INTERNATIONAL STANDARD

ISO/IEC 18477-2

First edition
2016-07-15

Information technology — Scalable compression and coding of continuous-tone still images —

Part 2: Coding of high dynamic range images

Technologies de l'information — Compression échelonnable et codage d'images plates en ton continu —

Partie 2: Codage d'images à gamme dynamique élevée

Reference number
ISO/IEC 18477-2:2016(E)

ISO copyright office
Ch. de Blandonnet 8 • CP 401
CH-1214 Vernier, Geneva, Switzerland
Tel. +41 22 749 01 11
Fax +41 22 749 09 47
copyright@iso.org
www.iso.org

Contents

Foreword

ISO (the International Organization for Standardization) and IEC (the International Electrotechnical Commission) form the specialized system for worldwide standardization. National bodies that are members of ISO or IEC participate in the development of International Standards through technical committees established by the respective organization to deal with particular fields of technical activity. ISO and IEC technical committees collaborate in fields of mutual interest. Other international organizations, governmental and non-governmental, in liaison with ISO and IEC, also take part in the work. In the field of information technology, ISO and IEC have established a joint technical committee, ISO/IEC JTC 1.

The procedures used to develop this document and those intended for its further maintenance are described in the ISO/IEC Directives, Part 1. In particular the different approval criteria needed for the different types of document should be noted. This document was drafted in accordance with the editorial rules of the ISO/IEC Directives, Part 2 (see www.iso.org/directives).

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. ISO and IEC shall not be held responsible for identifying any or all such patent rights. Details of any patent rights identified during the development of the document will be in the Introduction and/or on the ISO list of patent declarations received (see www.iso.org/patents).

Any trade name used in this document is information given for the convenience of users and does not constitute an endorsement.

For an explanation on the meaning of ISO specific terms and expressions related to conformity assessment, as well as information about ISO's adherence to the World Trade Organization (WTO) principles in the Technical Barriers to Trade (TBT) see the following URL: www.iso.org/iso/foreword.html.

The committee responsible for this document is ISO/IEC JTC 1, *Information technology*, SC 29, *Coding of audio, picture, multimedia and hypermedia information*.

A list of all parts in the ISO 18477 series, published under the general title *Information technology -- Scalable compression and coding of continuous-tone still images*, can be found on the ISO website.

Introduction

This document is an extension of ISO/IEC 18477-1, a compression system for continuous tone digital still images which is backward compatible with Rec. ITU-T T.81 | ISO/IEC 10918-1. That is, legacy applications conforming to Rec. ITU-T T.81 | ISO/IEC 10918-1 will be able to reconstruct streams generated by an encoder conforming to this document, but will possibly not be able to reconstruct such streams in full dynamic range, full quality or other features defined in this document.

The aim of this document is to provide a migration path for legacy applications to support coding of high-dynamic range images. Existing tools depending on the existing standards will continue to work, but will only be able to reconstruct a lossy and/or a low-dynamic range version of the image contained in the codestream. This document specifies a codestream, referred to as JPEG XT, which is designed primarily for storage and interchange of continuous-tone photographic content.

This document specifies a coded codestream format for storage of continuous-tone high and low dynamic range photographic content. JPEG XT Part 2 is a scalable image coding system supporting multiple component images in floating point. It is by itself an extension of the coding tools defined in ISO/IEC 18477-1; the codestream is composed in such a way that legacy applications conforming to Rec. ITU-T T.81 | ISO/IEC 10918-1 are able to reconstruct a lower quality, low dynamic range, eight bits per sample version of the image.

Today, the most widely used digital photography format, a minimal implementation of JPEG (specified in Rec. ITU-T T.81 | ISO/IEC 10918-1), uses a bit depth of 8; each of the three channels that together compose an image pixel is represented by 8 bits, providing 256 representable values per channel. For more demanding applications, it is not uncommon to use a bit depth of 16 or higher, providing greater than 65 536 representable values to describe each channel within a pixel, resulting on over 2.8 × 10^{14} representable colour values. In some less common scenarios, even greater bit depths are used.

