediction: This coding technique predicts a picture in the current view from synthesized references generated from neighboring views [37]-[39].

It was shown that additional coding gains could be achieved by using these block-level coding tools. In an analysis of the coding gains offered by both illumination compensation and motion skip mode that was reported in [40], an average bit rate

[2] In that comparison, the anchor bitstreams used for the subjective evaluation testing did not use a multi-level hierarchical prediction referencing structure (as this type of referencing had not yet become well established in industry practice). If such hierarchical referencing had been used in the anchors, the estimated bit rate gains would likely have been more modest.

reduction of 10% (relative to an MVC coding design without these tools) was reported over a significant set of sequences – with a maximum sequence-specific reduction of approximately 18%. While the gains were notable, these tools were not adopted into the MVC standard since they would require syntax and design changes affecting low levels of the encoding and decoding process (within the macroblock level). It was believed that these implementation concerns outweighed the coding gain benefits at the time. There was also some concern that the benefits of the proposed techniques might be reduced by higher quality video acquisition and pre-processing practices. However, as the 3D market matures, the benefits of block-level coding tools may be revisited in the specification of future 3D video formats.

IV. H.264/MPEG-4 AVC BASICS

MVC was standardized as an extension of H.264/MPEG-4 AVC. In order to keep the paper self-contained, the following brief description of H.264/MPEG-4 AVC is limited to those key features that are relevant for understanding the concepts of extending H.264/MPEG-4 AVC towards multiview video coding. For more detailed information about H.264/MPEG-4 AVC, the reader is referred to the standard itself [1] and the various overview papers that have discussed it (e.g., [41]-[43]).

Conceptually, the design of H.264/MPEG-4 AVC covers a *Video Coding Layer* (VCL) and a *Network Abstraction Layer* (NAL). While the VCL creates a coded representation of the source content, the NAL formats these data and provides header information in a way that enables simple and effective customization of the use of VCL data for a broad variety of systems.

A. Network Abstraction Layer (NAL)

A coded H.264/MPEG-4 AVC video data stream is organized into NAL units, which are packets that each contain an integer number of bytes. A NAL unit starts with a one-byte indicator of the type of data in the NAL unit. The remaining bytes represent payload data. NAL units are classified into *video coding layer* (VCL) NAL units, which contain coded data for areas of the picture content (coded slices or slice data partitions), and non-VCL NAL units, which contain associated additional information. Two key types of non-VCL NAL units are the *parameter sets* and the *supplemental enhancement information* (SEI) messages. The sequence and picture parameter sets contain infrequently changing information for a coded video sequence. SEI messages do not affect the core decoding process of the samples of a coded video sequence. However, they provide additional information to assist the decoding process or affect subsequent processing such as bitstream manipulation or display. The set of consecutive NAL units associated with a single coded picture is referred to as an *access unit*. A set of consecutive access units with certain properties is referred to as a *coded video sequence*. A coded video sequence (together with the associated parameter sets) represents an independently decodable part of a video bitstream. A coded video sequence always starts with an *instantaneous decoding refresh* (IDR) access unit, which signals that the IDR access unit and all access units that follow it in the bitstream can be decoded without decoding any of the pictures that preceded it.

B. Video Coding Layer (VCL)

The VCL of H.264/MPEG-4 AVC follows the so-called block-based hybrid video coding approach. Although its basic design is very similar to that of prior video coding standards such as H.261, MPEG-1, H.262/MPEG-2, H.263, or MPEG-4 Visual, H.264/MPEG-4 AVC includes new features that enable it to achieve a significant improvement in compression efficiency relative to any prior video coding standard [41]-[43]. The main difference relative to previous standards is the greatly increased flexibility and adaptability of the H.264/MPEG-4 AVC design.

The way pictures are partitioned into smaller coding units in H.264/MPEG-4 AVC, however, follows the rather traditional concept of subdivision into *slices* which in turn are subdivided into *macroblocks.* Each slice can be parsed independently of the other slices in the picture. Each picture is partitioned into macroblocks that each covers a rectangular picture area of 16×16 luma samples and, in the case of video in 4:2:0 chroma sampling format, 8×8 sample areas of each of the two chroma components. The samples of a macroblock are either spatially or temporally predicted, and the resulting prediction residual signal is represented using transform coding. Depending on the degree of freedom for generating the prediction signal, H.264/MPEG-4 AVC supports three basic slice coding types that specify the types of coding supported for the macroblocks within the slice:

- *I* slices, in which each macroblock uses *intra-picture* coding using spatial prediction from neighboring regions,
- *P* slices, which support both intra-picture coding and inter-picture *predictive* coding using one prediction signal for each predicted region,
- *B* slices, which support intra-picture coding, inter-picture predictive coding, and also inter-picture *bi-predictive* coding using two prediction signals that are combined with a weighted average to form the region prediction.

For *I* slices, H.264/MPEG-4 AVC provides several directional spatial intra-picture prediction modes, in which the prediction signal is generated by using the decoded samples of neighboring blocks that precede the block to be predicted (in coding and decoding order). For the luma component, the intra-picture prediction can be applied to individual 4×4 or 8×8 luma blocks within the macroblock, or to the full 16×16 luma array for the macroblock; whereas for the chroma components, it is applied on a full-macroblock region basis.

For *P* and *B* slices, H.264/MPEG-4 AVC additionally permits variable block size motion-compensated prediction with multiple reference pictures. The macroblock type signals the partitioning of a macroblock into blocks of 16×16, 16×8, 8×16, or 8×8 luma samples. When a macroblock type specifies partitioning into four 8×8 blocks, each of these so-called *sub-*

macroblocks can be further split into 8×4, 4×8, or 4×4 blocks, as determined by a sub-macroblock type indication. For *P* slices, one motion vector is transmitted for each inter-picture prediction block. The reference picture to be used for inter-picture prediction can be independently chosen for each 16×16, 16×8, or 8×16 macroblock motion partition or 8×8 sub-macroblock. The selection of the reference picture is signaled by a reference index parameter, which is an index into a list (referred to as *list* 0) of previously coded reference pictures that are stored by the decoder for such use after they have been decoded.

In *B* slices, two distinct reference picture lists are used, and for each 16×16, 16×8, or 8×16 macroblock partition or 8×8 sub-macroblock, the prediction method can be selected between *list 0*, *list 1*, or *bi-prediction*. List 0 and list 1 prediction refer to inter-picture prediction using the reference picture at the reference index position in reference picture list 0 and 1, respectively, in a manner similar to that supported in *P* slices. However, in the bi-predictive mode the prediction signal is formed by a weighted sum of the prediction values from both a list 0 and list 1 prediction signal. In addition, special modes referred to as *direct modes* in *B* slices and *skip modes* in *P* and *B* slices are provided, which operate similarly to the other modes, but in which such data as motion vectors and reference indices are derived from properties of neighboring previously-coded regions rather than being indicated explicitly by syntax for the direct or skip mode macroblock.

For transform coding of the spatial-domain residual difference signal remaining after the prediction process, H.264/MPEG-4 AVC specifies a set of *integer transforms* of different block sizes. While for intra-picture coded macroblocks the transform size is directly coupled to the prediction block size, the luma signal of motion-compensated macroblocks that do not contain blocks smaller than 8×8 can be coded by using either a 4×4 or 8×8 transform. For the chroma components, a two-stage transform is employed, consisting of 4×4 transforms and an additional Hadamard transform of the resulting DC coefficients. A similar hierarchical transform is also used for the luma component of macroblocks coded in the 16×16 intra-picture macroblock coding mode. All inverse transforms are specified by exact integer operations, so that inverse-transform mismatches are avoided. H.264/MPEG-4 AVC uses *uniform reconstruction quantizers*. The reconstruction step size for the quantizer is controlled for each macroblock by a quantization parameter *QP*. For 8-bit-per-sample video, 52 values of QP can be selected. The QP value is multiplied by an entry in a scaling matrix to determine a transform-frequency-specific quantization reconstruction step size The scaling operations for the quantization step sizes are arranged with logarithmic step size increments, such that an increment of the *QP* by 6 corresponds to a doubling of quantization step size.

For reducing blocking artifacts, which are typically the most disturbing artifacts in block-based coding, H.264/MPEG-4 AVC specifies an *adaptive deblocking filter* that operates within the motion-compensated inter-picture prediction loop.

