**INTERNATIONAL ORGANISATION FOR STANDARDISATION**

**ORGANISATION INTERNATIONALE DE NORMALISATION**

**ISO/IEC JTC1/SC29/WG 7**

**CODING OF MOVING PICTURES AND AUDIO**

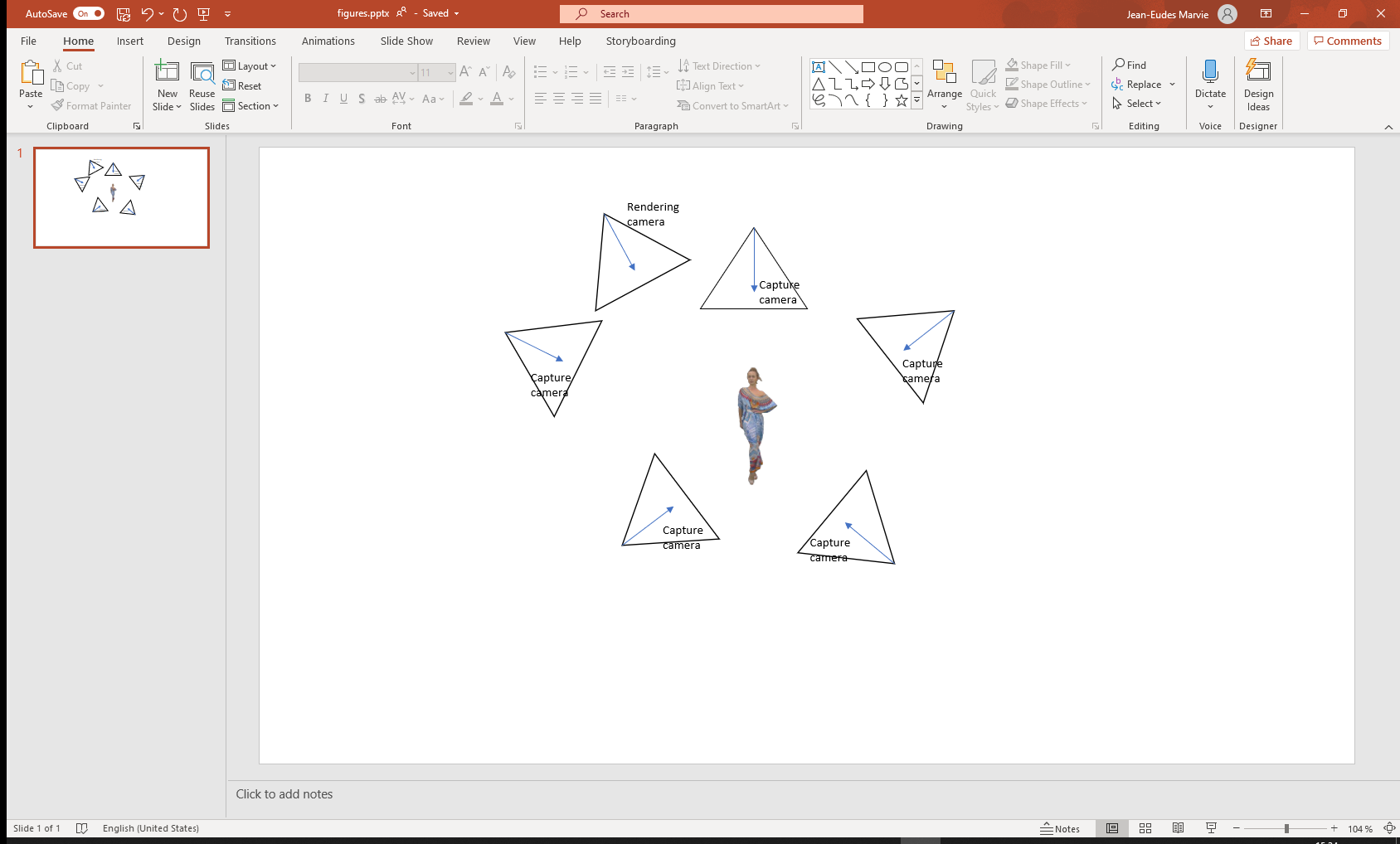
**ISO/IEC JTC 1/SC 29/WG 7/m55xxx**

**Online – July 2021**

|  |  |
| --- | --- |
| **Source** | **InterDigital** |
| **Status** | **Input Contribution** |
| **Title** | **[V-PCC] Capture position SEI message for view-dependent attributes** |
| **Author**  **Abstract** | Pierre Andrivon, Julien Ricard, David Gendron  View-dependent attributes per 3D position, depicted in V-PCC requirements document, is only partially addressed in current version of V-PCC specification. The proposed SEI message fully enables this feature by carrying attributes capture positions and associating them to the attributes data contained in the V-PCC bitstream. This contribution proposes and implements in TMC2 such a mechanism. |

# Problem statement

This document follows contribution m55328 [2] presented in MPEG132. Basically, the view-dependent feature -- i.e. the property that the surface of a material composing an object may render differently the reflected light (attributes values) according to the viewing angle -- is not fully covered in current V-PCC specification [1]. Indeed, the camera capture position and their relation to compressed attributes data are not carried in the V-PCC bitstream. Thus, a faithful rendering of 3D point cloud with multiple attributes according to viewing angle is uncertain (see **Figure 1**). It is also noted that such a feature may be desirable for Surface Light Field or plenoptic content appropriate rendering.



**Figure 1 . Example of a capture camera rig and a virtal camera doing the rendering. At rendering, the system must know the set of capture cameras parameters in order to select the proper point samples within the point cloud.**

# Proposition

As attributes of a point may change according to the viewer viewpoint, the following elements should be considered:

1) the **position of the viewer**,

*It is already documented in V-PCC specification by viewport SEI messages family which enable rendering view-dependent attributes but unfaithfully to the original point cloud as there is no indication about the position of where the attributes were captured from.*

2) a collection of **attribute values for each given point** according to different angles of capture,

*It is covered by V-PCC specification as up to 127 attributes may be carried in the V-PCC stream and indicated by ai\_attribute\_count.*

3) the **position of the capture camera** for a given set of captured attribute value (capture position) and its **association** to the attribute data contained in the V-PCC stream.

*This is the proposal of this contribution: carry capture camera position and orientation and their association to the attribute data in a descriptive metadata.*

**capture position SEI message** (descriptive metadata)

The position and orientation of the camera used to capture attributes are provided in an SEI message aligned with existing viewport position SEI message. Thus, this SEI has the same syntax elements and the same semantics as the viewport position SEI message except that, as it qualifies capture camera position and orientation:

- "viewport" is replaced in the semantics by "capture"

Besides, three syntax elements associate the attributes (viewpoint capture) index coded in the V-PCC stream with the SEI message.

- cp\_atlas\_id specifies the ID of the atlas that corresponds to the associated current V3C unit. The value of cp\_atlas\_id shall be in the range of 0 to 63, inclusive.

- cp\_attribute\_index indicates the index of the attribute data associated to camera position (i.e. equal to the matching vuh\_attribute\_index). The value of cp\_attribute\_index shall be in the range of 0 to (ai\_attribute\_count[ cp\_atlas\_id ] – 1).

