ISO /IEC JTC 1/SC 29 /WG 1 **N71031**
71th Meeting – La Jolla, CA, USA – 11 March 2016

ISO/IEC JTC1/SC29/WG1
(ITU-T SG16)

Coding of Still Pictures

JBIG	**JPEG**
Joint Bi-level Image Experts Group	Joint Photographic Experts Group

TITLE: Call for Proposals for a low-latency lightweight image coding system

SOURCE: WG1 (Editor: Antonin Descampe, a.descampe@intopix.com)

PROJECT: ISO/IEC 21122 (JPEG XS)

STATUS: Approved

REQUESTED ACTION: For distribution

DISTRIBUTION: Public

Contact:
ISO/IEC JTC 1/SC 29/WG 1 Convener – Prof. Touradj Ebrahimi
EPFL/STI/IEL/GR-EB, Station 11, CH-1015 Lausanne, Switzerland
Tel: +41 21 693 2606, Fax: +41 21 693 7600, E-mail: Touradj.Ebrahimi@epfl.ch

Abstract

Image sequences have been transmitted and stored in uncompressed form in many cases, such as in professional video links (3G/6G/12G-SDI), IP transport (SMPTE2022 5/6 & proprietary uncompressed RTPs), Ethernet transport (IEEE/AVB), and memory buffers. A low-latency lightweight image coding system allows for an increased resolution and frame rate, while offering visually lossless quality with reduced amount of resources such as power and bandwidth. This document is a call for proposals in view of standardization of a low-latency lightweight image coding system.

1 Introduction

Infrastructures and systems capabilities increase at a lower pace than content resolution and are therefore progressively becoming the bottleneck in many applications. As an example, NHK has announced plans to test 8K TV broadcast in 2016 and foresees a full service by 2020 Tokyo Olympics. Transmitting uncompressed 8K content over existing links or over soon-to-be-available alternatives channels is not feasible. A lightweight low-latency coding system appears to be a smart and affordable solution to meet market needs.

Even for transmitting content fitting currently available systems, the use of a lightweight and low-latency coding system could be beneficial as it allows for reduced bandwidth, and consequently results in lowering corresponding cost or enable longer cable runs (for example, usual 3G-SDI cable run is 14m while it reaches 21m for HD-SDI, 50m for SD-SDI).

In a nutshell, the key advantage of a lightweight and low-latency image coding system is to allow increasing resolution and frame rate in a cost-effective manner, i.e.

- **safeguarding all advantages of an uncompressed stream**
    - low power consumption (through lightweight image processing)
    - low-latency in coding and decoding
    - easy to implement (through low complexity algorithm)
    - small size on chip and fast software running on general purpose CPU with the use of SIMD and GPU.
- **without significant increase in required bandwidth**
    - low power consumption (through reasonable bandwidth interfaces)
    - longer cable runs
    - SRAM size & frequency reduction with a frame buffer compression
    - more adequate for current infrastructures

WG1 requests the national bodies, affiliated entities and organizations to submit proposals for testing and evaluation in response to this call. No commitment to any subsequent course of action regarding the proposed technology will be made in advance of the evaluation of proposals.

The aggregate results of the tests will be made public. However, no direct identification of any of the proponents will be made.

Descriptions of proposals must be registered and submitted as input documents to the proposal evaluation at 72nd WG1 meeting (see timeline in Section 2). Moreover, proponents are required to present their proposals during this meeting. Consequently, they are encouraged to attend this meeting. Alternatively, a WebEx presentation can be organized. Further information about logistical steps to attend the meeting can be obtained from the listed contact persons (see Section 10).

2 Timeline

The intended timeline for the evaluation of the proposals is the following:

11/03/2016	CfP Issued
23/05/2016	Deadline for the indication of interest and registration. - Send emails to the people listed in Section 10. - Submit the proposal's overview as input document of 72nd WG1 meeting in Geneva (and send it to the people listed in Section 10). See Section 7.1 for more information on the proposal's overview.
30/05 => 03/06/2016	WG1 meeting, Geneva: review of proposals overviews (attendance to the meeting is recommended)
03/06 => 24/06/2016	Objective and subjective evaluation of the anchors
27/06 => 28/06/2016	JPEG XS AhG meeting: review of anchors evaluation results and agreement on final test set and evaluation procedures. The final list of sequences will be communicated to proponents having registered.
23/09/2016	Deadline for submission of binaries, algorithm description and design (complexity and latency analysis), and encoded-decoded test material.
24/09 => 14/10/2016	Objective and subjective evaluation of the proposals and anchors
17/10 => 21/10/2016	WG1 meeting, Chengdu: proposals assessment, WD1 based on technical proposals and assessment results (attendance to the meeting is required).
28/10/2016	Deadline for submission of the Verification Model source code and the full technical proposal for selected technologies.
11/11/2016	End of editing period for WD1.

Please refer to Section 7 for further details on submission requirements.

The intended timeline for the whole standardisation process is as follows:

CfP	16/02
WD1	16/10
WD2	17/02
CD	17/06
DIS	17/10
IS	18/06

3 Use Cases

3.1 Transport over video links and IP network

In such use cases, a video link and transport protocol is employed to transport video streams at a higher throughput than its physical throughput, thanks to a lightweight compression with a compression ratio ranging from 2:1 to 6:1. Several examples are given in Table 1.

Table 1 – Transport over video links use cases

Video stream	Video throughput[1]	Physical link	Available throughput	Comp. ratio
2K / 60p / 422 / 10 bits	2.7 Gbps	HD-SDI	1.33 Gbps	~ 2
2K / 120p / 422 / 10 bits	5.4 Gbps	HD-SDI	1.33 Gbps	~ 4
4K / 60p / 422 / 10 bits	10.8 Gbps	3G-SDI	2.65 Gbps	~ 4
2K / 60p / 422 / 10 bits	2.7 Gbps	1G Ethernet (SMPTE2022 1/2)	0.85 Gbps	~ 3
2K / 60p / 444 / 12 bits	4.8 Gbps	1G Ethernet (SMPTE 2022 6)	0.85 Gbps	~ 6
4K / 60p / 422 / 10 bits	10.8 Gbps	10G Ethernet (SMPTE2022 1/2)	8.5 Gbps	~ 1.3
3x [4K / 60p / 422 / 10 bits]	32.4 Gbps	10G Ethernet (SMPTE2022 6)	7.96 Gbps	~ 4
4K / 60p / 444 / 12 bits	19 Gbps	10G Ethernet (SMPTE2022 1/2)	8.5 Gbps	~ 2.2
2x [4K / 60p / 444 / 12 bits]	37.9 Gbps	10G Ethernet (SMPTE2022 6)	7.96 Gbps	~ 5
8K / 120p / 422 / 10 bits	85 Gbps	25G Ethernet	21,25 Gbps	~ 4

As shown in the table, the main applications targeted by these use cases are broadcast, digital cinema, and industrial vision applications.

