

**ISO/IEC JTC 1/SC 29/WG 11**

**Coding of moving pictures and audio**

**Convenorship: UNI (Italy)**

**ISO/IEC JTC 1/SC 29/WG 11 N18940**

**Document type: Approved WG 11 document**

**Title: V-PCC Core Experiment 2.30 on Patch information signalling**

**Status: Draft**

**Date of document: 2019-10-25**

**Source: 3DG**

**Expected action:**

**No. of pages: 3**

**Email of convenor: leonardo@chiariglione.org**

**Committee URL: mpeg.chiariglione.org**

**INTERNATIONAL ORGANISATION FOR STANDARDISATION**

**ORGANISATION INTERNATIONALE DE NORMALISATION**

**ISO/IEC JTC 1/SC 29/WG 11**

**CODING OF MOVING PICTURES AND AUDIO**

**ISO/IEC JTC 1/SC 29/WG 11 N18940**

**Geneva, CH – October 2019**

|  |  |
| --- | --- |
| **Source:** | **3DG** |
| **Title:** | **V-PCC Core Experiment 2.** **30 on Patch information signalling** |

**V-PCC Core Experiment 2.30 on Patch information signalling**

|  |  |
| --- | --- |
| **Editor** | Jungsun Kim, Alexis Michael Tourapis, Khaled Mammou(Apple), Vladyslav Zakharchenko (Futurewei) |

# Abstract

This document describes the goals and mandates of PCC Core Experiment 2. 30 on Patch information signalling.

# Introduction

The goal of CE4FE 2.30 is to study a variety of alternative methods for signalling patch information parameters in the context of V-PCC, and in particular the syntax elements corresponding to the 2D patch size, 2D patch location, and 3D patch location for non-inter predicted patches. It has been suggested that these methods could simplify the specification as well as enable, under certain conditions, easier rewriting/editing of encoded point cloud bitstreams. More details can be found in [1].

The experimental results of this CE will be evaluated by the 3DG/PCC AhG.

# Mandates

The mandates for CE4FE 2.30 are as follows:

1. Study the impact to editing a point cloud stream and coding performance and of signalling patch size syntax elements in intra coded patches using a u(v) representation where the value of v, i.e. the bit count for the representation of each syntax element, is either derived or signalled.
2. Study the impact and coding performance of coding the syntax elements that correspond to the 2D and 3D patch location coordinates using different representations. In particular, the following representations will be studied:
   1. a u(v) representation with explicitly signalled bit counts,
   2. a u(v) representation with derived bit counts,
   3. ue(v) (Exponential Golomb) representation.

# Participants

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ***Participant*** | ***Contact*** | ***Email*** | ***Type*** | |
| Apple | Jungsun Kim | [jungsun\_kim@apple.com](mailto:jungsun_kim@apple.com) | P |
| Vladyslav Zakharchenko | Vladyslav Zakharchenko | vladyslav.zakharchenko@futurewei.com | C |

(P=proponent, C=crosss checker)

# Test Model, anchors and CTC

Tests will be performed using the VPCC test conditions specified in the Common Test Conditions document [2] on top of the PCC test model [3]. Objective results will be provided using the result spreadsheet template.

# Proposed tests

## Signalling absolute patch size with derived bitcounts

Evaluate the performance of signalling the absolute value of a patch's size instead of its difference from the size of the previous in coding order patch. In this case. the bitcounts for the size will be derived from the size of the current tile group. The syntax changes required for this modification are the same as those needed for 5.1, except that the bit count signalling in the AFPS is no more required. Furthermore, the semantics for the new elements are modified as follows:

#### Patch data unit syntax

|  |  |
| --- | --- |
| patch\_data\_unit( patchIdx ) { | **Descriptor** |
| **pdu\_2d\_pos\_x**[ patchIdx ] | u(v) |
| **pdu\_2d\_pos\_y**[ patchIdx ] | u(v) |
| **pdu\_2d\_size\_x\_minus1**[ patchIdx ] | u(v) |
| **pdu\_2d\_size\_y\_minus1**[ patchIdx ] | u(v) |
| … |  |
| } |  |

**pdu\_2d\_ size\_x\_minus1**[ p ] plus 1 specifies the width value of the patch with index p in the current atlas tile group. The value of pdu\_2d\_size\_x)\_minus1[ p ] shall be in the range of 0 to Min( 2afps2dSizeXBitCount – 1, asps\_frame\_width – 1), inclusive. The number of bits used to represent pdu\_2d\_size\_x\_minus1[ p ], afps2dSizeXBitCount, shall be equal to log2(afti\_tile\_cols\_width\_minus1 +1 ) - atgh\_patch\_size\_x\_info\_quantizer – 1.

