|  |  |
| --- | --- |
| **A picture containing drawing  Description automatically generatedA picture containing drawing  Description automatically generatedJoint Collaborative Team on Video Coding (JCT-VC)**  **of ITU-T SG 16 WP 3 and ISO/IEC JTC 1/SC 29/WG 11**  38th Meeting: Brussels, BE, 10–17 January 2020 | Document: JCTVC-AL0022-v1 |

|  |  |  |  |
| --- | --- | --- | --- |
| *Title:* | **Film Grain Synthesis Support in AVC and HEVC** | | |
| *Status:* | Input document to JCT-VC | | |
| *Purpose:* | Proposal | | |
| *Author(s) or Contact(s):* | A. M. Tourapis K. Rapaka  D. Singer  K. Kolarov | Tel: |  |
| Email: | [atourapis@apple.com](mailto:atourapis@apple.com)  [krapaka@apple.com](mailto:krapaka@apple.com)  [singer@apple.com](mailto:singer@apple.com)  [kolarov@apple.com](mailto:kolarov@apple.com) |
| *Source:* | Apple, Inc. | | |

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

# Abstract

Inspired by contribution JVET-Q0424, this contribution proposes that if the AV1 film grain synthesis model is to be adopted in VVC, such adoption should be done again in a form of a new film grain SEI message and not, as suggested by JVET-Q0424, be normative. It is also suggested that the same SEI message is also introduced in the AVC and HEVC specifications since that may better fit existing deployed implementations.

# Introduction – Problem statement

Film grain synthesis was originally proposed and adopted in AVC as a method that could help enhance the quality/compression efficiency of certain content types, especially for content containing film grain noise. This was done by allowing an encoder implementation to extract the noise from the video signal, encode the denoised video signal and, using clearly specified film grain synthesis models, permit the synthesis and the addition of a noise signal at the decoder that approximates the characteristics of the extracted noise. These models were signalled in both AVC and HEVC using the film grain characteristics SEI message, while the use of this SEI message from the perspective of those specifications, was completely optional at the decoder. The implementation details of the film grain modeling and the addition process are left to the decoder manufacturer and were considered outside the scope of those specifications.

Unfortunately, because of the non-normative nature of the film grain characteristics SEI message as well as other issues such as the complexity of the specified models, most deployed AVC and HEVC decoders did not include support for such SEI messages. This ended up also discouraging the use of film grain synthesis with such standards, even though there was some early traction with the adoption of Film Grain in, the later abandoned, HD DVD format.

On the other hand, it could be argued that the benefits of film grain synthesis [2] were more clearly recognized during the development of the AOM AV1 video coding standard [1]. To avoid the implementation issues of film grain synthesis in AVC and HEVC, it was decided that film grain synthesis should be mandated in all implementations so as to encourage its use. A simpler model compared to the ones supported in AVC and HEVC was also specified. In principle, however, film grain is still a post-processing stage, while even the AV1 specification recognizes that implementations of film grain modeling may vary given the design of the decoding and processing pipeline in especially HW designs. To this point, although AV1 encourages a bit exact film grain synthesis and addition implementation, it also allows decoders to support and implement their own "interpretation" of the film grain synthesis model, while the generated noise could be added at any stage in the display processing pipeline and after the decoding process, that the decoder deems suitable. For example, film grain noise could be added after proprietary artifact removal or the consideration of frame rate upconversion, which may potentially also be in a different representation model than that of the decoded signal. For example, the decoded signal may be in a 4:2:0 10 bit YCbCr representation, but the noise addition may be done in a 4:2:2 or 4:4:4 YCbCr representation.

Since quite likely some VVC decoder implementations may also support AV1 decoding, contribution JVET-Q0424 [3] recognized that there may be now an opportunity to also have VVC decoders more readily supporting film grain synthesis. Furthermore, it allows different codecs, in this case VVC and AV1, to be able to reuse the same filtered video signals and film grain models for compressing a video signal, which could be an advantage for some services. The proposal in JVET-Q0424 seems to assume that the film grain synthesis module is readily available for other decoders or could be easily added in a VVC decoder module. However, it makes the mistake, in our opinion, of mandating the support of this film grain model in the specification. We see the following reasons why this is not necessary or problematic:

1. The VVC specification also supports the pre-existing film grain modeling schemes in an SEI message form. By making this new method mandatory this could result in conflicts between the existing and new methods especially if both methods are used jointly. It also biases the use of this new method in favor of the existing methods, which might be considered by some as maybe an "unfair" practice, especially since the noise characteristics and behavior of these models have not been carefully evaluated.
2. JVET-Q0424 does not make any mention of the "soft" conformance requirements in AV1 and seems to maybe require that the process is done in a bit exact manner. If so, that would make the design less flexible compared to that in AV1.
3. It creates maybe a dangerous precedence where post-filtering methods are normatively adopted in video coding specifications.
4. It should be pointed out that the AV1 film grain model has certain limitations given that it is initialized at the picture level. That means that processing at the sub-picture level may result in different performance and noise generation compared to the noise being added at the picture level.

