



# JVET-Q0199 AHG8: Support of ROI (Region-of-Interest) RPR

---

17th JVET Meeting: Brussels, BE, 7–17 January 2020

Dolby Laboratories, Inc.

# Outline

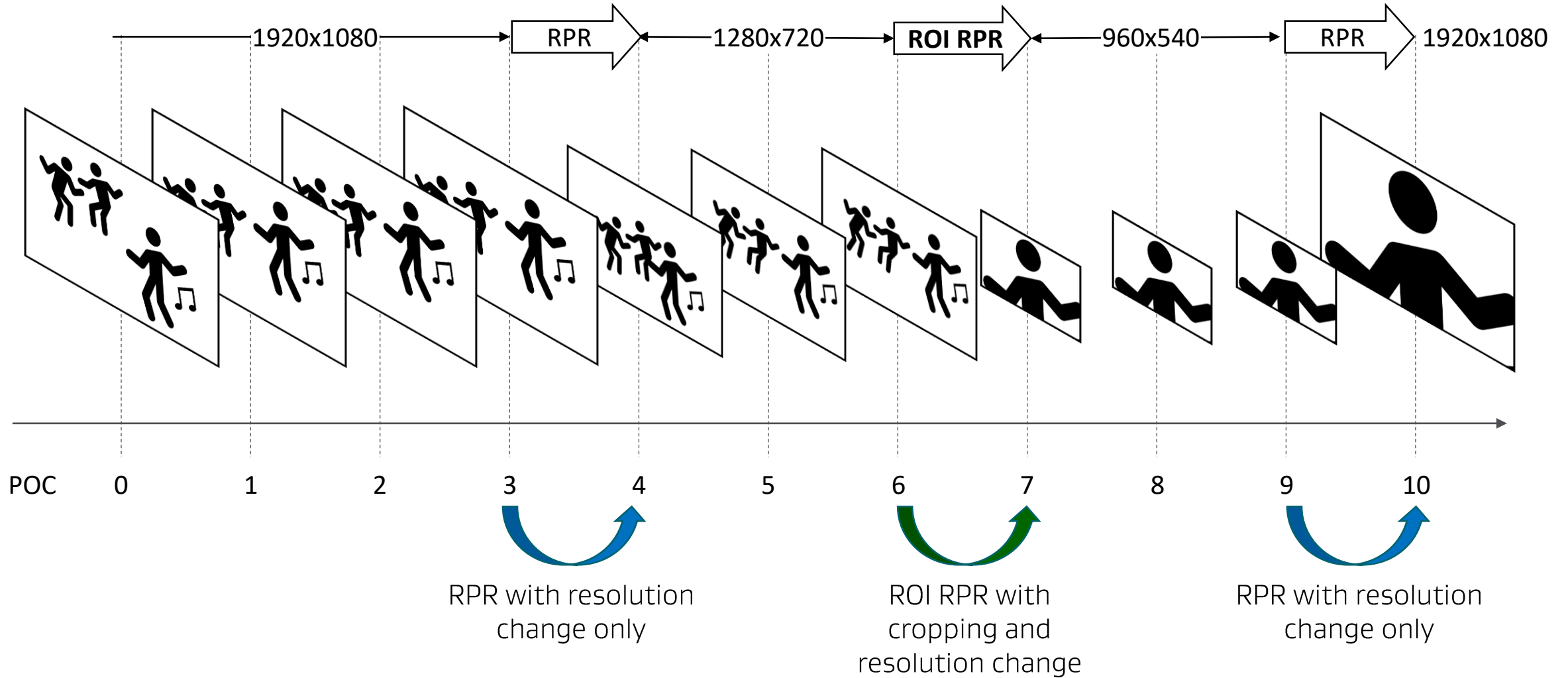
- Conceptual overview of region-of-interest (ROI) RPR
- ROI RPR design and constraints
- ROI RPR syntax, semantics and changes to decoding processes
- ROI RPR software implementation and simulations
- Memory bandwidth constraints
- Conclusion

# Key concepts for ROI RPR

- ROI RPR leverages RPR as currently specified in VVC:
  - ROI RPR is a generalization of RPR: extend scaling window to ROI window.
- Proposes syntax and semantics to support RPR for subregions (ROI) within a picture.
  - Decoding process change is minor => following exact scaling window process.
- ROI RPR addresses meeting notes for JVET-P0336 on support of ROI scalability using RPR.
  - Concern was expressed about whether this functionality would have a complexity problem.
  - There is interest in this in concept, further study with provision of text and software was encouraged.
- ROI RPR supports spatial scalability without additional syntax or semantics.