3.2 Real-time video storage

Embedded devices such as cameras use internal storage to store large streams of images. These devices offer limited access rates (i.e. approx 500 MBytes/s (4Gbit/s) for SSD drives, approx 50-90 30MBytes/s (400-720 Mbit/s for SD cards). Lightweight compression would allow real-time storage of video streams with throughputs higher than these access rates.

[1] Gbps = Giga bit per second

Table 2 – Real-time video storage use cases

Video stream	Video throughput	Physical storage	Access rate	Comp. ratio
UHD / 60p / 422 / 10 bits	10 Gbps	SSD Drive	~ 4 Gbps	2.5
HD / 30p / 422 / 10 bits	1.2 Gbps	SD card	~ 0.5 Gbps	2.4

3.3 Video memory buffer

Buffer compression reduces the system form factor's weight, decreases the number of interconnect wires and extends the battery life for battery powered systems

- Upscaler/downscaler
- buffer for high refresh rate displays (120~600 Hz, Triple Flash)
- storage and replay buffer for high speed camera
- key frame buffer for AVC/HEVC 4K decoder

3.4 Omnidirectional video capture system

Omnidirectional video capture systems are assembled from a multitude of cameras mounted on a platform. Each camera covers a certain field of view which tends to overlap with that of its adjacent cameras in order to facilitate image stitching.

The proposed use case addresses the challenge of concurrently transferring and storing the image streams from each camera to a front-end processing system. In order to reduce the required bandwidth and therefore allow multiple cameras to send their data over a shared physical link, a lightweight, real-time compression between 2:1 and 6:1 of the image data at the camera is desired. Furthermore, this compression should be visually lossless. Applying such compression will furthermore reduce both the required storage size and throughput demands of the storage sub-system on the front-end processing system.

3.5 Head mounted display for Virtual or Augmented Reality (VR/AR)

Omnidirectional VR and AR content is highly suitable for viewing through head mounted displays ("HMD"). HMDs are either tethered (i.e. connected through a cable) or wireless in which case the display is battery powered. Furthermore, with omnidirectional content, the HMD will only show that portion of the media stream which is within the viewer's field of vision. Given the computational (and power) constraints of such a display, it can not be expected to receive the full image stream and then locally perform the required filtering onto the viewer's field of vision – this needs to be done upstream and based on spatial data received from the HMD.

From the viewer's perspective, the quality of experience is crucially tied to the latency with which the system reacts to changes in his spatial gaze. An immersive experience requires very high resolution video - well beyond HD. These requirements lead to the need for adaptive strategies which allow to transmit, switch between and decode multiple high resolution image streams (each covering a certain spatial region) while decoding the video streams with imperceptible latency.

4 Target markets

There are several target markets, among which:

- Broadcast applications and live production
- Live-production
- Digital Cinema applications
- Industrial vision
- Professional audio visual systems
- Consumer TV
- Mobile video applications
- Camera array based recordings
- Ultra high frame rate cameras
- Medical Imaging
- Video Surveillance and security
- Automotive Infotainment
- Camera manufacturers
- Set-top boxes
- Low-cost visual sensors in Internet of Things (IoT)
- HMD displays

5 Requirements

This Section presents the requirements that should be met by the proposals so as to be suited for the above described use cases. Requirements are split between "core coding requirements" and "optional features". The latter are not strictly required for a proposal to be accepted and evaluated. However, if some optional features are present, this will be taken into account in their assessment (described in Section 6).

5.1 Core coding requirements

5.1.1 Uncompressed image attributes

- *Image resolution*: from VGA (640x480) up to 8K (8192x4320)
- *Component subsampling*: 422, 444
- *Component type*: RGB, YCbCr
    - Input type of the encoder shall match output type of the decoder.
    - Internal color space conversion is permitted (as part of the proposal).
- *Component bit-depth*: 8 to 16 bits per component (bpc), integer (up to 12 bits in the first phase of the specification).
- *Frame rate*: from 24 fps to 120 fps, progressive content.
- *Content*: natural, synthetic, screen content
- *Supporting different color spaces*, including Rec. BT 709 [1], Rec. BT2020 [2], P3D65 [3], LogC.

5.1.2 Compressed bitstream requirements

- Visually lossless picture quality. Visually lossless means that the difference between the original image or image sequence and the same image or image sequence after compression and decompression is not detectable by a human observer under normal viewing conditions.
- Compression ratio ranging from 2:1 to 6:1. Objective and subjective evaluation procedures in Section 6.3 define specific compressed bitrates that will be used for technology comparison.
- Support of a variable bitrate (VBR) & constant bitrate (CBR) mode. CBR mode means that the target compressed file size is met exactly including necessary header, control and metadata bits. Variable bit rate in this context means that the actual compressed file size is less or equal than the target compressed file size.
- Ability to define a strict maximum compressed size per frame
- Guaranteed avoidance of target rate exceedance
- Self-contained compressed frame: a compressed frame shall contain all information required to completely recover the corresponding uncompressed frame.
- Robustness to multiple encoding-decoding cycles, equal to or above 7 cycles.
  - Live-acquisition requires several encoding/decoding cycles, with different kinds of intermediate processing operations (overlay, crop, editing, pan&scan).
  - Proposals will be compared on their ability to prevent quality losses along successive encoding-decoding cycles, as described in Section 6.3.1.2.

5.1.3 Design requirements

- Low-latency: a maximum algorithmic latency of 32 video lines for a combined encoder-decoder suite is required that is connected with an ideal transmission channel whose transmission speed corresponds to the defined bits per pixel. This algorithmic latency includes the possibly necessary smoothing buffer of the encoder and decoder in order to cope with the limited channel transmission speed. Algorithmic latency means the latency caused by data dependencies of the coding algorithm, excluding implementation delays. In other words, an infinitely fast processing is assumed.
- Low complexity in hardware and software: the algorithm shall be defined in such a way to allow for low complexity implementations on multiple platforms. As an indication, to process 4k 4:4:4 8bit 60p in real-time with a compression ratio compliant with the above requirement, neither encoder nor decoder should require more than 50% of an FPGA similar to Xilinx Artix 7 [5] or 25% of an FPGA similar to Altera Cyclon 5 [6]. The target of an optimized software implementation able to real-time process 4k 4:4:4 8bit 60p should be an i7 processor, or equivalent. The complexity analysis is further defined in Section 6.3.4.
- Support of different kinds of end-to-end parallelization, for CPU, SIMD, GPU, FPGA and ASIC. Hence, there shall be no serial bottleneck in the encoding and decoding process.
- Implementation scalability: the resources required by the encoder and the decoder shall scale depending on required throughput.