**pdu\_2d\_size\_y\_minus1**[ p ] plus 1 specifies the height value of the patch with index p in the current atlas tile group. The value of pdu\_2d\_size\_y\_minus1[ p ] shall be in the range of 0 to Min( 2afps2dSizeYBitCount – 1, asps\_frame\_height – 1), inclusive. The number of bits used to represent pdu\_2d\_size\_y[ p ], afps2dSizeYBitCount, shall be equal to log2(afti\_tile\_rows\_height\_minus1 +1 ) - atgh\_patch\_size\_y\_info\_quantizer – 1.

#### Raw patch data unit syntax

|  |  |
| --- | --- |
| raw\_patch\_data\_unit( patchIdx ) { | **Descriptor** |
| **…** |  |
| **rpdu\_2d\_size\_x\_minus1**[ patchIdx   ] | u(v) |
| **rpdu\_2d\_size\_y\_minus1**[ patchIdx   ] | u(v) |
| **…** |  |
| } |  |

**rpdu\_2d\_ size\_ x\_minus1**[ p ] plus 1 specifies the width value of the raw coded patch with index 0 in the current atlas tile group. The value of pdu\_2d\_size\_x[ p ] shall be in the range of 0 to Min( 2afps\_2d\_size\_x\_bit\_count – 1, asps\_frame\_width -1), inclusive. The number of bits used to represent pdu\_2d\_size\_x\_minus1[ p ], afps2dSizeXBitCount, shall be equal to log2(afti\_tile\_cols\_width\_minus1 +1 ) - atgh\_patch\_size\_x\_info\_quantizer – 1.

**rpdu\_2d\_ size\_y\_ minus1**[ p ] plus 1 specifies the height value of the raw coded patch with index 0 in the current atlas tile group. The value of pdu\_2d\_size\_y[ p ] shall be in the range of 0 to Min( 2afps\_2d\_size\_y\_bit\_count – 1, asps\_frame\_height -1), inclusive. The number of bits used to represent pdu\_2d\_size\_y[ p ], afps2dSizeYBitCount, shall be equal to log2(afti\_tile\_rows\_height\_minus1 +1 ) - atgh\_patch\_size\_y\_info\_quantizer – 1.

#### EOM patch data unit syntax

|  |  |
| --- | --- |
| eom\_patch\_data\_unit( patchIdx ) { | **Descriptor** |
| **…** |  |
| **epdu\_2d\_size\_ minus1\_x**[ patchIdx   ] | u(v) |
| **epdu\_2d\_size\_ minus1\_y**[ patchIdx   ] | u(v) |
| … |  |
| } |  |

**epdu\_2d\_ size\_ x\_minus1**[ p ] plus 1 specifies the width value of the EOM attribute coded patch with index 0 in the current atlas tile group. The value of pdu\_2d\_size\_x[ p ] shall be in the range of 0 to Min( 2afps\_2d\_size\_x\_bit\_count – 1, asps\_frame\_width -1), inclusive. The number of bits used to represent pdu\_2d\_size\_x\_minus1[ p ], afps2dSizeXBitCount, shall be equal to log2(afti\_tile\_cols\_width\_minus1 +1 ) - atgh\_patch\_size\_x\_info\_quantizer – 1.

**epdu\_2d\_ size\_y\_minus1**[ p ] plus 1 specifies the height value of the EOM attribute coded patch with index 0 in the current atlas tile group. The value of pdu\_2d\_size\_y[ p ] shall be in the range of 0 to Min( 2afps\_2d\_size\_y\_bit\_count – 1, asps\_frame\_height -1 ), inclusive. The number of bits used to represent pdu\_2d\_size\_y[ p ], afps2dSizeYBitCount, shall be equal to log2(afti\_tile\_rows\_height\_minus1 +1 ) - atgh\_patch\_size\_y\_info\_quantizer – 1.

## Signalling 2D and 3D location with derived bitcounts

Evaluate the performance of signalling the syntax elements corresponding to the 2D locations of patches with derived bitcounts from the size of the tile group and signalling the syntax elements corresponding to 3D locations of patches with derived bitcounts from the bitdepth of the point cloud geometry.