# Example: RPR and ROI RPR in single layer low-delay P

Example: Low-delay P, each frame uses previous frame for inter prediction



# Proposal: ROI Window

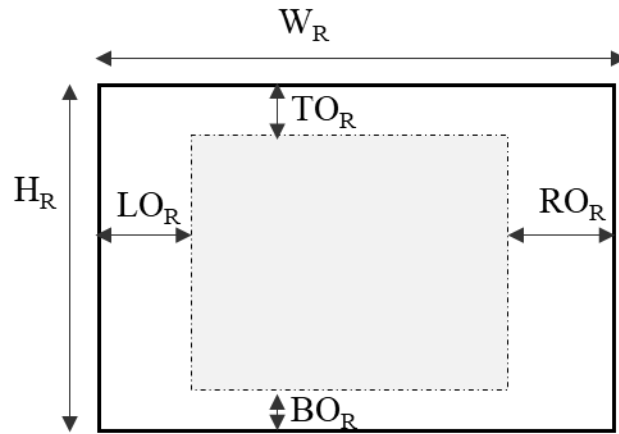
RPR as currently specified

- JVT-P0590 (adopted in VVC draft 7): scaling window offset parameters in PPS
    - Derive scaling ratio of ref pic and cur pic
    - Locate inter prediction correspondence of ref pic and cur pic (fractional sample interpolation process)
- 

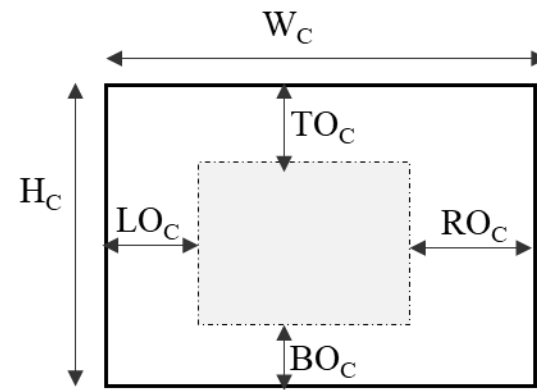
Proposed ROI RPR

- Key concept: replace scaling window offset parameters with ROI window offset parameters when ROI RPR is used.
- New ROI window offset parameters syntax are needed to support ROI RPR
  - ROI window cannot be supported directly by scaling window offset parameters in PPS
  - Scaling window offset parameters are a property of the picture itself (enable pic size of multiple of 8).
  - ROI window offset parameters are specified with respect to both cur and ref pics.

# ROI RPR Design



Reference picture (R)



Current picture (C)

- Specify ROI window offset parameters for current and reference pictures
  - Ref pic: ( $LO_R$ ,  $RO_R$ ,  $TO_R$ ,  $BO_R$ )
  - Cur pic: ( $LO_C$ ,  $RO_C$ ,  $TO_C$ ,  $BO_C$ )
- Determine horizontal and vertical scaling ratios:
  - $hor\_scale = (W_R - LO_R - RO_R) / (W_C - LO_C - RO_C)$
  - $ver\_scale = (H_R - TO_R - BO_R) / (H_C - TO_C - BO_C)$

# ROI RPR Design

Scaling window is used as in default RPR case for a pair of cur pic and ref pic.