- No external memory for hardware implementations (FPGA / ASIC).
- Multiple platform interoperability (FPGA / ASIC / GPU / CPU): for frequencies ranging from 100MHz to 3GHz, circuits of different type shall produce the same codestream and be interoperable, so as to enable massive adoption.
- Configurability:
  - Image size, frame rate, bit-depth (bpc), component type, subsampling
  - CBR or VBR mode
    - Targeted compressed bitrate (bpp) for CBR
    - Maximum compressed bitrate (bpp) for VBR
  - Option has to be given to disable optional features (see Section 5.2 hereunder), as disabling features might lead to smaller hardware or software footprint.

5.2 Optional features

- Support for 4224/4444 Alpha.
- Support for Raw-Bayer.
- Robustness to post-processing operations (such as subsequent editing operations, color transform or gamma conversion that shall not induce visual artefacts).
- Handling of different transfer functions: the proposed algorithm should optimize its performance by taking into account the transfer function being used by the content to be processed.
- Avoidance of SDI forbidden values
  - According to SMPTE 292 and 425, (10 bit) video data values 000h – 003h and 3FCh – 3FFh are excluded and reserved for sync words (EAV,SAV and ancillary data start). These markers play a specific role for the descrambling and synchronization of the SDI data.
  - The proposed algorithm is welcome to define more efficient SDI mapping operations.
- Robustness to error: independently from error protection mechanisms available at transport level, the proposed algorithm should minimize the impact of random bit flips (in the case of transport over SDI links) and packet losses (in the case of transport over IP).

6 Proposals evaluation

6.1 Test material

Test material is made of both test sequences (Section 6.1.1) and still images (Section 6.1.2). All test material has been converted to the required input format and is made available to proponents on an FTP server for the purpose of this standardisation project only.

Proponents shall contact Antonin Descampe (see Section 10) to receive the login information required to access the sequences, together with copyright information related to this test material.

Note that some of the sequences and still images listed below might not be part of the final evaluation test material. This final set will be communicated to proponents having registered by the end of June 2016 (see timeline in Section 2). This includes the addition of test material, depending on the concrete needs.

6.1.1 Test sequences

6.1.1.1 Test sequences properties and file format

Table 3 gives an overview of all properties covered by the selected test sequences. As indicated in this table, no HDR content has currently been selected (in particular, bitdepth goes up to 12 bpc). Specific HDR core experiments might be defined later on in the standardisation process.

Table 3: Overview of the input test sequences properties

Content type	Natural content Computer generated content Screen content
Resolution	from 1920x1080 to 4096x2160
Frame rate	from 24 to 60
Component bit-depth	from 8 to 12
Component type	RGB YCbCr
Transfer function	Rec. BT 709 [1] P3D65 [3]
Chroma subsampling	444 422

For RGB content, sequences are made available in PPM (binary format)[2].

For YCbCr content, sequences are made available in a simple planar YUV format: one file per frame, no header, components concatenated one after the other, zero-padding to 16 bits on the most significant side if bit-depth is greater than 8, little-endian.

Any subsequent conversion made from these input sequences will be considered as part of the proposed compression scheme (an RGB to YCbCr conversion for instance). Decoded sequences format shall match the input format.

[2] http://netpbm.sourceforge.net/doc/ppm.html

For convenience, Annex E introduces a freely available conversion and measurement tool, difftest_ng, developed by a WG1 expert. In particular, specifiers for the above described YUV input format are given.

Sequences filenames are constructed with the following syntax:

Origin_Name_Resolution_Fps_Bpc_TransferFunction_ChromaFormat_xxx.yyy

with

- Origin: organization having provided the original sequence
- Name: sequence name
- Resolution: picture size (e.g. 1920x1080p)
- Fps: frame rate in frames per second
- Bpc: number of bits per component sample
- Transfer Function: transfer function used for the sequence
- Chroma format: e.g. 4:2:2, or 4:4:4
- xxx: frame number
- yyy: *ppm* for RGB content, or *yuv* for YCbCr content

6.1.1.2 Test sequences list

Table 4 lists all sequences considered for the JPEG XS evaluation process.

Table 4: JPEG XS test sequences

#	Sequence filename
S01_a	VQEG_CrowdRun_3840x2160p_50_10b_bt709_422_xxx.yuv
S01_b	VQEG_CrowdRun_3840x2160p_50_8b_bt709_444_xxx.ppm
S02_a	VQEG_ParkJoy_3840x2160p_50_10b_bt709_422_xxx.yuv
S02_b	VQEG_ParkJoy_3840x2160p_50_8b_bt709_444_xxx.ppm
S03_a	ARRI_AlexaHelicopterView_3840x2160p_24_12b_P3_444_xxx.ppm
S03_b	ARRI_AlexaHelicopterView_3840x2160p_24_12b_P3_422_xxx.yuv
S03_c	ARRI_AlexaHelicopterView_3840x2160p_24_12b_logC_444_xxx.ppm
S04_a	ARRI_AlexaDrums_3840x2160p_24_12b_P3_444_xxx.ppm
S04_b	ARRI_AlexaDrums_3840x2160p_24_12b_P3_422_xxx.yuv
S04_c	ARRI_AlexaDrums_3840x2160p_24_12b_logC_444_xxx.ppm
S05_a	ARRI_PublicUniversity_2880x1620p_24_8b_bt709_444_xxx.ppm
S06_a	ARRI_Lake2_2880x1620p_24_8b_bt709_444_xxx.ppm
S05_b	ARRI_PublicUniversity_2880x1620p_24_10b_bt709_422_xxx.yuv
S06_b	ARRI_Lake2_2880x1620p_24_10b_bt709_422_xxx.yuv
S07	BLENDER_Sintel1_4096x1744p_24_8b_sRGB_444_xxx.ppm