It could also be argued that there may be more value in supporting this new model not only in VVC but also in older video standards, such as AVC and HEVC. For example, a service that supports AV1 decoders may also have to support legacy AVC and HEVC decoders, and may need to transcode AV1 bitstreams to those different formats. Without the new film grain modeling support one would most likely have to transcode such bitstreams after fully reconstructing the videos with the additive film grain noise. Such would likely have a big negative impact in coding efficiency. Alternatively, the film grain characteristics SEIs may be used but would require their implementations in those systems, which might not be possible or desirable because of its higher complexity.

# Proposed alternative

Unlike JVET-Q0424, we would like to request that if the AV1 film grain model is added in VVC, that should be added also in an SEI message form and not be mandatory. If film grain should be mandated, that should be left at the application level. The same message should then also be added in AVC and HEVC. We feel that if already AV1 decoders support film grain modeling, then the reasoning of making this mandatory in VVC as was done in AV1, as to encourage adoption and implementations of film grain modeling, goes away. The use of an SEI also avoids biasing the VVC specification towards this particular scheme.

# Editorial changes

We have borrowed the syntax proposed in [4] and modified it into an SEI message form. It should be noted that this SEI message, as also the original film grain characteristics SEI message, has a parsing dependency with chroma\_format\_idc. We did not try to remove this dependency, however we are wondering, if this was removed, if it would maybe make it easier to parse such SEI messages. We will leave that decision to the group assuming that such proposal was adopted into the VVC, AVC, and/or HEVC specifications.

## Alternative film grain model SEI message syntax

|  |  |
| --- | --- |
| alternative\_film\_grain\_model( payloadSize ){ | **Descriptor** |
| **afgm\_alternative\_film\_grain\_model\_cancel\_flag** | u(1) |
| if ( !afgm\_alternative\_film\_grain\_model\_cancel\_flag ) { |  |
| **apply\_grain** | u(1) |
| **grain\_seed** | u(16) |
| **afgm\_update\_model\_flag** | u(1) |
| if ( afgm\_update\_model\_flag ) { | u(1) |
| **num\_y\_points** | u(4) |
| for ( i = 0; i < num\_y\_points; i++ ) { |  |
| **point\_y\_value[ i ]** | u(8) |
| **point\_y\_scaling[ i ]** | u(8) |
| } |  |
| **chroma\_scaling\_from\_luma** | u(1) |
| if ( chromaGrainScalingSignaling) { |  |
| **num\_cb\_points** | u(4) |
| for ( i = 0; i < num\_cb\_points; i++ ) { |  |
| **point\_cb\_value[ i ]** | u(8) |
| **point\_cb\_scaling[ i ]** | u(8) |
| } |  |
| **num\_cr\_points** | u(4) |
| for ( i = 0; i < num\_cr\_points; i++ ) { |  |
| **point\_cr\_value[ i ]** | u(8) |
| **point\_cr\_scaling[ i ]** | u(8) |
| } |  |
| } |  |
| **grain\_scaling\_minus\_8** | u(2) |
| **ar\_coeff\_lag** | u(2) |
| if ( num\_y\_points ) { |  |
| for ( i = 0; i < numPosLuma; i++ ) |  |
| **ar\_coeffs\_y\_plus\_128[ i ]** | u(8) |
| } |  |
| if ( chroma\_scaling\_from\_luma || num\_cb\_points ) { |  |
| for ( i = 0; i < numPosChroma; i++ ) |  |
| **ar\_coeffs\_cb\_plus\_128[ i ]** | u(8) |
| } |  |
| if ( chroma\_scaling\_from\_luma || num\_cr\_points ) { |  |
| for ( i = 0; i < numPosChroma; i++ ) |  |
| **ar\_coeffs\_cr\_plus\_128[ i ]** | u(8) |
| } |  |
| **ar\_coeff\_shift\_minus\_6** | u(2) |
| **grain\_scale\_shift** | u(2) |
| if ( num\_cb\_points ) { |  |
| **cb\_mult** | u(8) |
| **cb\_luma\_mult** | u(8) |
| **cb\_offset** | u(9) |
| } |  |
| if ( num\_cr\_points ) { |  |
| **cr\_mult** | u(8) |
| **cr\_luma\_mult** | u(8) |
| **cr\_offset** | u(9) |
| } |  |
| **grain\_blocks\_overlap\_flag** | u(1) |
| **clip\_to\_restricted\_range** | u(1) |
| } |  |
| **afgm\_alternative\_film\_grain\_model\_persistence\_flag** | u(1) |
| } |  |
| } |  |

## Alternative film grain model SEI message semantics

This SEI message provides the decoder with a parameterized model for an alternative film grain synthesis model.