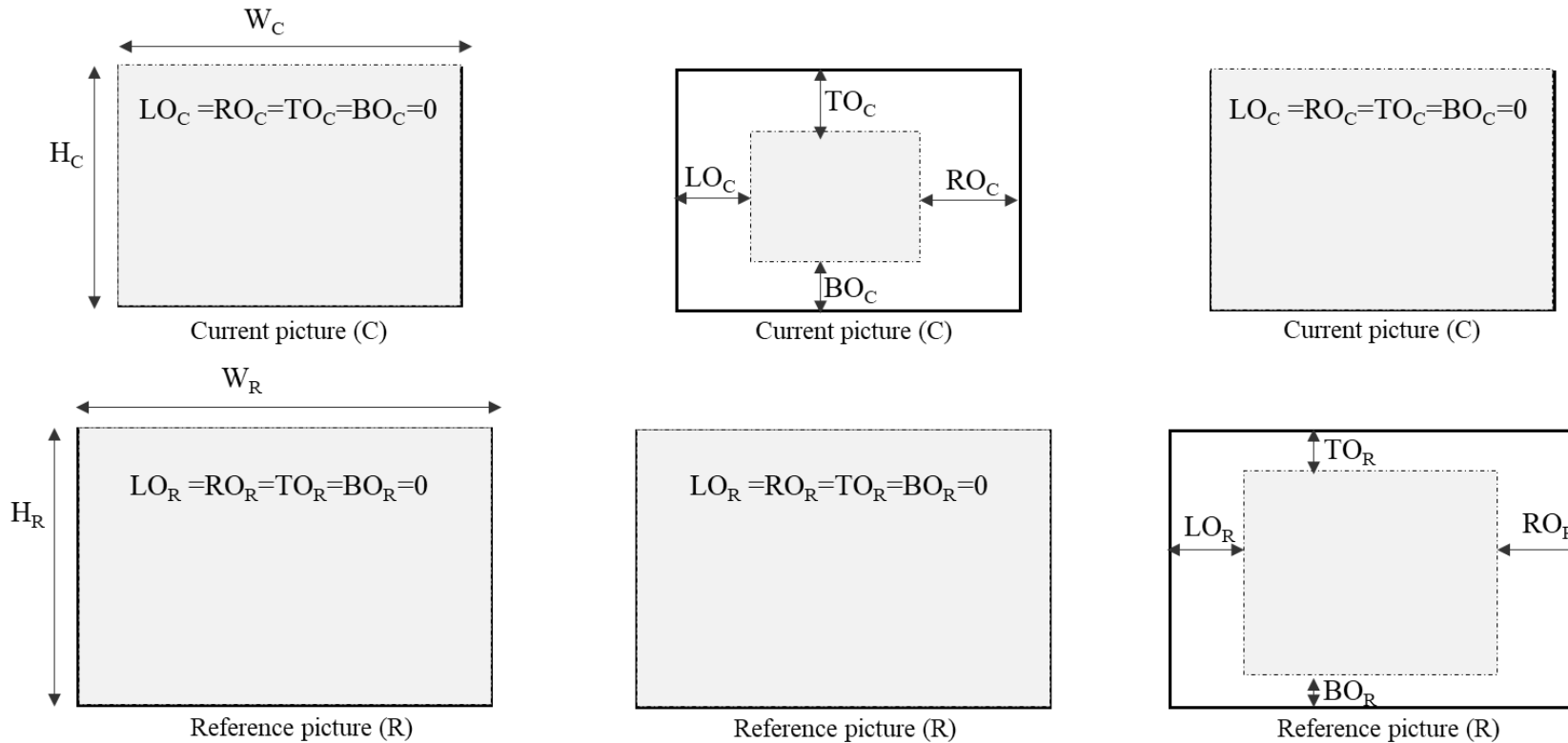
- If ROI RPR is enabled, ROI window offset parameters are used in derivation of scaling ratio and in offset handling in fractional interpolation process;
  - Otherwise (ROI RPR is disabled), scaling window offset parameters are applied as in current VVC design.
- 

Constraints proposed to simplify design and implementation:

- 1) Any ROI picture is completely contained in the full picture: ROI window is no bigger than scaling window.
- 2) Any picture, current or reference, may contain at most one ROI window: avoid complexity of determining which of several ROI windows should be used for each sample

# ROI RPR with Constraints 1 and 2

Constraints: 1) Any ROI picture is completely contained in the full picture  
2) Any picture, current or reference, may contain at most one ROI window



**Case 1:** ROI RPR same as default RPR mode

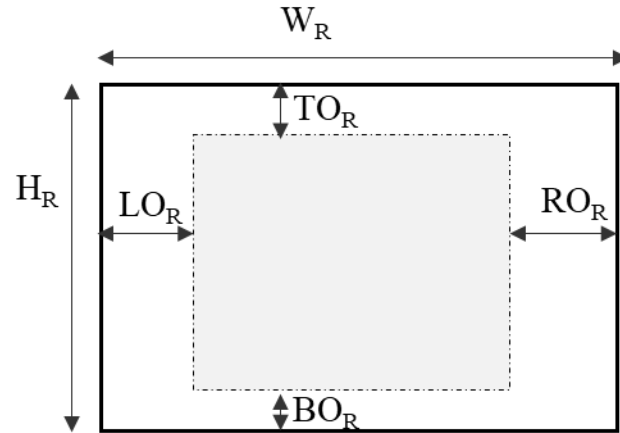
**Case 2:** Entire ref pic corresponds to ROI window in cur pic