S08	BLENDER_Sintel1_4096x1744p_24_10b_sRGB_444_xxx.ppm
S09	BLENDER_TearsOfSteel_4096x1714p_24_12b_sRGB_444_xxx.ppm
S10_a	HDM_CarouselFireworks1_1920x1080p_25_10b_bt709_444_xxx.ppm
S10_b	HDM_CarouselFireworks1_1920x1080p_25_10b_bt709_422_xxx.yuv
S11_a	HDM_Showgirl2_1920x1080p_25_10b_bt709_444_xxx.ppm
S11_b	HDM_Showgirl2_1920x1080p_25_10b_bt709_422_xxx.yuv
S12_a	EBU_PendulusWide_3840x2160p_25_10b_bt709_444_xxx.ppm
S12_b	EBU_PendulusWide_3840x2160p_25_10b_bt709_422_xxx.yuv
S13_a	EBU_RainFruits_3840x2160p_25_10b_bt709_444_xxx.ppm
S13_b	EBU_RainFruits_3840x2160p_25_10b_bt709_422_xxx.yuv
S14	APPLE_BasketBallScreen_2560x1440p_60_8b_sRGB_444_xxx.ppm
S15	HUAWEI_ScMap_1280x720p_60_8b_sRGB_444_xxx.ppm
S16	RICHTER_ScreenContent_3840x2160_15_8b_sRGB_444_xxx.ppm

6.1.2 Test still images

The input test images considered for the evaluation tests are listed in Table 5.

Table 5: JPEG XS test still images

#	Image filename
I01	Zoneplate-gbf_1920x1080_10b.ppm
I02	Zoneplate-rbf_1920x1080_10b.ppm
I03	Zoneplate-rgf_1920x1080_10b.ppm
I04	FemaleStripedHorseFly_1920x1080_8b.ppm
I05	Tools_1524x1200_8b.ppm
I06	Peacock_1920x1080_8b.ppm
I07	HintergrundMusik_1920x1080_8b.ppm
I08	LeavesIso200_3008x2000_8b.ppm

6.2 Anchors

Proposals will be compared against several anchors. Each anchor has been selected because it can fulfil to a certain extent the requirements described above. Even if some requirements are not met (low complexity requirement for instance), the comparison is still meaningful to see how each proposal performs when compared to well-known solutions. Anchors have been configured to be as close as possible to JPEG XS requirements.

Following anchors have been selected:

- JPEG (ISO/IEC 10918-1 | ITU-T Rec. T.81)
- JPEG 2000 (ISO/IEC 15444-1 | ITU-T Rec. T.800)
- HEVC (ISO 23008-2:2015 | ITU-T Rec. H.265 (V3))
- VC-2 (SMPTE ST 2042-1)

Information on available software and configuration to be used for each of these anchors is given in Annex A.

6.3 Evaluation procedures

Objective and subjective quality evaluation of the proposals will each be done by at least two independent members of the Qualinet Network-of-Excellence, following procedures described hereunder in Sections 6.3.1 and 6.3.2, and based on the encoded-decoded test material provided by each proponent. Submitted binaries will be used for verification purpose. All practical details, including the final set of the test material that will be used, the name of the labs that will perform the evaluations, the procedure to submit binaries and sequences, will be communicated to registered proponents by the end of June.

For objective quality testing, evaluation tools described in Annex E are made freely available to let proponents perform their own assessments.

Beside objective and subjective quality evaluation, the proposals will also be evaluated by analysing the error robustness (Section 6.3.3) and the complexity (Section 6.3.4) of the proposed algorithm. For this, proponents are invited to provide all relevant information allowing WG1 to accurately estimate if a given proposal fulfils the requirements and how it performs compared to anchors and other proposals.

In the evaluation procedures described hereunder, definitions in Annex B are used.

6.3.1 Objective quality testing

Objective quality testing shall be done by computing PSNR between compressed and original image sequences, at different target bitrates.

6.3.1.1 Single encoding & decoding experiment

Proposals will be assessed on (a subset of) test sequences listed in Table 4, for the following target bitrates: 3, 4, 5, 6, 8, 10, 12, 14.

Depending on the computing time required by this objective assessment, test sequences might be subsampled in the time direction.

6.3.1.2 Multiple encoding & decoding experiment

To assess the required robustness to multiple encoding-decoding cycles, the following experiment will be done: proposals will be assessed on (a subset of) test sequences listed in Table 4 (possibly subsampled in the time direction), on a succession of 10 encoding-decoding cycles. Each encoding cycle will be done at a target bitrate of 6 bpp. PSNR will be computed between original and compressed sequence after each encoding-decoding cycle and PSNR degradation will be observed.

6.3.2 Subjective quality testing

6.3.2.1 Sequences

The subjective quality evaluation of test sequences will use the methodologies described in *ISO/IEC 29170-2* [4], adapted for image sequences. For this evaluation, a subset of approximately 5 test sequences will be selected from Table 4 based on various criteria including color gradiance, contrast edges, high frequency content, face tones with different ethnicity, CGI / natural / screen content, and objective quality assessment of the anchors. Input format for this subset of sequences will be RGB444 10 bits for CGI, RGB444 8 bits for screen content, and YCbCr422 10 bits for other sequences. All sequences will be evaluated in a single color space (either Rec BT 709, or P3D65). Registered proponents will be informed about the performed selection such that the appropriate files can be submitted.

The present Call for Proposal only defines SDR subjective assessment, up to 10 bpc. HDR subjective assessment might be specified later on.

The evaluation procedure can be described in the following way:

- The evaluation procedure is a forced-choice experiment: the observer is asked to choose between two options A and B, one of the option being the original sequence, the other one being the encoded-decoded sequence.
- The sequence is displayed on a single 4k 10-bit display: the left half of the display corresponds to a selected crop of option A and the right half of the display corresponds to the same crop of option B.
- The viewing distance shall be 60 pixels/degree
- Sequences to be used as compressed sequences will be the ones after 7 encoding-decoding cycles
- Target bit-rates [bpp] (corresponding compression ratios are 3, 4, 5 and 6 respectively):
  - 444 8 bit sequences: 8, 6, 4.8, 4 [bpp]
  - 444 10 bit sequences: 10, 7.5, 6, 5 [bpp]
  - 422 10 bit sequences: 6.7, 5, 4, 3.3 [bpp]
- The goal of this experiment is to identify the visual threshold where transparency is lost. Consequently, evaluation of bit-rates being lower than this threshold might be skipped.

For this experiment, proposals can promote modes optimizing compression for improving visual quality.

