INVITED PAPER

MPEG Immersive Video Coding Standard

Immersive volumetric content, which can be captured by multiple cameras, enables six degrees of freedom (6DoF) for the end users. This article provides a comprehensive overview of the MPEG Immersive Video (MIV) codec as well as a description of reference software assets including experimental results.

By Jill M. Boyce, *Fellow IEEE*, Renaud Doré, Adrian Dziembowski, Julien Fleureau, Joel Jung, Bart Kroon, *Member IEEE*, Basel Salahieh, *Senior Member IEEE*, Vinod Kumar Malamal Vadakital, and Lu Yu, *Senior Member IEEE*

ABSTRACT | This article introduces the ISO/IEC MPEG Immersive Video (MIV) standard, MPEG-I Part 12, which is undergoing standardization. The draft MIV standard provides support for viewing immersive volumetric content captured by multiple cameras with six degrees of freedom (6DoF) within a viewing space that is determined by the camera arrangement in the capture rig. The bitstream format and decoding processes of the draft specification along with aspects of the Test Model for Immersive Video (TMIV) reference software encoder, decoder, and renderer are described. The use cases, test conditions, quality assessment methods, and experimental results are provided. In the TMIV, multiple texture and geometry views are coded as atlases of patches using a legacy 2-D video codec, while optimizing for bitrate, pixel rate, and quality. The design of the bitstream format and decoder is based on the visual volumetric video-based coding (V3C) and video-based point cloud compression (V-PCC) standard, MPEG-I Part 5.

KEYWORDS | Immersive media; MPEG-I; multiview compression; video-based point cloud compression (V-PCC); visual volumetric video-based coding (V3C); volumetric representation.

I. INTRODUCTION

MPEG is developing a standard for coding immersive video, called MPEG Immersive Video (MIV), as part 12 of the ISO/IEC MPEG-I family of standards. The MIV standard is being designed to provide the capability to compress a representation of a real or virtual 3-D scene captured by multiple real or virtual cameras.

A key characteristic of immersive video playback is that the viewer is in control of the view position and orientation of the content. Unlike 360° video that is limited to three degrees of freedom, representing the orientation, immersive video provides playback with six degrees of freedom (6DoF) of view position and orientation within a limited range of motion. The orientation can be represented as three angles, yaw, pitch, and roll, starting from an initial direction, or equivalently with a nonambiguous unit quaternion. With 360° video, the viewer's perspective of the video content is as if the viewer were in the center of a sphere, looking out, with all content at the same distance away from the viewer to the sphere surface. The viewer may change orientation, to select which portion of the sphere's surface can be seen. 360° video is not capable of supporting motion parallax, in which the relative position of objects changes based on the viewer's position with respect to the objects. The lack of motion parallax is contrary to viewers' experience in the real world, which can lead to discomfort and even sickness for some viewers.

Manuscript received May 31, 2020; revised October 17, 2020; accepted February 19, 2021. Date of publication March 10, 2021; date of current version August 20, 2021. The work of Adrian Dziembowski was supported by the Ministry of Education and Science of Republic of Poland. *(Corresponding author: Jill M. Boyce.)*

Jill M. Boyce is with Intel Corporation, Hillsboro, OR 97124 USA (e-mail: jill.boyce@intel.com).

Renaud Doré and **Julien Fleureau** are with Interdigital, 35576 Cesson-Sévigné, France (e-mail: renaud.dore@interdigital.com; julien.fleureau@interdigital.com).

Adrian Dziembowski is with the Institute of Multimedia Telecommunications, Poznań University of Technology, 60-965 Poznań, Poland (e-mail: adrian.dziembowski@put.poznan.pl).

Joel Jung is with Tencent MediaLab, Palo Alto, CA 94306 USA (e-mail: joejung@tencent.com).

Bart Kroon is with Philips Research Eindhoven, 5656 AE Eindhoven, The Netherlands (e-mail: bart.kroon@philips.com).

Basel Salahieh is with Intel Corporation, Santa Clara, CA 95054 USA (e-mail: basel.salahieh@intel.com).

Vinod Kumar Malamal Vadakital is with Nokia Technologies, 33100 Tampere, Finland (e-mail: vinod.malamalvadakital@nokia.com).

Lu Yu is with Zhejiang University, Hangzhou 310027, China (e-mail: yul@zju.edu.cn).

This article has supplementary downloadable material available at https://doi.org/10.1109/JPROC.2021.3062590, provided by the authors.

Digital Object Identifier 10.1109/JPROC.2021.3062590

Immersive video with 6DoF supports motion parallax, providing a viewing experience with greater similarity to that experienced in the real world, akin to that provided by a virtual reality (VR) game. In addition, immersive video may be acquired by physical video camera systems, allowing a viewer to traverse a real-world 3-D scene captured by cameras with high fidelity and resolution.

The MIV standard can be used in many VR, augmented reality (AR), and mixed reality (MR) use cases. Immersive video playback devices include Head-Mounted Displays (HMDs), holographic displays, or ordinary 2-D displays with input of the viewers' position and orientation. Sports viewing is an example use case which benefits from immersive video. A viewer can choose to watch a sporting event from any desired position and orientation. Education and training use cases provide a student with a 3-D view of objects and scenes, seen from a variety of perspectives. Immersive video makes video conferencing/telepresence and virtual tourism more realistic.

An immersive video encoder processes multiple input views, to enable the rendering of any intermediate viewpoint selected by the viewer. Using existing Multiview and 3-D video codecs such as MV-HEVC or 3-D HEVC would require a very high number of samples, likely exceeding the capability of devices, or would require embedding a nontrivial automatic and flexible view selector. In addition, these standards were designed in another context, with more restricted viewing area, and views from a narrower direction. For instance, the MV-HEVC block-based inter-view motion compensation uses block translational motion for inter-view texture prediction, which is optimal only when cameras are coplanar and have the same intrinsics, with significant overlap between the cameras, and all samples within a block have the same depth.

The MIV coding framework has been designed to accommodate any camera arrangement, with the encoder selecting the most appropriate information to be signaled, to enable the rendering of any intermediate, noncaptured, view.

The development of the MIV standard began with the definition of requirements for MPEG-I Phase 1b [1] including head-motion parallax. An exploration activity, called as 3DoF+, led to the issuance of a Call for Proposals (CfP) [2] by the MPEG-I Visual group in January 2019, with responses reviewed in March 2019. Five responses were received for the call, which were evaluated based on objective metrics and subjective viewing. A first version of the working draft (WD) of the standard was defined, based on combining aspects from multiple of the CfP responses. Like other video codecs standardized by MPEG, the specification defines a normative bitstream format and decoding process, while leaving flexibility to nonnormative encoder and post-processing operations. To enable collaborative development within MPEG, a Test Model for Immersive Video (TMIV) was defined, which provides software and documentation for a reference encoder, decoder, and renderer.

Fig. 1. *Two input views with corresponding depth maps: views 0 and 15, sequence Painter.*

MPEG holds four meetings per year, and in each meeting cycle, an updated version of the WD and TMIV is provided. Since the fourth version of the WD, the MIV specification is closely aligned with the visual volumetric video-based coding (V3C)/video-based point cloud compression (V-PCC) specification [3], because of the existence of technical overlap between the bitstream format and decoder definitions. MIV normatively references V3C and provides extensions to it. Significant differences remain between the MIV and V3C/V-PCC input and output formats, reference encoders, and reference renderers.

The MIV reached Committee Draft status in the July 2020 MPEG meeting [4]. Draft International Standard (DIS) is expected in April 2021, and Final Draft International Standard (FDIS), signifying finalization of the standard, is expected in 2021.

The organization of this article is as follows. A high-level overview of the MIV codec is provided in Section II. A description of the TMIV is provided in Section III. Section IV describes the MIV rendering process. Section V provides additional information related to alignment with the V3C/V-PCC standard. The common test conditions (CTCs) as well as experimental results using the TMIV are provided in Section VI, and Section VII concludes this article and highlights future work.

II. MIV CODEC OVERVIEW

A. Source Material

The inputs to the MIV codec are based on the Multiview Video + Depth (MVD) [6] representation of video data, with each source view represented by frames of geometry (e.g., spatial information) and attribute samples (e.g., texture, transparency, surface normals, reflectance), and with view parameters to enable 3-D reconstruction.

For each input view, spatial information is provided in a geometry map that combines depth and occupancy. Sample values equal to zero indicate that the sample is unoccupied, whereas nonzero values represent depth information for the sample. The depth information represents the distance between the camera and the objects in the scene, in "scene units" defined in the specification. In order to better suit the human visual system, depth information is

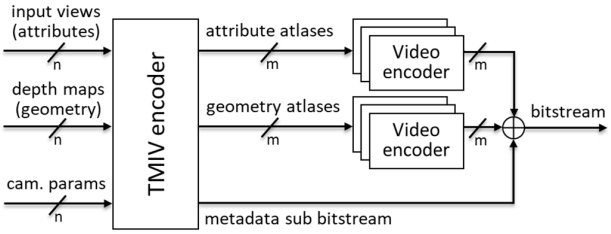


Fig. 2. ***High-level block diagram of an MIV encoder.***

typically stored as normalized disparity instead of distance in meters [7]. Fig. 1 presents two views with corresponding depth maps. Brighter samples correspond to higher normalized disparity and thus smaller distance between an object and the camera.

Geometry/depth maps may be obtained in different ways. For computer-generated (CG) sequences they may be synthetic, for example, rendered by tools such as Blender.[1] Depth maps may also be acquired by depth sensors, for instance, by illuminating a scene in infrared using defined patterns of points (e.g., Microsoft Kinect device [8]) or by measuring time of flight (ToF) of an emitted infrared light (ToF cameras [9]).