The related changes to the syntax and semantics are as follows:

#### Atlas frame parameter set RBSP syntax

|  |  |
| --- | --- |
| atlas\_frame\_parameter\_set\_rbsp( ) { | **Descriptor** |
| **…** |  |
| **~~afps\_2d\_pos\_x\_bit\_count\_minus1~~** | ~~u(4)~~ |
| **~~afps\_2d\_pos\_y\_bit\_count\_minus1~~** | ~~u(4)~~ |
| **afps\_3d\_pos\_x\_bit\_count\_minus1** | u(5) |
| **afps\_3d\_pos\_y\_bit\_count\_minus1** | u(5) |
| … |  |
| } |  |

**pdu\_2d\_pos\_x**[ p ] specifies the x-coordinate of the top-left corner of the patch bounding box for patch p in the current atlas tile group, expressed as a multiple of PatchPackingBlockSize.

The value of pdu\_2d\_pos\_x[ p ] shall be in the range of 0 to Min( 2afps\_2d\_pos\_x\_bit\_count\_minus1 + 1 – 1, ColWidth[ i ] / PatchPackingBlockSize − 1), inclusive when the patch is in i-th tile. The number of bits used to represent pdu\_2d\_pos\_x[ p ] is log2(afti\_tile\_cols\_width\_minus1 +1 ) - asps\_log2\_patch\_packing\_block\_size – 1.

**pdu\_2d\_pos\_y**[ p ] specifies the y-coordinate of the top-left corner of the patch bounding box for patch p in the current atlas tile group, expressed as a multiple of PatchPackingBlockSize.

The value of pdu\_2d\_pos\_y[ p ] shall be in the range of 0 to Min( afti\_tile\_rows\_height\_minus1, RowHeight[ i ] / PatchPackingBlockSize − 1), inclusive when the patch is in i-th tile. The number of bits used to represent pdu\_2d\_pos\_y[ p ] is log2(afti\_tile\_rows\_height\_minus1 +1 ) - asps\_log2\_patch\_packing\_block\_size - 1.

#### Atlas frame parameter set RBSP syntax

|  |  |
| --- | --- |
| atlas\_frame\_parameter\_set\_rbsp( ) { | **Descriptor** |
| **…** |  |
| **afps\_2d\_pos\_x\_bit\_count\_minus1** | u(4) |
| **afps\_2d\_pos\_y\_bit\_count\_minus1** | u(4) |
| **~~afps\_3d\_pos\_x\_bit\_count\_minus1~~** | ~~u(5)~~ |
| **~~afps\_3d\_pos\_y\_bit\_count\_minus1~~** | ~~u(5)~~ |
| … |  |
| } |  |

**pdu\_3d\_pos\_x**[ p ] specifies the shift to be applied to the reconstructed patch points in the patch with index p of the current atlas tile group along the tangent axis. The value of pdu\_3d\_pos\_x[ p ] shall be in the range of 0 to Min( 2afps\_3d\_pos\_x\_bit\_count\_minus1 + 1, 2gi\_geometry\_3d\_coordinates\_bitdepth\_minus1 + 1 ) − 1, inclusive. The number of bits used to represent pdu\_3d\_pos\_x[ p ] is gi\_geometry\_3d\_coordinates\_bitdepth\_minus1.

**pdu\_3d\_pos\_y**[ p ] specifies the shift to be applied to the reconstructed patch points in the patch with index p of the current atlas tile group along the bitangent axis. The value of pdu\_3d\_pos\_y[ p ] shall be in the range of 0 to Min(2afps\_3d\_pos\_y\_bit\_count\_minus1 + 1, 2gi\_geometry\_3d\_coordinates\_bitdepth\_minus1 + 1 ) − 1, inclusive. The number of bits used to represent pdu\_3d\_pos\_y[ p ] is gi\_geometry\_3d\_coordinates\_bitdepth\_minus1.

## Signalling 2D size, 2D location and 3D location with the derived bitcounts

Evaluate the performance of the combination of 5.1 and 5.2.

# Timeline

2019/11/xx New CTC and VPCC software available

2019/12/xx Code base and reference results delivered to the cross-checkers

2019/12/xx Preliminary feedback from cross-checkers to the proponents  
2020/01/xx Input contribution upload deadline

# Document and software references

1. J. Kim, A. Tourapis, K. Mammou, [V-PCC] Simplified signalling for non-inter predicted patch data unit,” ISO/IEC JTC1/SC29 WG11 Doc. M51076, Oct. 2019, Geneva, CH
2. Common test conditions for PCC, ISO/IEC JTC1/SC29 WG11 Doc. Nxxxxx, 2019, Geneva, CH
3. PCC Test Model Category 2 v8.0, ISO/IEC JTC1/SC29 WG11 Doc. Nxxxxx, 2019, Geneva, CH