**Case 3:** ROI window in ref pic corresponds to entire cur pic

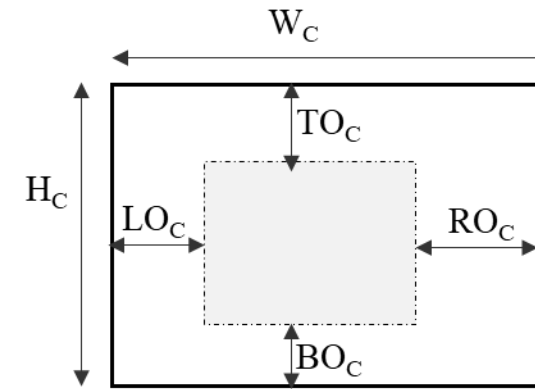


# ROI RPR with Constraints 2 Only

Constraints: 1) Any ROI picture is completely contained in the full picture  
2) Any picture, current or reference, may contain at most one ROI window



Reference picture (R)

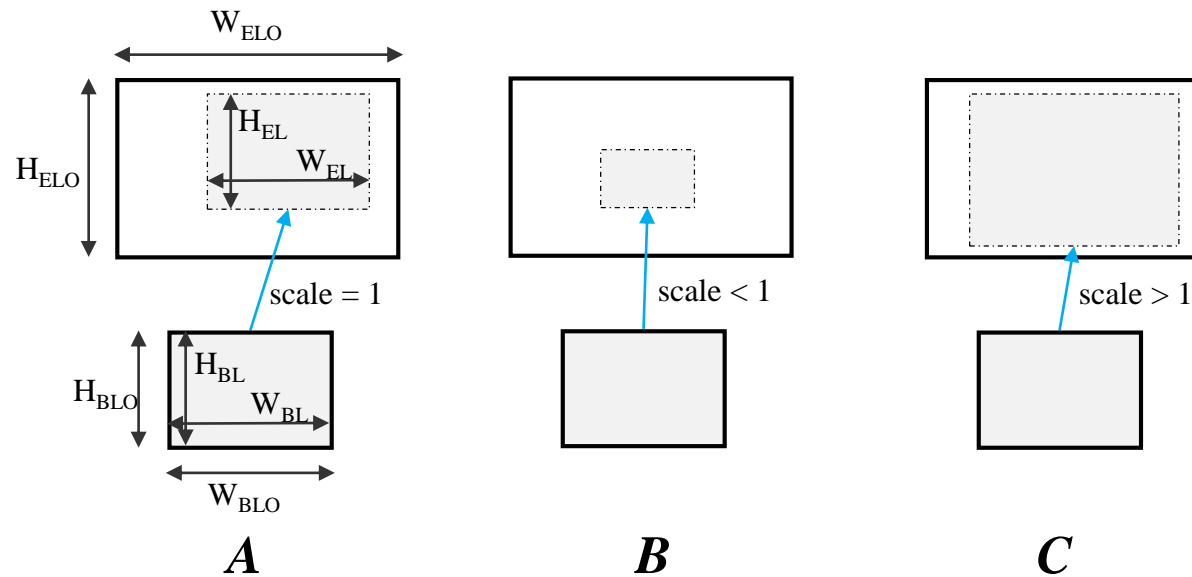


Current picture (C)

- Decoder: same process for all samples as in ROI RPR with both constraint 1 and 2
- Encoder: extra care for ME in VTM7.0 software

# ROI RPR and Spatial Scalability

- VVC draft7 supports spatial scalability using RPR framework with BL used as LTRP.
- Adding support for ROI RPR supports ROI scalability automatically when the BL (ROI picture) is completely included in EL full picture.



ROI scalability support

# ROI Syntax: SPS and Picture Header

## Proposal

- Signal candidate ROI window offset parameters as a list in SPS
- Signal ROI window pairs in Picture Header (PH) and send index of ROI pairs in slice header (SH)

## Rationale

- Signaling an ROI window with 4 offsets for every pair of cur pic and ref pic is costly.
- Provides flexibility to have a different ROI window for cur pic or code cur pic using multiple ref pics each with a different ROI window.
- Compared to APS, SPS implementation is simpler when ROI window does not change frequently.