H.264/MPEG-4 AVC supports two methods of entropy coding, which both use context-based adaptivity to improve performance relative to prior standards. While *CAVLC* (context-based adaptive variable-length coding) uses variable-length codes and its adaptivity is restricted to the coding of transform coefficient levels, *CABAC* (context-based adaptive binary arithmetic coding) uses arithmetic coding and a more sophisticated mechanism for employing statistical dependencies, which leads to typical bit rate savings of 10-15% relative to CAVLC.

In addition to increased flexibility at the macroblock level and the lower levels within it, H.264/MPEG-4 AVC also allows much more flexibility on a picture and sequence level compared to prior video coding standards. Here we primarily refer to reference picture buffering and the associated buffering memory control. In H.264/MPEG-4 AVC, the coding and display order of pictures is completely decoupled. Furthermore, any picture can be used as reference picture for motion-compensated prediction of subsequent pictures, independent of its slice coding types. The behavior of the *decoded picture buffer* (DPB), which can hold up to 16 frames (depending on the supported conformance point and the decoded picture size), can be adaptively controlled by *memory management control operation* (MMCO) commands, and the reference picture lists that are used for coding of *P* or *B* slices can be arbitrarily constructed from the pictures available in the DPB via *reference picture list modification* (RPLM) commands.

For efficient support of the coding of interlaced-scan video, in a manner similar to prior video coding standards, a coded *picture* may either comprise the set of slices representing a complete video frame or of just one of the two fields of alternating lines in such a frame. Additionally, H.264/MPEG-4 AVC supports a macroblock-adaptive switching between frame and field coding. In this adaptive operation, each 16×32 region in a frame is treated as a single coding unit referred to as a *macroblock pair*, which can be either transmitted as two macroblocks representing vertically-neighboring 16×16 rectangular areas in the frame, or as macroblocks formed from the de-interleaved lines of the top and bottom fields in the 16×32 region. This scheme is referred to as macroblock-adaptive frame-field coding (MBAFF). Together the single-field picture coding and MBAFF coding features are sometimes referred to as *interlace coding tools*.

V. EXTENDING H.264/MPEG-4 AVC FOR MULTIVIEW

The most recent major extension of the H.264/MPEG-4 AVC standard [1] is the Multiview Video Coding (MVC) design [11]. Several key features of MVC are reviewed below; some of which have also been covered in [10] and [44]. Several other aspects of the MVC design were further elaborated on in [44], including random access and view switching, extraction of *operation points* (sets of coded views at particular levels of a nested temporal referencing structure) of an MVC bitstream for adaptation to network and device constraints, parallel processing, and a description of several newly adopted

SEI messages that are relevant for multiview video bitstreams. An analysis of MVC decoded picture buffer requirements was also provided in that work.

A. Bitstream Structure

A key aspect of the MVC design is that it is mandatory for the compressed multiview stream to include a base view bitstream, which is coded independently from all other views in a manner compatible with decoders for single-view profile of the standard, such as the High profile or the Constrained Baseline profile. This requirement enables a variety of uses cases that need a 2D version of the content to be easily extracted and decoded. For instance, in television broadcast, the base view could be extracted and decoded by legacy receivers, while newer 3D receivers could decode the complete 3D bitstream including non-base views.

As described in Section IV.A, coded data in H.264/MPEG-4 AVC is organized into NAL units. There exist various types of NAL units, some of which are designated for coded video pictures, while others for non-picture data such as parameter sets and SEI messages. MVC makes use of the NAL unit type structure to provide backward compatibility for multiview video.

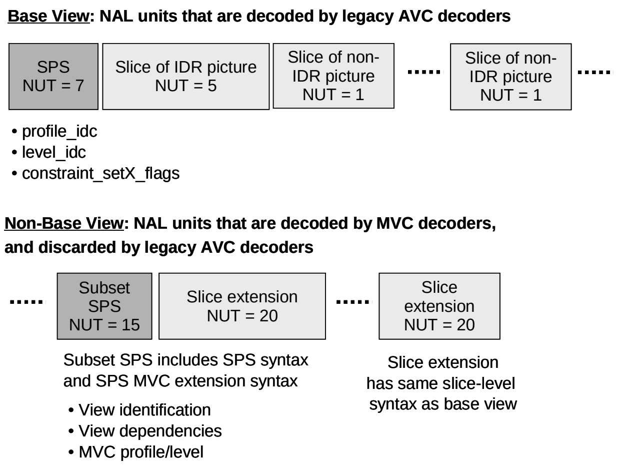

Base View: NAL units that are decoded by legacy AVC decoders
SPS NUT = 7
Slice of IDR picture NUT = 5
Slice of non-IDR picture NUT = 1
.....
Slice of non-IDR picture NUT = 1
.....
• profile_idc
• level_idc
• constraint_setX_flags
Non-Base View: NAL units that are decoded by MVC decoders, and discarded by legacy AVC decoders
.....
Subset SPS NUT = 15
Slice extension NUT = 20
.....
Slice extension NUT = 20
Subset SPS includes SPS syntax and SPS MVC extension syntax
• View identification
• View dependencies
• MVC profile/level
Slice extension has same slice-level syntax as base view

Fig. 2. Structure of an MVC bitstream including NAL units that are associated with a base view and NAL units that are associated with a non-base view. *NAL unit type* (NUT) indicators are used to distinguish different types of data that are carried in the bitstream.

To achieve this compatibility, the video data associated with a base view is encapsulated in NAL units that have previously been defined for the 2D video, while the video data associated with the additional views are encapsulated in an extension NAL unit type that is used for both scalable video coding (SVC) [45] and multiview video. A flag is specified to distinguish whether the NAL unit is associated with an SVC or MVC bitstream. The base view bitstream conforms to existing H.264/MPEG-4 AVC profiles for single-view video, e.g., High profile, and decoders conforming to an existing single view profile will ignore and discard the NAL units that contain the data for the non-base views since they would not recognize those NAL unit types. Decoding the additional views with these new NAL unit types would require a decoder that recognizes the extension NAL unit type and conforms to one of the MVC profiles. The basic structure of the MVC bitstream including some NAL units associated with a base view and some NAL units associated with a non-base view is shown in Fig. 2. Further discussion of the high-level syntax is given below. MVC profiles and levels are also discussed later in this section.

B. Enabling Inter-view Prediction

The basic concept of inter-view prediction, which is employed in all of the described designs for efficient multiview video coding, is to exploit both spatial and temporal redundancy for compression. Since the cameras (or rendered viewpoint perspectives) of a multiview scenario typically capture the same scene from nearby viewpoints, substantial inter-view redundancy is present. A sample prediction structure is shown in Fig. 3. Pictures are not only predicted from temporal references, but also from inter-view references. 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.

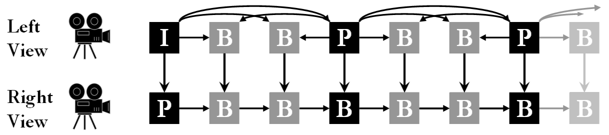


Fig. 3. Illustration of inter-view prediction in MVC.

Inter-view prediction is a key feature of the MVC design, and it is enabled in a way that makes use of the flexible reference picture management capabilities that had already been designed into H.264/MPEG-4 AVC, by making the decoded pictures from other views available in the reference picture lists for use by the inter-picture prediction processing. Specifically, the reference picture lists are maintained for each picture to be decoded in a given view. Each such list is initialized as usual for single-view video, which would include the temporal reference pictures that may be used to predict the current picture. Additionally, inter-view reference pictures are included in the list and are thereby also made available for prediction of the current picture.

According to the MVC specification, inter-view reference pictures must be contained within the same access unit as the current picture, where an access unit contains all the NAL units pertaining to a certain capture or display time instant. The MVC design does not allow the prediction of a picture in one view at a given time using a picture from another view at a different time. This would involve inter-view prediction across different access units, which would incur additional complexity for limited coding benefits.