However, multiview sequences are typically captured by multicamera systems [10]–[13]. In such systems, depth maps are estimated from input views [14]–[16]. Depth estimation is a crucial step impacting immersive video coding performance and quality of content presented to the viewer, but it is out of the scope of the MIV standard and of this article.

The MIV standard allows any arrangement of cameras, however the quality of the representation of the 3-D scene is highly dependent on the density of cameras and their placement. The available range of supported viewer motion is limited by the range captured by the cameras and represented in the coded MIV bitstream. Interpolation quality is impacted by the sampling density of the 3-D scene by cameras, based on the distances between cameras and the distances from the cameras to the objects in the scene. Hollow shapes or thin geometries are also more challenging to capture. Such situations can be addressed by increasing the camera density and by improving algorithms at the decoder and renderer to bring graceful degradation. The different source materials used for testing during the MIV development have good camera density based on 10–25 cameras located within a volume corresponding to a realistic range of viewer motion. For the natural contents (NCs), the test content conforms to camera configurations that could be available in practical applications.

B. Codec Structure, Atlases, and Patches

Fig. 2 shows a high-level block diagram of an encoder for the MIV. A key function of an MIV encoder is to form one or more attribute and geometry atlases and generate metadata to describe the atlases, by compositing patches extracted from the input views. The attribute and geometry atlases are encoded as a video with a video encoder, while the metadata is encoded following the MIV standard, using the TMIV reference software encoder.

As with the V3C standard, the MIV can be used with any video coding standard. HEVC has been used during the development process, because it has been widely deployed in products, and the HEVC HM reference software [17] is used in the CTCs [24]. V3C explicitly defines codec profiles, which enables support for HEVC, AVC, and VVC, and provides a mechanism to indicate other codecs. This flexible codec profile mechanism could be used to indicate MV-HEVC or 3-D HEVC in conjunction with the MIV, but those codecs have not been widely deployed in products, so have not been the focus of this work. The rationale for forming patches and atlases is to reduce inter-view redundancy of data while preserving the quality of content presented to the user of immersive video system. An atlas is a picture containing visual data from many input views. In order to produce atlases, two main steps are performed: pruning and packing.

Inter-view redundancy is removed in a pruning step, where pixels are reprojected between different views. If an object is visible in two or more views, it is pruned from all views except for one. The left column in Fig. 3 shows three input views, one in each row. The middle column shows the pruned views in which all the samples of the center view are retained, and the samples from the top and bottom views which were also visible in the central view were pruned. This representation is similar to the layered depth video (LDV) representation [18]–[20], which contains a layer of texture and depth of foreground objects, and additional layers of regions occluded by the foreground objects. The pruned views of the MIV are a representation of the occluded parts not visible from the center view. A segmentation algorithm is then used to create patches from the pruned views and pack the patches into an atlas along with the center view. The right column shows an atlas generated by packing patches from the three pruned

Fig. 3. ***Input views (left), views after pruning (middle), and after packing into atlas (right); sequence Kitchen.***

[1] https://www.blender.org/

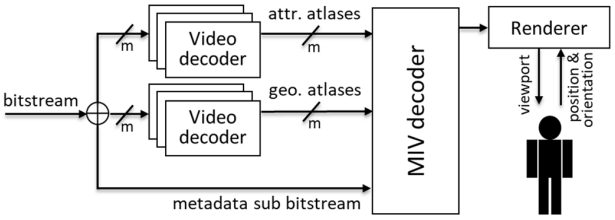


Fig. 4. *High-level block diagram of the MIV decoder/renderer.*

views. This process of pruning views, forming patches, and packing them into atlases allows for a compact representation of the scene with minimal pixel redundancies.

The decoder/renderer shown in Fig. 4 first performs video decoding, that is, HEVC decoding, then reconstructs views by reversing the atlas packing process. The MIV bitstream contains metadata indicating the packing order, position, rotation, and source view number of each patch in the atlas, which are used in the reconstruction process.

The MIV specification normatively specifies the operation of an MIV decoder, with conformance points defined. A nonnormative hypothetical reference renderer (HRR) is also described in the specification, but renderer implementations are not required to follow the operations of the HRR.

C. View Parameters

The rendering process requires knowledge of parameters for each view, for example, representing the real or virtual camera corresponding to the view. These view parameters are carried in the MIV bitstream and include the projection plane size, projection type, camera intrinsics (specific to the projection type), camera extrinsics, and depth quantization parameters (QPs). Perspective, equirectangular, and orthographic projections are supported.

The intrinsic parameters of a camera provide the relationship between a sample position within an image frame and a ray origin and direction. For perspective cameras, the intrinsic parameters are represented as a projection matrix that contains focal length and the position of the principal point of the camera matrix. The camera model presumes that distortion handling is done as a preprocessing step. For equirectangular projection, the image row and column translate to a latitude and longitude, respectively, and the intrinsic parameters are the latitude and longitude range of the projection plane. For orthographic projection, all samples have the same ray direction, but the ray origins of the samples form a plane in scene space. As such, the orthographic camera model differs from perspective and equirectangular models in that it does not have a cardinal point. In this case, the intrinsic parameters are the width and height of the orthographic plane.

The extrinsic parameters of a camera represent the camera pose where the position is a 3-D Cartesian coordinate, and orientation is a unit quaternion. The camera extrinsics allow the cameras to be located in a common coordinate system, enabling view interpolation from multiple views.

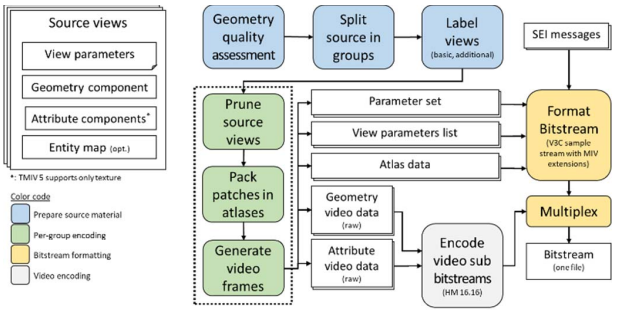


Fig. 5. *High-level scheme of the TMIV group-based encoder.*

III. TMIV DESCRIPTION

A. Overview

The test model consists of the TMIV document and publicly available reference software,[2] providing an encoder and decoder/renderer. The document serves as a source of general tutorial information on the MIV design. It defines terminology used, process and data flow, operating modes, and description of algorithmic components accepted by the MPEG Video subgroup for the test model.

1) Encoder Process Overview: As depicted in Fig. 5, the TMIV encoder assesses the geometry quality, optionally splits the source views into groups, and labels each view as "basic" or "additional." All samples in a basic view are represented in an atlas, while additional views may have samples pruned and packed into one or more other atlas(es). Then, each group is encoded separately, as depicted in Fig. 6. The texture and geometry atlases are encoded separately as videos using the HM HEVC reference model [17], and the coded video bitstreams are multiplexed together with metadata sub-bitstream to generate the MIV-compliant bitstream.

The encoding of each group (Fig. 6) consists of automatic selection of parameters, including the number of atlases and atlas frame sizes, optionally separating the

[2]https://gitlab.com/mpeg-i-visual/tmiv

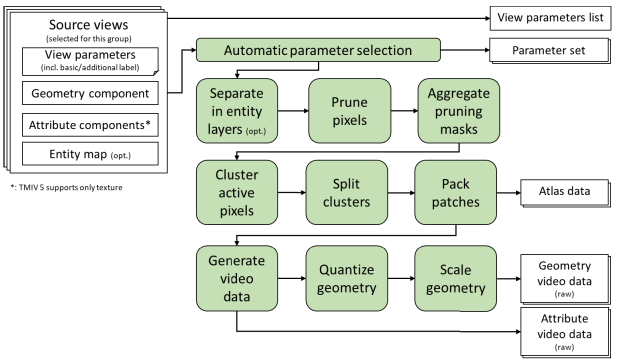


Fig. 6. *High-level scheme of the TMIV single-group encoder.*

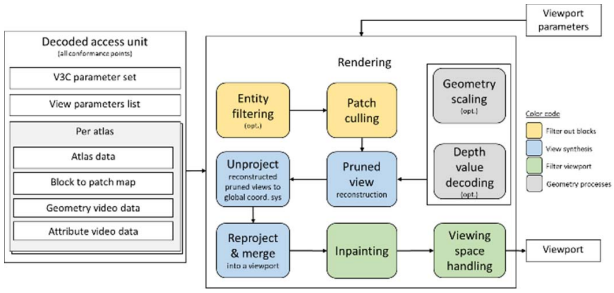


Fig. 7. ***High-level block diagram of the TMIV renderer.***

views into entity layers, pruning of the redundant information, aggregation of the pruning masks, and clustering of the preserved pixels. Patches are formed and packed, and video data is generated per group. The geometry maps are quantized and scaled.

2) Decoder Process Overview: The TMIV decoder/renderer follows the MIV decoding process described in the specification and implements a renderer which, to a large extent, keeps on track with the successive versions of the nonnormative HRR. The rendering is composed of an optional entity filtering stage, a patch culling process that speeds up the rendering, reconstruction of the pruned views, synthesis of the requested intermediate view, the inpainting of occluded areas, and final viewing space handling.

3) Hypothetical Reference Renderer: An HRR is described in the specification although it is beyond the conformance point and therefore nonnormative. Implementations of the MIV standard are not required to exactly match its operation but may be supported by some MIV metadata. The nonnormative HRR description was included within the specification in order to clearly describe the intended usage of the MIV syntax elements which do not affect the normative operations but would negatively impact rendered video quality if interpreted in a significantly different manner than intended. As illustrated in Fig. 7, inputs are the geometry and attributes decoded atlas, and all the parameters embedded in the atlas data (AD) according to V3C specification and to the MIV extension.

For each atlas, the video-decoded geometry output (optionally downscaled) is converted back to metric depth through inverse of the $1/z$ quantization law detailed in (2). This also includes the identification of pixel validity if this information is embedded in the depth coding.

