



## JVET-O0438

Non-CE4: Cleanup for affine AMVR  
enabling flag in SPS level

G. Ko, D. Kim, J. Jung, J. Son, J. Kwak

July, 2019

WILUS Inc.

# ○ Introduction

- In the current VVC, there are affine-related enabling flags in SPS level including **sps\_affine\_amvr\_enabled\_flag** and **sps\_affine\_enabled\_flag**.

| seq parameter set rbsp( ) {                           | Descriptor |
|-------------------------------------------------------|------------|
| // ...                                                | u(1)       |
| <b>sps affine amvr enabled flag</b>                   | u(1)       |
| <b>sps dmvr enabled flag</b>                          | u(1)       |
| <b>sps mmvd enabled flag</b>                          | u(1)       |
| <b>sps isp enabled flag</b>                           | u(1)       |
| <b>sps mrl enabled flag</b>                           | u(1)       |
| <b>sps mip enabled flag</b>                           | u(1)       |
| <b>sps cclm enabled flag</b>                          | u(1)       |
| if( sps cclm enabled flag && chroma format idc == 1 ) |            |
| <b>sps cclm colocated chroma flag</b>                 | u(1)       |
| <b>sps mts enabled flag</b>                           | u(1)       |
| if( sps mts enabled flag ) {                          |            |
| <b>sps explicit mts intra enabled flag</b>            | u(1)       |
| <b>sps explicit mts inter enabled flag</b>            | u(1)       |
| }                                                     |            |
| <b>sps sbt enabled flag</b>                           | u(1)       |
| if( sps sbt enabled flag )                            |            |
| <b>sps sbt max size 64 flag</b>                       | u(1)       |
| <b>sps affine enabled flag</b>                        | u(1)       |
| if( sps affine enabled flag )                         |            |
| <b>sps affine type flag</b>                           | u(1)       |
| // ...                                                |            |
| }                                                     |            |



# ○ Introduction

- **sps\_affine\_amvr\_enabled\_flag** is used only when parsing **amvr\_flag** in CU level.

| coding_unit( x0, y0, cbWidth, cbHeight, treeType ) {                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            | Descriptor |
|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------|
| // ...                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          |            |
| if( ( sps_amvr_enabled_flag && inter_affine_flag[ x0 ][ y0 ] == 0 &&<br>( MvdL0[ x0 ][ y0 ][ 0 ] != 0    MvdL0[ x0 ][ y0 ][ 1 ] != 0   <br>MvdL1[ x0 ][ y0 ][ 0 ] != 0    MvdL1[ x0 ][ y0 ][ 1 ] != 0 ) )   <br>( <b>sps_affine_amvr_enabled_flag</b> && inter_affine_flag[ x0 ][ y0 ] == 1 &&<br>( MvdCpL0[ x0 ][ y0 ][ 0 ][ 0 ] != 0    MvdCpL0[ x0 ][ y0 ][ 0 ][ 1 ] != 0   <br>MvdCpL1[ x0 ][ y0 ][ 0 ][ 0 ] != 0    MvdCpL1[ x0 ][ y0 ][ 0 ][ 1 ] != 0   <br>MvdCpL0[ x0 ][ y0 ][ 1 ][ 0 ] != 0    MvdCpL0[ x0 ][ y0 ][ 1 ][ 1 ] != 0   <br>MvdCpL1[ x0 ][ y0 ][ 1 ][ 0 ] != 0    MvdCpL1[ x0 ][ y0 ][ 1 ][ 1 ] != 0   <br>MvdCpL0[ x0 ][ y0 ][ 2 ][ 0 ] != 0    MvdCpL0[ x0 ][ y0 ][ 2 ][ 1 ] != 0   <br>MvdCpL1[ x0 ][ y0 ][ 2 ][ 0 ] != 0    MvdCpL1[ x0 ][ y0 ][ 2 ][ 1 ] != 0 ) ) ) { |            |
| <b>amvr_flag</b> [ x0 ][ y0 ]                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   | ae(v)      |
| if( amvr_flag[ x0 ][ y0 ] )                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |            |
| <b>amvr_precision_flag</b> [ x0 ][ y0 ]                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         | ae(v)      |
| }                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               |            |
| // ...                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          |            |
| }                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               |            |



# ○ Proposed method

- It is proposed to signal `sps_affine_amvr_enabled_flag` when `sps_affine_enabled_flag` is 1.

| seq parameter set rbsp() {                            | Descriptor      |
|-------------------------------------------------------|-----------------|
| // ...                                                | u(1)            |
| <del>sps affine amvr enabled flag</del>               | <del>u(1)</del> |
| sps dmvr enabled flag                                 | u(1)            |
| sps mmvd enabled flag                                 | u(1)            |
| sps isp enabled flag                                  | u(1)            |
| sps mrl enabled flag                                  | u(1)            |
| sps mip enabled flag                                  | u(1)            |
| sps cclm enabled flag                                 | u(1)            |
| if( sps cclm enabled flag && chroma format idc == 1 ) |                 |
| sps cclm colocated chroma flag                        | u(1)            |
| sps mts enabled flag                                  | u(1)            |
| if( sps mts enabled flag ) {                          |                 |
| sps explicit mts intra enabled flag                   | u(1)            |
| sps explicit mts inter enabled flag                   | u(1)            |
| }                                                     |                 |
| sps sbt enabled flag                                  | u(1)            |
| if( sps sbt enabled flag )                            |                 |
| sps sbt max size 64 flag                              | u(1)            |
| sps affine enabled flag                               | u(1)            |
| if( sps affine enabled flag ) {                       |                 |
| sps affine type flag                                  | u(1)            |
| sps affine amvr enabled flag                          | u(1)            |
| }                                                     |                 |
| // ...                                                |                 |
| }                                                     |                 |



# Experimental results

- Under CTCs

| Random access Main10 |       |       |       |      |      |
|----------------------|-------|-------|-------|------|------|
| Over VTM-5.0         |       |       |       |      |      |
|                      | Y     | U     | V     | EncT | DecT |
| Class A1             | 0.00% | 0.00% | 0.00% | 100% | 100% |
| Class A2             | 0.00% | 0.00% | 0.00% | 100% | 100% |
| Class B              | 0.00% | 0.00% | 0.00% | 100% | 100% |
| Class C              | 0.00% | 0.00% | 0.00% | 100% | 100% |
| Class E              |       |       |       |      |      |
| <b>Overall</b>       | 0.00% | 0.00% | 0.00% | 100% | 100% |
| Class D              | 0.00% | 0.00% | 0.00% | 100% | 100% |
| Class F              | 0.00% | 0.00% | 0.00% | 100% | 100% |

| Low delay B Main10 |       |       |       |      |      |
|--------------------|-------|-------|-------|------|------|
| Over VTM-5.0       |       |       |       |      |      |
|                    | Y     | U     | V     | EncT | DecT |
| Class A1           |       |       |       |      |      |
| Class A2           |       |       |       |      |      |
| Class B            | 0.00% | 0.00% | 0.00% | 100% | 100% |
| Class C            | 0.00% | 0.00% | 0.00% | 101% | 101% |
| Class E            | 0.00% | 0.00% | 0.00% | 100% | 101% |
| <b>Overall</b>     | 0.00% | 0.00% | 0.00% | 100% | 100% |
| Class D            | 0.00% | 0.00% | 0.00% | 101% | 100% |
| Class F            | 0.00% | 0.00% | 0.00% | 101% | 100% |



# ○ Conclusion

- The contribution proposes to signal `sps_affine_amvr_enabled_flag` if affine motion compensation is enabled in SPS level.
- Thank Chosun University for crosscheck