To keep the management of reference pictures consistent with that for single-view video, all the memory management control operation commands that may be signaled through an H.264/MPEG-4 AVC bitstream apply to one particular view in which the associated syntax elements appear. The same is true for the sliding window and adaptive memory control processes that can be used to mark pictures as not being used for reference. The reference picture marking process of H.264/MPEG-4 AVC is applied independently for each view, so that the encoder can use the available decoder memory capacity in a flexible manner. Moreover, just as it is possible for an encoder to re-order the positions of the reference pictures in a reference picture list that includes temporal reference pictures, it can also place the inter-view reference pictures at any desired positions in the lists. An extended set of re-ordering commands are provided in the MVC specification for this purpose.

It is important to emphasize that the core macroblock-level and lower-level decoding modules of an MVC decoder are the same, regardless of whether a reference picture is a temporal reference or an inter-view reference. This distinction is managed at a higher level of the decoding process.

In terms of syntax, supporting MVC only involves small changes to high-level syntax, e.g., an indication of the prediction dependency as discussed in the next subsection. A major consequence of not requiring changes to lower levels of the syntax (at the macroblock level and below it) is that MVC is compatible with existing hardware for decoding single-view video with H.264/MPEG-4 AVC. In other words, supporting MVC as part of an existing H.264/MPEG-4 AVC decoder should not require substantial design changes.

Since MVC introduces dependencies between views, random access must also be considered in the view dimension. Specifically, in addition to the views to be accessed (called the *target* views), any views on which they depend for purposes of inter-view referencing also need to be accessed and decoded, which typically requires some additional decoding time or delay. For applications in which random access or view switching is important, the prediction structure can be designed to minimize access delay, and the MVC design provides a way for an encoder to describe the prediction structure for this purpose.

To achieve access to a particular picture in a given view, the decoder should first determine an appropriate access point. In H.264/MPEG-4 AVC, each *instantaneous decoding refresh* (IDR) picture provides a clean random access point, since these pictures can be independently decoded and all the coded pictures that follow them in bitstream order can also be decoded without temporal prediction from any picture decoded prior to the IDR picture. In the context of MVC, an IDR picture in a given view prohibits the use of temporal prediction for any of the views on which a particular view depends at that particular instant of time; however, inter-view prediction may be used for encoding the non-base views of an IDR picture. This ability to use inter-view prediction for encoding an IDR picture reduces the bit rate needed to encode the non-base views, while still enabling random access at that temporal location in the bitstream. Additionally, MVC also introduces an additional picture type, referred to as an *anchor picture* for a view. Anchor pictures are similar to IDR pictures in that they do not use temporal prediction for the encoding of any view on which a given view depends, although they do allow inter-view prediction from other views within the same access unit. Moreover, it is prohibited for any picture that follows the anchor picture in both bitstream order and display order to use any picture that precedes the anchor picture in bitstream order as a reference for inter-picture prediction, and for any picture that precedes the anchor picture in decoding order to follow it in display order. This provides a clean random access point for access to a given view. The difference between anchor pictures and IDR pictures is similar to the difference between the "open GOP" and "closed GOP" concepts that previously applied in the H.262/MPEG-2 context[3], with closed GOPs being associated with IDR pictures and open GOPs being associated with anchor pictures [44]. With an anchor picture, it is permissible to use pictures that precede the anchor picture in bitstream order as reference pictures for inter-picture prediction of pictures that follow after the anchor picture in bitstream order, but only if the pictures that use this type of referencing precede the anchor picture in display order. In MVC, both IDR and anchor pictures are efficiently coded, and they enable random access in the time and view dimensions.

C. High-level Syntax

The decoding process of MVC requires several additions to the high-level syntax, which are primarily signaled through a multiview extension of the *sequence parameter set* (SPS) defined by H.264/MPEG-4 AVC. Three important pieces of information are carried in the SPS extension:

- View identification
- View dependency information
- Level index for operation points

The view identification part includes an indication of the total number of views, as well as a listing of view identifiers. The view identifiers are important for associating a particular view to a specific index, while the order of the view identifiers signals the view order index. The view order index is critical to the decoding process as it defines the order in which views are decoded.

The view dependency information is composed of a set of signals that indicate the number of inter-view reference pictures for each of the two reference picture lists that are used in the prediction process, as well as the views that may be used for predicting a particular view. Separate view dependency information is provided for anchor and non-anchor pictures to provide some flexibility in the prediction while not overburdening decoders with dependency information that could change for each unit of time. For non-anchor pictures, the view dependency only indicates that a given set of views may be used for inter-view prediction. There is additional signaling in the NAL unit header indicating whether a particular view at a given time may be used for inter-view reference for any other

[3] For those familiar with the more modern version of this concept as found in H.264/MPEG-4, an MVC anchor picture is also analogous to the use of the H.264/MPEG-4 AVC recovery point SEI message with a recovery frame count equal to 0.

picture in the same access unit. The view dependency information in the SPS is used together with this syntax element in the NAL unit header to create reference picture lists that include inter-view references, as described in the previous subsection.

The final portion of the SPS extension is the signaling of *level information* and information about the *operating points* to which it correspond. The level index is an indicator of the resource requirements for a decoder that conforms to a particular level; it is mainly used to establish a bound on the complexity of a decoder and is discussed further below. In the context of MVC, an operating point corresponds to a specific temporal subset and a set of views including those intended for output and the views that they depend on. For example, an MVC bitstream with 8 views may provide information for several operating points, e.g., one corresponding to all 8 views together, another corresponding to a stereo pair, and another corresponding to a set of three particular views. According to the MVC standard, multiple level values could be signaled as part of the SPS extension, with each level being associated with a particular operating point. The syntax indicates the number of views that are targeted for output as well as the number of views that would be required for decoding particular operating points.

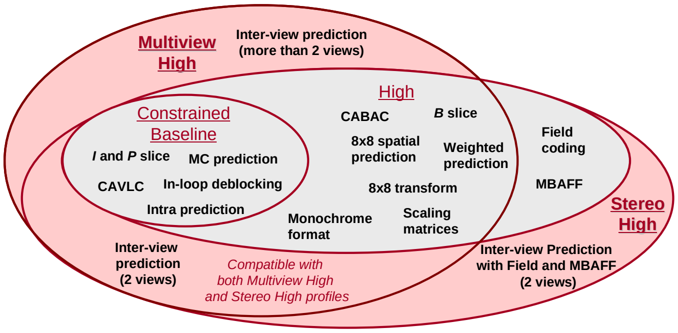


Fig. 4. Illustration of MVC profiles, consisting of the Multiview High and Stereo High profiles, together with illustration of the features compatible with both profiles and profiles that can be used for the encoding of the base view.

D. Profiles and Levels

As with prior video coding standards, *profiles* determine the subset of coding tools that must be supported by conforming decoders. There are two profiles currently defined by MVC with support for more than one view: the Multiview High profile and the Stereo High profile. Both are based on the High profile of H.264/MPEG-4 AVC, with a few differences.

- The Multiview High profile supports multiple views and does not support interlace coding tools.
- The Stereo High profile is limited to two views, but does support interlace coding tools.

For either of these profiles, the base view can be encoded using either the High profile of H.264/MPEG-4 AVC, or a more constrained profile known as the Constrained Baseline profile which was added to the standard more recently [12]. When the High profile is used for the base view for the Multiview High profile, the interlace coding tools (field picture coding and MBAFF), which are ordinarily supported in the High profile, cannot be used in the base layer since they are not supported in the Multiview High profile. (The Constrained Baseline profile does not support interlace coding tools.)

An illustration of these profile specifications relative to the High and Constrained Baseline profiles of H.264/MPEG-4 AVC is provided in Fig. 4. It is possible to have a bitstream that conforms to both the Stereo High profile and Multiview High profile, when there are only two views that are coded and the interlace coding tools are not used. In this case, a flag signaling their compatibility is set.

Levels impose constraints on the bitstreams produced by MVC encoders, to establish bounds on the necessary decoder resources and complexity. The level limits include limits on the amount of frame memory required for the decoding of a bitstream, the maximum throughput in terms of macroblocks per second, maximum picture size, overall bit rate, etc.

