



AHG9: Multiplane Image Information SEI

—

JVET-AE0066

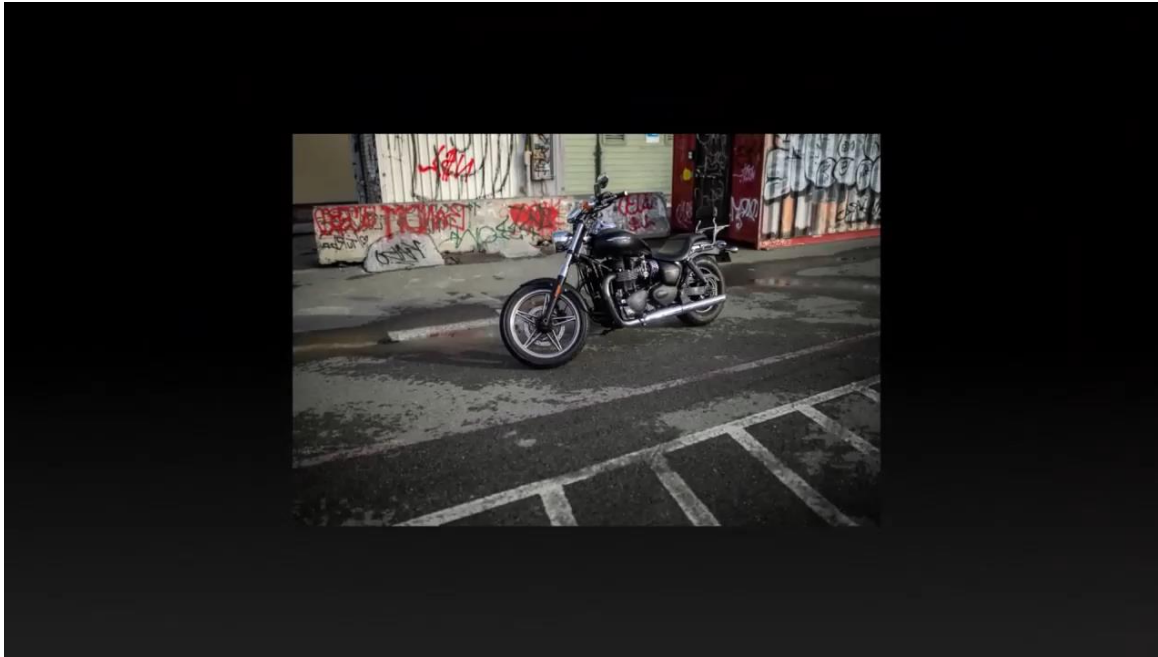
TAORAN LU, PENG YIN, GUAN-MING SU, DAE YEOL LEE, TSUNG-WEI HUANG,
SEAN MCCARTHY, WALT HUSAK, GARY J. SULLIVAN

Outline

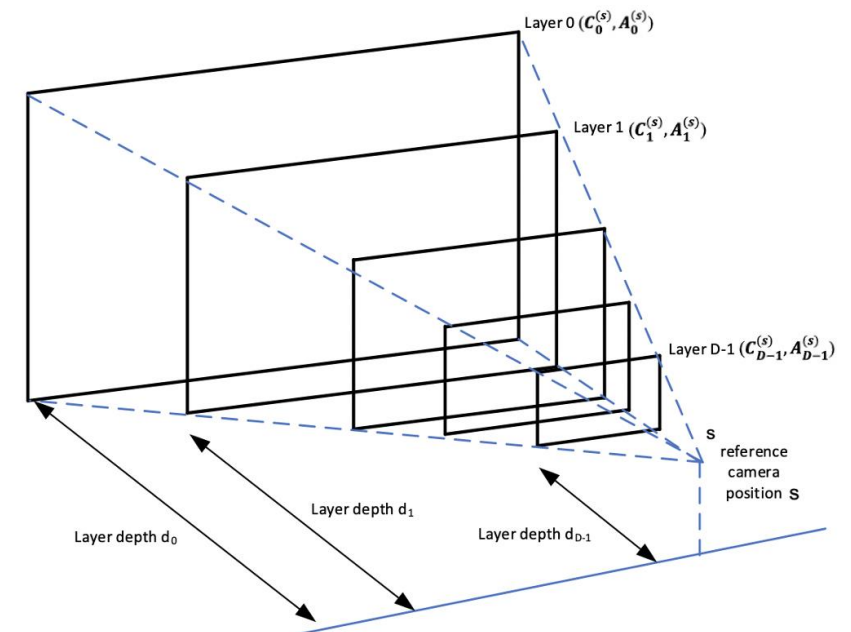
- Multiplane image
- Proposed multiplane image information SEI
 - Design philosophy
 - Syntax and semantics
- Examples and demo
- Conclusions

