|  |  |
| --- | --- |
| **Joint Collaborative Team on Video Coding (JCT-VC)**  **of ITU-T SG 16 WP 3 and ISO/IEC JTC 1/SC 29/WG 11**  28th Meeting: Torino, IT, 15–21 July 2017 | Document: JCTVC-AB0032 |

|  |  |  |  |
| --- | --- | --- | --- |
| *Title:* | **Proposed design of high-level syntax for spatial relation between independent HEVC sub bitstreams** | | |
| *Status:* | Input document to JCT-VC | | |
| *Purpose:* | Proposal | | |
| *Author(s) or Contact(s):* | Alexandre Gabriel, Emmanuel Thomas Anna van Buerenplein 1 2595 DA Den Haag The Netherlands | Tel: Email: | 0031 611 843 111 {alexandre.dasilvapratasgabriel , emmanuel.thomas}@tno.nl |
| *Source:* | TNO | | |

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

# Abstract

This contribution presents a method of defining the spatial relation between independent sub video bit streams depicting spatial regions of a video. In the context of this document, these independent sub video bit streams are called tiles and they are orthogonal to the existing HEVC tile features. To do so, it proposes a new parameter set – Tile-positioning Parameter Set (TPS). The proposed design breaks the direct relationship between the Sequence Parameter Set (SPS) and Video Parameter Set (VPS) to introduce the TPS and serves as an intermediate parameter set between the two. TPS permits describing the relation between different tiles by using the concept of hooks. Hooks permit describing the relative position between tiles without a coordinate system. Matching hook ids identify two neighbour tiles and express the spatial relations between the different sub video bit streams.

# Problem Statement

## Background

There are a several applications where the spatial subdivision of a frame can be useful. One example is large 8K videos in which the user may want to zoom in and pan. In this case, accessing only a part of the video is an efficient implemenation since it would not be necessary to decode the whole frame for displaying a sub part of it and streaming the whole frame might also not be necessary. To this end, the image can be divided into tiles and the decoder can only decode the necessary tiles to render the viewpoint the user desires. Figure 1 below depitcs a situation where a user zooms in to see the first cyclists in the *peloton*. In this case, decoding the whole frame would not be an efficient use of computing power.

Figure – Panorama with a particular Region of Interest

Another application of tiles can be video mixing which aggregates a series of bitstreams to produce a new video such is the case of a telefonference, see Figure 2.



Figure – Multiparty conference call

One last case where spatial random access is helpful is in 360 VR applications. Typically, VR applications do not render the whole video but only a small part of it corresponding to the user's viewport. Figure 3 shows an example of how a tiling system allows for only half of the image to be sent and decoded (tiles numbered 3 and 4) in order for the application to render the user's viewport. In this case, this would allow for approximately 50% saving in transmitted bitrate compared to sending the whole bitstream.



Figure – 360 video partitioned into 4 tiles (red) and the user's viewport (yellow)

## Example solutions based on HEVC tiles and independent tile video

There are currently several known implementations that take advantage of the existing HEVC tiles for the use cases described above. A quick overview is given below.

In [1], the authors propose an implementation using HEVC tiles with the objective of achieving more efficient panoramic streaming with changing ROIs. In [2], the authors use independent video tiles for optimising the delivery of high resolution panoramas where the user can then pan to their personalised ROI. In [3], experiments were done to use HEVC tiles for enabling video mixing a different video streams similar to Figure 2.

## Problems with current solution based on HEVC tiles

Although HEVC tiles gained momentum for enabling spatial random access over a sequence of frames, the original design of HEVC tiles did not anticipate this usage. As a result, there are several problems with this approach. Essentially, these issues are related to the fact that HEVC tiles are designed for in-frame parallelisation but not for true spatial random access, not even within a frame.

One of the impediments results in HEVC tiles always having to be in a grid. Figure 4 shows a frame partition that it is not possible to have with HEVC tiles, i.e. one tile above another with different widths (region 2 and 4), or tiles side by side with different heights (region 1 and 2). Looking at the ERP video example in Figure 3, a different tile distribution might be more suitable with pole region being composed of one tile.