-cp\_attribute\_partition\_index indicates the index of the attribute dimension group associated to camera position

Mapping attribute data and capture camera position

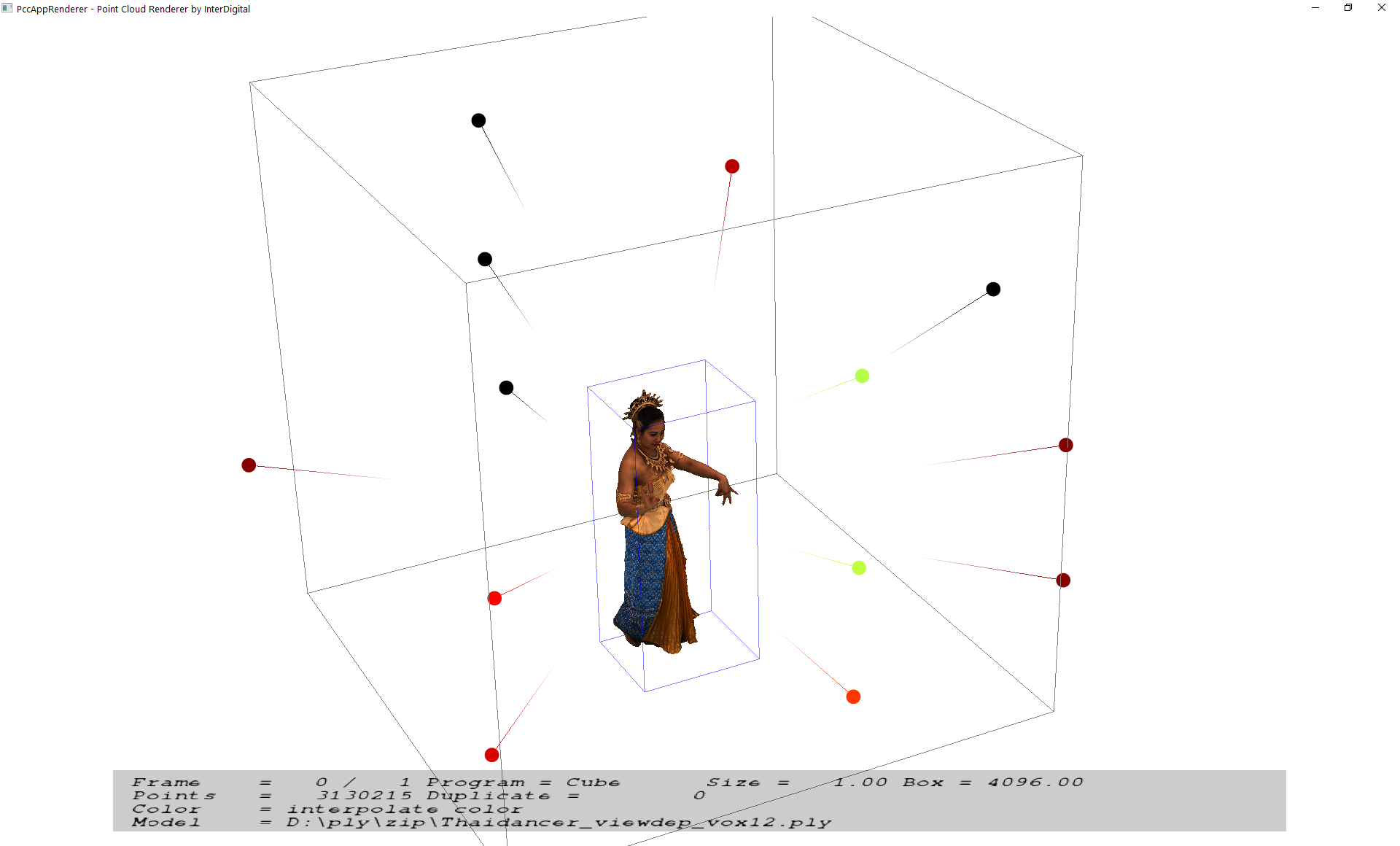
Capture camera position

(same syntax as viewport position SEI position)