A Chroma Key test might be used later on to differentiate proposals.

6.3.2.2 Still images

The subjective quality evaluation of still images will be done on images listed in Table 5 and will use the methodologies described in *ISO/IEC 29170-2* [4].

In addition to what is described in the standard, the following procedures will be applied

- The image is displayed on a HD 10-bit display using 444 component subsampling only.
- Images to be used as compressed images will be the ones after 7 encoding-decoding cycles
- Target bit-rates [bpp] (corresponding compression ratios are 3, 4, 5 and 6 respectively):
    - 444 8 bit images: 8, 6, 4.8, 4 [bpp]
    - 444 10 bit images: 10, 7.5, 6, 5 [bpp]

6.3.3 Error robustness analysis

Proponents are invited to submit an evaluation of the error robustness of their algorithm. All relevant information related to error robustness is welcome, including (without being limited to):

- Ratio of sensitive data and induced damage if sensitive data is impacted by bit errors
- Possibility of error concealment mechanisms

As part of this evaluation, proponents can also submit the results of the bit-flip error robustness experiment available in the evaluation tools described in Annex E. This experiment assesses the robustness of a given software against random bit-flip errors[3]. If error concealment modes are available, note that objective and subjective evaluation described above will be done without these modes.

6.3.4 Complexity and latency analysis

Proponents are invited to submit an evaluation of the complexity and the latency of their algorithm. Such evaluation shall include:

- A detailed block diagram of the proposed algorithm showing the flow of the data, including possible feedback loops or additional dependencies.
- For each processing unit from the block diagram, a statement on
    - The granularity of data simultaneously processed by the unit (i.e. sample, bit, byte, blocks, bit-planes, etc)
    - The presence of any feedback loop inside this unit.

[3] As results of this experiment are also dependent on the quality of the implementation being tested, it is not formally part of the evaluation protocol. However, good results would prove both the implementation and the algorithm to be robust to such bit-flip errors.

    - The number and kind of operation executed in this unit (i.e. multiplication, division, addition, logic operators, non-linear operations like square root or logarithmic operations, any floating point operation). The provided number of operations will be used to determine more precise complexity values, while for missing values, worst case complexity will be estimated. Accompanying text should explain how the values have been derived.
    - The amount of any large (> 8 kbytes) memory required by the unit (i.e. presence of memory blocks exceeding 8 kbytes). Accompanying text should explain how the values have been derived.
- The end-to-end latency, in number of lines, including a text for justification.
- An explanation on how the proposal scales to higher throughputs. In particular, it shall be shown that higher (resp. lower) throughput induce an approximately linear increase (resp. decrease) of the amount of required resources to sustain target throughput.
- An explanation of the achievable parallelism of the algorithm for both the encoder and the decoder.
- All information available at the time of submission showing the performances of the algorithm once implemented in software and hardware, as described in Annex C.

As an example, the JPEG 2000 anchor has been evaluated according to these guidelines. The block diagram and the list of processing units is available in Annex D.

6.3.5 Overall Evaluation

The ranking of the received proposals will be done for each individual compression ratio, first on the quality, second on the complexity, and third on optional features.

7 Requirements on submissions

The process to evaluate proposals will be done following the timeline defined in Section 2. The successive deliverables are further defined hereunder.

In addition to documents and binaries to be submitted, proponents are reminded that they are expected to contribute to the standardisation process, as described in Section 9.

7.1 Proposal overview [due 23/05/2016]

The proposal overview shall include:

- A high-level description of the proposal including block diagrams of encoder and decoder.
- Arguments on why the proposal is meeting the requirements.

Convenient formats include Word document, PDF format, PowerPoint presentations or example pictures.

7.2 Binary encoder and decoder executables [due 23/09/2016]

Proponents need to submit separate encoder and decoder executable programs (statically linked Linux executables with all required libraries and system dependencies), configurable via command line or configuration file. Binaries should preferably be optimized software meeting the performances requirements described above in order to speed up the evaluation process.

Proponents can choose to use executable compression or similar tools to prevent reverse engineering or disassembly of the submitted executable files.

Proponents shall provide the command-line parameters intended to be used for the evaluation procedures described above. Scripts for generating the test content shall also be provided for every test case. A detailed list of test cases will be communicated by the end of June to proponents having registered.

Configurability shall be as described in Section 5.1.3.

7.3 Encoded-decoded material [due 23/09/2016]

Proponents need to submit the final test material processed by their coding system:

- Single encoding-decoding results
- Multi-pass results for all ten iterations (1^{st} to 10^{th})
- Encoded-decoded material for subjective evaluation

The exact list of processed material to provide will be communicated by the end of June to proponents having registered, together with naming conventions and the address where to ship the hard-drive containing the material.

7.4 Algorithm and design description [due 23/09/2016]

Each proposal shall include a presentation that provides a detailed description of the proposed algorithm. This presentation shall be in Word document and PDF format.

The presentation shall clearly explain how the proposed algorithm meets the above described requirements: quality, complexity, latency, and additional features (low-latency, architecture scalability, rate control mechanism, etc).

In particular, complexity and latency shall be described following the guidelines detailed in Section 6.3.4. Information used to perform this analysis shall be provided. In particular, description of operations and detailed algorithm steps shall be given. If available, FPGA and ASIC synthesis and place-and-route reports should be provided.

Proponents are encouraged to list all features, benefits and performance advantages of their architecture.

7.5 Technical documentation [due 28/10/2016]

If (part of the) the proposal has been selected to be part of the upcoming standard, a technical description of the selected technology shall be provided. This includes:

- Description of operations, as described in algorithm and design description
- Coded bitstream syntax
- Coding process (encoding and decoding) methodology, as described in algorithm and design description
- Proposed standard structure

The description shall include all necessary processing (including performance optimizations) that are used to create the bitstream in a bit-exact manner.

7.6 Verification model source code [due 28/10/2016]

Proponents agree to release source code to serve as a Verification Model (VM), written in a high-level language, such as C or C++, if they are selected in the evaluation process. Source code shall be documented and understandable. It does not need to correspond to the accelerated evaluation software mentioned in Section 7.2. Hence, assembly language or GPU code is not permitted. All libraries used by the source code shall be either public or provided in source code form with ISO/IEC and ITU-T compliant terms. Make files or project files need to support compilation on both Windows and Linux systems. The VM decoder should correctly decode any codestream generated by the submitted encoder executable binary. Moreover, the VM decoder and the submitted decoder executable binary shall both generate the exact same output.

