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

ISO/IEC JTC1/SC29/WG11 MPEG2013/M31442
October 2013, Geneva (CH)

Source **Telecom ParisTech, Canon Research Centre France**
Status **For consideration at the 106th MPEG meeting**
Title **On Independent HEVC Slices**
Author Jean Le Feuvre, Cyril Concolato, Nael Ouedraogo, Franck Denoual, Frédéric Mazé, Eric Nassor

1 Discussion

During the 105th MPEG meeting in Vienna, concerns were raised in m30303 / JCTVC-N0378 about restrictions in the slice segment order in case of independent slices or tiles, which have to be in raster scan order. The video group position is that the profile being approved, no changes are possible but that indeed a compliant decoder could decode such a bitstream if well interfaced with the systems layer.
The problem is twofold: first slice segment marker and slice segment order.

In HEVC, the `first_slice_segment_in_pic_flag` is used to indicate the first slice segment in a coded picture. However, when first slice segment is lost, a decoder can only detect new picture start by checking POC changes If a loss happens between consecutive IDRs, then POC is always 0 and the decoder has to rely on systems information to detect the frame change (.

Usually, the systems layer is able to reassemble all slices of a single coded picture before handing them to the decoder, and signal the NALU set as being a new picture. The decoder may use this to start parallel processing of slices, by assigning independent slices to different threads.
It is therefore simple for a decoder to handle all slices as part of a new picture.

Regarding slice segment order, the HEVC standard imposes that all slice segments are fed to the decoder in raster scan order. While this may not be very important from a decoder point of view, the constraint is very high for the overall chain: a parallel HEVC architecture may very well encode and produce independent tiles or slices in various order for a given picture, and these should be transportable as soon as they are produced. Imposing a decoding order constraint in such case is likely to increase the latency of the chain, as reordering has to happen either at the sender or at the receiver side.
If reordering happens at the encoder side, there is no real benefit of doing independent slices, the encoder has to produce them in order. If reordering does not happen at the encoder side, there is a need to signal this at the systems layer, so that if the decoder is not able to process slices out of order, the systems layer re-orders the slices before providing them to the decoder.

The questions to the groups are:
- should this be addressed at the systems layer ?

- if so, for which systems techno is this useful ? ISOBMFF? RTP? Both?