The general approach to defining level limits in MVC was to enable the repurposing of the decoding resources of single-view decoders for the creation of multiview decoders. In this way, some level limits are unchanged, such as the overall bit rate; in this way, an input bitstream can be processed by a decoder regardless of whether it encodes a single view or multiple views. However, other level limits are increased, such as for the maximum decoded picture buffer capacity and throughput; a fixed scale factor of two was applied to these decoder resource requirements. Assuming a fixed resolution, this scale factor enables the decoding of stereo video using the same level as is specified for single-view video at the same resolution. For instance, the same Level 4.0 designation is used for single-view video at 1920×1080p at 24 Hz using the High profile and for stereo-view video at 1920×1080p at 24 Hz for each of the two views using the Stereo High profile. To decode a higher number of views, one would either use a higher level and/or reduce the spatial or temporal resolution of the multiview video.

E. Coding Performance

It has been shown that coding multiview video with inter-view prediction does give significantly better results compared to independent coding [47]. For some cases, gains as high as 3 dB, roughly corresponding to a 50% savings in bit rate, have been reported. A comprehensive set of results for multiview video coding over a broad range of test material was presented in [40] according to a set of common test conditions and test material specified in [48]. For multiview video with up to 8 views, an average of 20% reduction in bit rate was reported, relative to the total simulcast bit rate, based on Bjøntegaard delta measures [49]. In other studies [50], an average reduction of 20-30% of the bit rate for the second (dependent) view of typical stereo movie content was reported, with a peak reduction for an individual test sequence of 43% of the bit rate of the dependent view. Fig. 5 shows sample rate-distortion curves comparing the performance of simulcast coding with the performance of MVC reference software that employs hierarchical predictions in both the temporal and view dimensions.

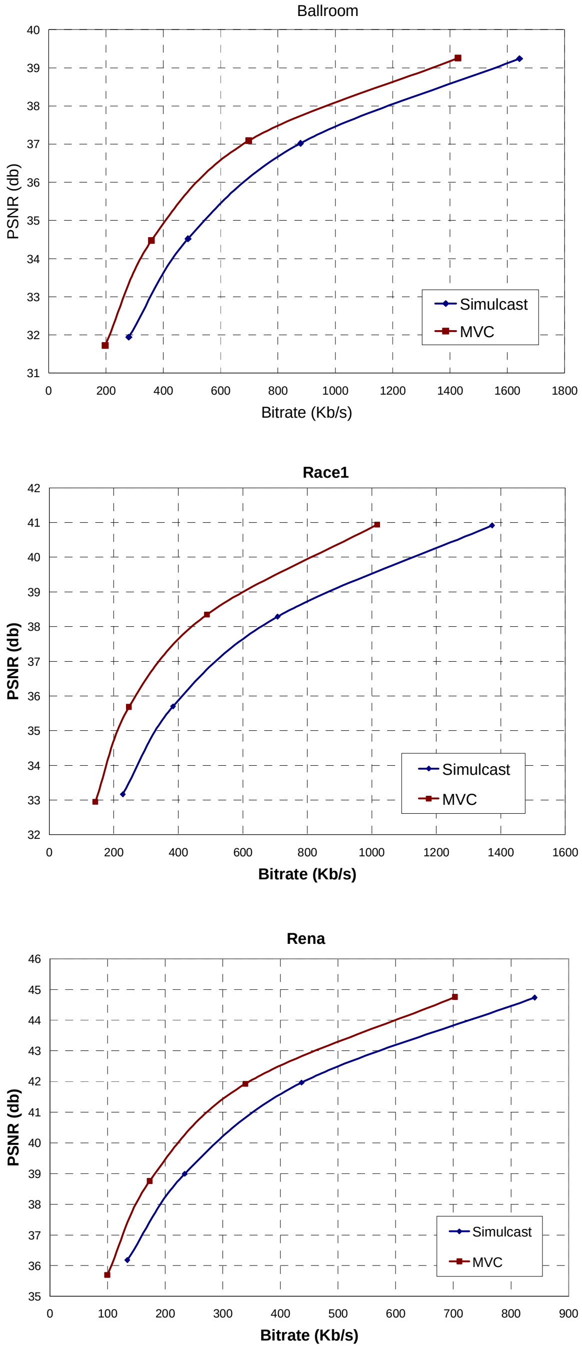


Fig. 5. Sample coding results for several MVC test sequences, including Ballroom, Race1, and Rena sequences, according to common test conditions [48].

There are many possible variations on the prediction structure considering both temporal and inter-view dependencies. The structure not only affects coding performance, but has notable impact on delay, memory requirements and random access. It has been confirmed that the majority of gains are obtained using inter-view prediction at anchor positions. An average decrease in bit rate of approximately 5-15% at equivalent quality could be expected if the inter-view predictions at non-anchor positions are not used [51]. The upside is that delay and required memory would also be reduced.

Prior studies on asymmetrical coding of stereo video, in which one of the views is encoded with lower quality than the other, suggest that a further substantial savings in bit rate for the non-base view could be achieved using that technique. In this scheme, one of the views is significantly blurred or more coarsely quantized than the other [52], or is coded with a reduced spatial resolution [53][54], with an impact on the stereo quality that may be imperceptible. With mixed resolution coding, it has been reported that an additional view could be supported with minimal rate overhead, e.g., on the order of 25-30% additional rate added to a base view encoding for coding the other view at quarter resolution. Further study is needed to understand how this phenomenon extends to multiview video with more than two views. The currently-standardized MVC design provides the encoder with a great deal of freedom to select the encoded fidelity for each view and to perform preprocessing such as blurring if desired; however, it uses the same sample array resolution for the encoding of all views.

F. *SEI Messages for Multiview Video*

Several new SEI messages for multiview video applications have also been specified as part of the MVC extension of H.264/MPEG-4 AVC. However, it should be noted that, in general, SEI messages only supply supplemental information that is not used within the standardized process for the decoding of the sample values of the coded pictures, and the use of any given SEI message may not be necessary or appropriate in some particular MVC application environment. A brief summary of these messages and their primary intended uses are included below.

Parallel decoding information SEI message: indicates that the views of an access unit are encoded with certain constraints that enable parallel decoding. Specifically, it signals a limitation that has been imposed by the MVC encoder whereby a macroblock in a certain view is only allowed to depend on reconstruction values of a subset of macroblocks in other views. By constraining the reference area, it is possible to enable better parallelization in the decoding process [44].

MVC scalable nesting SEI message: enables the reuse of existing SEI messages in the multiview video context by indicating the views or temporal levels to which the messages apply.

View scalability information SEI message: contains view and scalability information for particular operation points (sets of coded views at particular levels of a nested temporal referencing structure) in the coded video sequence. Information such as bit rate and frame rate, among others, are signaled as part of the message for the subset of the operation points. This information can be useful to guide a bitstream extraction process [44].

Multiview scene information SEI message: indicates the maximum disparity among multiple view components in an access unit. This message can be used for processing the decoded view components prior to rendering on a 3D display. It may also be useful in the placement of graphic overlays, subtitles, and captions in a 3D scene.

Multiview acquisition information SEI message: this SEI message specifies various parameters of the acquisition environment, and specifically, the intrinsic and extrinsic camera parameters. These parameters are useful for view warping and interpolation, as well as solving other correspondence problems mentioned above in section II.B.

Non-required view component SEI message: indicates that a particular view component is not needed for decoding. This may occur if a particular set of views have been identified for output and there are other views in the bitstream that these target output views do not depend on.

View dependency change SEI message: with this SEI message, it is possible to signal changes in the view dependency structure.

Operation point not present SEI message: indicates operation points that are not present in the bitstream. This may be useful in streaming and networking scenarios that are considering available operation points in the current bitstream that could satisfy network or device constraints.

Base view temporal HRD SEI message: when present, this SEI message is associated with an IDR access unit and signals information relevant to the hypothetical reference decoder (HRD) parameters associated with the base view.

VI. FRAME-COMPATIBLE STEREO ENCODING FORMATS

Frame compatible formats refer to a class of stereo video formats in which the two stereo views are essentially multiplexed into a single coded frame or sequence of frames. Some common such formats are shown in Fig. 6. Other common names include stereo interleaving or spatial/temporal multiplexing formats. In the following, a general overview of these formats along with the key benefits and drawbacks are discussed. The signaling for these formats that has been standardized as part of the H.264/MPEG-4 AVC standard is also described.