Multi-Plane Image (MPI)

- A volumetric scene representation using multiple layers of RGBA images.
- Computationally efficient novel view rendering with state of art view synthesis performance
- Enable new viewing experiences using existing image/video transmission infrastructures and viewing devices.



"Local Light Field Fusion: Practical View Synthesis with Prescriptive Sampling Guidelines", Siggraph 2019



layers are shown in perspective projection.
they are essentially the same size

MPI-based view synthesis (rendering)

- Given D-layer MPI image $\{(\mathbf{C}_i, \mathbf{A}_i) \text{ for } i=0, \dots, D-1\}$
- Warping** $T_{v_s, v_t}()$ current view point position (v_s) to new view point position (v_t)

$$- \begin{bmatrix} u_s \\ v_s \\ 1 \end{bmatrix} = \mathbf{K}_s \left(\mathbf{R} - \frac{\mathbf{t} \mathbf{n}^T}{d} \right) (\mathbf{K}_t)^{-1} \begin{bmatrix} u_t \\ v_t \\ 1 \end{bmatrix}$$

$$- \mathbf{C}_i^t = T_{v_s, v_t}(d_i, \mathbf{C}_i)$$

$$- \mathbf{A}_i^t = T_{v_s, v_t}(d_i, \mathbf{A}_i)$$

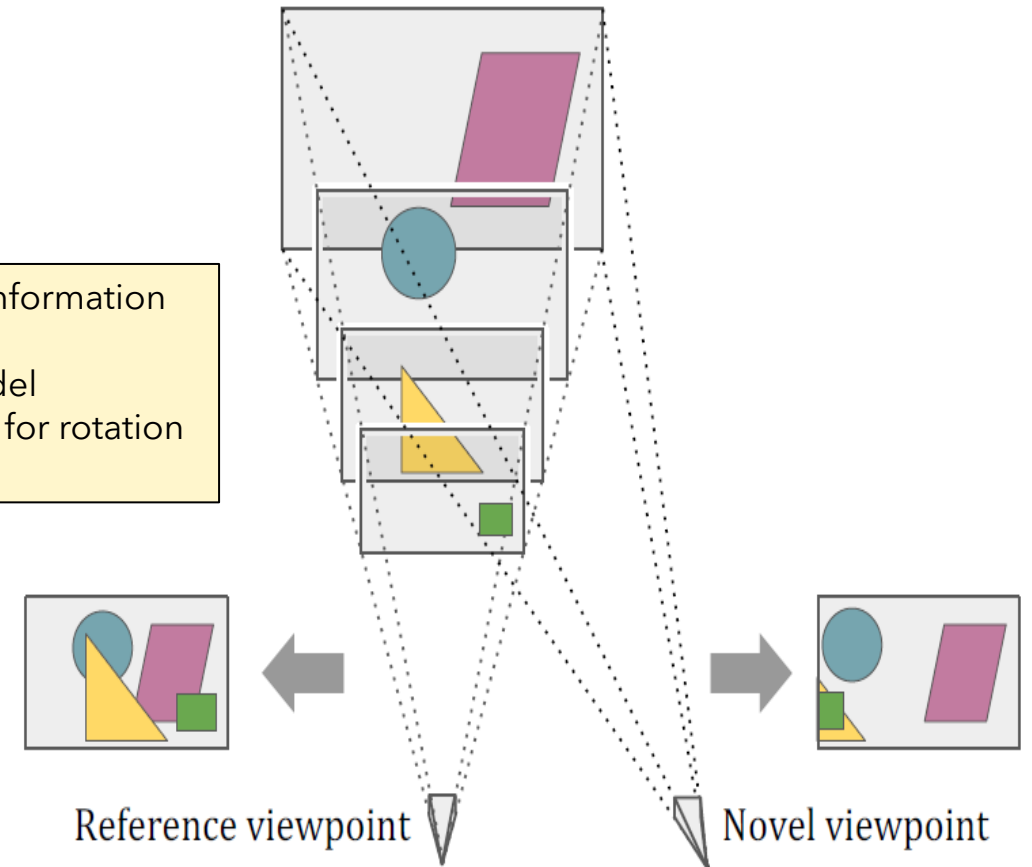
- Compositing**

$$- \mathbf{W}_i^t = (\mathbf{A}_i^t \cdot \prod_{j=i+1}^{D-1} (1 - \mathbf{A}_j^t))$$

$$- \text{Texture in new view } \mathbf{C}^t = \sum_{i=0}^{D-1} \mathbf{C}_i^t \mathbf{W}_i^t$$