Geometry and attributes atlas pairs are inputs to the central block of Fig. 7, used to reconstruct all of the source views by processing the patches. Patch information from the V3C AD access unit and the related view parameters such as the view pose and projection type are used in the reconstruction process. The association of each atlas pixel to its corresponding patch is determined by the patch information list. A pixel-wise patch index map is first generated for each atlas to link them with their corresponding patch. In practice, the patches do not need to be defined at the pixel level and the pixel-wise patch index map is replaced by a patch index block-wise map, with the block size selected by the encoder.

Finally, a viewport is rendered by a deprojection from each related source followed by a reprojection according to the viewport coordinates input at each frame by the application, for example, the HMD pose coordinates. Since geometry conveys the depth value from the view center for perspective and equirectangular projections or orthogonally from the projection plane otherwise, a deprojection simply consists in placing a point for each pixel texture in 3-D space at that related geometry distance, while the following and reverse reprojection should cope with the fractional position of the projected point onto the target viewport image. The blending of all these source view contributions is made within a view synthesis in charge of occlusion handling and smooth rendering.

B. Distribution of Source Views in Groups

The TMIV encoder divides the source views into multiple groups: an automatic process is implemented to select the views of each group, based on the view parameters list and the number of groups. Each group is encoded independent of each other, to allow parallel processing. The grouping feature improves local coherence of projections of important regions (e.g., foreground objects/occlusions) in the atlases which improves the rendering quality. This is also beneficial in the case of multicamera rig systems where distant views have less in common, hence can be processed more efficiently in separate groups and still be multiplexed in the bitstream.

It is also possible to render from atlases of a given group separately since each group's atlas includes patches carrying its own basic view(s) and pruned additional views separately. Fig. 8 shows an example of the atlases generated using three groups for the *Frog* content. Each group

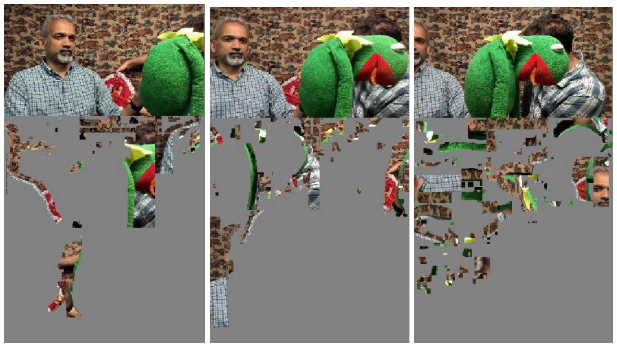


Fig. 8. ***Group-based encoding results using three groups (ordered left to right) for the Frog content. Each group has one atlas, each with one whole basic view (on the top) plus patches of multiple additional views.***

has one atlas, each with one whole basic view (on the top) plus patches of multiple additional view.

C. View Labeling

This process classifies the input views within groups into two categories: the "basic" views, corresponding to full views that are packed in an atlas as a single patch, and "additional" views, corresponding to views pruned and packed in multiple patches. It includes two steps: first, the number of basic views is determined, considering the direction deviation, the field of view, the distances, and overlap between the views. Second, the basic views are selected, considering the distance and overlap to/with a central view position.

D. Automatic Parameter Selection

1) Geometry Quality Assessment: Because of different methods used to generate depth maps, the quality of the source content geometry varies. Knowledge of the accuracy of the geometry is used to select the behavior of the MIV encoder and is signaled to the decoder. A simple quality assessment of the geometry is applied. Based on the first frame, each input view is reprojected to the position of the other views. For every reprojected pixel, it is checked if the reprojected geometry value is higher than the geometry value of the collocated pixel or any of its neighbors in the target view (in a 3×3 neighborhood). If this condition is not fulfilled, the pixel is counted as inconsistent because it appears in front of the target view, which is contradictory. This condition is checked up to a tolerance of 97% to empirically take into account quantization effects. Studies have shown that the percentage of pixels counted as inconsistent varies greatly among the set of test sequences used in the MIV CTC, with significant differences between the synthetic content and the NC. A threshold default value of 0.1% was selected, and if the inconsistent pixel percentage exceeds the threshold, the quality of the geometry is set to low.

2) Atlas Frame Size Computation: The encoder calculates the number of atlases per group and atlas frame size. This computation is related to constraints on the maximum size of a picture (in luma samples), the maximum sample rate (in Hz) of the luma samples, and a total number of allowed decoder instantiations. The following principle is applied.

1) The atlas frame width is set to the widest source view.
2) The number of atlases per group is set high enough to reach the maximum luma sample rate, without exceeding it.
3) The atlas frame height is set as large as possible within the constraints.

E. Separation in Entity Layers

The TMIV can operate in the optional entity coding mode, where entities may refer to objects, materials, or other compositions of the scene. In this mode, an input entity map is provided per source view indicating what entity each pixel in the view belongs to (i.e., an entity map is of same resolution as its source view with values representing indices, called entity indices from 0 to *maxEntities*-1). The encoder uses them to perform entity-based pruning, aggregation, clustering, and packing. Here, each patch packed within the atlases has active pixels that belong to a single entity at a time, hence each patch can be tagged by its entity ID. This enables selective encoding and/or decoding such that only entities that are of interest are transmitted and/or processed, enabling new applications and potentially resulting in bitrate savings. Samples of entity-based encoding and decoding/rendering are shown in Fig. 9 for the *Museum* content.

Fig. 9. *Entity-based TMIV coding of Museum showing three atlases (out of four) in the top row, coding all entities. Rendering results of selected entities (representing foreground entities) in the bottom row.*

F. Pruning of Redundant Pixels

Multiview content typically has inter-view redundancy. The pruner determines whether individual pixels are removed or preserved, based on their importance in rendering any intermediate viewpoint.

The pruner performs data projection between input views considering a hierarchy of views. In general, each view is pruned using information projected from views higher up in the view hierarchy. At the top of the view hierarchy, there is a basic view. It is used to prune one additional view. Then, a basic view and a view pruned in the previous step are used to prune a second additional

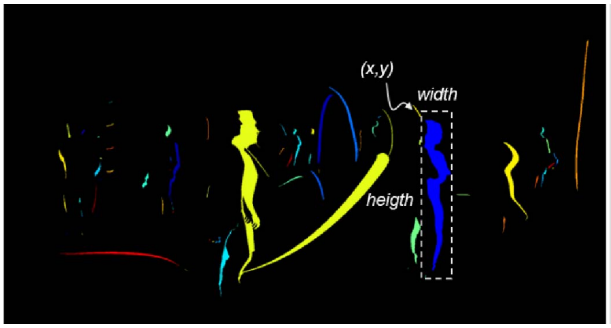


Fig. 10. *Clusters obtained on a pruned view (false colors).*

view. This algorithm repeats until all additional views are pruned.

The pruner uses two criteria to determine whether a pixel may be pruned: the pixel is synthesized from the views higher up in the hierarchy, and the difference between the source and synthesized geometry is less than a threshold. The information of whether a pixel was pruned or preserved is stored in the pruning mask, separately for each view.

In order to clean up holes and irregularities in the pruning mask, an erosion and dilation process is applied. An example of pruned views is given in Fig. 3.

When entities are used, all active pixels of a patch must belong to a single entity. Thus, the pruning masks are updated based on the entity maps such that pixels are turned on only for pixels belonging to the processed entity.

G. Pixel Clustering Into Patches

Pruning masks are accumulated at the pixel level over an intra-period. As a result, the contours in the pruning masks become thicker on areas of the geometry maps that have motion. Then, "clusters" are created, representing sets of connected pixels in a mask. When entities are used, this clustering is done separately for each entity. An example of clusters is depicted in Fig. 10.

Patches are formed from rectangular bounding boxes around the clusters. Some clusters are bounded by a large rectangle despite the number of pixels in the cluster being relatively low (such as the irregular, large yellow one in Fig. 10). To save space in the atlas, these patches are recursively split into two smaller clusters so that the resulting total area becomes smaller.

H. Patch Packing

Initially, all patches are sorted by decreasing size order. Then, each patch is packed sequentially into atlases, using the *MaxRect* algorithm [18]–[20]. Rotation of 90°, 180°, and 270° are supported, as well as vertical flip. To allow proper view restoration at the decoder side, packing must be reversible. Therefore, the packing order, position, rotation, and source view number for each patch are included in the metadata. An example of packed clusters into an atlas is given in Fig. 3. In general, the encoder outputs m atlases from n input views. An atlas contains visual data from several input views.

I. Geometry Coding

The TMIV encoder outputs pairs of geometry and attribute atlases to be coded by the video coding stage using the number of bits determined by the MIV profile. Although the MIV specification is video codec-agnostic, the HEVC Main 10 profile is utilized in the CTCs, using 10-bit depth for geometry and attributes.

In order to minimize the geometry quantization error as seen from the original view locations, the geometry quantization law is based on normalized disparity for perspective and equirectangular type and requires the transport of metadata, indicating their near and far limits $(d_{\text{near}}, d_{\text{far}})$ corresponding to the $(z_{\min}, z_{\max})$ range per view [6]. In the case of 10 bits depth, the formula to quantize the disparity as a function of the normalized disparity d is therefore

$$d_{\text{quantized}} = 1023\frac{(d - d_{\text{far}})}{(d_{\text{near}} - d_{\text{far}})}. \tag{1}$$

In the case of orthographic projection, (1) is replaced by a simple offset and scaling law. The specification conceptually allows for the mix of view projection types, but this has not been studied.

