

**INTERNATIONAL ORGANISATION FOR STANDARDISATION
ORGANISATION INTERNATIONALE DE NORMALISATION
ISO/IEC JTC1/SC29/WG11
CODING OF MOVING PICTURES AND AUDIO**

ISO/IEC JTC1/SC29/WG11 MPEG/m53685
April 2020, Online

<i>Source:</i>	Apple Inc.	
<i>Status:</i>	Input document	
<i>Title:</i>	G-PCC: Review of slice point count signalling	
<i>Author(s):</i>	David Flynn Khaled Mammou	<i>davidflynn@apple.com</i> <i>kmammou@apple.com</i>

Abstract

At the 129th meeting, an initial adoption on point count signalling was made. This contribution restates the issues raised and seeks to clarify the semantics of what is adopted.

Introduction

The current design includes a per-slice point count signalled in the geometry data unit header.

At the 129th meeting, following discussions of [1, 2, 3], the following was adopted:

Number of points per slice: Further study. Apple, BlackBerry, Tencent and Qualcomm to sit together. Come back on topic at the end of the G-PCC review this meeting. Thu 17:25: Adopt to signal it at the beginning with fix length coding.

However, this resolution does not cover all aspects of what was discussed.

Confirmation

The current draft text [4] and test model [5] implement the signalling as described in the resolution: A fixed-width value at the start of the slice.

What is omitted is the discussion on any constraints. The motivation to use a fixed-width value is to permit easy modification of the header after geometry coding is complete.

Question: is it mandatory that the signalled `num_points` is exact?

Background

When signalling a length value n that relates a process which truthfully codes p items, two cases must be considered:

$n < p$: Signalled length too short. Should a decoder:

- truncate the output to n ?
- undefined? (but prohibit)
- overflow/crash $\leftarrow$ TM behaviour

$n > p$: Signalled length too long. Should a decoder:

- truncate the missing points? ← TM behaviour
- undefined? (but prohibit)

A decoder may use the number of points to allocate buffers for both output and for the octree fifo.

If the signalled size is too small, the fifo may overflow (in a ring buffer, the write-pointer overtakes the read pointer). Alternatively, the output buffer may overflow. While the latter case is easily guarded against, the former is not and is irrecoverable.

Conversely, if the signalled size is too large, not all output points are defined. It is easy to indicate the number of decoded points in this instance. However, this may be an annoyance for systems that try to determine the number of points in a frame (over multiple slices) or file.

Removal doesn't solve anything

The number of points is not strictly required by the decoding process¹.

Since a level defines the maximum number of points per slice, n could be inferred as the level limit, almost all slices would hit the $n > p$ case above. However, malformed bitstreams can still trigger the $n < p$ case.

The downside to removal is not being able to estimate the number of points in the file from the high-level syntax.

Signalling at the start or end

There are two places that a fixed-width point count can be signalled:

- Slice header (the start).
- Slice footer (the end).

Signalling at the start of the slice (the current design) provides information immediately to the decoder. However, some encoders must wait until encoding is complete before being able to write out the header. The worst case is a one-slice latency.

Signalling at the end of the slice may be an appropriate hybrid. It is simple for the encoder with no latency implications. For a decoder there are two cases. If a decoder needs the exact value, it can wait for the full slice and seek to the end (same effective latency as the signalled-at-start case). Otherwise, if a decoder needs to minimise latency, it may assume the level limit which guarantees that $n \geq p$.

Question: signal at the end of the slice?

References

- [1] D. Flynn and K. Mammou, “G-PCC: High-level syntax issues,” ISO/IEC JTC1/SC29/WG11, 129th meeting, Brussels, Tech. Rep. m52527, Jan. 2020.
- [2] X. Zhang, W. Gao, and S. Liu, “[G-PCC][New Proposal] on signaling number of points in HLS,” ISO/IEC JTC1/SC29/WG11, 129th meeting, Brussels, Tech. Rep. m52401, Jan. 2020.
- [3] B. Ray, A. K. Ramasubramanian, G. V. der Auwera, and M. Karczewicz, “[G-PCC] High level syntax cleanup of G-PCC,” ISO/IEC JTC1/SC29/WG11, 129th meeting, Brussels, Tech. Rep. m52342, Jan. 2020.
- [4] 3DG, “Text of ISO/IEC 23090-9 DIS Geometry-based PCC,” ISO/IEC JTC1/SC29/WG11, 129th meeting, Brussels, Tech. Rep. w19088, Jan. 2020.

¹ Scalable attribute coding does make use of this value

- [5] —, “G-PCC performance evaluation and anchor results,” ISO/IEC JTC1/SC29/WG11, 129th meeting, Brussels, Tech. Rep. w19086, Jan. 2020.