- $\mathbf{C}_i, \mathbf{A}_i$ texture and opacity information
- d layer depth information
- $\mathbf{K}_s, \mathbf{K}_t$ camera intrinsic model
- $\mathbf{R}, \mathbf{t}$ extrinsic camera model for rotation and translation (novel view)

layers are shown in perspective projection. they are essentially the same size



Single-View View Synthesis With Multiplane Images ([link](#))

Outline

- Multiplane image
- Proposed multiplane image information SEI
 - Design philosophy
 - Syntax and semantics
- Examples and demo
- Conclusions

MPI Information SEI: design philosophy

- **Goal**

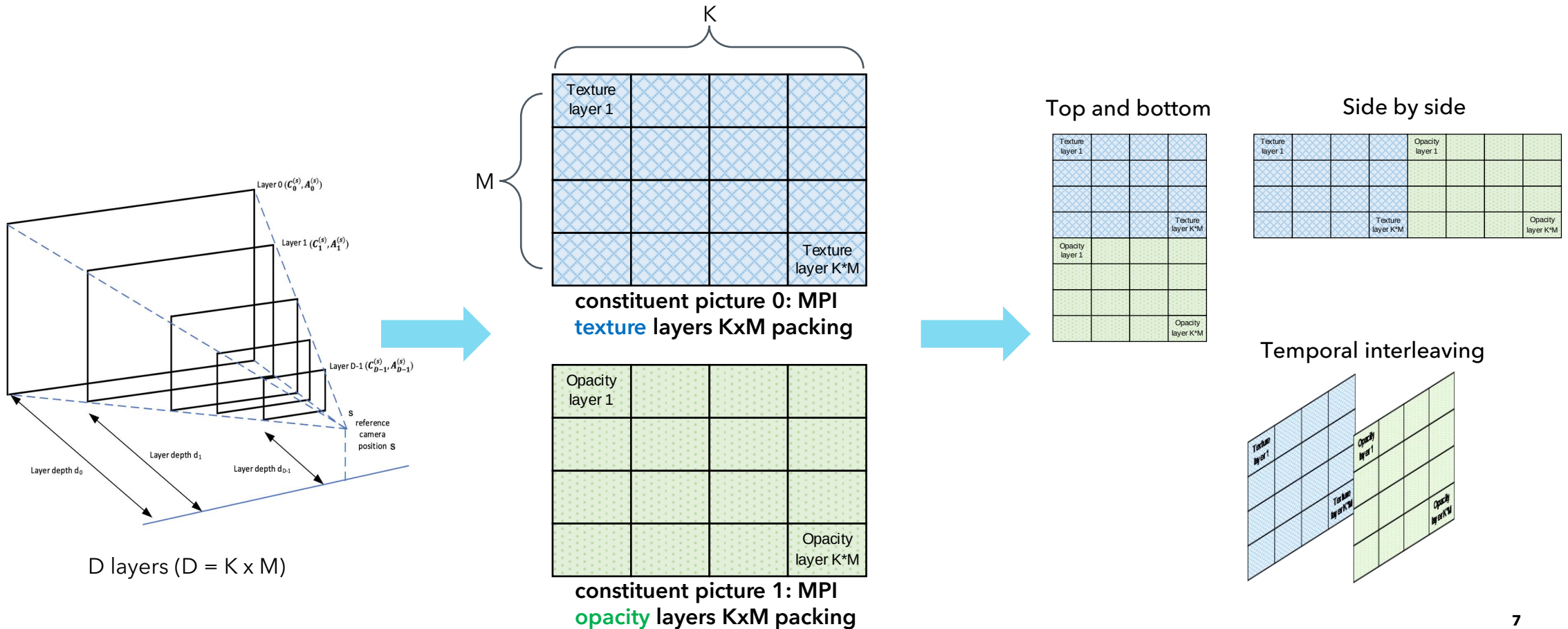
- To create a container/carrier for the MPI based volumetric representation
- To enable distribution of this “container” with existing video distribution infrastructure.