The patches may have invalid pixels within their rectangular area, either as a result of the pruning process or because these pixels were unoccupied in the source views, as may occur in synthetic source content when partial rendering by layers is activated. The pixel-level validity information is embedded in the quantized depth directly by reserving a $[0, T]$ range at the lower part of the 10 bits range, with T value signaled in the bitstream, thus compressing the 1023 range of (2) to $(1023 - T)$ without significantly increasing the geometry quantization error. The zero values in the geometry map indicating invalid pixels are, however, expected to be modified by the video compression and decompression processes and this video compression additive noise will be even more important at high QP close to the patch contours. The $[0, T]$ guard band, therefore, enables to apply a threshold of value T for reidentifying invalid pixels, thus ensuring contours quality without the cost for another dedicated occupancy bitstream.

J. Geometry Downscaling

The MIV enables reduction of pixel rate by allowing downscaling of geometry video data (GVD) for some or all atlases (Fig. 11). It has been observed that video compression artifacts cause visible view synthesis artifacts at foreground-to-background transitions [22]. By lowering the resolution of the geometry frames while also lowering the QP of the HEVC encoder, it is possible to use fewer pixels for a similar end-to-end rate distortion (RD) characteristic.

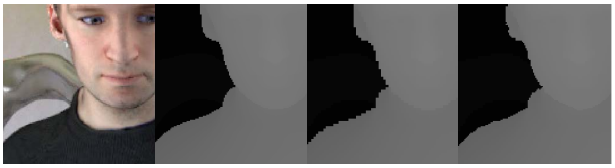


Fig. 11. *Geometry scaling with, from left to right, texture, original geometry, max-pool down plus nearest neighbor upscale, and selective erosion.*

Regular image scaling methods such as cubic interpolation would be detrimental because they cause intermediate values connecting foreground to background objects, and they may distort thin foreground objects to the point that thin lines would disappear. To counter this, the TMIV encoder performs a max-pooling operation: for a group of $N \times N$ pixels, with $1/N$ being the scaling factor, the nearest depth value is selected. When none of the pixels are occupied, the output pixel is set to nonoccupied.

The encoder thus deliberately grows the apparent size of foreground objects. This enables the decoder to approximately reconstruct the full resolution geometry frame by a sequence of bilateral morphological operations: 1) nearest-neighbor up-sampling; 2) attribute-aligned selective geometry erosion; and 3) geometry contour smoothening [22].

IV. MIV RENDERING PROCESS

The MIV draft specification includes a nonnormative reference rendering process, which is described in this section. Implementations of the MIV standard are not required to exactly match the reference rending process, providing flexibility for products to improve quality or reduce complexity.

A. Optional Entity Filtering

When the encoder has used the optional entity coding mode, use cases are supported where the application renders only a subset of the objects, for example, foreground objects only. Patches are selected by the filter if their entity index is within the subset of targeted entities.

B. Patch Culling

Patches that have no overlap with the requested target viewport can be culled to reduce the computational cost of the rendering. The patch culler follows the same sequential order as the patch creation at the encoder: the four corners of a patch are reprojected to the target view by using the minimum and maximum geometry values of the patch. The patch is culled if the area enclosed by the eight reprojected points has no overlap with the target viewport.

C. Auxiliary Atlases

Optional auxiliary atlases convey patches which will not have any direct impact on the rendering as defined in the HRR description. For example, auxiliary patches may represent shapes in 3-D needed for a quick shadow computation in case an application would implement a simplistic scene relighting. The MIV 3-D scenes are typically composed of open forms which are not sufficient to describe the complete object shapes. Another example is to convey purely geometrical information needed for collision detection or haptic application. The task of the MIV rendering process is simply to discard these auxiliary patches.

D. Pruned View Reconstruction

Source views are reconstructed separately by a process that aggregates all the patches among the different atlases belonging to a given source view. The patches which have not been culled are copied with possible rotation and flip from their position in the atlas to their position in the views, as an exact inverse operation to the patch packing in the encoder described in Section III.

This process regenerates all source views for which camera parameters were signaled, except for basic views carried as a single patch.

This can be illustrated by referring back to the packing operation example of Fig. 3 and considering that the atlas as shown in the right column and now possibly degraded during the video coding stage is converted to three views, as shown in the middle column of the same figure, of which two are pruned views.

E. View Synthesis

The final synthesis performs a visibility pass based only on the depth information followed by a shading pass adding the attribute information. A global weight factor per view is used in the blending of the shader pass to give more importance to the source views close to the viewport, by computing the distance between the viewport and each view. This is illustrated with an example in the upper part of Fig. 12 where the synthesized viewport is more heavily weighted by views 1 and 2 than by view 3, because of the closer proximity of the viewport to views 1 and 2. This weighting strategy allows to mitigate the effect of depth inconsistencies between remote cameras due to potentially

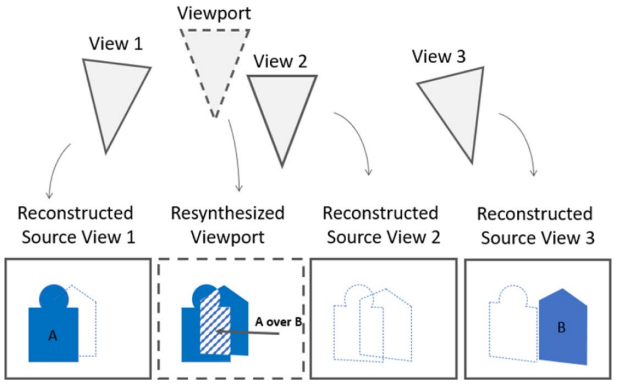


Fig. 12. *Bias in viewport synthesis when only based on unpruned parts (blue).*

insufficient calibrations and avoids sharp transitions as the viewport position moves between the camera positions. Weighting based on the distance of the rendered viewport from the reconstructed views also provides limited support for specularities in rendered viewports.

The global weight factor per view is combined with a factor computed at each fragment level of the scene to be rendered. The so-called Weight Weighting Synthesizer used for the MIV content considers that the reconstructed views are possibly pruned, sometimes extensively, and that the depth of the objects across views may be inconsistent. The lower part of Fig. 12 shows an example of the reconstruction of a viewport for a scene with three views containing two flat objects, A and B, in which views 1 and 3 transport the A and B shapes, respectively, while view 2 is completely pruned, based on view 1 containing the A object. In this example, because of depth inconsistency between the views, views 1 and 2 would see object A slightly in front of B, while view 3 would see the opposite, with object B in front of object A. For each fragment of the scene, the visibility pass uses a criterion typically based on the number of views seeing that object fragment. A simple weighting strategy based on the sole pruned information would not reflect this criterion. This is illustrated in Fig. 12 where view 2 would be completely discarded by the encoder, leading to possible transparency artifacts in which object B would appear through object A.

In order to mimic view synthesis as would be done from full unpruned source views, optional metadata signals a description of the pruning graph hierarchy used at the encoder, which helps to virtually reconstitute the original weight of each pixel of each source view [23].

The quality of the viewport rendering is related to the conjunction of pruning and synthesis algorithms. They come with several adjusted parameters which are adapted for various types of content. Since the contours of an MIV-rendered scene have a primary impact on visual saliency, it is challenging to make them smooth and clean without any transparency artifacts and sacrifices on image quality. The handling of true contours zone, therefore, constitutes the true benchmark for a synthesizer implementation.

While the conceptual renderer design illustrated in Fig. 7 shows per view operations, in practice, the renderer implementation in the TMIV operates on a per patch basis, where patches are deprojected just as before but directly reprojected onto the final viewport. This approach saves a synthesis pass and accelerates the rendering process when real-time synthesis is targeted.

V. ALIGNMENT WITH V3C/V-PCC SPECIFICATION

A. Alignment With V-PCC Specification

The MIV bitstream syntax is an extension to the V3C bitstream format [3] that is in common with V-PCC. At the highest level, there is a V3C unit stream where each V3C unit is composed of an integer number of bytes. The standard specifies the V3C sample stream to concatenate V3C units into a single bitstream. Every V3C unit is composed of a header and a payload, where the first element of the header is the unit type. The current text of the standard specifies five unit types, namely the V3C parameter set (VPS), AD, GVD, attribute video data (AVD), and occupancy video data (OVD).

The VPS is an essential unit which signals the start of a new sequence and provides information that allows the decoder to setup, such as the atlas count, frame sizes, presence of unit types, map count (i.e., number of geometry layers), attribute count and definition, and so on. The VPS can either be provided in-band or out-of-band.

For other V3C units, the header links the unit to VPS and atlas by ID. In addition, GVD and AVD link to a map, and AVD also links to an attribute (e.g., texture) and attribute partition (i.e., set of video planes) by index.

The payload of GVD, AVD, and OVD units is a video sub-bitstream. The AD unit contains an AD sub-bitstream that is structured as a network abstraction layer (NAL) unit stream, which is a generic format suitable for use in both packet-oriented and bitstream-oriented systems. Every NAL unit is either an atlas coding layer (ACL) NAL containing patch data or a non-ACL NAL. The atlas sequence parameter set (ASPS), atlas frame parameter set (AFPS), and atlas adaptation parameter set (AAPS) are non-ACL NAL units that carry infrequently changing information of the coded atlases in the V3C bitstream. These parameter sets have V-PCC and MIV extensions. The ASPS may also carry the volumetric usability information (VUI) which provides rendering hints. There are also NAL units that carry supplemental enhancement information (SEI) messages (like in HEVC).

In an MIV, a dedicated V3C unit, V3C_CAD, is reserved to carry information that is common to all atlases in the bitstream. V3C_CAD carries the AAPS and the Common Atlas Frame NAL unit. The MIV allows updating of camera- and depth-related parameters at any time within a sequence, signaled in the Common Atlas Frame NAL unit.

