



JCTVC-D126

Syntax for AMVP Parsing Error Control

Jian-Liang Lin, Yu-Wen Huang, Chih-Ming Fu, Ching-Yeh Chen,
Yu-Pao Tsai and Shawmin Lei



Presented by Jian-Liang Lin
4th Meeting: Daegu, KR
20-28 January, 2011

Problem

- AMVP includes the **temporal motion vector predictor** in the candidate set
- When a temporal motion vector predictor (MV) from a previous picture cannot be decoded correctly
- A **mismatch** between the candidate set on the encoder side and that on the decoder side occurs
- Result in **parsing error** of the index of the best MVP candidate
- The rest of the current picture and following pictures can not be parsed correctly
- One small decoding error of an MV causes **parsing error propagation** of many subsequent pictures

Overview

- In order to control this parsing error propagation, the use of the **temporal MVP** needs to be constrained
- 3 new syntax elements: 2 in the sequence parameter set (SPS) and 1 in the slice header
- The parsing error can thus be **controlled** within a few pictures instead of uncontrolled propagation to many pictures
- The proposed syntax design provides multiple trade-off points between parsing error resilience and coding efficiency

Proposed Syntax Description in SPS level

- When **temporal_mvp_candidates_not_used** = 1
 - no temporal MVPs are allowed and no parsing error propagation due to AMVP will occur
 - but the coding efficiency may be significantly decreased.
- When **temporal_mvp_candidates_not_used_in_ref_pic** = 1
 - the parsing error may occurs in a non-reference picture
 - parsing error propagation will be terminated by the next reference picture
 - better coding efficiency
 - can not be used in the LD configuration

seq_parameter_set_rbsp() {	C	Descriptor
...		
temporal_mvp_candidates_not_used	1	u(1)
if(!temporal_mvp_candidates_not_used)		
temporal_mvp_candidates_not_used_in_ref_pic	1	u(1)
...		
}		

Proposed Syntax Description in Slice level

- When **temporal_mvp_candidate_not_used_in_slice** = 1
 - the parsing error propagation caused from any prior slice will be terminated by the current slice.
 - Most flexible and can be used for the LD configuration
 - Different frequency of setting this flag provides full flexibility of trade-off points between the parsing error resilience and the coding efficiency

slice_header() {	C	Descriptor
...		
if(!temporal_mvp_candidates_not_used && !temporal_mvp_candidates_not_used_in_ref_pic)		
temporal_mvp_candidates_not_used_in_slice	2	u(1)
...		
}		

Conclusion

- Propose two flags in the SPS
 - temporal_mvp_candidates_not_used
 - temporal_mvp_candidates_not_used_in_ref_pic
- Propose one flag in the slice header
 - temporal_mvp_candidates_not_used_in_slic
- Allow flexible trade-off points between parsing error resilience and coding efficiency for AMVP.
- Propose to adopt the syntax design