A. Basic Principles

With a frame-compatible format, the left and right views are packed together in the samples of a single video frame. In such a format, half of the coded samples represent the left view and the other half represent the right view. Thus, each coded view has half the resolution of the full coded frame. There is a variety of options available for how the packing can be performed. For example, each view may have half horizontal resolution or half vertical resolution. The two such half-resolution views can be interleaved in alternating samples of each column or row, respectively, or can be placed next to each other in arrangements known as the *side-by-side* and *top-bottom* packings (see Fig. 6). The top-bottom packing is also sometimes referred to as *over-under* packing [55]. Alternatively, a "checkerboard" (quincunx) sampling may be applied to each view, with the two views interleaved in alternating samples in both the horizontal and vertical dimensions (as also shown in Fig. 6).

Temporal multiplexing is also possible. In this approach, the left and right views would be interleaved as alternating frames or fields of a coded video sequence. These formats are referred to as *frame sequential* and *field sequential*. The frame rate of each view may be reduced so that the amount of data is equivalent to that of a single view.

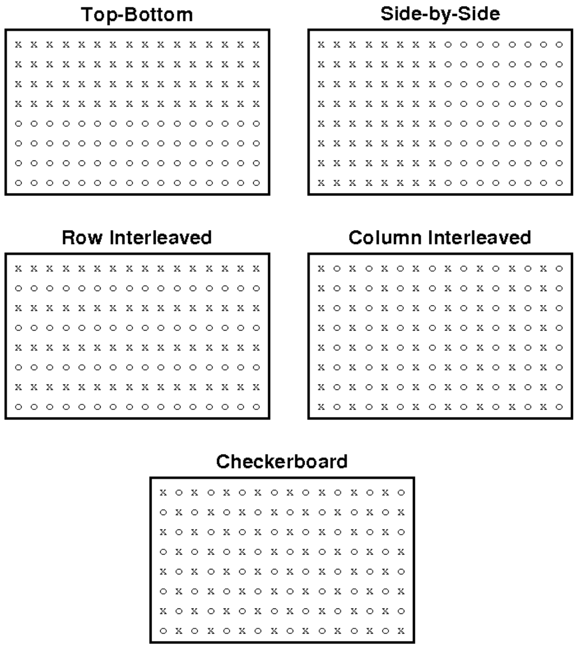


Fig. 6. Common frame-compatible formats where ‘x’ represents the samples from one view and ‘o’ represents the samples from the other view.

Frame-compatible formats have received considerable attention from the broadcast industry since they facilitate the introduction of stereoscopic services through existing infrastructure and equipment. The coded video can be processed by encoders and decoders that were not specifically designed to handle stereo video – such that only the display subsystem that follows the decoding process needs to be altered to support 3D. Representing the stereo video in a way that is maximally compatible with existing encoding, decoding and delivery infrastructure is the major advantage of this format. The video can be compressed with existing encoders, transmitted through existing channels, and decoded by existing receivers. Only the final display stage requires some customization for recognizing and properly rendering the video to enable a 3D viewing experience. Although compression performance may vary depending on the content, the acquisition and pre-processing technology, and the frame packing arrangement that are used, the bit rates for supporting stereo video in this manner may not need to be substantially higher than for a compressed single view at an equivalent spatial resolution (although a somewhat higher bit rate may be desirable, since the frame-compatible stereo video would tend to have higher spatial frequency content characteristics). This format essentially tunnels the stereo video through existing hardware and delivery channels. Due to these minimal changes, stereo video service can be quickly deployed to 3D capable displays (which are already available in the market – e.g., using the HDMI 1.4a specification [56]).

The drawback of representing the stereo signal in this way is that spatial or temporal resolution would be only half of that used for 2D video with the same (total) encoded resolution. The key additional issue with frame-compatible formats is distinguishing the left and right views. To perform the de-interleaving, it is necessary for receivers to be able to parse and interpret some signal that indicates that the frame packing is being used. Since this signaling may not be understood by legacy receivers, it may not even be possible for such devices to extract, decode and display a 2D version of the 3D program. However, this may not necessarily be considered so problematic, as it is not always considered desirable to enable 2D video extraction from a 3D stream. The content production practices for 2D and 3D programs may be different, and 2D and 3D versions of a program may be edited differently (e.g., using more frequent scene cuts and more global motion for 2D programming than for 3D). Moreover, the firmware on some devices, such as cable set-top boxes, could be upgraded to understand the new signaling that describes the video format (although the same is not necessarily true for broadcast receivers and all types of equipment).

B. Signaling

The signaling for a complete set of frame-compatible formats has been standardized within the H.264/MPEG-4 AVC standard as supplemental enhancement information (SEI) messages. A decoder that understands the SEI message can interpret the format of the decoded video and display the stereo content appropriately.

An earlier edition of the standard that was completed in 2004 specified a *stereo video information* (SVI) SEI message that could identify two types of frame-compatible encoding for left and right views. More specifically, it was able to indicate either a row-based interleaving of views that would be represented as individual fields of a video frame or a temporal multiplexing of views where the left and right views would be in a temporally alternating sequence of frames. The SVI SEI message also had the capability of indicating whether the encoding of a particular view is self-contained, i.e., whether the frames or fields corresponding to the left view are only predicted from other frames or fields of the left view. Inter-view prediction for stereo is possible when the self-contained flag is disabled.

Although the specification of the SVI SEI message is still included in the current version of the standard [1], the functionality of this SEI message has recently been incorporated, along with additional signaling capabilities and support of various other spatially multiplexed formats (as described above), into a new SEI message. Thus the new edition of the standard expresses a preference for the use of the new SEI message rather than the SVI SEI message. The new SEI message is referred to as the *frame packing arrangement* (FPA) SEI message. It was specified in an amendment of the H.264/MPEG-4 AVC standard [12] and was incorporated into the latest edition [1]. This new SEI message is the current suggested way to signal frame-compatible stereo video information, and it is able to signal all of the various frame packing arrangements shown in Fig. 6. With the side-by-side and top-bottom arrangements, it is also possible to signal whether one of the views has been flipped so as to create a mirror image in the horizontal or vertical direction, respectively. Independent of the frame packing arrangement, the SEI message also indicates whether the left and right views have been subject to a quincunx (checkerboard) sampling. For instance, it is possible to apply a quincunx filter and sub-sampling process, but then rearrange the video samples into a side-by-side format. Such schemes are also supported in the FPA SEI message. Finally, the SEI message indicates whether the upper-left sample of a packed frame is for the left or right view and it also supports additional syntax to indicate the precise relative grid alignment positions of the samples of the left and right views, using a precision of one sixteenth of the sample grid spacing between the rows and columns of the decoded video array.

C. Discussion

Industry is now preparing for the introduction of new 3D services. With the exception of Blu-ray Discs, which will offer a stereo format with HD resolution for each view based on the Stereo High profile of the MVC extensions, the majority of services will start based on frame-compatible formats that will have a lower resolution for each coded view than the full resolution of the coded frame [57]. Some benefits and drawbacks of the various formats are discussed below; further discussion can also be found in [57].

In the production and distribution domains, the side-by-side and top-bottom formats currently appear to be the most favored (e.g., in [55] and [58]). Relative to row or column interleaving and the checkerboard format, the quality of the reconstructed stereo signal after compression can be better maintained. The interleaved formats introduce significant high frequency content into the frame-compatible signal – thereby requiring a higher bit rate for encoding with adequate quality. Also, the interleaving and compression process can create cross-talk artifacts and color bleeding across views.

From the pure sampling perspective, there have been some studies that advocated benefits of quincunx sampling. In particular, quincunx sampling preserves more of the original signal and its frequency-domain representation is similar to that of the human visual system. The resolution loss is equally distributed in the vertical and horizontal directions. So, while it may not be a distribution-friendly format, quincunx sampling followed by a rearrangement to side-by-side or top-bottom format could potentially lead to higher quality compared to direct horizontal or vertical decimation of the left and right views by a factor of two. On the other hand, quincunx sampling may introduce high frequencies into the video signal that are difficult to encode, since it creates frequency content that is neither purely vertical nor purely horizontal. This may result in a signal that requires a higher bit rate to encode with adequate quality [55].

Another issue to consider regarding frame-compatible formats is whether the source material is interlaced. Since the top-bottom format incurs a resolution loss in the vertical dimension and an interlaced field is already half the resolution of the decoded frame, the side-by-side format is generally preferred over the top-bottom format for interlaced content [55][58].