B. High-Level Similarities/Differences With V-PCC

The alignment of the MIV standard to MPEG-I Part 5 as a normative reference was motivated by the large similarities in the technical building blocks used in coding of point clouds based on video and the coding of immersive video as studied within MPEG. Keeping two standards with similar purpose would have increased the general implementation effort and reduced the adoption of either standard. The alignment benefit is reflected in that the majority of syntax, semantics, and decoding processes of the specification draft of the MIV are referenced from Part 5. To facilitate the MIV and allow for future volumetric video standards, the main text of Part 5 is called V3C, and the V-PCC-specific part has become an annex of Part 5. Both V-PCC and MIV are specified by a combination of V3C extension mechanisms with additional decoding and rendering processes.

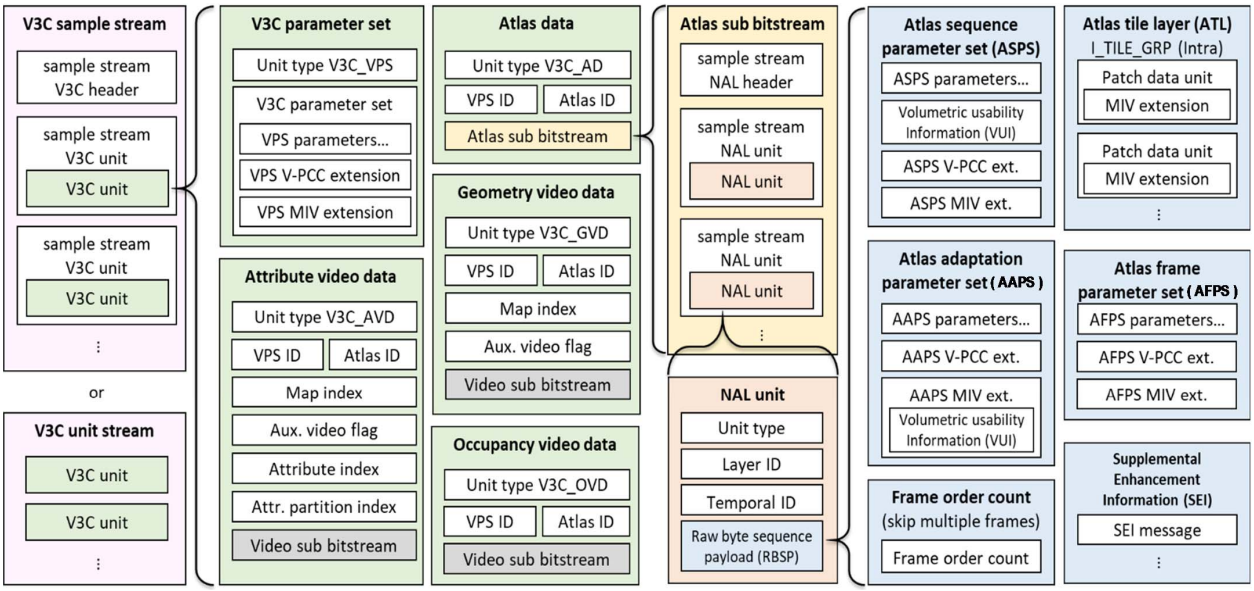


Fig. 13. ***Overview of V3C bitstream structure with V-PCC and MIV extensions.***

V3C (Fig. 13) specifies how to transmit volumetric video through the combination of AD, GVD, OVD, AVD, VUI, and SEI messages. Hereby, AD describes the position of patches per frame plus projection information that enables 3-D reconstruction. There may be multiple atlases and each atlas may have multiple attributes and multiple maps (depth planes).

The MIV and V-PCC differ in their source and output formats. The input to V-PCC is a temporal sequence of point clouds, with each point cloud containing a set of points with a scene coordinate and optional attributes per point, while the MIV source material is represented as an MVD format. The V3C portion of the specification, which is common to MIV and V-PCC, does not define reconstruction. The MIV and V-PCC describe different reconstruction methods, corresponding to their respective input formats. In the MIV, the reconstruction process is informative and nonnormative, allowing implementations to differ from the described process. The intended output of an MIV decoder/renderer is a viewport within the viewing space of the source material, with the resolution of viewport determined by the device rather than the content. V-PCC, in contrast, reconstructs the original point cloud, supporting both lossy and lossless encoder modes.

Not all aspects of V3C are supported by both derived standards, and for different reasons:

1) V-PCC uses only one atlas because pixel rate is typically not a concern when transmitting single volumetric objects. The MIV, on the other hand, needs multiple atlases to stay within the luma picture size constraint of practical video decoders when conveying full scenes albeit from a limiting viewing space. V-PCC has fixed configurations of orthographic projection planes with the volumetric object within those planes, where MIV signals view parameters for an arbitrary amount of views in the common atlas.
2) The MIV uses only one map per atlas because deocclusion functionality is implemented by partial transmission of multiple views.
3) V-PCC uses occupancy maps and attributes padding between patches, while the MIV combines depth and occupancy coding within the geometry data. Experiments in this area are ongoing and convergence may be possible.
4) The MIV does not have a lossless coding mode, restricting the related V3C tools. The reason is that perfect MVD reconstruction can only be achieved for source view positions, but the goal of the MIV is to render intermediate viewports through view interpolation.

VI. EXPERIMENTAL RESULTS

A. Common Test Conditions

During the development of a new video coding standard, multiple new tools are proposed and selected for inclusion in the draft standard when they have a positive impact on the coding framework relative to their complexity. To ensure that all participants in the standardization process use the same evaluation methods and competing proposals may be fairly compared, CTCs are defined [24] and may be updated each meeting cycle. The CTC defines test content, specifies how to generate an anchor, and provides methods and a template to report experimental results.

1) Test Content Description: The MIV standard is intended to be used in a wide range of applications, so the test sequences used for evaluating proposals have varying

Table 1 Test Sequences

Test sequence	Ref.	Type		Resolution	Frame rate	Views
ClassroomVideo	[25]	O	CG	4096 × 2048	30	15
Museum	[26]	O	CG	2048 × 2048	30	24
Hijack	[26]	O	CG	4096 × 2048	30	10
Kitchen	[27]	P	CG	1920 × 1080	30	25
Painter	[28]	P	NC	2048 × 1088	30	16
Frog	[29]	P	NC	1920 × 1080	30	13
Fencing	[30]	P	NC	1920 × 1080	25	10

O: omnidirectional (ERP), P: perspective,
CG: computer-generated, NC: natural content

characteristics, as summarized in Table 1, and illustrated in Fig. 14. The test set contains both CG content with close-to-perfect depth maps and NC with depth maps that were estimated, and in some cases post-processed. Among the sequences, three are omnidirectional, with angles of view from 180° × 90° (*Hijack*) up to 360° × 180° (*ClassroomVideo*). Four sequences were captured with perspective cameras. The resolution of the test sequences varies from Full-HD to 4K. Several different camera arrangements are considered: some sequences were captured by linear multicamera systems (*Frog*) and others by camera arrays (*Painter*, *Kitchen*) or cameras arranged within stereopairs placed on an arc (*Fencing*).

Test sequences also differ in processing difficulty. For example, there are specular and translucent objects in *Kitchen,* heavy noise in *ClassroomVideo*, fast movement in *Fencing,* and imperfect depth maps for natural sequences.

2) Coding Setup: A challenging subset of 97 frames is selected for each test sequence, corresponding to three Group of Pictures plus one I-Frame at the end.

The content is encoded using HEVC reference software HM16.16 [17]. In Section II-B, two operating modes of the standard are compared: MIV Atlas and MIV View. While in the MIV Atlas mode, views are pruned as described in Section II-B, in the MIV View mode a subset of views has been manually selected for each sequence, and the atlases are made of basic views only (no pruning is applied), with one view per atlas.

Fig. 14. ***Test sequences.***

Five rate points are considered, corresponding to a set of QPs for the attributes QP = {22, 27, 32, 37, 42} and a set of QPs for the geometry QP_d = {4, 7, 11, 15, 20} for the MIV Atlas mode, and QP_d = {9, 9, 14, 17, 21} for the MIV View mode. Two rate modes are considered, High-BR and Low-BR, corresponding, respectively, to the four lowest QPs and for highest QPs, to reflect improvements at different bitrate ranges. The QPs for the geometry where empirically selected low enough to maintain a good compromise between rendering efficiency and bit consumption.

3) Pixel Rate Constraints: The purpose of any video encoder is to minimize bitrate while preserving quality. Of course, this statement is true also for the MIV standard described in this article. However, in immersive video applications, multiple input views are processed while commercial video decoders cannot handle an unlimited amount of data. The MIV is designed with the assumption of the reuse of existing video codec implementations. Therefore, a critical constraint to consider when evaluating proposals is the "pixel rate." The CTC imposes the following constraints in contributions.

1) The combined luma sample rate across all decoders shall not exceed 1 069 547 520 samples per second, corresponding to HEVC Main10 profile level 5.2.
2) Each coded video picture size shall not exceed 8 912 896 pixels (i.e., 4096 × 2048).
3) The number of simultaneous decoder instantiations shall not exceed 4.

4) Objective Quality Assessment: In the MIV development process, quality is assessed based on the video quality of rendered viewports. This differs from the V-PCC development process, which uses quality metrics based on the accuracy of the position of points in 3-D space.

Video quality may be assessed objectively and subjectively. Conducting subjective tests is time-consuming and thus impractical for testing each proposal. In general, the objective metric should mimic subjective perception of the video quality as much as possible. Therefore, several objective quality metrics were tested during the development of the immersive video codec presented in this article. To increase the reliability of the results, only full-reference metrics were used.

