**3GPP TSG SA WG4#117e S4-220045-Constraints-Google**

**E-meeting, 14th – 23rd February 2022**

|  |
| --- |
| *CR-Form-v12.0* |
| **PSEUDO CHANGE REQUEST** |
|  |
|  | **26**.**955** | **CR** | pseudo | **rev** | **-** | **Current version:** | **1.4.6** |  |
|  |
| *For* [***HE******LP***](http://www.3gpp.org/3G_Specs/CRs.htm#_blank)*on using this form: comprehensive instructions can be found at* [*http://www.3gpp.org/Change-Requests*](http://www.3gpp.org/Change-Requests)*.* |
|  |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| ***Proposed change affects:*** | UICC apps |  | ME |  | Radio Access Network |  | Core Network |  |

|  |
| --- |
|  |
| ***Title:***  | **[FS\_5G\_Video] Proposed General Definitions for Coding constraints** |
|  |  |
| ***Source to WG:*** | Qualcomm Incorporated, Tencent, Google |
| ***Source to TSG:*** |  |
|  |  |
| ***Work item code:*** | FS\_5GVideo |  | ***Date:*** | 04/02/2022 |
|  |  |  |  |  |
| ***Category:*** | **B** |  | ***Release:*** |   |
|  | *Use one of the following categories:****F*** *(correction)****A*** *(mirror corresponding to a change in an earlier release)****B*** *(addition of feature),* ***C*** *(functional modification of feature)****D*** *(editorial modification)*Detailed explanations of the above categories canbe found in 3GPP [TR 21.900](http://www.3gpp.org/ftp/Specs/html-info/21900.htm). | *Use one of the following releases:Rel-8 (Release 8)Rel-9 (Release 9)Rel-10 (Release 10)Rel-11 (Release 11)Rel-12 (Release 12)**Rel-13 (Release 13)Rel-14 (Release 14)Rel-15 (Release 15)Rel-16 (Release 16)* |
|  |  |
| ***Reason for change:*** |  |
|  |  |
| ***Summary of change:*** |  |
|  |  |
| ***Consequences if not approved:*** |  |
|  |  |
| ***Clauses affected:*** |  |
|  |  |
|  | **Y** | **N** |  |  |
| ***Other specs*** |  |  |  Other core specifications  | TS/TR ... CR ...  |
| ***affected:*** |  |  |  Test specifications | TS/TR ... CR ...  |
| ***(show related CRs)*** |  |  |  O&M Specifications | TS/TR ... CR ...  |
|  |  |
| ***Other comments:*** |  |
|  |  |
| ***This CR's revision history:*** |  |

**===== CHANGE =====**

## 5.1 Overview

This clause defines the characterization framework for video codecs for relevant 3GPP scenarios. For this purpose, the following is applied:

- A set of relevant scenarios is defined. The scenarios reflect a typical application for video codecs in 5G systems and networks. The scenarios are introduced in clause 6 and may be extended in future versions of this document.

- For each scenario, one or several reference sequences are defined that serve as the baseline for anchor generation.

- For each test scenario, one or several anchors are defined and generated. For details on anchors, please refer to clause 5.3. For anchor generation, reference software tools are used as introduced in clause 5.4. Anchors for specific scenarios follow certain general encoding constraints documented in clause 5.6.

- For each of the anchors, metrics are provided. Metrics are documented in detail in clause 5.5.

- Tests for new codecs can be developed and generated. They are equivalent to anchor generation, but possibly for other codecs. Tests are introduced in clause 5.7.

- Codecs are to be characterized against anchors. Characterization is documented in terms of expected bitrate savings for a codec, and may include additional comparison parameters such as complexity increase, etc. The basic characterization framework is introduced in clause 5.8.

- Verification of the provided anchors and tests is needed. A process for verification is introduced in clause 5.9.

An overview of the anchor generation framework and the anchor metrics is provided in Figure 5.1-1. An integral part of this Technical Report is the following information:

- Formats to store reference and anchor sequences

- Reference sequences for each of the defined scenarios

- Reference software encoders

- Anchor configuration files

- Anchor bitstreams in a well-defined anchor bitstream format

- A anchor metric computation based on a reference sequence, anchor bitstream and an anchor sequence.

