#### Adaptive loop filter unit syntax

|  |  |
| --- | --- |
| alf\_unit(rx, ry, cIdx, lcuIdx, endrX, oneUnitFlag ) { | Descriptor |
| if( oneUnitFlag ) { |  |
| if( lcuIdx = = 0 ) { |  |
| **alf\_lcu\_enable\_flag**[ cIdx ][ ry ][ rx ] | u(1) |
| if( alf\_lcu\_enable\_flag[ cIdx ][ ry ][ rx ] ) |  |
| alf\_info( rx, ry, cIdx ) |  |
| } |  |
| } else { |  |
| if( !alf\_repeat\_row \_flag[ cIdx ] ) { |  |
| if( rx = = 0 || alfRun[ cIdx ][ ry ][ rx ] < 0 ) |  |
|  |  |
| alfRun[ cIdx ][ ry ][ rx ] = **alf\_run** | u(v) |
|  |  |
|  |  |
| … |  |
| } else |  |
| alfRun[ cIdx ][ ry ][ rx ] = alfRun[ cIdx ][ ry − 1 ][ rx ] |  |
| } |  |
| } |  |

#### 7.4.3.5 Adaptive loop filter unit semantics

The number of times the ALF parameters corresponding to an coding treeblock are repeated for subsequent coding treeblocks in the same row is represented by alfRun[ cIdx ][ rx ][ ry ]. The array index cIdx specifies the colour component; cIdx is equal to 0 for luma, equal to 1 for Cb, and equal to 2 for Cr. The array indices rx and ry specify the location ( rx, ry ) of the considered coding treeblock relative to the top-left coding treeblock of the picture.

**alf\_lcu\_enable\_flag**[ cIdx ][ ry ][ rx ]equal to 1 specifies that the adaptive loop filter shall be applied on current coding treeblock at position rx and ry in the colour component cIdx; equal to 0 specifies that adaptive loop filter shall be disabled for the current coding treeblock at position rx and ry in the colour component cIdx.

**alf\_run** specifies the alfRun of current coding treeblock When alfRun is greater than or equal to 0, the syntax elements in alf\_info() are derived from the corresponding syntax elements of the left coding treeblock. The length of the alf\_run syntax element is Ceil( Log2(alf\_num\_lcu\_in\_width\_minus1 − rx + 2) ) bits.

**alf\_merge\_up\_flag** equal to 1 specifies that the syntax elements in alf\_info() are derived from the corresponding syntax elements of the above coding treeblock; equal to 0 specifies that the syntax elements in alf\_info() are not derived from the corresponding syntax elements of the above coding treeblock. When alf\_merge\_up\_flag is not present, it is inferred to be equal to 0.