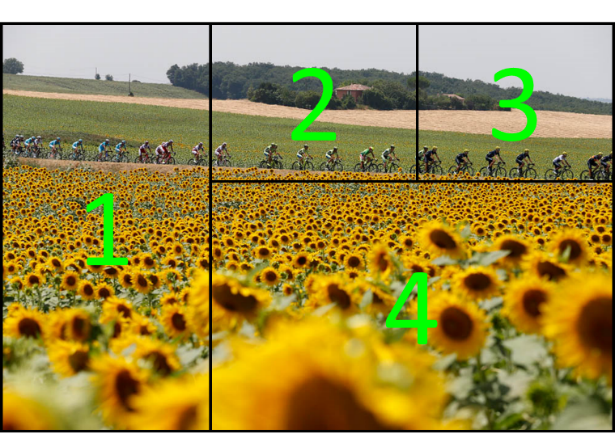


Figure – Example of heterogeneous tile size that is not currently possible

Additionally, the current HEVC syntax specifies the spatial relations between HEVC tiles at a low level in the bitstream. This means that if an alteration of the bitstream has to be made, for example on-the-fly by a network element, all the syntax has to be read to find the point where the tile position and relations are described which is a cumbersome and time-consuming task. A possible workaround is to constrain one HEVC tile per slice. However, the number of slices going up causes a higher overhead due to slice headers in the frame.

Furthermore, the encoding of the HEVC content for such applications using HEVC tile-based access requires careful configuration of the encoders. From the literature in the domain, there are well-known specific configuration to be applied :

* in-loop filtering needs to be disabled
* motion vectors needs to be constrained within the same HEVC tiles over a desired number of frames
* the tile grid needs to be invariant over successive frames (by the HEVC definition, tile grid can change for each frame)
* the invariance of the tile grid needs to be signalled to the decoder
* HEVC tiles position and address may have to be updated if a HEVC tile changes position in the frame
* HEVC tiles are not a bitstream partition which makes them invisible to high level syntax of the HEVC bitstream

All of these problems are being addressed in ongoing standardisation work. However, the amount and the complexity of these measures to be taken advocate for a more fundamental design of the bitstream to enable spatial random access by design which HEVC lacks at the moment.

# Proposed solution

## Hooks concept

The solution proposed uses the concept of hooks to describe the spatial relationship between tiles. Hooks serve as descriptors of points of contact between different tiles. A tile can have up to 4 hooks, top, bottom, left and right. If the identifier of two hooks from different tiles match then the tiles are adjacent to each other at this common edge. In the case that there is no match then it means that the tiles are not adjacent. Figure 5 shows an example of matching and non matching hooks.

