INTERDIGITAL®

JCTVC-K0248
ROI TILE SECTIONS

InterDigital Communications and eBrisk Video

11th JCT-VC meeting, Oct 2012

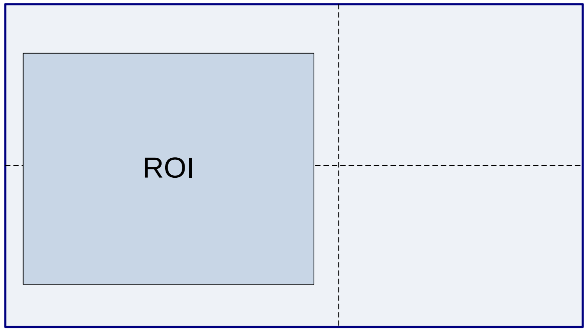
Motivation

- ROI coding is a useful feature for many video applications, especially on mobile
  - Example: ClassX Mobile, an e-learning system from Stanford, allows users to pan/zoom into a portion of the video
  - ClassX uses H.264/AVC as underlying codec

- Goal: to provide better ROI coding support in HEVC
  - ROI only decoding → reduced complexity and power consumption on mobile devices

ROI and tiles

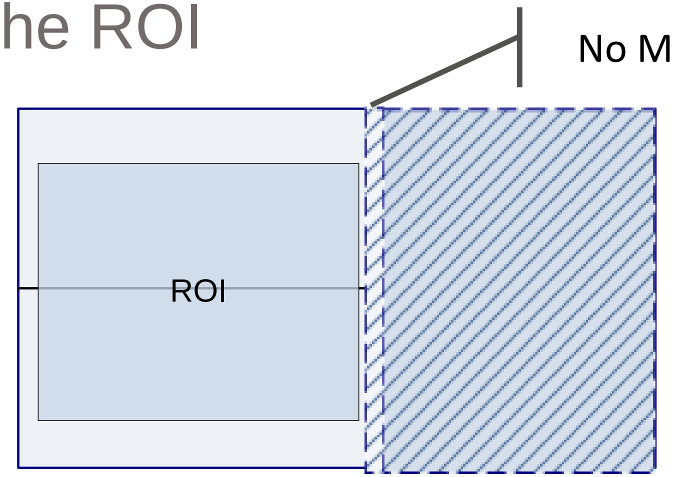
- Tile is a natural choice for ROI support
  - Tiles are independently decodable units
  - Tiles are similarly shaped as ROI



- Only two tiles on the left needed to display ROI
- But, in the current HEVC spec, all tiles must be decoded for decoding of future pictures
- Proposal: to constrain the range of MCP and to signal such constraint in the bitstream to enable ROI only decoding

MCP range constraints

- MCP cannot be based on samples outside of the tiles containing the ROI



No MCP, 4 samples wide

- Additionally, MCP cannot be based on samples within a 4-pixel-wide band on the boundary between ROI tiles and non-ROI tiles
    - If loop_filter_across_tiles_enabled_flag = 1
- Any deviation of sample values from correctly decoded values is limited within the 4-sample-wide band

ROI tile sections

ROI_tile_section () {	**Descriptor**
num_ROI_tile_sections	ue(v)
if(num_ROI_tile_sections > 0)	
for(i = 0; i < num_ROI_tile_sections; i++) {	
num_tiles_in_ROI_section_minus1[i]	u(v)
for(j = 0; j <= num_tiles_in_ROI_section_minus1[i]; j++)	
tile_index[i][j]	u(v)
}	
}	

- Combine one or more tiles into ROI tile sections
- For each ROI tile section, MCP range constraint is imposed by the encoder

Option 1: using VUI to signal ROI tile sections

vui_parameters( ) {	**Descriptor**
…	
if(bitstream_restriction_flag) {	
tiles_fixed_structure_flag	u(1)
num_ROI_tile_sections	u(v)
if(num_ROI_tile_sections > 0)	
for(i = 0; i < num_ROI_tile_sections; i++) {	
num_tiles_in_ROI_section_minus1[i]	u(v)
for(j = 0; j <= num_tiles_in_ROI_section_minus1[i]; j++)	
tile_index[i][j]	u(v)
}	
motion_vectors_over_pic_boundaries_flag	u(1)
…	
}	
}	