# Proposed ROI syntax ( 1 ): ROI window list in SPS

*ROI window list in sequence parameter set RBSP syntax*

seq_parameter_set_rbsp() {	Descriptor
...	
if( ref_pic_resampling_enabled_flag ) {	
sps_rpr_roi_window_present_flag	u(1)
if( sps_rpr_roi_window_present_flag ) {	
num_candidate_roi_windows_minus1	ue(v)
for( i = 0; i <= num_candidate_roi_windows_minus1; i++ ) {	
roi_offset_prec_shift[ i ]	ue(v)
roi_offset_list[ i ][ 0 ] /*left */	ue(v)
roi_offset_list[ i ][ 1 ] /*right */	ue(v)
roi_offset_list[ i ][ 2 ] /*top */	ue(v)
roi_offset_list[ i ][ 3 ] /*bottom */	ue(v)
}	
}	
}	

- Same ROI window parameters definition as scaling window
- Maximum number of ROI window to support is 4.

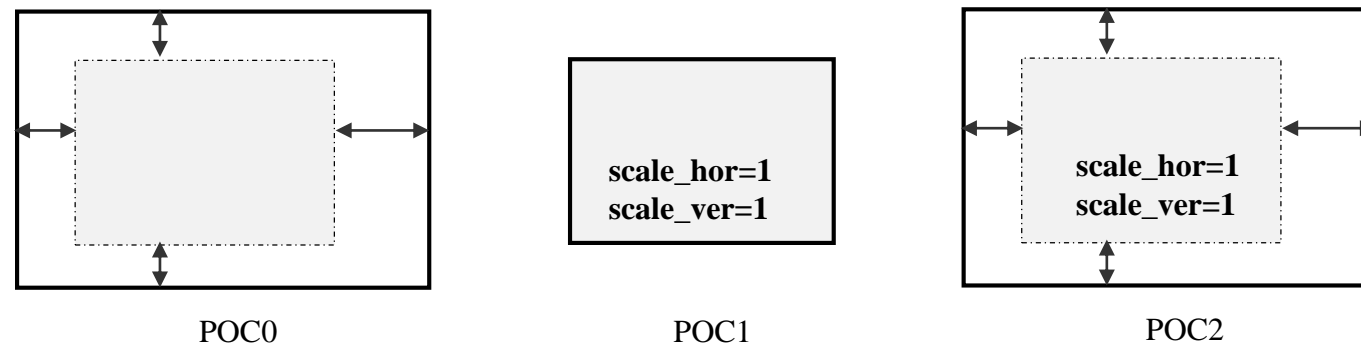
# Proposed ROI Syntax (2): ROI window signaling PH and SH

picture_header_rbsp() {	Descriptor
...	u(1)
if(sps_rpr_roi_window_present_flag) {	
pic_rpr_roi_window_enabled_flag	u(1)
if( pic_rpr_roi_window_enabled_flag ) {	
pic_num_curr_ref_pairs_minus1	ue(v)
for( i = 0; i < pic_num_curr_ref_pairs; i++ ) {	
curr_roi_window_present_flag[ i ]	u(1)
if( curr_roi_window_present_flag[ i ] && num_candidate_roi_windows_minus1 > 0 )	
curr_roi_offset_idx[ i ]	u(v)
ref_roi_window_present_flag[ i ]	u(1)
if( ref_roi_window_present_flag[ i ] && num_candidate_roi_windows_minus1 > 0 )	
ref_roi_offset_idx[ i ]	u(v)
}	
}	
...	
}	

slice_header() {	Descriptor
...	u(v)
if( pic_rpr_roi_window_enabled_flag )	
for( i = 0; i < 2; i++ )	
for( j = 0; j < NumRefIdxActive[ i ]; j++ )	
if( pic_num_curr_ref_pairs_minus1 > 0 )	
slice_curr_ref_pair_idx[ i ][ j ]	ue(v)
...	
}	

# Enabling ROI RPR when scaling ratio is equal to 1

- In VVC draft 7, RPR is disabled when ref pic and cur pic are the same size (ScalingRatio=1.0)
- However, ROI RPR is meaningful when ScalingRatio=1.0
  - ROI is created by cropping alone without additional re-sampling
- For ROI RPR, it is proposed to additionally check if the PicOutputWidthL of ref and cur pic or PicOutputHeightL of ref and cur pic is not equal when ScalingRatio=1.0 (which indicates there is an ROI either in cur or ref pic)
  - To determine the applicability of other coding tools (TMVP, BDOF/PROF, DMVP).