|  |  |
| --- | --- |
| capture\_position ( payloadSize ) { | **Descriptor** |
| **cp\_capture\_id** | ue(v) |
| **cp\_atlas\_id** | u(6) |
| **cp\_attribute\_index** | u(7) |
| **cp\_attribute\_partition\_index** | u(7) |
| **cp\_cancel\_flag** | u(1) |
| if( !cp\_cancel\_flag ) { |  |
| **cp\_persistence\_flag** | u(1) |
| for( d = 0 ; d < 3; d++) |  |
| **cp\_position**[d] | fl(32) |
| **cp\_rotation\_qx** | i(16) |
| **cp\_rotation\_qy** | i(16) |
| **cp\_rotation\_qz** | i(16) |
| **cp\_center\_view\_flag** | u(1) |
| if( !cp\_center\_view\_flag ) |  |
| **cp\_left\_view\_flag** | u(1) |
| } |  |
| } |  |

In this proposed version, one SEI message is instantiated per capture camera position. For some applications this may help parsing and sorting of particular views at the price of an increase of SEI message number. An alternative to decrease the possible number of SEI messages is to have a loop on the attribute data count.

The proposed SEI message has been implemented in TMC2 to showcase the usage of view-dependent attributes. The “Thai dancer” is a typical multi-view dependent sequence counting 13 different views and is used to demonstrate the interest of the proposed SEI message. The provided demonstration shows that appropriate attribute data are rendered (with MPEG renderer) according to the proximity of camera capture positions and the viewer position.



**Figure 2 . “Thai dancer” multi-view dependent sequence, attributes are captured by 13 cameras which position and orientation are depicted**

It is noted that although V-PCC specification includes multiple attribute data sub-stream, this feature is not yet supported in TMC2 and could be planned in TMC2 roadmap.

Camera position and orientations may be stored as comments in the PLY format (see below). At the encoder, the proposed demonstration parses the PLY format, converts coordinates matrix in quaternion system, associate each camera position and orientation to the appropriate attribute data stream and code the information in the proposed SEI message syntax. The decoder performs dual operations before the renderer to use this piece of information for enabling the multi-view dependent content.

ply

format ascii 1.0

comment Copyright 2018, 8i Labs, Inc.

comment frame\_to\_world\_scale 0.083114

comment frame\_to\_world\_translation -165.348541 -5.000000 -173.269196

comment width 4095

comment rig\_count 13

comment rig\_matrix\_0 0.999914 -0.012484 0.003989 0.000000 0.013103 0.945876 -0.324264 0.000000 0.000275 0.324289 0.945958 0.000000 6.904244 169.576767 230.972504 1.000000

comment rig\_matrix\_1 0.707785 -0.013902 -0.706291 0.000000 -0.193665 0.957683 -0.212924 0.000000 0.679363 0.287488 0.675141 0.000000 174.571976 167.907944 166.235916 1.000000

comment rig\_matrix\_2 0.006098 -0.017496 -0.999828 0.000000 -0.352905 0.935476 -0.018522 0.000000 0.935639 0.352958 -0.000470 0.000000 235.781128 185.011444 -2.387628 1.000000

comment rig\_matrix\_3 -0.999773 -0.002537 0.021155 0.000000 0.003554 0.959126 0.282958 0.000000 -0.021008 0.282969 -0.958899 0.000000 -4.308383 168.462265 -241.115295 1.000000

comment rig\_matrix\_4 -0.025752 -0.027582 0.999288 0.000000 0.335752 0.941313 0.034634 0.000000 -0.941598 0.336405 -0.014980 0.000000 -236.043106 185.440933 4.168575 1.000000

comment rig\_matrix\_5 0.695137 -0.003555 0.718868 0.000000 0.224927 0.950855 -0.212799 0.000000 -0.682782 0.309617 0.661774 0.000000 -162.440231 166.962631 166.894409 1.000000

comment rig\_matrix\_6 0.999419 0.034089 0.000588 0.000000 -0.034001 0.997827 -0.056432 0.000000 -0.002510 0.056379 0.998406 0.000000 0.396569 99.431526 237.511856 1.000000