- **Our contribution**

- Design method to pack high-dimensional volumetric data into “conventional” 2D video
- Provide the minimal set of metadata for de-packing and enable view synthesis

MPI layer packing and arrangement

- MPI texture and opacity layers are first spatially packed by $D=K \times M$ arrangement to form constituent pictures
- two constituent pictures can be packed by different options



MPI Information SEI: syntax and semantics

	Descriptor
multiplane_image_information(payloadSize) {	
mpii_num_layers_minus1	ue(v)
mpii_layer_depth_equal_distance_flag	u(1)
if(mpii_layer_depth_equal_distance_flag) {	
depth_rep_info_element(ZNearSign, ZNearExp, ZNearMantissa, ZNearManLen)	
depth_rep_info_element(ZFarSign, ZFarExp, ZFarMantissa, ZFarManLen)	
} else	
for(i = 0; i <= mpii_num_layer_minus1; i++)	
depth_rep_info_element(ZSign[i], ZExp[i], ZMantissa[i], ZManLen[i])	
mpii_texture_opacity_interleave_flag	u(1)
if(mpii_texture_opacity_interleave_flag == 0)	
mpii_texture_opacity_arrangement_flag /* 0: TaB, 1:SbS */	u(1)
mpii_frame_num_layers_in_height_minus1	ue(v)
}	

Note: Camera related information is not included in MPII SEI for minimum set. Camera parameters can be signaled via Multiview acquisition information SEI (ITU-T Rec. H.274 clause 8.20) to reuse what we already defined in VSEI.

Layer depth related:
depth_rep_info_element() has exact same definition as in DRI SEI (ITU-T Rec. H.274 clause 8.22.1.1)

Layer packing & arrangement related

Complete semantics are provided in contribution document including decoding process to recover MPI representation from packed format.

Outline

- Multiplane image
- Proposed multiplane image information SEI
 - Design philosophy
 - Syntax and semantics
- Examples and demo
- Conclusions

Content and compression rates for MPI demo



GuitarMan ©Dolby



Barn ©InterDigital



Breakfast ©InterDigital



CBAbasketball ©Alibaba

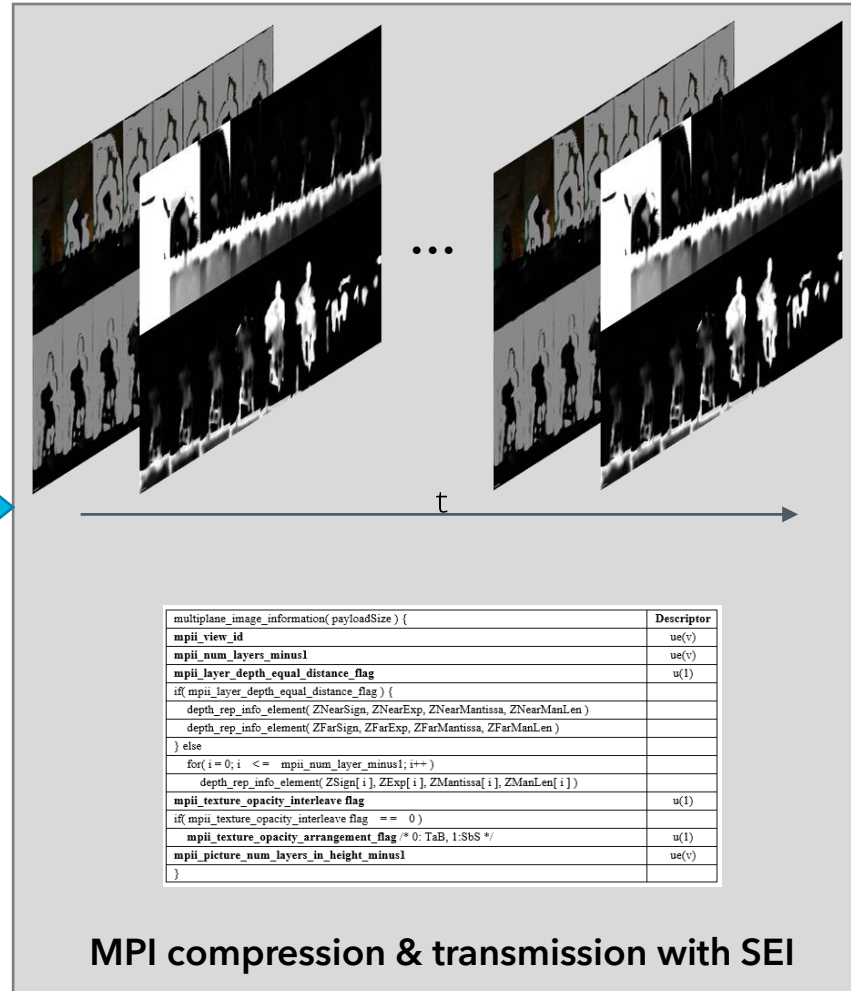
	image width	image height	MPI layer arrangement	constituent picture width	constituent picture height	High BR Mbps @ 30fps	Mid BR Mbps @ 30fps	Low BR Mbps @ 30fps
GuitarMan	384	768	2x8, temporal interleaved	3072	1536	10.2	5.8	3.2
Barn	896	512	4x4, temporal interleaved	3584	2048	12.3	4.1	1.9
Breakfast	896	512	4x4, temporal interleaved	3584	2048	8.6	3.1	1.6
CBAbasketball	896	512	4x4, temporal interleaved	3584	2048	8.4	2.5	1.2