# Specification: Changes to decoding process

- Derivation of scaling ratio
  - 8.3.2 (Decoding process for reference picture lists construction)
    - Add text to construct reference and current ROI window offset parameters
    - Add text of ROI RPR enabled (if not, RPR default)
      - Add additional text for RefPicsScaled to allow scaling ratio equal to 1 if ROI RPR is enabled
- Offset handling in fractional interpolation process
  - 8.5.6.3.1 (Fractional sample interpolation process: General subsection):
    - Add text to specify reference and current ROI window offset parameters as inputs to the process
    - Change variable names for reference picture scale and offsets, and add variables for current picture offsets
    - Change variable names for determination of locations pointed to by motion vector (equations 937, 938, 946, and 948)

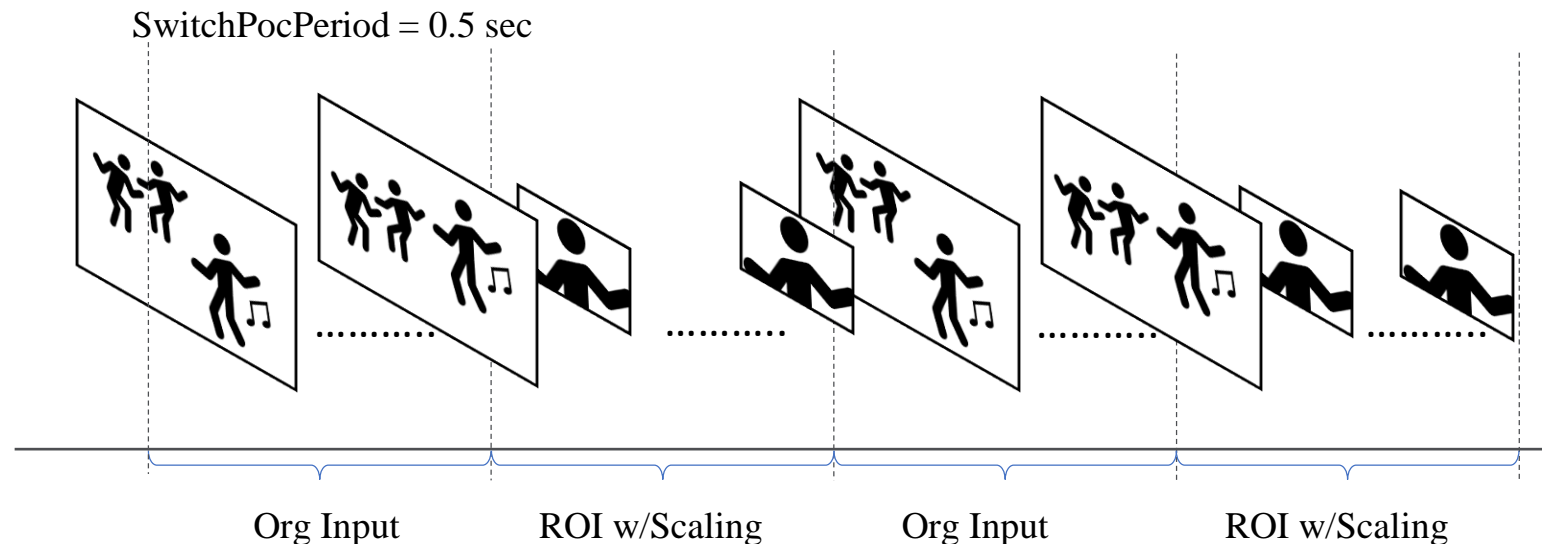
# Software Implementation

- Based on VTM7.0 and MR1089
  - MR1089: ([https://vcgit.hhi.fraunhofer.de/jvet/VVCSsoftware\\_VTM/merge\\_requests/1089](https://vcgit.hhi.fraunhofer.de/jvet/VVCSsoftware_VTM/merge_requests/1089)) JVET-P0590: scaling window implementation and JVET-P0592: chroma phase offset implementation.
  - Thanks Vadim Seregin for providing software
- Changes are mainly modifications of existing RPR framework.
  - Most software changes are made at encoder side.
  - No additional memory storage is introduced.