8 Intellectual Property Rights

Proponents are advised that this call is being made subject to the common patent policy of ITU-T/ITU-R/ISO/IEC and other established policies of these standardization organizations. The persons named below as contacts can assist potential submitters in identifying the relevant policy information.

9 Contribution to Standardization

Proponents are informed that based on the submitted proposals, a unified standard will be created. If they submit a proposal and (part of) the proposed technology is accepted for inclusion in the standard, they will hence have to attend subsequent WG1 meetings and contribute to the creation of the different standard documents. Within this process, evolution and changes are possible.

Concerning the proposals evaluation process, attendance to the 72nd meeting in Geneva (May 30th – June 3rd, 2016) is recommended and attendance to the 73rd meeting in Chengdu (Oct 17th – Oct 21st) is mandatory.

10 Contacts

Gaël Rouvroy - intoPIX SA
Rue Emile Francqui, 9
1435 Mont-Saint-Guibert, Belgium
g.rouvroy@intopix.com
+32 10 23 84 70

Antonin Descampe - intoPIX SA
Rue Emile Francqui, 9
1435 Mont-Saint-Guibert, Belgium
a.descampe@intopix.com
+32 477 77 38 50

Joachim Keinert
Fraunhofer IIS
Department Moving Picture Technologies
Am Wolfsmantel 33
91058 Erlangen
Germany
Joachim.keinert@iis.fraunhofer.de
+49 9131 776 5152

ANNEX A

Anchors configurations

A.1. JPEG (ISO/IEC 10918-1 | ITU-T Rec. T.81)

- Configuration
    - JPEG does not specify a rate allocation mechanism allowing to target a specific bitrate. Hence, an external rate-control loop is required to achieve targeted bitrate. This additional loop is part of the evaluation tools described in Annex E.
    - Irreversible RGB to YCbCr conversion has to be disabled when dealing with YCbCr content
    - Subsampled content (i.e. 422) is first upsampled to 444 before being encoded. The decoded content is then downsampled to 422 before PSNR to be computed.
- Available software: JPEG XT Demo Codec v1.51 (GPL v3)
    - Available at http://jpeg.org/jpegxt/software.html
    - License: GPLv3
    - Only supports 8 bpc and 12 bpc content
    - Command-line examples (to use within rate-control loop)
        - RGB

```
jpeg -q [QUALITY_PARAMETER] [INPUTFILE] [OUTPUTFILE]
```

        - YCbCr

```
jpeg -c -q [QUALITY_PARAMETER] [INPUTFILE] [OUTPUTFILE]
```

A.2. JPEG 2000 (ISO/IEC 15444-1 | ITU-T Rec. T.800)

- Configuration
    - Slice vertical size[4]: 8
    - Tile vertical size: 2
    - Slice/tile horizontal size: unbounded
    - Tile-part length marker segments in the main header (TLM)
    - Tile-parts per component
    - Progression order: CPRL

[4] Division of the frame in several horizontal slices, each one being an independent JPEG 2000 code-stream (see Ultra Low Latency mode used in broadcast applications).

    - Precincts: {256,256} for all levels but the last one where it is {128,128}
    - Code-block size: 1024x4 (width x height)
    - DWT: 9-7
    - # of decomposition levels: 6
    - YCC conversion if RGB input, no MCT otherwise
    - No visual weighting
    - Code-block style:
        - Selective arithmetic coding bypass enabled
        - Reset context probabilities on coding pass boundaries
        - Termination on each coding pass
        - Vertically causal context
- Available softwares:
    - OpenJPEG
        - Available at http://www.openjpeg.org
        - License: BSD
    - Kakadu
        - Available at http://www.kakadusoftware.com
        - License: demo binaries freely available for non-commercial use
        - Command-line for RGB 8bpc

```
kdu_compress -i [INPUTFILE] -o [OUTPUTFILE] Stiles={2,8192}
ORGgen_tlm=3 ORGtparts=C Corder={CPRL} Clevels=6
Mprecision=8,8,8 Sprecision=8,8,8 Ssigned=no,no,no
Cprecincts={256,256},{256,256},{256,256},{256,256},{256,256},{25
6,256},{128,128} Cblk={4,1024} Creversible=no Cycc=yes
Cmodes={RESET|RESTART|CAUSAL|BYPASS} -precise Qstep=.0001 -rate
16 -no_weights -num_threads 0 -fprec 8,8,8
```

        - Command-line for YCbCr 422 10bpc

```
kdu_compress -i[INPUTFILE_Y],[INPUTFILE_Cb],[INPUTFILE_Cr] -o
[OUTPUTFILE] Mprecision=10,10,10 Sprecision=10,10,10
Ssigned=no,no,no
Sdims=[RESOLUTION_INPUTFILE_Y],[RESOLUTION_INPUTFILE_Cb],[RESOLU
TION_INPUTFILE_Cr] Scomponents=3 Stiles={2,8192} ORGgen_tlm=3
ORGtparts=C Corder={CPRL} Clevels=6
Cprecincts={256,256},{256,256},{256,256},{256,256},{256,256},{25
6,256},{128,128} Cblk={4,1024} Creversible=no Cycc=no
Cmodes={RESET|RESTART|CAUSAL|BYPASS} -precise Qstep=.0001 -rate
16 -no_weights -num_threads 0
```

A.3. HEVC (ISO 23008-2:2015 | ITU-T Rec. H.265 (V3))

- Configuration:
    - An external rate-control loop is required to achieve targeted bitrate.
    - encoder_intra_main_rext.cfg to allow for 444 and 422 content
    - Max CTU size: 16
    - One slice per CTU row and tiles disabled

- Available software: HEVC Test Model (HM)
    - Available at https://hevc.hhi.fraunhofer.de/
    - License: BSD
    - Configuration files to be used are available in the repository of the evaluation tools described in Annex E, at the following link:

      https://github.com/uclouvain/opentestbench/tree/master/codecs/HEVC

A.4. VC-2 (SMPTE ST 2042-1)

- Configuration:
    - Low-Delay syntax
    - HQCBR profile
    - Wavelet kernel: LeGall
    - Wavelet depth: 3
    - Slice surface: 2x1
        - which corresponds, with a wavelet depth of 3, to 16x8 coefficients per slice.
- Available software: VC-2 reference encoder-decoder
    - Available at https://github.com/bbc/vc2-reference
    - License: Apache-2
    - Command-line for RGB 8bpc

```
EncodeHQ-CBR -v --bytes 1 --framerate 8 --width [IMAGE_WIDTH]--height
[IMAGE_HEIGHT] --format RGB --bitDepth 8 -p -k LeGall --waveletDepth 3
--vSlice 1 --hSlice 2 --compressedBytes [COMPRESSED_FRAME_SIZE]
[INPUTFILE] [OUTPUTFILE]
```