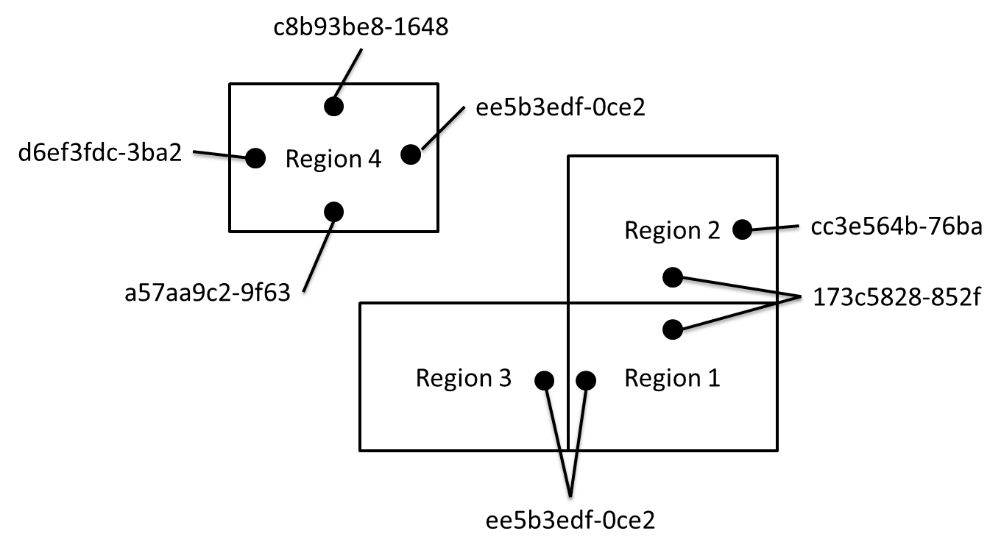


Figure – Example of hooks and their identification numbers

One technical benefit of hooks is that they allow for variable tile size.. The way a decoder can reassemble the tiles is described below with examples from Figure 5, the other two cases would be done with the same logic.

* Tiles vertically adjacent – region 1 and 2 (bottom hook of 2 and top hook of 1 have same ID)
  + The decoder will place region 1’s top left pixel below region 2’s bottom left pixel
* Tiles horizontally adjacent – region 3 and 1 (right hook of 3 and left hook of 1 have same ID)
  + The decoder will place region 1’s top left pixel to the right of region 3’s top right pixel

There is also the possibility of having multiple hook configurations for a layout of tiles as can be seen in Figure 6, which shows the case of multiple tile sizes as well as different hook matching configurations. The decoder would be able to deduce the image on the left by looking at the pixel resolutions of each of the tiles and following the logic explained in the previous paragraph.

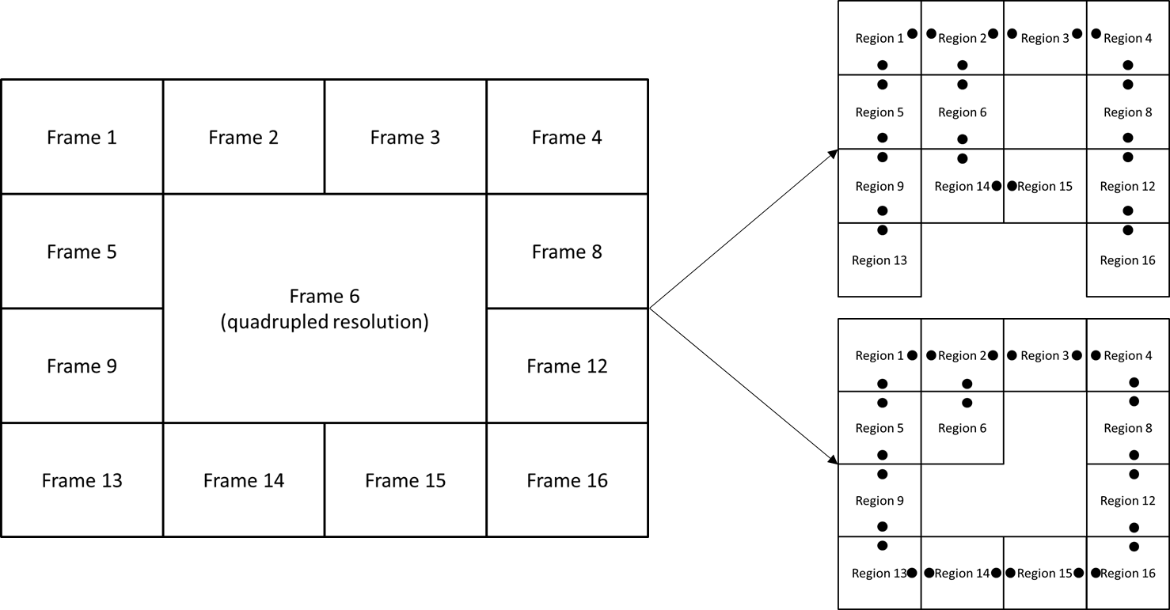


Figure – Multiple hook configuration for the same tile layout

## Proposed syntax

Currently HEVC parameter sets have a relation that can be seen in Figure 7.