High BR: VVC Main10, QPT=22, QPA=27
 Mid BR: VVC Main10, QPT=32, QPA=37
 Low BR: VVC Main10, QPT=37, QPA=42

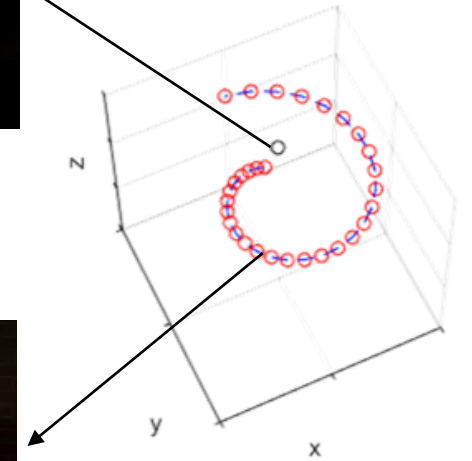
Example: GuitarMan



source video to MPI representation and SEI

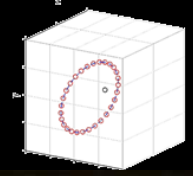
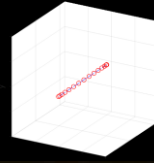
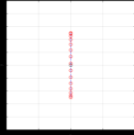
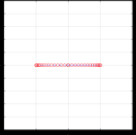


MPI rendering (at ref camera position)

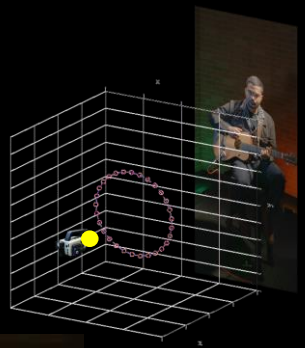


MPI rendering (along a virtual camera position path)

Novel view rendering examples: virtual path can be set freely at device-side



Reference view rendering at different compression quality



uncompressed



High BR
10.2 Mbps



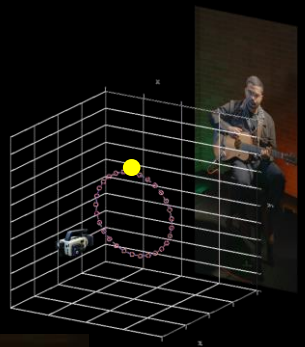
Mid BR
5.8 Mbps



Low BR
3.2 Mbps



Novel view rendering at different compression quality



uncompressed



High BR
10.2 Mbps



Mid BR
5.8 Mbps



Low BR
3.2 Mbps



Example: Barn



source video to MPI representation and SEI



...