    - Command-line for YCbCr 422 10bpc

```
EncodeHQ-CBR -v --framerate 8 --width [IMAGE_WIDTH]--height
[IMAGE_HEIGHT] --format 4:2:2 --lumaDepth 10 --chromaDepth 10 -p -k
LeGall --waveletDepth 3 --vSlice 1 --hSlice 2 --compressedBytes
[COMPRESSED_FRAME_SIZE] [INPUTFILE] [OUTPUTFILE]
```

ANNEX B

Definitions

B.1. Mean Square Error (MSE)

For a given image component *C* of size *W*x*H,* the Mean Square Error (MSE) between this original image component *C* and the encoded and decoded image *C'* is given by :

$$MSE_{C'} = \frac{\sum_{i=0}^{W-1}\sum_{j=0}^{H-1}\left(C(i,j) - C'(i,j)\right)^2}{W \cdot H}$$

For a given image *I* made of 3 components C_1, C_2 and C_3, without chroma subsampling (444), the Mean Square Error (MSE) between this original image *I* and the encoded and decoded image *I'* is given by :

$$MSE_{I'} = \frac{\sum_{k=0}^{3} MSE_{C'_k}}{3}$$

For a given image *I* made of 3 components *Y*, C_b and C_r, with 422 chroma subsampling, the Mean Square Error (MSE) between this original image *I* and the encoded and decoded image *I'* is given by :

$$MSE_{I'} = \frac{MSE_{Y'}}{2} + \frac{MSE_{C'_b}}{4} + \frac{MSE_{C'_r}}{4}$$

For a given sequence *S* made of *N* images, the average $MSE_{S'}$ over the whole sequence is given by :

$$MSE_{S'} = \frac{\sum_{L=0}^{N-1} MSE_{L'}}{N}$$

B.2. Peak Signal-to-Noise Ratio (PSNR)

For a given image *I* with a maximum component sample value of *B*, the Peak Signal-to-Noise Ratio (PSNR) between this original image *I* and the encoded and decoded image *I'* is given by :

$$PSNR_{I'} = 10 \cdot \log\left(\frac{(2^B - 1)^2}{MSE_{I'}}\right)$$

For a given image sequence *S* with a maximum component sample value of *B*, the Peak Signal-to-Noise Ratio (PSNR) between this original sequence *S* and the encoded and decoded sequence *S'* is given by :

$$PSNR_{S'} = 10 \cdot \log\left(\frac{(2^B - 1)^2}{MSE_{S'}}\right)$$

B.3. Target bitrate and bits per pixel (bpp)

The target bit-rate for objective and subjective quality experiments is given in “bits per pixel”.

For a given compressed image, the bit-rate in bits per pixel (bpp) is given by the length of the compressed image (in bits) divided by the number of pixels in the original image. If the original image is a chroma-subsampled image, the number of pixels corresponds to the number of samples in the Y component.

ANNEX C

Software and hardware complexity indicators

The software and hardware complexity indicators are the amount of resources required by a proposal, for each configuration specified below, to sustain real-time encoding and decoding of 4k, 4:4:4 60p, 8 bits/sample.

C.1. Software

Configuration

- i7 processor, or equivalent

Indicators such as

- Configuration of processor(s) used, RAM, and GPU
- Number of required threads

C.2. FPGA

Configuration

- Artix-7 / Cyclon-5

Indicators such as

- Frequency
- Speedgrade
- FPGA model
- # luts
- # registers
- # BRAM
- # DSP

C.3. ASIC

Indicators such as

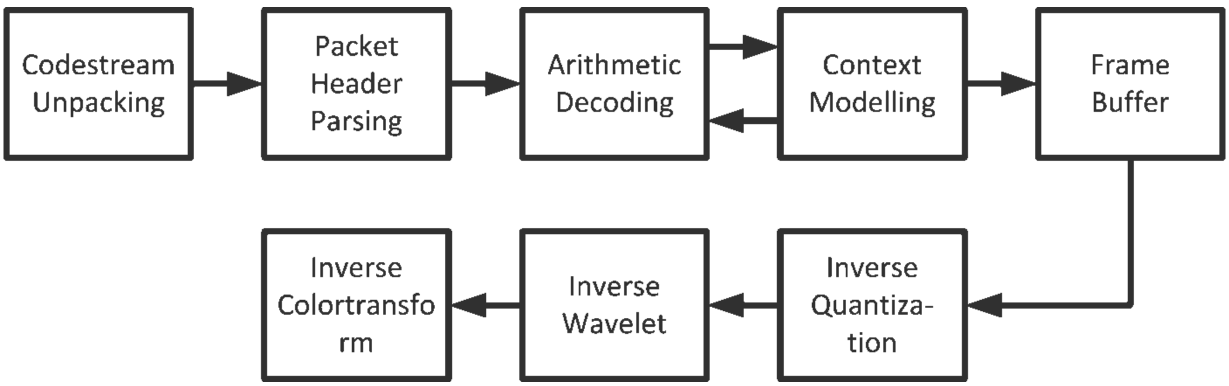
- Technology node
- Frequency
- # gates
- Amount of RAM