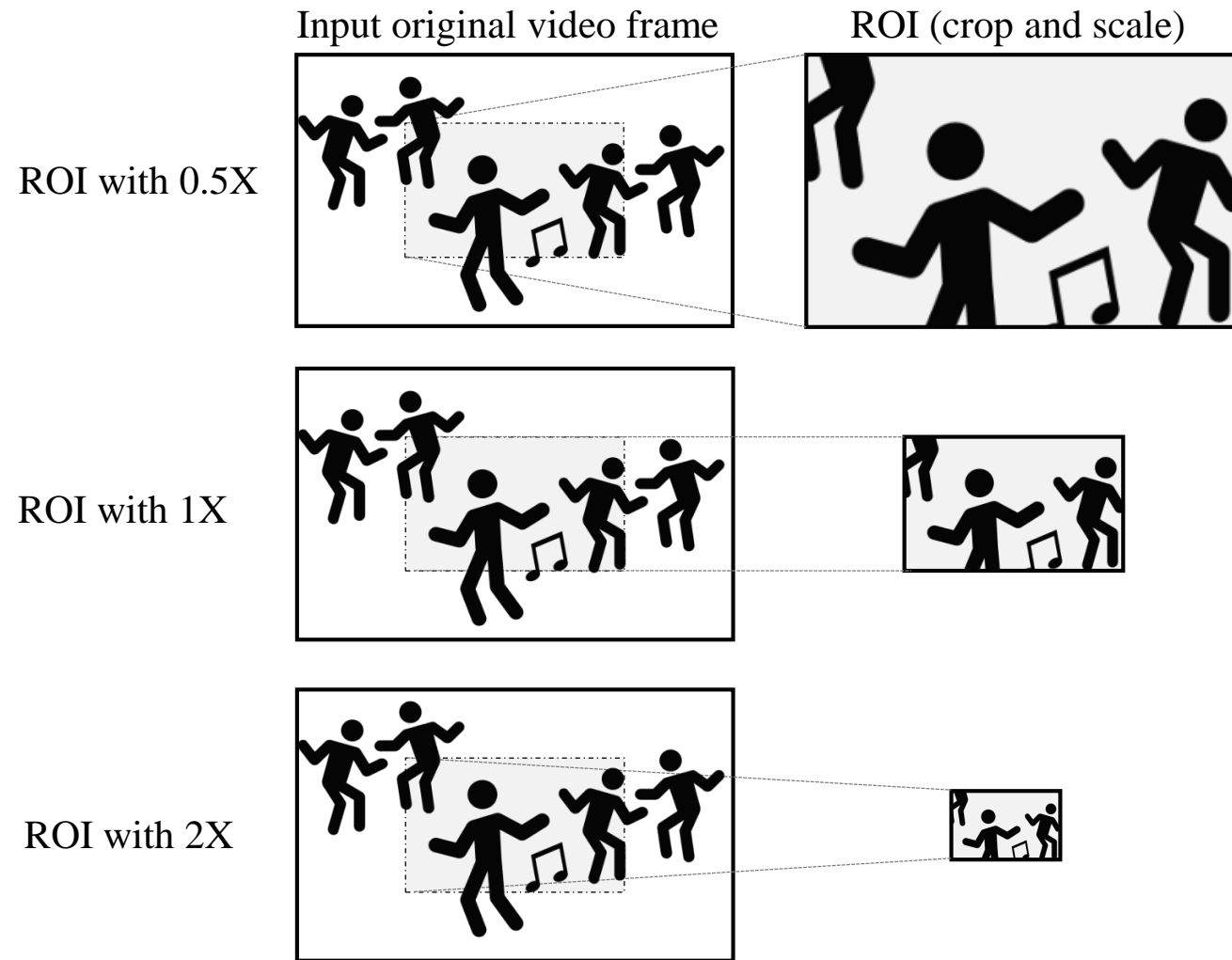


# Functionality Test (1)

- Test condition follows RPR CTC.
  - Switching between original input and ROI happens in every 0.5 seconds
  - FractionNumFrames is set to 0.5 as in RPR tests to encode only half of the whole clip
- Experiment setup
  - ROI is taken as center of picture
  - ROI with 3 different scaling factors: 0.5X, 1X, 2X



# Functionality Test (2)



# Bit Rate Overhead

- ROI window list in SPS:
  - LDB ROI 0.5x: One ROI window
  - Overhead depends on resolution and ROI offsets value
- PH and SH bits overhead
  - For CTC LDB configuration (up to 8 ref pics), slice header overhead is 1~25 bits per slice.
  - For CTC LDP configuration (up to 4 ref pics), slice header overhead is 1~13 bits per slice.