- Conforming decoders to generate an anchor sequence from an anchor bitstream

- Anchor metrics in a well-defined storage format.

NOTE: as the anchor sequences can be generated by conforming decoders, anchor sequences are not included in this document.



Figure 5.1-1 Anchor Generation Framework and Anchor Metrics Generation

**===== CHANGE =====**

## 5.3 Anchors

Anchors provide a baseline that a tested method can be compared against. Anchors defined in this specification use a codec/profile/level that exists in an existing 3GPP specification as introduced in clause 4.

Anchor tuples are collected to address different quality and bitrates that can then be used for evaluation over a larger set of operation points.

The following principle apply to anchor definitions:

- Each scenario typically has several well-defined anchors

- An anchor is a combination of:

- Explanation on anchor relevance

- Reference sequence

- Reference encoder

- Encoder configuration matching scenario requirements

- Encoding complexity estimation, if available

- Variable encoder configuration to create multiple quality/bitrate variants (using for example QP variations or other bitrate/quality evaluation tools).

- Anchor tuples creating multiple variants, each including

- Anchor bitstream

- Anchor Metrics

- Additional recommended anchor information includes

- MD5 check sum of the complete reconstructed yuv file (anchor sequence)

- Output picture log from reference encoder

- Output picture log from reference decoder

Anchors and anchor tuples are an integral part of this document.

Anchor tuples should be created over a wide range of parameters to provide sufficient data and overlap with expected test results to support the generation of characterization results (see clause 5.7).

The workflow for the generation of anchor tuples is shown in Figure 5.3-1.

 

Figure 5.3-1: Anchor Tuple Generation Framework and Anchor Tuple Metrics Generation

Anchors are provided according to the format as defined in Annex B.3.

**===== CHANGE =====**

## 5.6 General encoding constraints

This clause defined general encoding constraints for encoders that are used in scenarios and configurations in order to ensure a fair comparison between the anchor and the test stream. These encoding constraints may for example address latency requirements, encoding complexity restrictions, functional properties or data management considerations.

Given how the anchors are produced (Clause 6), it is expected that the test encoders should follow similar configuration settings, which would enable similar functionality as per the defined applications.

* *Single-pass encoding*: Only fixed periodic (temporal) QP and coding structures (e.g. GOP order, coding order) are permitted, without the use of any lookahead multi pass encoding of a current picture or multiple pictures ahead encoding that would alter encoding, rate-distortion optimization processes, coding tools settings, the QP or coding structures (e.g. GOP order, coding order) dynamically per content. For HDR content, the QP settings may be adjusted within a frame as a function of the local, average luma and chroma values. QP and coding structures may differ from those used by the anchors, but such differences should be consistent for all content in a given scenario and should be described. Preprocessing in the form of Motion-Compensated Temporal Filtering (MCTF) and residual energy/distortion-based decisions that can adapt the coding type of the current frame are allowed.
* *Multiple variants*: Variable test encoder configuration to create multiple quality/bitrate variants (using for example QP variations or other bitrate/quality evaluation tools) provide similar quality.
* *Random Access Functionality*: When decoding a bitstream from a specific bitstream position, i.e. a random access point, the resulting output pictures are identical as if the bitstream would be decoded from the beginning.
* *Random Access Period*: For a given scenario the random access period of the test encoders matches the random access period of the anchors exactly to ensure the same number of Intra frames in the anchor and test bitstreams.
* *Low-Delay*: encoding configuration that is defined to produce a bitstream only containing pictures with decoding order matching the presentation order.
* *Low-Delay P (Constrained):* Low-delay encoding configuration that is defined such that the encoding process includes performing a constrained prediction of the current block, where all samples within a block are predicted using a single hypothesis.

NOTE:  This configuration illustrates the most largely deployed service configurations observed at the time of producing this report.