Option 2: using SEI to signal ROI tile sections

ROI_tile_section (payloadSize) {	**Descriptor**
num_ROI_tile_sections	ue(v)
if(num_ROI_tile_sections > 0)	
for(i = 0; i < num_ROI_tile_sections; i++) {	
num_tiles_in_ROI_section_minus1[i]	u(v)
for(j = 0; j <= num_tiles_in_ROI_section_minus1[i]; j++)	
tile_index[i][j]	u(v)
}	
}	

Proposed semantics

num_ROI_tile_sections specifies the number of ROI tile sections in the picture. num_ROI_tile_sections shall be in the range of 0 to (num_tile_columns_minus1 + 1) * (num_tile_rows_minus1 + 1), inclusive. If num_ROI_tile_sections is not present, it is inferred to be 0.

NOTE – an ROI tile section defines an area within a video picture for which motion compensated prediction is constrained. Motion compensated prediction of any samples within an ROI tile section should not depend on any samples from outside the ROI tile section, or from no more than 4 samples away from any ROI tile section boundary, unless the ROI tile section boundary is also a picture boundary. s

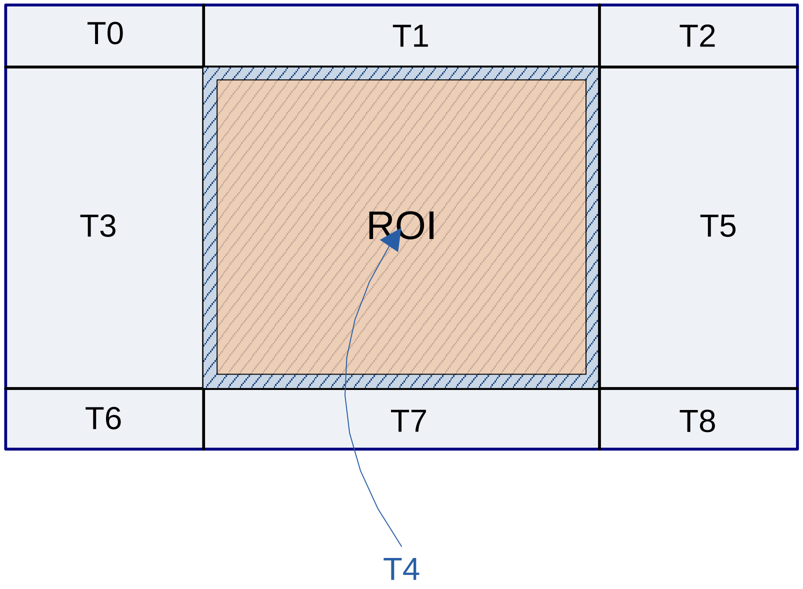
num_tiles_in_ROI_section_minus1[i] plus 1 specifies the number of tiles in the i-th ROI tile section. num_tiles_in_ROI_section_minus1[i] shall be in the range of 0 to (num_tile_columns_minus1 + 1) * (num_tile_rows_minus1 + 1) − 1, inclusive.

tile_index[i] [j] specifies the index of the j-th tile in the i-th ROI tile section in the picture. tile_index[i] [j] shall be in the range of 0 to (num_tile_columns_minus1 + 1) * (num_tile_rows_minus1 + 1) − 1, inclusive. The index values of the tiles in a picture are assigned by starting from 0 in the top left corner, and incrementing by 1 following the raster scan order.

NOTE – When num_ROI_tile_section is equal to 0, no ROI tile section is defined for the pictures in the video sequence. That is, there is no constraint on the range of motion compensated prediction for all samples in the picture. When num_ROI_tile_section is equal to or greater than 1, composition of each of the ROI tile sections is signaled using tile_index[i][j].

Example use case

- One ROI, tile partitions are aligned with the position and dimension of ROI



num_ROI_tile_sections = 1
num_tiles_in_ROI_section_minus1[0] = 0
tile_index[0][0] = 4

Conclusion

- Proposed to add ROI coding support in HEVC
  - ROI only decoding is useful feature, esp. for mobile devices
- Discussed appropriate MCP range constraint at encoder that can limit the decoding error
- Proposed to use ROI tile section to signal the MCP range constraint
  - Add ROI tile section syntax in VUI or as a new SEI message