The most common objective quality metric used in video processing applications is peak signal-to-noise ratio (PSNR). However, it is not adapted for omnidirectional video, where information from 3-D space is projected into the 2-D image in a nonuniform way (e.g., objects at the poles of ERP image are more stretched than objects at the equator [31]). For this reason, the weighted-to-spherically-uniform PSNR (WS-PSNR) [32] is used for omnidirectional content. Other tested quality metrics were the visual information fidelity (VIF) [33], the multiscale SSIM (MS-SSIM) [34], the video multimethod assessment fusion (VMAF) [35], and the immersive video PSNR

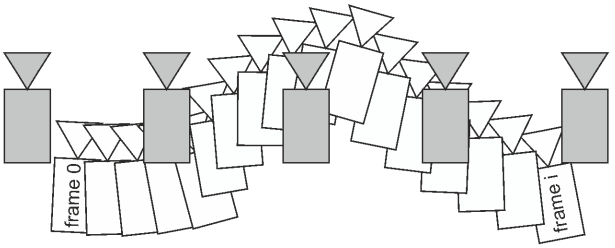


Fig. 15. ***Conceptual example pose trace (gray: input views, white: virtual views).***

(IV-PSNR) [36] designed specifically to handle common rendering artifacts unnoticeable by human perception. The Bjøntegaard delta [37] is used for each metric.

After several meeting cycles, and tens of informal subjective viewing sessions, the group observed a slightly higher correlation with two metrics, VMAF and IV-PSNR, that were eventually maintained in the evaluation process. This "observed" correlation does, however, not rely on any standard recommendation. It must be remarked that at this stage of the immersive video codec development, some tools provide gain and changes that significantly influence the nature itself of the artifacts, which makes the task of objective metrics even more difficult, having to compare two sources with different kinds of impairments. Still, the quality is first assessed objectively, to filter contributions, and to select which ones later deserve further subjective evaluation, before adoption in the draft standard and/or the reference software.

5) Subjective Quality Assessment: A user of the immersive video system can freely navigate within a scene (e.g., using VR headsets). Therefore, in order to assess the quality of virtual movement of the user, the subjective quality cannot be assessed by simply comparing synthesized input views. However, assessment with VR headsets has two major flaws: first, it is more time-consuming and requires more effort than typical subjective tests, and second, each participant would arbitrarily choose where to look at, so results from different participants would not be comparable.

Considering the above-mentioned issues, a middle ground for subjective tests was established: participants assess the quality of pose trace videos. A pose trace is a predefined virtual trajectory of a viewer, including both shifts and rotations that are modified smoothly over time, represented by virtual camera positions. Fig. 15 illustrates an example pose trace, where the gray cameras indicate static source camera positions and the white cameras indicate dynamic virtual camera positions with each individual white camera corresponding to a particular frame time. Each participant assesses the quality of virtual navigation within the scene while judging viewports generated using exactly the same path of rendered view position and orientation. The CTC defines several pose traces for each test sequence and the resolution of the rendered viewport.

A pose trace for the anchor, generated with the latest test model [5], is compared side by side with the pose trace resulting from the proposal. Such an approach allows to perform a quick "visual check" comparing the proposal to the anchor. Informal expert viewing is performed using these side-by-side pose traces, rather than a subjective evaluation, such as those described in the recommendations of ITU-T P.910 [38], or P.360VR [39] dedicated to subjective evaluation with VR headsets. It is infeasible to do formal subjective evaluations during the MPEG meetings for each contribution to the meeting, and the informal expert viewing is considered to provide some additional information to the objective quality results.

B. Experimental Results

In this article, we demonstrate the coding results in two modes of operation: the MIV Atlas mode where all views per sequence are pruned and packed to produce atlases satisfying the CTC pixel rate constraint, and the MIV View mode where a handpicked subset of views are coded as complete views without removing redundancy among those views. The comparison between these experiments reveals the impact of inter-view redundancy removal using the MIV Atlas mode versus increasing view sparsity as in the MIV View mode. Table 2 illustrates the number of selected input views, number of atlases, and atlas resolution for both attribute and geometry video sub bitstreams, per coding case per sequence. Objective quality assessment is performed according to Section VI-A4, with 97 frames for both experiments.

Table 2 MIV Atlas Versus MIV View Configurations per Sequence for Attribute (A) and Geometry (G) Video Sub-bitstreams

Test sequence	Mode	#Views	#Atlases	Atlas Resolution
ClassroomVideo	MIV Atlas	15	2	A: 4096 × 2176 G: 2048 × 1088
	MIV View	9	9	A: 4096 × 2048 G: 4096 × 2048
Museum	MIV Atlas	24	2	A: 2048 × 4352 G: 1024 × 2176
	MIV View	8	8	A: 2048 × 2048 G: 2048 × 2048
Hijack	MIV Atlas	10	2	A: 4096 × 2176 G: 2048 × 1088
	MIV View	5	5	A: 4096 × 2048 G: 4096 × 2048
Kitchen	MIV Atlas	25	3	A: 1920 × 3280 G: 960 × 1640
	MIV View	9	9	A: 1920 × 1080 G: 1920 × 1080
Painter	MIV Atlas	16	3	A: 2048 × 3072 G: 1024 × 1536
	MIV View	8	8	A: 2048 × 1088 G: 2048 × 1088
Frog	MIV Atlas	13	3	A: 1920 × 3280 G: 960 × 1640
	MIV View	7	7	A: 1920 × 1080 G: 1920 × 1080
Fencing	MIV Atlas	10	3	A: 1920 × 3280 G: 960 × 1640
	MIV View	5	5	A: 1920 × 1080 G: 1920 × 1080

Table 3 Pixel Rate Comparison of MIV Atlas Versus MIV View [GP/s]

	Classroom Video	*Museum*	*Hijack*	*Kitchen*	*Painter*	*Frog*	*Fencing*
MIV Atlas	0.67	0.67	0.67	0.71	0.71	0.71	0.59
MIV View	4.53	2.01	2.52	1.12	1.07	0.87	0.52
Ratio	0.15	0.34	0.27	0.63	0.66	0.82	1.13

Pixel rate, as measured by the combined luma sample rate of all video sub-bitstreams, is reported in Table 3 for both modes. The MIV Atlas mode defines a pixel rate limit below that of a single HEVC Level 5.2 decoder (1.07 GP/s), whereas the MIV View mode does not always meet the CTC pixel rate constraint. The pixel rate ratio of the MIV Atlas mode compared to the MIV View mode, as shown in the third row of Table 3, shows significant reduction in pixel rate for the MIV Atlas mode. Even so, the MIV Atlas mode outperforms the MIV View mode on most sequences

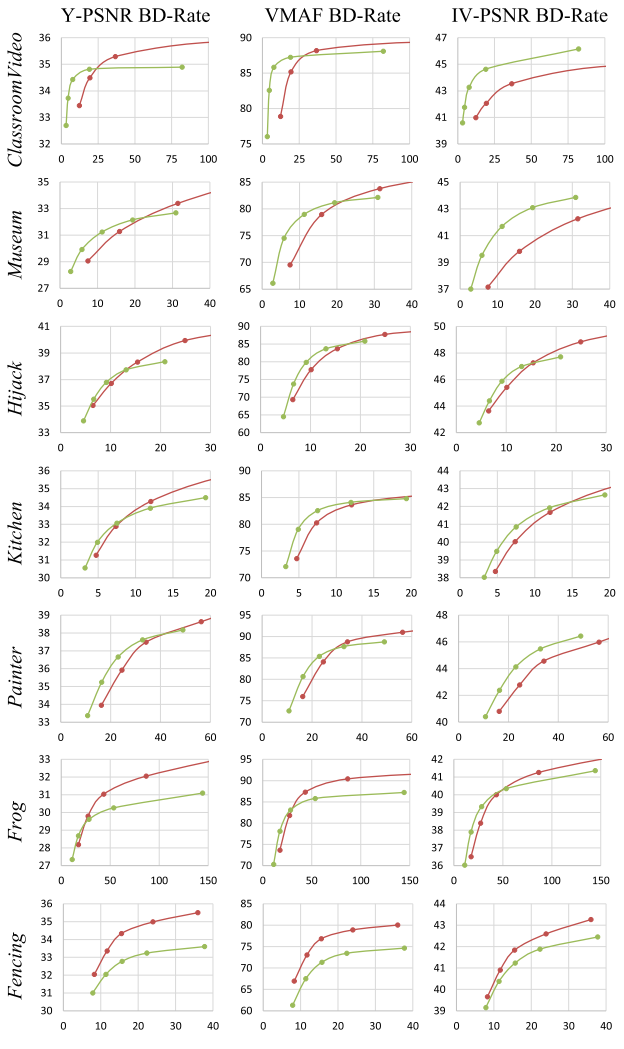


Fig. 16. *Rate–distortion curves averaged over 97 frames and all reconstructed source views, for PSNR (left column), VMAF (center column), and IV-PSNR (right column) metrics. The MIV Atlas mode is indicated by the green curve and the MIV View mode is by the red curve. Horizontal axis: bitrate [Mb/s], vertical axes for consecutive columns: Y-PSNR [dB], VMAF [%], and IV-PSNR [dB].*

Table 4 BD rate of MIV Atlas Versus MIV View (Negative Means MIV Atlas Is Better)

Sequence	*High-BR BD rate Y-PSNR*	*Low-BR BD rate Y-PSNR*	*High-BR BD rate VMAF*	*Low-BR BD rate VMAF*	*High-BR BD rate IV-PSNR*	*Low-BR BD rate IV-PSNR*
ClassroomVideo	-33.3%	-59.6%	-40.5%	-66.3%	-77.0%	-74.7%
TechnicolorMuseum	-11.0%	-34.2%	-12.0%	-39.5%	-56.0%	-58.8%
InterdigitalHijack	-0.2%	-9.6%	-15.3%	-18.2%	-10.2%	-16.1%
OrangeKitchen	11.3%	-7.2%	-19.6%	-24.5%	-13.0%	-20.3%
TechnicolorPainter	-12.7%	-20.4%	-0.6%	-16.3%	-27.2%	-27.1%
IntelFrog	96.6%	3.5%	29.9%	-11.8%	-2.7%	-21.9%
PoznanFencing	160.3%	56.0%	119.7%	46.8%	37.2%	18.9%

in terms of end-to-end RD characteristics. The MIV Atlas mode performs slightly behind the MIV View mode only for the *Fencing* sequence, which is challenging NC with

Fig. 17. *Subjective evaluation of the reconstruction results (zoomed in) of MIV View (left) and MIV Atlas (right) coding approaches at selected frames from pose traces, that is, intermediate views at approximately matched bitrate per sequence.*

low-quality depth maps, captured by ten cameras covering a large arc.