Low Delay B Main 10, 0.5X				ROI offsets				SPS overhead (bits)
		Width	Height	L	R	T	B	
Class B	MarketPlace	1920	1080	480	480	270	270	65
	RitualDance	1920	1080	480	480	270	270	65
	Cactus	1920	1080	480	480	270	270	65
	BasketballDrive	1920	1080	480	480	270	270	65
	BQTerrace	1920	1080	480	480	270	270	65
Class C	BasketballDrill	832	480	208	208	120	120	43
	BQMall	832	480	208	208	120	120	43
	PartyScene	832	480	208	208	120	120	43
	RaceHorses	832	480	208	208	120	120	43
Class D	BasketballPass	416	240	104	104	60	60	41
	BQSquare	416	240	104	104	60	60	41
	BlowingBubbles	416	240	104	104	60	60	41
	RaceHorses	416	240	104	104	60	60	41
Class E	FourPeople	1280	720	320	320	180	180	53
	Johnny	1280	720	320	320	180	180	53
	KristenAndSara	1280	720	320	320	180	180	53
Class F	BasketballDrillText	832	480	208	208	120	120	43
	ArenaOfValor	1920	1080	480	480	270	270	65
	SlideEditing	1280	720	320	320	180	180	53
	SlideShow	1280	720	320	320	180	180	53

# Scaling Ratio Constraints

- The constraints suggested in JVET-Q0331[5] item 1 can be applied for ROI RPR summarized below:
- When  $\text{RefPicIsScaled}[i][j]$  is equal to 1, it is a requirement of bitstream conformance that the following constraints apply:
  - $\text{RefPicScale}[i][j][0]$  shall be smaller than or equal to  $2^M$
  - $\text{RefPicScale}[i][j][0]$  shall be larger than or equal to  $2^N$
  - $\text{RefPicScale}[i][j][1]$  shall be smaller than or equal to  $2^M$
  - $\text{RefPicScale}[i][j][1]$  shall be larger than or equal to  $2^N$

Where  $M = 15$ ,  $N = 11$

# Worst Case MC Memory Constraints

- Whatever agreed constraints for RPR can be used for ROI RPR.
  - ROI window is no bigger than scaling window.
- Using scaling ratio directly in constraints
- Example: JVET-Q0179: AHG9: Bitstream conformance requirement related to RPR scaling ratio for worst case MC memory bandwidth reduction
  - Using method 1 as example, constraints are rewritten in the context of ROI RPR

$$\text{pic\_width\_in\_luma\_samples} * \text{pic\_height\_in\_luma\_samples} * (16 * \text{RefPicScale}[i][j][0] / (1 \ll 14) + 7) * (4 * \text{RefPicScale}[i][j][1] / (1 \ll 14) + 7) \leq \text{pic\_width\_max\_in\_luma\_samples} * \text{pic\_height\_max\_in\_luma\_samples} * 253$$
$$\text{pic\_width\_in\_luma\_samples} * \text{pic\_height\_in\_luma\_samples} * (4 * \text{RefPicScale}[i][j][0] / (1 \ll 14) + 7) * (16 * \text{RefPicScale}[i][j][1] / (1 \ll 14) + 7) \leq \text{pic\_width\_max\_in\_luma\_samples} * \text{pic\_height\_max\_in\_luma\_samples} * 253$$

# Conclusion

- Proposal to add ROI support in RPR
  - Syntax change to add ROI window offsets parameter list in SPS, ROI window pairs for ref pic or cur pic in PH, index to pairs in SH
  - Decoding process changes are minor: replace scaling window offset parameters with ROI window offset parameters
  - Functionality: extend use cases of RPR
  - ROI scalability is automatically supported
- Specification and software implementation are provided
- Propose for adoption

# Acknowledgement

- We would like to thank Alibaba for the crosscheck JVET-Q0567
- We would like to thank Editor Y.K Wang to help revise specification.



# JVET-Q0199 AHG8: Support of ROI (Region-of-Interest) RPR

---

17th JVET Meeting: Brussels, BE, 7–17 January 2020

Dolby Laboratories, Inc.