ANNEX A

Anchors configurations

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

A.1.1. 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 Opentestbench evaluation tools.
- 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.

A.1.2. 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)

A.2.1. Configuration

The JPEG 2000 anchor is used in three different configurations. The first one follows the Broadcast contribution single tile profile, as described in ISO/IEC 15444-1 :2004 - AMD3, and does not meet the low latency and CBR requirements. Therefore it is only used for the objective evaluations.

The second configuration is detailed in Table 8 and uses a tile-based allocation method allowing to fulfill the low-latency and CBR requirements.

The third one is also detailed in Table 8 and is only used to validate the evaluation setup and procedures as it induces obvious impairment to the content at the targeted bitrates.

Table 8: list of parameters for the ULL JPEG 2000 ULL configuration and for the controlled-anchor JPEG 2000 configuration

	JPEG 2000 ULL	Controlled-anchor JPEG 2000
Slice vertical size (1)	No slices	8
Slice horizontal size	No slices	Unbounded
Tile vertical size	8	2
Tile horizontal size	Unbounded	Unbounded
TLM markers	No	Yes
Tile-parts per component	No	Yes
Progression order	CPRL	CPRL
Precincts size	No precincts	{256,256} for all levels but the last one where it is {128,128}
Code-block size	1024 x 4 (w x h)	1024 x 4 (w x h)
Code-block style	BYPASS RESET RESTART	BYPASS RESET RESTART VCAUSAL
DWT	9x7	9x7
# of decomposition levels	6	6
YCC conversion	If RGB input	If RGB input
Visual weighting	No for objective evaluation Yes for subjective evaluation	No
Rate allocation	Tile-based	Slice-based
# Layers	1	1

Note:

(1) For the controlled-anchor JPEG 2000 configuration, each frame is divided in several horizontal slices, each one being an independent JPEG 2000 code-stream.

A.2.2. Available software

Accusoft JPEG 2000 software

- Accusoft is making its JPEG 2000 software available to WG1 experts for the sole purpose of JPEG XS evaluation, as indicated in input document wg1m72069.
- The following command lines shall be used for the JPEG 2000 ULL configuration and Broadcast configuration:
    - RGB 444 input:
        - ULL encoding, without visual weighting:

```
transcoder -qe 9 -b <bitrate in bpp> -cn 1 -al -tw 4096 -th 8 -cp 4 -cw
10 -ch 2 -cb 7 -cd 6 input.ppm out.j2k
```

        - Broadcast encoding, without visual weighting:

```
transcoder -b ${bpp} -qe 9 -cp 4 -cn 1 -cL 1 -cf 1 -ec 4 -et on -cW 7 -
cH 7 input.ppm out.j2k
```

- To enable visual weighting, simply add the “`-w 1000`” switch to the command lines above.

- Decode with:

```
transcoder out.j2k out.ppm
```

- YUV 422 input
  - First convert from 422YUV to 444YUV.ppm
    - For 10bpc:

```
difftest_ng --cocup 2 1 --convert temp.ppm
in.raw@${width}x${height}x3:[6-],[10-=0]:[6-]/2x1,[10-=1]/2x1:[6-
]/2x1,[10-=2]/2x1 -
```

- For 12bpc:

```
difftest_ng --cocup 2 1 --convert temp.ppm
in.raw@${width}x${height}x3:[4-],[12-=0]:[4-]/2x1,[12-=1]/2x1:[4-
]/2x1,[12-=2]/2x1 -
```

- Then encode 444YUV with 2,1 subsampling:
  - For ULL anchor:

```
transcoder -my on -Cx 2 -Cy 1 -Cq -qe 9 -b <bitrate> -cn 1 -al -tw 4096
-th 8 -cp 4 -cw 10 -ch 2 -cb 7 -cd 6 temp.ppm out.j2k
```

- For Broadcast anchor:

```
transcoder -my on -Cx 2 -Cy 1 -Cq -b ${bpp} -qe 9 -cp 4 -cn 1 -cL 1 -cf
1 -ec 4 -et on -cW 7 -cH 7 temp.ppm out.j2k
```

- To enable visual weighting, simply add the “`-w 1000`” switch to the command lines above.