t

	Descriptor
multiplane_image_information(payloadSize) {	
mpii_view_id	ue(v)
mpii_num_layers_minus1	ue(v)
mpii_layer_depth_equal_distance_flag	u(1)
if(mpii_layer_depth_equal_distance_flag) {	
depth_rep_info_element(ZNearSign, ZNearExp, ZNearMantissa, ZNearManLen)	
depth_rep_info_element(ZFarSign, ZFarExp, ZFarMantissa, ZFarManLen)	
} else	
for(i = 0; i <= mpii_num_layer_minus1; i++)	
depth_rep_info_element(ZSign[i], ZExp[i], ZMantissa[i], ZManLen[i])	
mpii_texture_opacity_interleave_flag	u(1)
if(mpii_texture_opacity_interleave_flag == 0)	
mpii_texture_opacity_arrangement_flag /* 0: TaB, 1: SbS */	u(1)
mpii_picture_num_layers_in_height_minus1	ue(v)
}	

MPI compression & transmission with SEI



MPI rendering (at ref camera position)



MPI rendering (along a virtual camera position path)

Novel view rendering at different compression quality

uncompressed



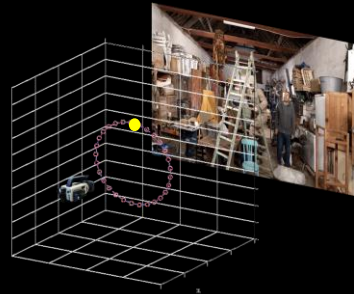
High BR: 12.3 Mbps



Mid BR: 4.1 Mbps



Low BR: 1.9 Mbps



SEI bit count

It is reasonable to update MPI info SEI at IRAP points to adapt to content variation

	Descriptor	non-equal depth example			equal depth example		
		value	bin strings	#bits	value	bin strings	#bits
multiplane_image_information(payloadSize) {							
mpii_num_layers_minus1	ue(v)	15	000010000'	9	15	000010000'	9
mpii_layer_depth_equal_distance_flag	u(1)	0	0'	1	1	1'	1
if(mpii_layer_depth_equal_distance_flag) {							
depth_rep_info_element(ZNearSign, ZNearExp, ZNearMantissa, ZNearManLen)					2.5		15
depth_rep_info_element(ZFarSign, ZFarExp, ZFarMantissa, ZFarManLen)					25		17
} else							
for(i = 0; i <= mpii_num_layer_minus1; i++)							
depth_rep_info_element(ZSign[i], ZExp[i], ZMantissa[i], ZManLen[i])		...		698			
mpii_texture_opacity_interleave_flag	u(1)	1	1'	1	1	1'	1
if(mpii_texture_opacity_interleave_flag == 0)							
mpii_texture_opacity_arrangement_flag /* 0: TaB, 1:SbS */	u(1)						
mpii_frame_num_layers_in_height_minus1	ue(v)	3	'00100'	5	3	'00100'	5
}							
		Total bits: 714			Total bits: 48		

Conclusion

This contribution:

- Introduces the concept of multiplane image (MPI) for view synthesis
- Presents the design of the MP11 SEI, with complete, simple, and clean syntax and semantics
- Illustrates the MP11 SEI usability with examples

We propose to adopt MP11 SEI into VSEI

MPII SEI vs. MIV MPI

	MIV MPI	MPI SEI
Definition of "MPI" in glossary	Set of pairs of texture and transparency attribute frames, each associated with an implicit constant geometry frame [1]	Set of pairs of texture and transparency layers with implicitly/explicitly signaled depth values
Generation of MPI	Out of scope (unknown algorithm by Interdigital in house GPU)	Out of scope (Open-source algorithm by Dolby in house GPU)
Encoder input	MPI format is one special input type of the MIV encoder. A special storage format (PCS [2]) is designed to store MPI in a compact way to save memory footprint.	The packed constituent picture is treated as conventional 2D video for encoder input
Number of MPI layers	Typically > 100 [3], <u>equal</u> distanced [1]	Recommend small value, e.g. 16, layers can be equal or <u>non-equal</u> distanced
Number of views supported	<u>single view</u> reference camera [2]	<u>single view</u> reference camera
Elemental bitstream	Texture and transparency layers are coded in <u>two</u> independent elemental bitstreams. The coded pictures are "atlas" with patches.	<u>One</u> elemental bitstream including both texture and transparency layers.
Decoding and post process	Decode <u>two bitstreams</u> following MIV spec, restore MPI representation from patches on atlas.	Decode <u>one bitstream</u> and restore MPI representation from packed constituent picture.