Use of this SEI message requires the definition of the following parameters and functions:

– A chroma format indicator, denoted herein by chroma\_format\_idc, as described in clause 7.3.

**afgm\_alternative\_film\_grain\_model\_cancel\_flag** equal to 1 indicates that the SEI message cancels the persistence of any previous alternative film grain model SEI message in output order that applies to the current layer. afgm\_alternative\_film\_grain\_model\_cancel\_flag equal to 0 indicates that alternative film grain model information follows.

**apply\_grain** equal to 1 specifies that film grain specified by this model shall be added to the reconstructed picture. apply\_grain equal to 0 specifies that film grain shall not be added.

**grain\_seed** specifies the starting value for the pseudo-random number generator used in film grain synthesis.

**afgm\_update\_model\_flag** equal to 1 specifies that the film grain model parameters shall be updated.afgm\_update\_model\_flagequal to 0 specifies that the film grain model parameters shall not be updated

**num\_y\_points** specifies the number of points for the piece-wise linear scaling function of the luma component. It is a requirement of bitstream conformance that num\_y\_points is less than or equal to 14.

**point\_y\_value[ i ]** represents the luma value for the i-th point of the piecewise linear scaling function for luma component. The values are signaled on the scale of 0..255. Note: (In 10-bit content, these values correspond to luma values divided by 4. In 12-bit content, these values correspond to luma values divided by 16).

If i is greater than 0, it is a requirement of bitstream conformance that point\_y\_value[ i ] is greater than point\_y\_value[ i - 1 ], which ensures x coordinates are signaled in the ascending order.

**point\_y\_scaling[ i ]** represents the scaling value for the i-th point of the piecewise linear scaling function for luma component.

**chroma\_scaling\_from\_luma** specifies that the chroma scaling function is inferred from the luma scaling function.

If any of the following statements is true, chromaGrainScalingSignaling is set to 0, otherwise chromaGrainScalingSignaling is set to 1:

* chroma\_scaling\_from\_luma is equal to 1,
* chroma\_format\_idc is equal to 0,
* chroma\_format\_idc is equal to 1 and num\_y\_points is equal to 0.

**num\_cb\_points** specifies the number of points for the piece-wise linear scaling function of the cb component. It is a requirement of bitstream conformance that num\_cb\_points is less than or equal to 10.

Note: When chroma\_scaling\_from\_luma is equal to 1, chroma scaling also needs to support up to 14 points since it is allowed for num\_y\_points to take values up to 14.

**point\_cb\_value[ i ]** represents the x coordinate for the i-th point of the piece-wise linear scaling function for cb component. The values are signaled on the scale of 0..255.

For values of i greater than 0, it is a requirement of bitstream conformance that point\_cb\_value[ i ] is greater than point\_cb\_value[ i - 1 ].

**point\_cb\_scaling[ i ]** represents the function value for the i-th point of the piecewise linear scaling function for cb component.

**num\_cr\_points** specifies the number of points for the piece-wise linear scaling function of the cr component. It is a requirement of bitstream conformance that num\_cr\_points is less than or equal to 10.

When chroma\_format\_idc is equal to 1 and num\_cb\_points is equal to 0, it is a requirement of bitstream conformance that num\_cr\_points is equal to 0.

When chroma\_format\_idc is equal to 1 and num\_cb\_points is not equal to 0, it is a requirement of bitstream conformance that num\_cr\_points is not equal to 0.

Note: This requirement ensures that for 4:2:0 chroma subsampling, film grain noise will be applied to both chroma components, or to neither. There is no restriction for 4:2:2 or 4:4:4 chroma subsampling.

**point\_cr\_value[ i ]** represents the x coordinate for the i-th point of the piece-wise linear scaling function for cr component. The values are signaled on the scale of 0..255.

For values of greater than 0, it is a requirement of bitstream conformance that point\_cr\_value[ i ] is greater than point\_cr\_value[ i - 1 ].

**point\_cr\_scaling[ i ]** represents the scaling value for the i-th point of the piecewise linear scaling function for cr component.

**grain\_scaling\_minus\_8** represents the shift – 8 applied to the values of the grain. The grain\_scaling\_minus\_8 can take values of 0..3 and determines the range and quantization step of the standard deviation of film grain.