Since there are displays in the market that support interleaved formats as their native display format, such as checkerboard for DLP televisions and row or column interleaving for some LCD-based displays, it is likely that the distribution formats will be converted to these display formats prior to reaching the display. The newest High-Definition Multimedia Interface specification between set-top boxes and displays, HDMI 1.4a [56], adds support for the following 3D video format structures: frame packing (for progressive and interlaced scan formats), side-by-side (half or full horizontal resolution), top-bottom (half vertical resolution only), field alternating (for interlaced formats), and line alternating (for progressive formats).[4] Therefore, the signaling of these formats over the display interface would be necessary along with the signaling of the various distribution formats.

The SEI message that has been specified in the latest version of the H.264/MPEG-4 AVC standard supports a broad set of possible frame-compatible formats. It is expected to be used throughout the delivery chain from production to distribution, through the receiving devices, and possibly all the way to the display in some cases.

A natural question that arises in regard to the deployment of frame-compatible stereo video is how to then migrate to a service that provides higher resolution for each view. Various approaches to this question are currently under study in the MPEG standardization working group – enhancing the resolution of each view with a coded resolution enhancement bitstream in a layered scalable fashion [59]. The best approach for this may involve some combination of MVC with another set of recent extensions of H.264/MPEG-4 AVC – namely the *scalable video coding* (SVC) extension [45] – perhaps along with additional new technology.

VII. CONCLUSIONS AND FURTHER WORK

3D video has drawn significant attention recently among industry, standardization forums, and academic researchers. The efficient representation and compression of stereo and multiview video is a central component of any 3D or multiview system since it defines the format to be produced, stored, transmitted and displayed. This article reviewed the recent extensions to the widely deployed H.264/MPEG-4 AVC standard that support 3D stereo and multiview video. The MVC standard includes support for improved compression of stereo and multiview video by enabling inter-view prediction as well as temporal inter-picture prediction. Another important development has been the efficient representation, coding and signaling of frame-compatible stereo video formats. Associated standards for the transport and storage of stereo and multiview video using H.222.0/MPEG-2 Systems, RTP and the ISO base media file format have also been specified, and are described in [60].

We are now witnessing the roll-out of new 3D services and equipment based on these technologies and standards. As the market evolves and new types of displays and services are offered, additional new technologies and standards will need to be introduced. For example, it is anticipated that a new 3D video format to support the generation of the large number of views required by auto-stereoscopic displays would be needed. Solutions that consider the inclusion of depth map information for this purpose are a significant area of focus for future designs, as discussed in [61].

[4] In addition to the HDMI formats relevant to this paper, also the formats left plus depth (for progressive-scan formats only), left plus depth, and graphics plus graphics-depth (for progressive-scan formats only) are specified.

ACKNOWLEDGMENT

The authors thank the experts of the Joint Video Team (JVT) of ISO/IEC MPEG and ITU-T VCEG for their contributions and fruitful discussions. We especially thank those that have contributed to the text editing, software development and conformance testing, namely Philipp Merkle, Karsten Müller, Ye-Kui Wang, Ying Chen, Purvin Pandit, Aljoscha Smolic, Sehoon Yea, Shinya Shimizu, Hideaki Kimata, Chong-Soon Lim, Dong Tian, and Teruhiko Suzuki.

REFERENCES

(Availability note: Joint Video Team (JVT) documents cited below are available at http://ftp3.itu.int/av-arch/jvt-site.)

[1] ITU-T and ISO/IEC JTC 1, "Advanced video coding for generic audio-visual services", ITU-T Recommendation H.264 and ISO/IEC 14496-10 (MPEG-4 AVC), 2010.

[2] M. E. Lukacs, "Predictive coding of multi-viewpoint image sets", *Proc. IEEE International Conf. on Acoustics, Speech and Signal Processing*, vol. 1, pp. 521-524, Tokyo, Japan, 1986.

[3] I. Dinstein, G. Guy, J. Rabany, J. Tzelgov, and A. Henik, "On the compression of stereo images: Preliminary results", Signal Processing: Image Communications, vol. 17, no. 4, pp. 373-382, Aug. 1989.

[4] M. G. Perkins, "Data compression of stereo pairs", *IEEE Trans. Communications*, vol. 40, no. 4, pp. 684-696, April 1992.

[5] ITU-T and ISO/IEC JTC 1, "Generic coding of moving pictures and associated audio information – Part 2: Video", ITU-T Recommendation H.262 and ISO/IEC 13818-2 (MPEG-2 Video), 1994.

[6] ITU-T and ISO/IEC JTC 1, "Final Draft Amendment 3", Amendment 3 to ITU-T Recommendation H.262 and ISO/IEC 13818-2 (MPEG-2 Video), ISO/IEC JTC 1/SC 29/WG 11 (MPEG) Doc. N1366, Sept. 1996.

[7] A. Puri, R. V. Kollarits, and B. G. Haskell. "Stereoscopic video compression using temporal scalability", *Proc. SPIE Conf. Visual Communications and Image Processing*, vol. 2501, pp. 745–756, 1995.

[8] X. Chen and A. Luthra, "MPEG-2 multi-view profile and its application in 3DTV", *Proc. SPIE IS&T Multimedia Hardware Architectures*, San Diego, USA, Vol. 3021, pp. 212-223, February 1997.

[9] J.-R. Ohm, "Stereo/Multiview Video Encoding Using the MPEG Family of Standards", *Proc. SPIE Conf. Stereoscopic Displays and Virtual Reality Systems VI*, San Jose, CA, Jan. 1999.

[10] G. J. Sullivan, "Standards-based approaches to 3D and multiview video coding", *Proc. SPIE Conf. Applications of Digital Image Processing XXXII*, San Diego, CA, Aug. 2009.

[11] A. Vetro, P. Pandit, H. Kimata, A. Smolic, and Y.-K. Wang, eds., "Joint Draft 8 of Multiview Video Coding", Joint Video Team (JVT) Doc. JVT-AB204, Hannover, Germany, July 2008.

[12] G. J. Sullivan, A. M. Tourapis, T. Yamakage, C. S. Lim, eds., "Draft AVC amendment text to specify Constrained Baseline profile, Stereo High profile, and frame packing SEI message", Joint Video Team (JVT) Doc. JVT-AE204, London, United Kingdom, July 2009.

[13] MPEG requirements sub-group, "Requirements on Multi-view Video Coding v.7", ISO/IEC JTC 1/SC 29/WG 11 (MPEG) Doc. N8218, Klagenfurt, Austria, July 2006.

[14] J. Konrad and M. Halle, "3-D Displays and Signal Processing – An Answer to 3-D Ills?", *IEEE Signal Processing Magazine*, Vol. 24, No. 6, Nov. 2007.

[15] N. A. Dodgson, "Autostereoscopic 3D Displays", *IEEE Computer*, vol. 38, no. 8, pp. 31-36, Aug. 2005.

[16] A. Smolic and P. Kauff, "Interactive 3-D video representation and coding technologies", *Proceedings of the IEEE*, vol. 93, no. 1, pp 98-110, Jan. 2005.

[17] T. Fujii, T. Kimono, and M. Tanimoto, "Free-viewpoint TV system based on ray-space representation", *Proc. SPIE ITCom*, vol. 4864-22, pp. 175-189, 2002.

[18] P. Kauff, O. Schreer, and R.Tanger, "Virtual Team User Environments - A Mixed Reality Approach for Immersive Tele-Collaboration", Int. Workshop on Immersive Telepresence (ITP 2002), pp.1-4, January 2002.

[19] I. Feldmann, O. Schreer, P. Kauff, R. Schäfer, Z. Fei, H.J.W. Belt, Ò. Divorra Escoda, "Immersive Multi-User 3D Video Communication", Proc. of International Broadcast Conference (IBC 2009), Amsterdam, Netherlands, September 2009.

[20] D. Florencio and C. Zhang, "Multiview video Compression and Streaming Based on Predicted Viewer Position", *Proc. IEEE International Conf. on Acoustics, Speech and Signal Processing*, Taipei, Taiwan, Apr. 2009.

[21] B. Wilburn, et al., "High Performance Imaging Using Large Camera Arrays", *ACM Transactions on Graphics*, vol. 24, no. 3, pp. 765-776, July 2005.