The objective quality assessment is presented in Fig. 16 as an end-to-end RD curve per mode, sequence, and metric. The total bitrate, including all video sub-bitstreams, patch and camera data, parameters and headers, is plotted versus the average quality over all (coded and noncoded) source views over the range of tested QPs.

From these RD curves, objective BD rate [37] metrics are reported in Table 4 over 97 frame sequences, averaged over all reconstructed source views, for low-bitrate and high-bitrate ranges. The green-highlighted numbers of the BD-rate values in Table 4 reflects the saving in bitrate and/or improvement in quality of results generated in the MIV Atlas mode compared to those in the MIV View one. We can infer from these results that objectively the MIV Atlas mode outperforms the MIV View mode in terms of end-to-end RD performance for most of the sequences and especially for the low-bitrate points. We also note that for some test points, perceptual metrics IV-PSNR and VMAF report gains, whereas PSNR indicates degradation (e.g., note, for instance, *Frog* and *Kitchen*). In our experience, in subjective viewing of contributions, the perceptual metrics have correlated better with subjective results. The only cases that results of MIV View are found better (i.e., red highlighted numbers in Table 4 or red dashed curves of Fig. 16) are at the high-bitrate region of the *Frog* sequence and the entire evaluated bitrate range for the *Fencing* sequence. These are, in particular, due to the fast motion, the proximity of objects to the capturing system, and the noisy depth maps estimated for those natural sequences which made it harder for the MIV Atlas encoder to perform pruning and patching efficiently.

For subjective comparison, Fig. 17 shows visual reconstructions of both modes per sequence at approximately matched bitrates, for selected regions. It can be observed that the MIV Atlas is better able to handle occlusions and maintain fine structures compared to the blurrier MIV View results.

Side-by-side synthesized pose trace videos at approximately matched bitrates of both MIV View (left) and MIV Atlas (right) for various sequences are available as a supplementary material to this article.

VII. CONCLUSION AND FUTURE WORK

The upcoming MIV standard will enable high-fidelity immersive experiences through playback of camera-captured 3-D scenes with 6DoF of viewer position and orientation. Test results demonstrate that the MIV can support such applications with affordable coded pixel rate and higher coding efficiency, especially for source content with high-quality depth information. Advancements are continuing to be made in the MIV test model, improving objective and subjective quality and making the MIV encoder more robust against the challenging NC. Studies are underway to test the use of the MIV with other video codecs than HEVC, including VVC and MV-HEVC. Improved methods of subjective quality evaluation for MIV content are under study. New features continue to be added to the standard specification to address a wider range of applications. New source content provides challenges such as non-Lambertian lighting variations and complex geometries.

The MPEG-I Visual group is progressing toward completion of the MIV standard in 2021. Even after the standard is finalized, improvements will continue to be made in the nonnormative encoder and renderer implementations. After publication by ISO, it is expected to see the first MIV deployments starting on content captured in well-controlled and -calibrated environment and expanding to more challenging content, to be displayed on a range of devices, including simple smartphones with the viewport controlled by touch screen, PCs, and VR devices. ■

REFERENCES

[1] R. Koenen and M. L. Champel, *Requirements MPEG-I Phase 1b l*, Standard ISO/IEC JTC1/SC29/WG11 MPEG/N7331, Gwangju, South Korea, Jan. 2018.

[2] *Call for Proposals on 3DoF+ Visual*, Standard ISO/IEC JTC1/SC29/WG11 MPEG/N18709, Marrakesh, MA, USA, Jan. 2019.

[3] *Information Technology—Coded Representation of Immersive Media—Part 5: Visual Volumetric Video-Based Coding (V3C) and Video-Based Point Cloud Compression (V-PCC)*, Standard ISO/IEC 23090-5, N19329, May 2020.

[4] J. Boyce, R. Doré, and V. K. M. Vadakital, *Committee Draft for Immersive Video*, Standard ISO/IECJTC1/SC29/WG11MPEG/N19482, Jul. 2020.

[5] B. Salahieh, B. Kroon, J. Jung, and A. Dziembowski, *Test Model 6 for Immersive Video*, Standard ISO/IEC JTC1/SC29/WG11MPEG/N19483, Jul. 2020.

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

[7] O. Stankiewicz, G. Lafruit, and M. Domański, "Multiview video: Acquisition, processing, compression and virtual view rendering," in *Academic Press Library in Signal Processing: Image and Video Processing and Analysis and Computer Vision*, vol. 6, R. Chellappa and S. Theodoridis, Eds. New York, NY, USA: Academic, 2018, ch. 1, p. 18.

[8] M. Camplani, T. Mantecón, and L. Salgado, "Depth-color fusion strategy for 3-D scene modeling with Kinect," *IEEE Trans. Cybern.*, vol. 43, no. 6, pp. 1560–1571, Dec. 2013.

[9] S. Foix, G. Alenya, and C. Torras, "Lock-in time-of-flight (ToF) cameras: A survey," *IEEE Sensors J.*, vol. 11, no. 9, pp. 1917–1926, Sep. 2011.

[10] M. Tanimoto, "FTV (free-viewpoint TV)," in *Proc. IEEE Int. Conf. Image Process.*, Hong Kong, Sep. 2010, pp. 2393–2396.

[11] A. Schenkel, D. Bonatto, S. Fachada, H. L. Guillaume, and G. Lafruit, "Natural scenes datasets for exploration in 6DoF navigation," in *Proc. Int. Conf. 3D Immersion*, Brussels, Belgium, Dec. 2018, pp. 1–8.

[12] O. Stankiewicz, M. Domański, A. Dziembowski, A. Grzelka, D. Mieloch, and J. Samelak, "A free-viewpoint television system for horizontal virtual navigation," *IEEE Trans. Multimedia*, vol. 20, no. 8, pp. 2182–2195, Aug. 2018.

[13] P. Goorts, M. Dumont, S. Rogmans, and P. Bekaert, "An end-to-end system for free viewpoint video for smooth camera transitions," in *Proc. Int. Conf. 3D Imag.*, Liege, Belgium, Dec. 2012, pp. 1–7.

[14] T. Senoh, N. Tetsutani, and H. Yasuda, "Depth estimation and view synthesis for immersive media," in *Proc. Int. Conf. 3D Immersion*, Brussels, Belgium, Dec. 2018, pp. 1–8.

[15] S. Rogge *et al.*, "MPEG-I depth estimation reference software," in *Proc. Int. Conf. 3D Immersion*, Brussels, Belgium, Dec. 2019, pp. 1–6.

[16] D. Mieloch, O. Stankiewicz, and M. Domański, "Depth map estimation for free-viewpoint television and virtual navigation," *IEEE Access*, vol. 8, pp. 5760–5776, 2020.

[17] C. Rosewarne, B. Bross, M. Naccari, K. Sharman, and G. J. Sullivan, *High Efficiency Video Coding (HEVC) Test Model 16 (HM 16) Encoder Description*, document JCTVC-AB1002, Joint Collaborative Team on Video Coding, Turin, Italy, Jul. 2017.

[18] J. Shade, S. Gortler, L. He, and R. Szeliski, "Layered depth images," in *Proc. 25th Annu. Conf. Comput. Graph. Interact. Techn.*, New York, NY, USA, 1998, pp. 231–242.

[19] K. Müller, A. Smolic, K. Dix, P. Kauff, and T. Wiegand, "Reliability-based generation and view synthesis in layered depth video," in *Proc. IEEE 10th Workshop Multimedia Signal Process.*, Oct. 2008, pp. 34–39.

[20] C. Zhu, Y. Zhao, L. Yu, and M. Tanimoto, Eds., *3D-TV System With Depth-Image-Based Rendering*.