**ar\_coeff\_lag** specifies the number of auto-regressive coefficients for luma and chroma.

numPosLuma = 2 \* ar\_coeff\_lag \* ( ar\_coeff\_lag + 1 )

numPosChroma = numPosLuma + 1

**ar\_coeffs\_y\_plus\_128[ i ]** specifies values of auto-regressive coefficients used for the luma plane.

**ar\_coeffs\_cb\_plus\_128[ i ]** specifies values of auto-regressive coefficients used for the cb plane.

**ar\_coeffs\_cr\_plus\_128[ i ]** specifies values of auto-regressive coefficients used for the cr plane.

**ar\_coeff\_shift\_minus\_6** specifies the range of auto-regressive coefficients. Values of 0, 1, 2, and 3 correspond to the auto-regressive coefficients range of [-2, 2), [-1, 1), [-0.5, 0.5) and [-0.25, 0.25) respectively.

**grain\_scale\_shift** specifies how much the Gaussian random numbers should be scaled down during the grain synthesis process.

**cb\_mult** represents a multiplier for the cb component used in derivation of the input index to the cb component scaling function.

**cb\_luma\_mult** represents a multiplier for the average luma component used in derivation of the input index to the cb component scaling function.

**cb\_offset** represents an offset used in derivation of the input index to the cb component scaling function.

**cr\_mult** represents a multiplier for the cr component used in derivation of the input index to the cr component scaling function.

**cr\_luma\_mult** represents a multiplier for the average luma component used in derivation of the input index to the cr component scaling function.

**cr\_offset** represents an offset used in derivation of the input index to the cr component scaling function.

**grain\_blocks\_overlap\_flag** equal to 1 indicates that the overlap between film grain blocks is applied. **g**rain\_blocks\_overlap\_flag equal to 0 indicates that the overlap between film grain blocks is not applied.

**clip\_to\_restricted\_range** equal to 1 indicates that clipping to the restricted (studio) range is applied to the sample values after adding the film grain. clip\_to\_restricted\_range equal to 0 indicates that clipping to the full range shall be applied to the sample values after adding the film grain.

**afgm\_alternative\_film\_grain\_model\_persistence\_flag** specifies the persistence of the alternative film grain model SEI message for the current layer.

afgm\_alternative\_film\_grain\_model\_persistence\_flagequal to 0 specifies that the alternative film grain model SEI message applies to the current decoded picture only.

Let picA be the current picture. afgm\_alternative\_film\_grain\_model\_persistence\_flagequal to 1 specifies that the alternative film grain model SEI message persists for the current layer in output order until any of the following conditions are true:

– A new CLVS of the current layer begins.

– The bitstream ends.

– A picture picB in the current layer in an AU containing a film grain characteristics SEI message that is applicable to the current layer is output for which PicOrderCnt( picB ) is greater than PicOrderCnt( picA ), where PicOrderCnt( picB ) and PicOrderCnt( picA ) are the PicOrderCntVal values of picB and picA, respectively, immediately after the invocation of the decoding process for picture order count for picB.

## Film grain synthesis process

As mentioned in [4], the recommended film grain synthesis process can be found in [1], Section 7.18.3.

# Conclusions

This contribution requests that if the AV1 film grain model is to be adopted in VVC, then such should be adopted in an SEI message format and not be mandatory. The same SEI message should also then be added in AVC and HEVC. Suggested syntax and semantics are provided.

# References

1. “AV1 Bitstream & Decoding Process Specification”, Version 1.0.0 with Errata 1, Jan. 8, 2019, https://aomediacodec.github.io/av1-spec/av1-spec.pdf
2. A. Norkin and N. Birkbeck, “Film Grain Synthesis for AV1 Video Codec”, in Proc. IEEE Data Compression Sonference (DCC), Snowbird, Utah, Mar. 2018.
3. R. Sjöberg, D. Saffar, M. Pettersson, M. Damghanian, “Mandatory film grain”, JVET-Q0424, Brussels, Jan. 2020.
4. A. Norkin, "Film Grain Synthesis Support", JVET-Q0533, Brussels, Jan. 2020.
5. https://aomedia.googlesource.com/aom

# Patent rights declaration(s)

**Apple, Inc. may have current or pending patent rights relating to the technology described in this contribution and, conditioned on reciprocity, is prepared to grant licenses under reasonable and non-discriminatory terms as necessary for implementation of the resulting ITU-T Recommendation | ISO/IEC International Standard (per box 2 of the ITU-T/ITU-R/ISO/IEC patent statement and licensing declaration form).**