[22] R. Raskar and J. Tumblin, Computational Photography: Mastering New Techniques for Lenses, Lighting, and Sensors, A K Peters, Ltd., ISBN 978-1-56881-313-4, 2010.

[23] MPEG video sub-group chair (J.-R. Ohm), "Submissions received in CfP on Multiview Video Coding", ISO/IEC JTC 1/SC 29/WG 11 (MPEG) Doc. M12969, Bangkok, Thailand, Jan. 2006.

[24] MPEG video and test sub-groups, "Subjective test results for the CfP on Multi-view Video Coding", ISO/IEC JTC 1/SC 29/WG 11 (MPEG) Doc. N7799, Bangkok, Thailand, Jan. 2006.

[25] K. Müller, P. Merkle, A. Smolic, and T. Wiegand, "Multiview Coding using AVC", ISO/IEC JTC 1/SC 29/WG 11 (MPEG) Doc. M12945, Bangkok, Thailand, Jan. 2006.

[26] E. Martinian, S. Yea, and A. Vetro, "Results of Core Experiment 1B on Multiview Coding", ISO/IEC JTC 1/SC 29/WG 11 (MPEG) Doc. M13122, Montreux, Switzerland, Apr. 2006.

[27] E. Martinian, A. Behrens, J. Xin, A. Vetro, and H. Sun, "Extensions of H.264/AVC for Multiview Video Compression", *Proc. IEEE International Conf. on Image Processing*, Atlanta, USA, Oct. 2006

[28] MPEG video sub-group, "Technologies under Study for Reference Picture Management and High-Level Syntax for Multiview Video Coding", ISO/IEC JTC 1/SC 29/WG 11 (MPEG) Doc. N8018, Montreux, Switzerland, Apr. 2006.

[29] Y.-K. Wang, Y. Chen, and M. M. Hannuksela, "Time-first coding for multi-view video coding", Joint Video Team (JVT) Doc. JVT-U104, Hangzhou, China, Oct. 2006.

[30] A. Vetro, Y. Su, H. Kimata, and A. Smolic, eds., "Joint Multiview Video Model 2.0", Joint Video Team (JVT) Doc. JVT-U207, Hangzhou, China, Oct. 2006.

[31] Y. L. Lee, J. H. Hur, Y. K. Lee, K. H. Han, S. H. Cho, N. H. Hur, J. W. Kim, J. H. Kim, P. Lai, A. Ortega, Y. Su, P. Yin, and C. Gomila, "CE11: Illumination compensation", Joint Video Team (JVT) Doc. JVT-U052, Hangzhou, China, 2006.

[32] J.H. Hur, S. Cho, and Y.L. Lee, "Adaptive local illumination change compensation method for H.264/AVC-based multiview video coding", *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 17, no. 11, pp. 1496-1505, Nov. 2007.

[33] P. Lai, A. Ortega, P. Pandit, P. Yin, and C. Gomila, "Adaptive reference filtering for MVC", Joint Video Team (JVT) Doc. JVT-W065, San Jose, CA, April 2007.

[34] P. Lai, A. Ortega, P. Pandit, P. Yin, and C. Gomila, "Focus mismatches in multiview systems and efficient adaptive reference filtering for multiview video coding", *Proc. SPIE Conference on Visual Communications and Image Processing*, San Jose, CA, Jan. 2008.

[35] H.S. Koo, Y.J. Jeon, and B.M. Jeon, "MVC motion skip mode", Joint Video Team (JVT) Doc. JVT-W081, San Jose, CA, April 2007.

[36] H.S. Koo, Y.J. Jeon, and B.M. Jeon, "Motion information inferring scheme for multi-view video coding. *IEICE Transactions on Communications*, E91-B(4), pp. 1247-1250, 2008.

[37] E. Martinian, A. Behrens, J. Xin, A. Vetro, "View synthesis for multiview video compression", *Proc. Picture Coding Symposium*, Beijing, China, 2006.

[38] S. Yea and A. Vetro, "View synthesis prediction for multiview video coding", *Image Communication*, vol. 24, no. 1-2, pp. 89-100, Jan. 2009.

[39] M. Kitahara, H. Kimata, S. Shimizu, K. Kamikura, Y. Yashima, K. Yamamoto, T. Yendo, T. Fujii, and M. Tanimoto, "Multi-view video coding using view interpolation and reference picture selection", *Proc. IEEE International Conference on Multimedia & Expo*, Toronto, Canada, pp. 97-100, July 2006.

[40] D. Tian, P. Pandit, P. Yin, and C. Gomila, "Study of MVC coding tools", Joint Video Team (JVT) Doc. JVT-Y044, Shenzhen, China, Oct. 2007.

[41] T. Wiegand, G. J. Sullivan, G. Bjøntegaard, and A. Luthra, "Overview of the H.264/AVC video coding standard", *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 13, no. 7, pp. 560-576, July 2003.

[42] G. J. Sullivan and T. Wiegand, "Video compression – from concepts to the H.264/AVC standard", *Proceedings of IEEE*, vol. 93, no. 1, pp. 18-31, Jan. 2005.

[43] D. Marpe, T. Wiegand, and G. J. Sullivan, "The H.264 / MPEG4 Advanced Video Coding standard and its applications", *IEEE Communications Magazine*, vol. 44, no. 8, pp. 134-144, Aug. 2006.

[44] Y. Chen, Y.-K. Wang, K. Ugur, M. M. Hannuksela, J. Lainema, and M. Gabbouj, "3D video services with the emerging MVC standard", EURASIP Journal on Advances in Signal Processing, 2009.

[45] H. Schwarz, D. Marpe, and T. Wiegand: "Overview of the Scalable Video Coding Extension of the H.264/AVC Standard", IEEE Transactions on Circuits and Systems for Video Technology, Special Issue on Scalable Video Coding, vol. 17, no. 9, pp. 1103-1120, September 2007.

[46] Y. Chen, P. Pandit, S. Yea, and C. S. Lim, eds., "Draft reference software for MVC (JMVC 6.0)", Joint Video Team (JVT) Doc. JVT-AE207, London, United Kingdom, July 2009.

[47] P. Merkle, A. Smolic, K. Mueller, and T. Wiegand, "Efficient Prediction Structures for Multiview Video Coding", *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 17, no. 11, Nov. 2007.

[48] Y. Su, A. Vetro, and A. Smolic, "Common Test Conditions for Multiview Video Coding", Joint Video Team (JVT) Doc. JVT-U211, Hangzhou, China, October 2006.

[49] G. Bjøntegaard, "Calculation of average PSNR differences between RD-curves", ITU-T SG16/Q.6, Doc. VCEG-M033, Austin, TX, April 2001.

[50] T. Chen, Y. Kashiwagi, C.S. Lim, and T. Nishi, "Coding performance of Stereo High Profile for movie sequences", Joint Video Team (JVT) Doc. JVT-AE022, London, United Kingdom, July 2009.

[51] M. Droese and C. Clemens, "Results of CE1-D on multiview video coding", ISO/IEC JTC 1/SC 29/WG 11 (MPEG) Doc. M13247, Montreux, Switzerland, Apr. 2006.

[52] L. Stelmach, W. J. Tam, D. Meegan, and A. Vincent, "Stereo image quality: effects of mixed spatio-temporal resolution", *IEEE Transactions on Circuits and Systems for Video Technology*, Vol. 10, No. 2, pp. 188-193, Mar. 2000.

[53] C. Fehn, P. Kauff, S. Cho, H. Kwon, N. Hur, and J. Kim, "Asymmetric coding of stereoscopic video for transmission over T-DMB", *Proc. 3DTV-CON 2007*, Kos, Greece, May 2007.

[54] H. Brust, A. Smolic, K. Müller, G. Tech, and T. Wiegand, "Mixed Resolution Coding of Stereoscopic Video for Mobile Devices", *Proc. 3DTV-CON 2009*, Potsdam, Germany, May 2009.

[55] Dolby Laboratories, "Dolby Open Specification for Frame-Compatible 3D Systems", Issue 1, available at http://www.dolby.com/uploadedFiles/Assets/US/Doc/Professional/3DFrameCompatibleOpenStandard.pdf, April 2010.