New York, NY, USA: Springer-Verlag, 2012.
[21] J. Jylänki, "A thousand ways to pack the bin—A practical approach to two-dimensional rectangle bin packing," Tech. Rep., 2010. [Online]. Available: http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.695.2918
[22] B. Sonneveldt and B. Kroon, *Depth-Map Scaling for Pixel-Rate Reduction*, Standard ISO/IEC JTC1/SC29/WG11 MPEG/M52365, Brussels, Belgium, Jan. 2020.
[23] J. Fleureau, R. Doré, F. Thudor, T. Tapie, G. Briand, and B. Chupeau, *Graph-Based Pruning for Natural Contents (Update)*, document ISO/IEC JTC1/SC29/WG11 MPEG/M52414, Brussels, Belgium, Jan. 2020.
[24] J. Jung, B. Kroon, and J. Boyce, *Common Test Conditions for Immersive Video*, Standard ISO/IECJTC1/SC29/WG11 MPEG/N19214, Alpbach, Austria, Apr. 2020.
[25] B. Kroon, *3DoF+ Test Sequence ClassroomVideo*, Standard ISO/IEC JTC1/SC29/WG11 MPEG/M42415, San Diego, CA, USA, Apr. 2018.
[26] R. Doré, *Technicolor 3DoF+ Test Materials*, Standard ISO/IEC JTC1/SC29/WG11 MPEG/M42349, San Diego, CA, USA, Apr. 2018.
[27] P. Boissonade and J. Jung, *Proposition of New Sequences for Windowed-6DoF Experiments on Compression, Synthesis, and Depth Estimation*, Standard ISO/IEC JTC1/SC29/WG11 MPEG/M43318, Ljubljana, Slovenia, Jul. 2018.
[28] D. Doyen *et al.*, *Light Field Content From 16-Camera Rig*, Standard ISO/IEC JTC1/SC29/WG11 MPEG/M40010, Geneva, Switzerland, Jan. 2017.
[29] B. Salahieh *et al.*, *Kermit Test Sequence for Windowed 6DoF Activities*, Standard ISO/IEC JTC1/SC29/WG11 MPEG/M43748, Ljubljana, Slovenia, Jul. 2018.
[30] M. Domański *et al.*, *Multiview Test Sequences for Free Navigation Exploration Obtained Using Pairs of Cameras*, Standard ISO/IEC JTC1/SC29/WG11 MPEG/M38247, Geneva, Switzerland, May 2016.
[31] Y. Ye, E. Alshina, and J. Boyce, *Algorithm Descriptions of Projection Format Conversion and Video Quality Metrics in 360Lib*, document ITU-T SG 16 WP3 ISO/IEC JTC1/SC29/W 11, JVET F-1003, Hobart, Australia, Apr. 2017.
[32] Y. Sun, A. Lu, and L. Yu, "Weighted-to-spherically-uniform quality evaluation for omnidirectional video," *IEEE Signal Process. Lett.*, vol. 24, no. 9, pp. 1408–1412, Sep. 2017.
[33] H. R. Sheikh and A. C. Bovik, "Image information and visual quality," *IEEE Trans. Image Process.*, vol. 15, no. 2, pp. 430–444, Feb. 2006.
[34] Z. Wang, E. P. Simoncelli, and A. C. Bovik, "Multiscale structural similarity for image quality assessment," in *Proc. 37th Asilomar Conf. Signals, Syst. Comput.*, vol. 2, 2003, pp. 1398–1402.
[35] Z. Li, A. Aaron, I. Katsavounidis, A. Moorthy, and M. Manohara, "Toward a practical perceptual video quality metric," Netflix Technol. Blog, Tech. Rep., 2016.
[36] *Software Manual of IV-PSNR for Immersive Video*, Standard ISO/IEC JTC1/SC29/WG11 MPEG/N18709, Göteborg, Sweden, Jul. 2019.
[37] G. Bjøntegaard, *Calculation of Average PSNR Differences Between RD-Curves*, document VCEG-M33, Austin, TX, USA, Mar. 2001.
[38] *Subjective Video Quality Assessment Methods for Multimedia Applications*, document ITU-T Rec. P910, Apr. 2008.
[39] *Subjective Test Methodologies for 360° Video on Head-Mounted Displays*, document ITU-T Rec. P919, Nov. 2020.

ABOUT THE AUTHORS

Jill M. Boyce (Fellow, IEEE) received the B.S. degree in electrical engineering from the University of Kansas, Lawrence, KS, USA, in 1988, and the M.S.E. degree in electrical engineering from Princeton University, Princeton, NJ, USA, in 1990.

She is currently an Intel Fellow and the Chief Media Architect at Intel Corporation, Hillsboro, OR, USA, where she is responsible for defining media hardware architectures for Intel's video hardware designs. She represents Intel at the Joint Video Exploration Team (JVET) of ITU-T SG16 and ISO/IEC MPEG. She was formerly the Director of Algorithms at Vidyo, Inc., Hackensack, NJ, USA, where she led video and audio coding and processing algorithm development. She was formerly the Vice President of Research and Innovation Princeton for Technicolor (formerly Thomson), Princeton. She was with Lucent Technologies Bell Labs, Holdmel, NJ, USA, AT&T Labs, Holmdel, NJ, USA, and Hitachi America, Princeton, NJ, USA.

Ms. Boyce was Associate Editor of IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY from 2006 to 2010. She serves as an Associate Rapporteur of ITU-T VCEG and is an Editor of the MIV specification.

Adrian Dziembowski was born in Poznań, Poland, in 1990. He received the M.Sc. and Ph.D. degrees from the Poznań University of Technology, Poznań, in 2014 and 2018, respectively.

Since 2019, he has been an Assistant Professor with the Institute of Multimedia Telecommunications, Poznań University of Technology. He authored or coauthored about 30 articles on various aspects of immersive video, free navigation, and free viewpoint television systems. He is also actively involved in ISO/IEC MPEG activities toward future MPEG immersive video coding standard.

Renaud Doré graduated from the Ecole Centrale de Marseille, Marseille, France, followed with a specialization grade in aerospace electronics from ENSAE in 1987.

He worked first on digital communications for military satellite system at Alcatel Espace, Toulouse, France. His domain has been the wireless and video processing fields at Thomson Research Center, Princeton, and the Technicolor Research Center, Princeton. In the frame of the big emerging momentum around the immersive media, he now leads projects related to graphics and video engineering for immersive experience with Interdigital, Cesson-Sévigné, France, a visual and wireless technology company.

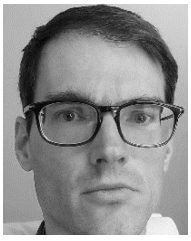
Mr. Doré is an active member of the MPEG-I Visual Group for the future immersive MPEG standard.

Julien Fleureau received the engineering and M.S. degrees in electrical engineering and computer science from the Ecole Centrale de Nantes, Nantes, France, in 2005, and the Ph.D. degree from the University of Rennes, Rennes, France, in 2008.

He is currently a Principal Scientist at the Immersive Laboratory, Research and Innovation Division, Interdigital, Rennes. His Ph.D. studies in image processing and biomedical modeling were followed by a two-year postdoctoral fellowship at the Laboratoire de Traitement du Signal et de l'Image (INSERM), Paris, France. His main research interests are related to signal processing, computer vision, and machine learning applied to virtual/augmented reality, human–machine interactions, biomedical engineering, and new technologies from a more general point of view.

Joel Jung received the habilitation degree in electrical engineering from Sorbonne University, Paris, France, in 2019, and the Ph.D. degree in electrical engineering from the University of Nice, Nice, France, in 2000.

From 1996 to 2000, he was with the CNRS Laboratory, Paris, where he was involved in the improvement of video decoders based on the correction of compression and transmission artifacts. In 2000, he joined Philips Research France, Paris, as a Research Scientist in video coding, postprocessing, perceptual models, objective quality metrics, and low-power codecs. He worked at Orange Labs, Cesson-Sévigné, France, from 2004 to 2020. He has contributed to the 2-D and 3-D video coding standard HEVC/3-D HEVC. He is currently a Principal Researcher with Tencent MediaLab, Palo Alto, CA, USA. His current research interests include video quality evaluation of cloud-gaming video services, contributing to ITU-T Study Group 12, and immersive video coding, view synthesis, and depth estimation with six degrees of freedom in user generated and professional contents, contributing to ISO-IEC MPEG.



Bart Kroon (Member, IEEE) was born in Leidschendam, The Netherlands, in 1979. He received the M.Sc. degree in electrical engineering from the Delft University of Technology, Delft, The Netherlands, in 2005.

He is currently a Senior Scientist with Philips Research Eindhoven, Eindhoven, The Netherlands, specialized in autostereoscopic display optics and rendering, and video processing for virtual and mixed reality. In this role, he is active in ISO/IEC MPEG activities toward future MPEG immersive video coding standard.

Basel Salahieh (Senior Member, IEEE) received the B.S. degree in communication engineering from Aleppo University, Aleppo, Syria, in 2007, the M.S. degree in electrical engineering from The University of Oklahoma, Norman, OK, USA, in 2010, and the M.S. degree in optical science and the Ph.D. degree in electrical and computer engineering from The University of Arizona, Tucson, AZ, USA, in 2015.

He is an Immersive Media Algorithms and Standards Architect at Intel Corporation, Santa Clara, CA, USA, where he is responsible for delivering immersive experiences on Intel devices. His research interests are related to light fields, point clouds, mixed reality, and immersive video systems.

Dr. Salahieh serves as an Editor to the Test Model of MPEG Immersive Video.

Vinod Kumar Malamal Vadakital received the B.E. degree in computer science and engineering from Bangalore University, Bengaluru, India, in 1998, and the M.S. degree in information technology and the Ph.D. degree in signal processing from the Tampere University of Technology, Tampere, Finland, in 2005 and 2012, respectively.

He is currently a Bell Labs Distinguished Member of Technical Staff and a Principal Researcher with the Media Transport Research Team, Nokia Technologies, Tampere. His research interests lie in the domains of video signal processing, computer vision, and XR technologies.

Dr. Malamal Vadakital has previously been an Editor of the High Efficiency Image File Format (HEIF) version 1 specification and is currently an Editor of the MPEG Immersive Video specification.

Lu Yu (Senior Member, IEEE) received the B.Eng. degree in radio engineering and the Ph.D. degree in communication and electronic systems from Zhejiang University, Hangzhou, China, in 1991 and 1996, respectively.

She is currently a Distinguished Professor with Zhejiang University. She developed a series of theories, algorithms, technologies, and architecture designs of video coding reflected in 150 peer-reviewed articles, over 80 granted patents, and more than 100 adopted technical contributions that help define a series of IEEE, ISO/IEC, as well as ISO/IEC and ITU-T joint standards.

Dr. Yu was the Video Subgroup Co-Chair and the Chair of the AVS Working Group from 2002 to 2017 and acted as the Video Subgroup Chair of ISO/IEC JTC1 SC29 WG11, known as MPEG, from January 2018 to July 2020. She also serves as the Chair for the Technical Committee of Education and Outreach of the IEEE Society of Circuits and Systems (CASS). She is serving as the Convener for ISO/IEC JTC 1/SC 29/WG 4, MPEG Video Coding, and leading standardization activities such as immersive video coding, essential video coding, and low-complexity enhancement video coding. She serves on the Editorial Board of IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY for 2020–2021.