- Decode with:

```
transcoder -U out.j2k out.pgx
```

- Use difftest_ng to convert pgx to raw and get the YUV422 file back.
  - For 10bpc:

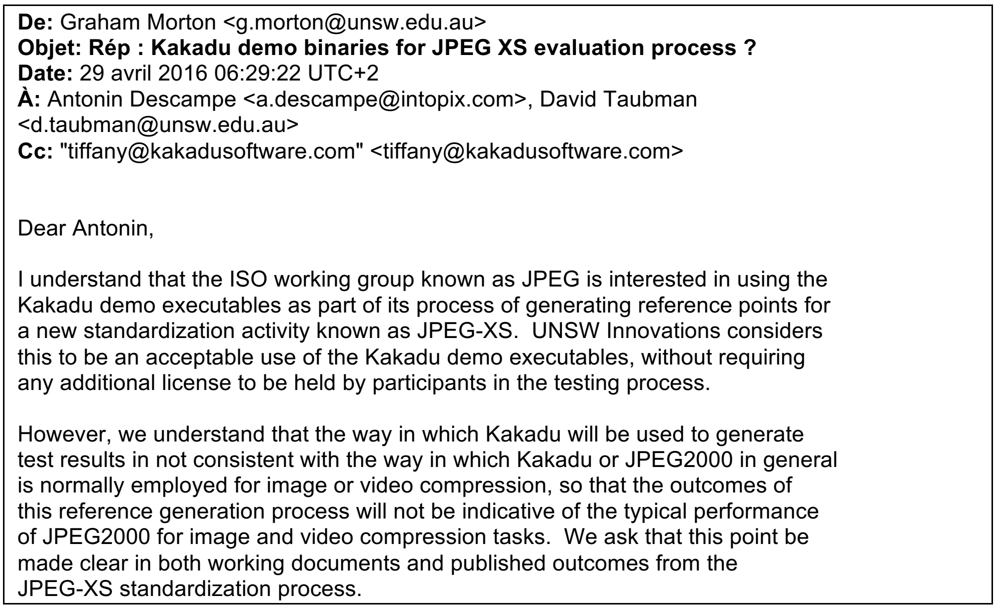
```
difftest_ng --convert out.raw@${width}x${height}x3:[6-],[10-=0]:[6-
]/2x1,[10-=1]/2x1:[6-]/2x1,[10-=2]/2x1 out.pgx -
```

- For 12bpc:

```
difftest_ng --convert out.raw@${width}x${height}x3:[4-],[12-=0]:[4-
]/2x1,[12-=1]/2x1:[4-]/2x1,[12-=2]/2x1 out.pgx -
```

Kakadu

- Available at http://www.kakadusoftware.com
- Only used for the controlled-anchor JPEG 2000 configuration above.
- License: demo binaries freely available for non-commercial use, UNSW Innovations explicitly gave WG1 the right to use these demo binaries for the JPEG XS project



De: Graham Morton <g.morton@unsw.edu.au>
Objet: Rép : Kakadu demo binaries for JPEG XS evaluation process ?
Date: 29 avril 2016 06:29:22 UTC+2
À: Antonin Descampe <a.descampe@intopix.com>, David Taubman <d.taubman@unsw.edu.au>
Cc: "tiffany@kakadusoftware.com" <tiffany@kakadusoftware.com>

Dear Antonin,

I understand that the ISO working group known as JPEG is interested in using the Kakadu demo executables as part of its process of generating reference points for a new standardization activity known as JPEG-XS. UNSW Innovations considers this to be an acceptable use of the Kakadu demo executables, without requiring any additional license to be held by participants in the testing process.

However, we understand that the way in which Kakadu will be used to generate test results in not consistent with the way in which Kakadu or JPEG2000 in general is normally employed for image or video compression, so that the outcomes of this reference generation process will not be indicative of the typical performance of JPEG2000 for image and video compression tasks. We ask that this point be made clear in both working documents and published outcomes from the JPEG-XS standardization process.

- 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},{256,256},{128,12
8} 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],[RESOLUTION_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},{256,256},{128,12
8} 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))

A.3.1. 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

A.3.2. 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 XX, at the following link:

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

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

A.4.1. 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.
- RGB => YCbCr transform in case of RGB content:
  - For 8-bit content, upshift from 8 to 10 bpc.
  - RGB => YCbCr transform done externally.
  - At the decoding side, perform the inverse transform followed by downshifting to 8 bpc.

A.4.2. Available software

VC-2 reference encoder-decoder

- Available at https://github.com/bbc/vc2-reference
- License: Apache-2

- This implementation does not include any component transformation in case of RGB content. For this anchor to provide relevant results when dealing with RGB content, the following external processing shall be used in case of RGB content
    - If bitdepth is 8bpc, change it from 8 to 10 and upshift all values accordingly.
    - Apply the irreversible RGB→YCbCr transformation
    - Encode and decode with VC-2
    - Apply the inverse transform YCbCr→RGB
    - Change back the bitdepth from 10 to 8 if bitdepth was initially 8bpc.
- 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]
```