[1] ISO/IEC 23090-12:2021(E)/AMD.1:2022, Information technology – Coded representation of immersive media – Part 12: MPEG Immersive video, H.9

[2] ISO/IEC JTC 1/SC 29/WG 4 N0271, "Test Model 15 for MPEG immersive video", section 4.6 & 5.6

[3] m55089, MIV CE1-Related - Activation of a transparency attribute and application to MPI encoding

Back up slides

Example: Breakfast



source video to MPI representation and SEI

...

t

	Descriptor
multiplane_image_information(payloadSize) {	
mpii_view_id	ue(v)
mpii_num_layers_minus1	ue(v)
mpii_layer_depth_equal_distance_flag	u(1)
if(mpii_layer_depth_equal_distance_flag) {	
depth_rep_info_element(ZNearSign, ZNearExp, ZNearMantissa, ZNearManLen)	
depth_rep_info_element(ZFarSign, ZFarExp, ZFarMantissa, ZFarManLen)	
} else	
for(i = 0; i <= mpii_num_layers_minus1; i++)	
depth_rep_info_element(ZSign[i], ZExp[i], ZMantissa[i], ZManLen[i])	
mpii_texture_opacity_interleave_flag	u(1)
if(mpii_texture_opacity_interleave_flag == 0)	
mpii_texture_opacity_arrangement_flag /* 0: TaB, 1: SbS */	u(1)
mpii_picture_num_layers_in_height_minus1	ue(v)
}	

MPI compression & transmission with SEI



MPI rendering (at ref camera position)



MPI rendering (along a virtual camera position path)

Novel view rendering at different compression quality

uncompressed



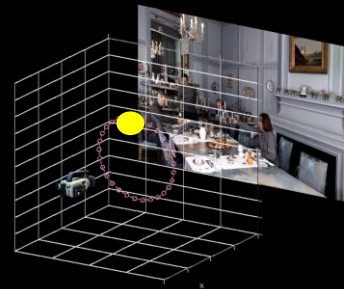
High BR: 8.6 Mbps



Mid BR: 3.1 Mbps



Low BR: 1.6 Mbps



Example: CBAbasketball



source video to MPI
representation and SEI

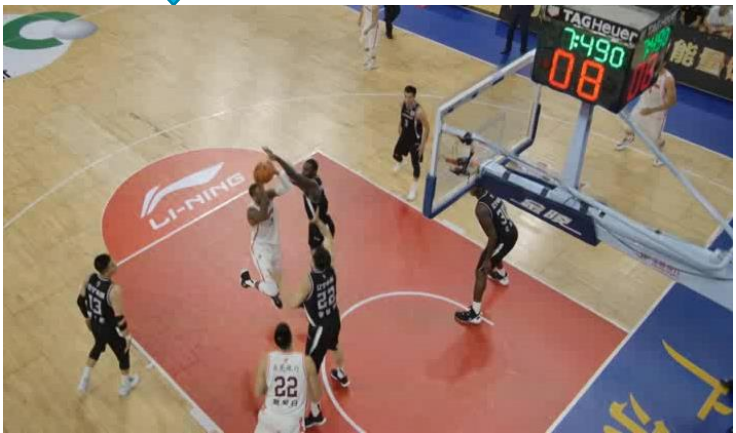
t

	Descriptor
multiplane_image_information(payloadSize) {	
mpii_view_id	ue(v)
mpii_num_layers_minus1	ue(v)
mpii_layer_depth_equal_distance_flag	u(1)
if(mpii_layer_depth_equal_distance_flag) {	
depth_rep_info_element(ZNearSign, ZNearExp, ZNearMantissa, ZNearManLen)	
depth_rep_info_element(ZFarSign, ZFarExp, ZFarMantissa, ZFarManLen)	
} else	
for(i = 0; i <= mpii_num_layers_minus1; i++)	
depth_rep_info_element(ZSign[i], ZExp[i], ZMantissa[i], ZManLen[i])	
mpii_texture_opacity_interleave_flag	u(1)
if(mpii_texture_opacity_interleave_flag == 0)	
mpii_texture_opacity_arrangement_flag /* 0: TaB, 1: SbS */	u(1)
mpii_picture_num_layers_in_height_minus1	ue(v)
}	

MPI compression & transmission with SEI



MPI rendering (at ref
camera position)



MPI rendering (along a virtual
camera position path)

Novel view rendering at different compression quality

uncompressed



High BR: 8.4 Mbps



Mid BR: 2.5 Mbps



Low BR: 1.2 Mbps