ANNEX D

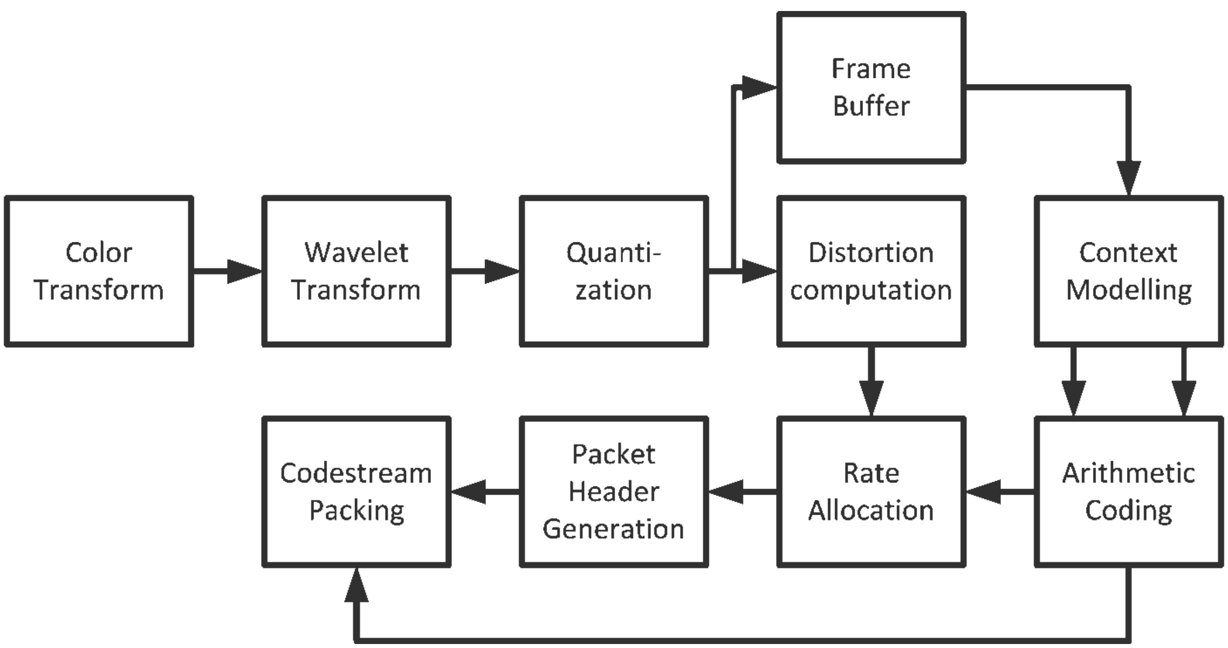
Example: complexity evaluation of JPEG 2000 anchor

D.1. Block diagram

D.1.1. Decoder



D.1.2. Encoder



D.1.3. Processing units

For each processing unit of JPEG 2000, Table 6 is listing the information required to be able to assess the complexity of the algorithm. For all values given below, an explanatory text has to be provided.

Table 6: complexity indicators for each processing unit of JPEG 2000

Processing units	Processing granularity	Internal Feedback Loop	Operations	Memory (>8Kbytes)
Color Transform	pixel	no	6 additions, 9 multipliers	no
Wavelet Transform (9/7)	sample	no	> 8 additions, > 8 multipliers	5 lines per decomposition level
Quantization	sample	no	1 multiplier, 1 rounding	no
Distortion computation	bit	no	1 addition, small LUT	no
Frame buffer	bitplane	no	logic	full image memory
Context Modeling	bit	yes	some small LUTs, many logic	4 states per coefficent per codeblock
Arithmetic Coding	bit	yes	2 additions, small LUTs, logic, shift	no
Rate Allocation	bitplane	yes	3 dividers, sorting of codeblock bitplanes, addition, …	number of codeblock * 3 passes * bitdepth
Packet Header generation	codeblock	yes	logic, LUT, shift	no
Codestream generation	byte	yes	shift, logic	full compressed image memory

ANNEX E

Evaluation tools

To ease the objective assessment of the different proposals, Université de Louvain (UCL, Belgium) has developed a set of tools (in Bash) to automatically perform the objective assessment of a given set of codecs.

Currently available features include

- Automatic installation script: the tool automatically downloads and configures all anchors codecs (11/03/16: HEVC missing but work in progress).
- Easy addition of new (proprietary) codecs by placing binaries in an appropriate folder.
- Supported input format: ppm for RGB content and yuv planar for YCbCr content.
- Easy addition of new test sequences
- Objective metric: PSNR
- Experiments implemented:
    - Single encoding and decoding experiment
    - Multiple encoding and decoding experiment
    - Bit-flip error robustness experiment
- Automatic generation of graphs based on results (matlab required)
- Currently only works under Linux platform

All conversion operations and metric measurements are done with difftest_ng, a C-utility developed by Thomas Richter (Stuttgart University) and freely available on github:

https://github.com/thorfdbg/difftest_ng

For convenience, Table 7 indicates the specifiers to be used with difftest_ng when dealing with the planar YUV format used for YCbCr test sequences.

Table 7: difftest_ng specifier for YUV format

YUV 422 12bit	[4-],[12-=0]:[4-]/2x1,[12-=1]/2x1:[4-]/2x1,[12-=2]/2x1
YUV 422 10bit	[6-],[10-=0]:[6-]/2x1,[10-=1]/2x1:[6-]/2x1,[10-=2]/2x1

Table 8 gives two commandline examples for such conversions.

Table 8: difftest_ng commandline examples

RGB 444 => YUV 422 12b	difftest_ng --toycbcr --csub 2 1 --convert out.raw@:[4-],[12-=0]:[4-]/2x1,[12-=1]/2x1:[4-]/2x1,[12-=2]/2x1 input.ppm -
YUV 422 => RGB 444 12b (HD)	difftest_ng --cup 2 1 --fromycbcr --convert out.ppm in.raw@1920x1080x3:[4-],[12-=0]:[4-]/2x1,[12-=1]/2x1:[4-]/2x1,[12-=2]/2x1 -

Difftest_ng is automatically downloaded and installed by Opentestbench.
The Opentestbench tool is freely available under GPLv3 license on github:
https://github.com/uclouvain/opentestbench

Support on these evaluation tools can be obtained from Alexandre Willème (alexander.willeme@uclouvain.be).

ANNEX F

References

[1] ITU-R BT.709-5 (2002), “Parameter values for the HDTV standards for production and international programme exchange”. https://www.itu.int/dms_pubrec/itu-r/rec/bt/R-REC-BT.709-5-200204-I!!PDF-E.pdf.

[2] ITU-R BT.2020 (2012), “Parameter Values for ultra-high definition television systems for production and internal programme exchange”. http://www.itu.int/dms_pubrec/itu-r/rec/bt/R-REC-BT.2020-0-201208-I!!PDF-E.pdf

[3] SMPTE ST 428-1:2006 “D-Cinema Distribution Master – Image Characteristics”. P3 with D65 white point.

[4] ISO/IEC 29170-2, “Information technology -- Advanced image coding and evaluation methodologies -- Part 2: Evaluation Procedure for nearly lossless coding”.

[5] http://www.xilinx.com/products/silicon-devices/fpga/artix-7.html#productTable

[6] https://www.altera.com/products/fpga/cyclone-series/cyclone-v/features.html

[7] Stefano Andriani, Harald Brendel, Tamara Seybold and Joseph Goldstone, “Beyond the Kodak Image Set: a new reference set of color image”, ICIP2013