* *Low-Delay B*: encoding configuration that is defined as a Low-delay configuration such that the encoding process performs an unrestricted prediction of the current block.

NOTE:  This configuration is intended to reflect the full capabilities of the video codec to be characterized.

**===== CHANGE =====**

## 5.7 Tests

Tests may be executed to compare codecs not yet in 3GPP specifications against anchors defined in this specification. Tests, equivalently to anchors, are collected in tuples to address different quality and bitrates that can then be used for evaluation over a larger set of operation points.

A test is developed against an anchor and is a combination of:

- The corresponding anchor, which includes

- Scenario

- Reference Sequence

- Test encoder

- Test encoder configuration that provides an equivalent setting to the anchor configuration based on the general encoding constraints in clause 5.6.

- Test tuples creating multiple variants, each including

- Test bitstream

- Test Metrics

- Additional recommended test information includes

- MD5 check sum of the complete reconstructed yuv file (reconstructed test sequence)

- Output picture log for reference encoder

- Output picture log for reference decoder

- Tests are an integral part of the Technical Report

The generation of test tuples is shown in Figure 5.7-1.



Figure 5.7-1: Test Tuple Generation Framework and Test Tuple Metrics Generation

For any coding technology being characterized in the study and reported in this document

* the evaluation is expected to be conducted consistent with the framework and test designs defined for the anchors as defined in clauses 5.5 and 6.
* technical documentation to conduct the study is expected to be available and provided. Such information includes normative specification text, reference software and description of configuration files, and codec description.
* additional data such as subjective test results (with description of test methodology and conditions) not conducted as part of this study item, encoding tool and configuration file description is also important information that could be provided

**===== CHANGE =====**

**Mapping to scenarios still need to be done**

**Scenario 1/2:**

* **Random Access**
* - "frameRate": 23.98 or 24.0 or 25 or 30 => IntraPeriod and IDRPeriod set to 32,
* - "frameRate": 50.0 or 59.94 or 60 => IntraPeriod and IDRPeriod set to 64
* **Single pass encoding**
* **4 bitrates**

**Scenario 3:**

* **Single pass encoding**
* **4 bitrates**
* **Random Access every second**
* **Low-Delay P**

**Scenario 3:**

* **Single pass encoding**
* **4 bitrates**
* **Random Access every second or none**
* **Low-Delay B**

**Scenario 5:**

* **Single pass encoding**
* **4 bitrates**
* **Random Access every second or none**
* **Low-Delay B**

8.4.2.4 Scenario 3: Screen Content Scenario

...

**===== CHANGE =====**

8.4.2.4.2 Common Parameters and Settings

The following common Scenario 3 command line options are provided in Table 8.4.2.4.2-1.

Note that this configuration follows the Low-delay P (constrained) configuration defined in Clause 5.6.

**Table 8.4.2.4.2-1 Common Scenario 3 command line options**

|  |  |
| --- | --- |
| --lag-in-frames=0 | No future frame is used as reference in low delay configuration.Key-frame-filtering is a motion compensated temporal filtering. It is only enabled in random access configuration. When --lag-in-frames=0, key frames will not be filtered. |
| --deltaq-mode=0 | disable block based adaptive quantization, so all blocks within a frame use the same quantization parameter |
| --enable-onesided-comp=0 | disable compound prediction for low delay content. Unidirectional prediction Inter frame, only. |
| --max-reference-frames=4 | restrict number of reference frames to 4. |
| --enable-obmc=0 | Disable OBMC |
| --enable-dist-wtd-comp=0 | Disable distance weighted inter compound prediction |
| --enable-diff-wtd-comp=0 | Disable difference weighted inter compound prediction |
| --enable-interintra-comp=0 | Disable inter/intra compound prediction |
| --enable-interinter-wedge=0 | Disable wedge-based inter compound prediction |
| --enable-interintra-wedge=0 | Disable wedge-based inter/intra compound prediction |
| --enable-masked-comp=0 | Disable mask based inter compound prediction |
| --enable-smooth-interintra=0 | Disable smooth inter/intra compound prediction |

====== End Change 6 =======