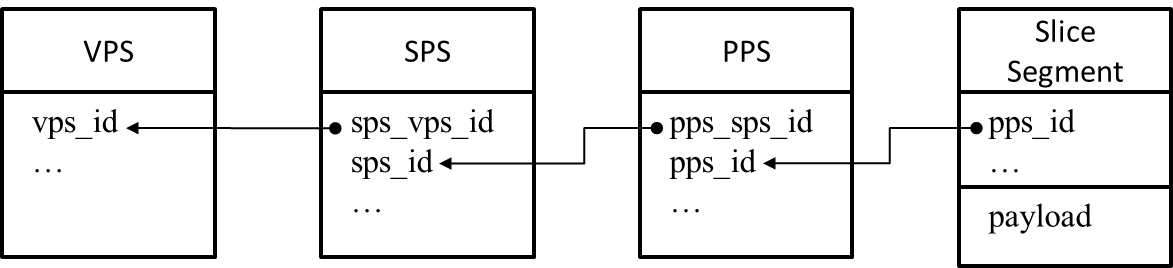


Figure - Current relation between parameter sets

Table 1 gives a quick reminder of the information they contain.

Table - Some of the purposes of parameter sets

|  |  |
| --- | --- |
| **Parameter Set** | **Type of information** |
| Video Parameter Set (VPS) | Information regarding possible layers in the bitstream |
| Sequence Parameter Set (SPS) | Information about all slices of a sequence; e.g., profile, picture size, etc. |
| Picture Parameter Set (PPS) | Information that can change from picture to picture, e.g. tile grid, loop filters, etc. |

Conceptually, the solution would be at a high-level in the syntax such that it can be easily read by a bitstream parser which can be a on-the-fly bitstream processer in the network or a decoder on devices. This contribution suggests the creation of a new parameter set – Tile-positioning Parameter Set (TPS) – that would be inserted between the Video Parameter Set (VPS) and the Sequence Parameter Set (SPS). The new relation between parameter sets can be seen in Figure 8 and would substitute the one in Figure 7.

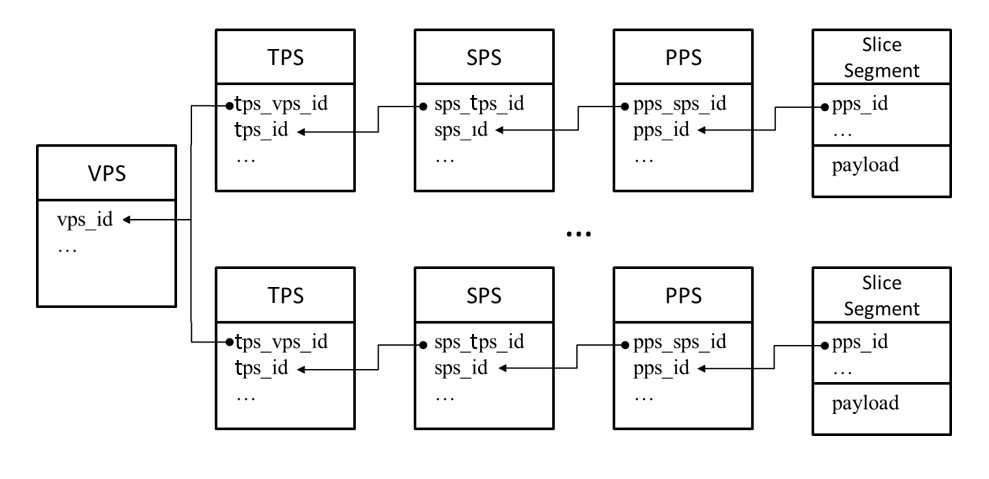


Figure – Proposed relation between parameter sets in HEVC

To achieve this, the Tile-positioning Parameter Set syntax and semantics is as follows:

**Syntax**

|  |  |
| --- | --- |
| tile\_positioning\_parameter\_set\_rbsp( ) { | Descriptor |
| tps\_video\_parameter\_set\_id | u(4) |
| tps\_tile\_positioning\_parameter\_set\_id | ue(6) |
| boundary\_identifier\_north | u(16) |
| boundary\_identifier\_east | u(16) |
| boundary\_identifier\_south | u(16) |
| boundary\_identifier\_west | u(16) |
| } |  |

**Semantics**

tps\_video\_parameter\_set\_id VPS NAL unit id that the TPS depends on;

tps\_tile\_positioning\_parameter\_set\_id Id of this Tile-positioning Parameter Set;

boundary\_identifier\_north boundary identifier located at the north boundary of the tile;

boundary\_identifier\_east boundary identifier located at the east boundary of the tile;

boundary\_identifier\_south boundary identifier located at the south boundary of the tile;

boundary\_identifier\_west boundary identifier located at the west boundary of the tile.

In addition, a nuh\_tile\_id is added to the NAL header, the syntax, semantics and advantages are explained below:

|  |  |
| --- | --- |
| nal\_unit\_header( ) { | Descriptor |
| forbidden\_zero\_bit | f(1) |
| nal\_unit\_type | u(6) |
| nuh\_layer\_id | u(6) |
| nuh\_temporal\_id\_plus1 | u(3) |
| nuh\_tile\_id | u(6) |
| } |  |

If nuh\_tile\_id ≠ 0 then it is the tps\_tile\_positioning\_parameter\_set\_id to which the NAL refers to.

If nuh\_tile\_id = 0 the NAL unit that has no relation with tiles and is invariant to merging and extraction operations of tile bitstreams

The advantages of having this new parameter, essentially linked to efficient bitstream parsing, are the following:

* NAL parsing is quite fast as it is at a high-level in the syntax, this means that it is possible to preselect the tiles wanted and only decode them in real time
* It is also possible to introduce merge 2 bitstreams that have the same characteristics into 1 by introducing the relevant NALs of the TPS, SPS and PPS

# Conclusion

The current tile feature in HEVC was originally designed for in-frame parallelisation and not for spatial random access as it is currently being used. This contribution aims at enabling a true spatial random access in HEVC video bitstreams to better satisfy the needs of video coding applications with high resolution (panoramic video, 360 VR video) as well as scenarios where multiple streams are mixed together. By bringing the tile syntax higher up in the bit stream, on the fly alterations can be made quicker and more efficiently. By introducing the concept of hooks in tiles, it is possible to have heterogenous tile sizes and the tile positions in the arrangement of tiles is no longer a parameter depending on the block partitioning of the frame.

If the video applications mentioned in the document are deemed to be relevant to be addressed, it is recommended to investigate the adoption of the current technical proposal for enabling by design in HEVC a true spatial random access over several frames. Among other things, the use of SEI messages for such purpose could be also investigated for more backward compatibility with respect to the current HEVC specification.

# Patent rights declaration(s)

**TNO may have current or pending patent rights relating to the technology described in this contribution and, conditioned on reciprocity, is prepared to grant licenses under reasonable and non-discriminatory terms as necessary for implementation of the resulting ITU-T Recommendation | ISO/IEC International Standard (per box 2 of the ITU-T/ITU-R/ISO/IEC patent statement and licensing declaration form).**

# References

|  |  |
| --- | --- |
| [1] | Y. Sánchez, R.Skupin and T. Schierl, "Compressed Domain Video Processing For Tile Based Panoramic Streaming Using HEVC". |
| [2] | R. v. Brandenburg, M. P. O. Niamut and H. Stokking, "Spatial Segmentation For Immersive Media Delivery," in *15th International Conference on Intelligence in Next Generation Networks*, 2011. |
| [3] | Y. Sanchez, R. Globisch, T. Schierl and T. Wiegand, "Low Complexity Cloud-video-Mixing Using HEVC". |