comment rig\_matrix\_7 0.655617 0.002151 -0.755091 0.000000 0.046210 0.998007 0.042966 0.000000 0.753679 -0.063062 0.654211 0.000000 184.014297 75.292534 157.264023 1.000000

comment rig\_matrix\_8 -0.999879 -0.010416 -0.011593 0.000000 -0.009392 0.996323 -0.085162 0.000000 0.012437 -0.085042 -0.996300 0.000000 5.579660 71.566063 -243.615128 1.000000

comment rig\_matrix\_9 0.688080 0.014701 0.725486 0.000000 -0.052116 0.998214 0.029202 0.000000 -0.723761 -0.057902 0.687618 0.000000 -168.745209 74.595894 163.582184 1.000000

comment rig\_matrix\_10 -0.999782 -0.017326 0.011658 0.000000 -0.009326 0.869927 0.493093 0.000000 -0.018685 0.492876 -0.869899 0.000000 -2.898166 264.470306 -238.955322 1.000000

comment rig\_matrix\_11 0.999697 -0.003365 -0.024365 0.000000 -0.010107 0.846887 -0.531677 0.000000 0.022424 0.531762 0.846597 0.000000 6.825474 275.690247 232.677536 1.000000

comment rig\_matrix\_12 0.999527 0.003657 -0.030523 0.000000 -0.030264 0.291397 -0.956123 0.000000 0.005397 0.956595 0.291370 0.000000 4.029252 341.354919 104.999611 1.000000

element vertex 3130215

property float x

property float y

property float z

property uchar red

property uchar green

property uchar blue

property float nx

property float ny

property float nz

property uchar red0

property uchar green0

property uchar blue0

property uchar red1

property uchar green1

property uchar blue1

property uchar red2

property uchar green2

property uchar blue2

property uchar red3

property uchar green3

property uchar blue3

property uchar red4

property uchar green4

property uchar blue4

property uchar red5

property uchar green5

property uchar blue5

property uchar red6

property uchar green6

property uchar blue6

property uchar red7

property uchar green7

property uchar blue7

property uchar red8

property uchar green8

property uchar blue8

property uchar red9

property uchar green9

property uchar blue9

property uchar red10

property uchar green10

property uchar blue10

property uchar red11

property uchar green11

property uchar blue11

property uchar red12

property uchar green12

property uchar blue12

end\_header

1791 159 1905 68 46 29 -0.787402 -0.590551 -0.125984 63 42 26 63 42 26 63 42 26 68 46 29 61 41 25 63 42 26 63 42 26 63 42 26 61 40 24 59 39 24 63 43 26 63 42 26 63 42 26

1791 159 1906 68 46 28 -0.787402 -0.590551 -0.125984 63 42 25 63 42 25 63 42 25 68 46 28 60 40 24 63 42 25 63 42 25 63 42 25 60 40 24 59 39 23 63 42 26 63 42 25 63 42 25

1791 159 1907 68 46 28 -0.80315 -0.582677 -0.0472441 63 42 26 63 42 26 63 42 26 68 46 28 61 40 24 63 42 26 63 42 26 63 42 26 61 40 24 59 39 23 63 42 26 63 42 26 63 42

# Conclusion

This contribution proposes to carry capture positions of attributes data and associate them with attribute data contained within the V-PCC bitstream thanks to an SEI message. It is claimed that this proposition fully enables view-dependent attributes feature for V-PCC, as demanded in the V-PCC requirements document.

# Recommendation

It is proposed to include this proposition in the current DIS version 2 of V3C V-PCC and the associated software in TMC2 v15.0.

# References

[1] V-PCC FDIS, ISO/IEC 23090-5, [w19579](https://dms.mpeg.expert/doc_end_user/current_document.php?id=75866&id_meeting=183), Video-based Point Cloud Compression, MPEG131 Online, July 2020

[2] Capture position SEI message for view-dependent attributes, m55328, MPEG132, Online, October 2020