[56] HDMI Founders, "HDMI Specification", version 1.4a, available at http://www.hdmi.org/manufacturer/specification.aspx, March 2010.

[57] D.K. Broberg, "Infrastructures for Home Delivery, Interfacing, Captioning, and Viewing of 3D Content", *this special issue.*

[58] Cable Television Laboratories, "Content Encoding Profiles 3.0 Specification OC-SP-CEP3.0-I01-100827", version I01, available at http://www.cablelabs.com/specifications/OC-SP-CEP3.0-I01-100827.pdf, August 2010.

[59] G. J. Sullivan, W. Husak, A. Luthra for MPEG Requirements Sub-Group, "Problem statement for scalable resolution enhancement of frame-compatible stereoscopic 3D video", ISO/IEC JTC 1/SC 29/WG 11 (MPEG) Doc. N11526, Geneva, Switzerland, July 2010.

[60] T. Schierl and S. Narasimhan, "Transport and storage systems for 3D video using MPEG-2 systems, RTP, and ISO file formats", *this special issue.*

[61] K. Müller, P. Merkle, and T. Wiegand, "3D video representation using depth maps", *this special issue*.

Anthony Vetro (S'92–M'96–SM'04–F'11) received the B.S., M.S., and Ph.D. degrees in electrical engineering from Polytechnic University, Brooklyn, NY. He joined Mitsubishi Electric Research Labs, Cambridge, MA, in 1996, where he is currently a Group Manager responsible for research and standardization on video coding, as well as work on display processing, information security, speech processing, and radar imaging. He has published more than 150 papers in these areas. He has also been an active member of the ISO/IEC and ITU-T standardization committees on video coding for many years, where he has served as an ad-hoc group chair and editor for several projects and specifications. Most recently, he was a key contributor to the Multiview Video Coding extension of the H.264/MPEG-4 AVC standard. He also serves as Vice-Chair of the U.S. delegation to MPEG.

Dr. Vetro is also active in various IEEE conferences, technical committees, and editorial boards. He currently serves on the Editorial Boards of IEEE Signal Processing Magazine and IEEE MultiMedia, and as an Associate Editor for IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY and IEEE TRANSACTIONS ON IMAGE PROCESSING. He served as Chair of the Technical Committee on Multimedia Signal Processing of the IEEE Signal Processing Society and on the steering committees for ICME and the IEEE TRANSACTIONS ON MULTIMEDIA. He served as an Associate Editor for IEEE Signal Processing Magazine (2006–2007), as Conference Chair for ICCE 2006, Tutorials Chair for ICME 2006, and as a member of the Publications Committee of the IEEE TRANSACTIONS ON CONSUMER ELECTRONICS (2002–2008). He is a member of the Technical Committees on Visual Signal Processing & Communications, and Multimedia Systems & Applications of the IEEE Circuits and Systems Society. He has also received several awards for his work on transcoding, including the 2003 IEEE Circuits and Systems CSVT Transactions Best Paper Award.

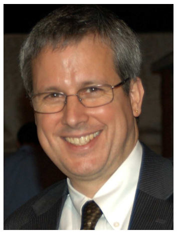
Thomas Wiegand (M'05–SM'08–F'11) is a professor at the department of Electrical Engineering and Computer Science at the Berlin Institute of Technology, chairing the Image Communication Laboratory, and is jointly heading the Image Processing department of the Fraunhofer Institute for Telecommunications - Heinrich Hertz Institute, Berlin, Germany. He received the Dipl.-Ing. degree in Electrical Engineering from the Technical University of Hamburg-Harburg, Germany, in 1995 and the Dr.-Ing. degree from the University of Erlangen-Nuremberg, Germany, in 2000. He joined the Heinrich Hertz Institute in 2000 as the head of the Image Communication group in the Image Processing department. His research interests include video processing and coding, multimedia transmission, as well as computer vision and graphics

From 1993 to 1994, he was a Visiting Researcher at Kobe University, Japan. In 1995, he was a Visiting Scholar at the University of California at Santa Barbara, USA. From 1997 to 1998, he was a Visiting Researcher at Stanford University, USA and served as a consultant to 8x8, Inc., Santa Clara, CA, USA. From 2006-2008, he was a consultant to Stream Processors, Inc., Sunnyvale, CA, USA. From 2007-2009, he was a consultant to Skyfire, Inc., Mountain View, CA, USA. Since 2006, he has been a member of the technical advisory board of Vidyo, Inc., Hackensack, NJ, USA.

Since 1995, he has been an active participant in standardization for multimedia with successful submissions to ITU-T VCEG, ISO/IEC MPEG, 3GPP, DVB, and IETF. In October 2000, he was appointed as the Associated Rapporteur of ITU-T VCEG. In December 2001, he was appointed as the Associated Rapporteur / Co-Chair of the JVT. In February 2002, he was appointed as the Editor of the H.264/MPEG-4 AVC video coding standard and its extensions (FRExt and SVC). From 2005-2009, he was Co-Chair of MPEG Video.

In 1998, he received the SPIE VCIP Best Student Paper Award. In 2004, he received the Fraunhofer Award and the ITG Award of the German Society for Information Technology. The projects that he co-chaired for development of the H.264/AVC standard have been recognized by the 2008 ATAS Primetime Emmy Engineering Award and a pair of NATAS Technology & Engineering Emmy Awards. In 2009, he received the Innovations Award of the Vodafone Foundation, the EURASIP Group Technical Achievement Award, and the Best Paper Award of IEEE Transactions on Circuits and Systems for Video Technology. In 2010, he received the Eduard Rhein Technology Award. Professor Wiegand was elected Fellow of the IEEE in 2011 'for his contributions to video coding and its standardization.'

He was a Guest Editor for the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY for its Special Issue on the H.264/AVC Video Coding Standard in July 2003, its Special Issue on Scalable Video Coding-Standardization and Beyond in September 2007, and its Special Section on the Joint Call for Proposals on High Efficiency Video Coding (HEVC) Standardization. Since January 2006, he has been an Associate Editor of IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY.



Gary J. Sullivan (S'83–M'91–SM'01–F'06) received the B.S. and M.Eng. degrees in electrical engineering from the University of Louisville J.B. Speed School of Engineering, Louisville, KY, in 1982 and 1983, respectively, and the Ph.D. and Engineer degrees in electrical engineering from the University of California, Los Angeles, in 1991. He has held leadership positions in a number of video and image coding standardization organizations since 1996, including chairmanship or co-chairmanship of the ITU-T Video Coding Experts Group (VCEG), the video subgroup of the ISO/IEC Moving Picture Experts Group (MPEG), the ITU-T/ISO/IEC Joint Video Team (JVT), the ITU-T/ISO/IEC Joint Collaborative Team on Video Coding (JCT-VC), and the JPEG XR subgroup of the ITU-T/ISO/IEC Joint Photographic Experts Group (JPEG). He is a video/image technology architect in the Windows Ecosystem Engagement team of Microsoft Corporation. At Microsoft he designed and remains lead engineer for the DirectX Video Acceleration (DXVA) video decoding feature of the Microsoft Windows operating system. Prior to joining Microsoft in 1999, he was the manager of Communications Core Research at PictureTel Corporation. He was previously a Howard Hughes Fellow and Member of the Technical Staff in the Advanced Systems Division of Hughes Aircraft Corporation and a Terrain-Following Radar (TFR) System Software Engineer for Texas Instruments. His research interests and areas of publication include image and video compression and rate-distortion optimization, video motion estimation and compensation, scalar and vector quantization, and scalable, multiview and loss-resilient video coding.

Dr. Sullivan has received the IEEE Consumer Electronics Engineering Excellence Award, the INCITS Technical Excellence Award, the IMTC Leadership Award, the J.B. Speed Professional Award in Engineering, the Microsoft Technical Achievement in Standardization Award, and the Microsoft Business Achievement in Standardization Award. The standardization projects that he led for development of the H.264/MPEG-4 AVC video coding standard have been recognized by an ATAS Primetime Emmy Engineering Award and a pair of NATAS Technology & Engineering Emmy Awards. He is a Fellow of the IEEE and SPIE. He was a Guest Editor for the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY for its Special Issue on the H.264/AVC Video Coding Standard in July 2003 and its Special Issue on Scalable Video Coding—Standardization and Beyond in September 2007.