The most common photo and image formats use an 8-bit or 16-bit unsigned integer value to represent some function of the intensity of each colour channel. While it might be theoretically possible to agree on one method for assigning specific numerical values to real world colours, doing so is not practical. Since any specific device has its own limited range for colour reproduction, the device's range may be a small portion of the agreed-upon universal colour range. As a result, such an approach is an extremely inefficient use of the available numerical values, especially when using only 8 bits (or 256 unique values) per channel. To represent pixel values as efficiently as possible, devices use a numeric encoding optimized for their own range of possible colours or gamut.

JPEG XT is primarily designed to provide coded data containing high dynamic range and wide colour gamut content while simultaneously providing 8 bits per pixel low dynamic range images using tools defined in ISO/IEC 18477-1. The goal is to provide a backward compatible coding specification that allows legacy applications and existing toolchains to continue to operate on codestreams conforming to this this document.

JPEG XT has been designed to be backward compatible to legacy applications while at the same time having a small coding complexity; JPEG XT uses, whenever possible, functional blocks of Rec. ITU-T T.81 | ISO/IEC 10918-1 to extend the functionality of the legacy JPEG Coding System. It is optimized for storage and transmission of high dynamic range and wide colour gamut 32 bit float images while also enabling low-complexity encoder and decoder implementations.

Information technology — Scalable compression and coding of continuous-tone still images —

Part 2:
Coding of high dynamic range images

1 Scope

This document specifies a coding format, referred to as JPEG XT, which is designed primarily for continuous-tone photographic content.

2 Normative references

The following documents are referred to in the text in such a way that some or all of their content constitutes requirements of this document. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

ISO/IEC 10918-1, *Information technology — Digital compression and coding of continuous-tone still images: Requirements and guidelines — Part 1*

ISO/IEC 18477-1, *Information technology — Scalable Compression and Coding of Continuous-Tone Still Images, Core Coding System Specification*

IEC 61966-2-1, *sRGB Colour management — Default RGB colour space — sRGB*

3 Terms and definitions

For the purposes of this document, the following terms and definitions apply.

ISO and IEC maintain terminological databases for use in standardization at the following addresses:

- IEC Electropedia: available at http://www.electropedia.org/
- ISO Online browsing platform: available at http://www.iso.org/obp

3.1
ASCII encoding
character encoding scheme defined by ANSI X3.4-1986

3.2
codestream
partially encoded or decoded sequence of bits comprising an entropy-coded segment

3.3
byte
group of 8 bits

3.4
coder
embodiment of a coding process

3.5
coding
encoding or decoding

3.6
(coding) process
general term for referring to an encoding process, a decoding process, or both

3.7
compression
reduction in the number of bits used to represent source image data

3.8
component
two-dimensional array of samples having the same designation in the output or display device

Note 1 to entry: An image typically consists of several components, e.g. red, green and blue.

3.9
continuous-tone image
image whose components have more than one bit per sample

3.10
discrete cosine transform
DCT
sum of cosine transforms at different frequencies

3.11
decoder
embodiment of a decoding process

3.12
decoding process
process which takes as its input compressed image data and outputs a continuous-tone image

3.13
downsampling
procedure by which the spatial resolution of a component is reduced

3.14
encoder
embodiment of an encoding process

3.15
encoding process
process which takes as its input a continuous-tone image and outputs compressed image data

3.16
grayscale image
continuous-tone image that has only one component

3.17
high dynamic range
image or image data comprised of more than eight bits per sample

3.18
Joint Photographic Experts Group
JPEG
informal name of the working group which created this part of ISO/IEC 18477

Note 1 to entry: The term “joint” comes from the ITU-T and ISO/IEC collaboration.

3.19
legacy decoder
embodiment of a decoding process conforming to Rec. ITU-T T.81 | ISO/IEC 10918-1, confined to the lossy discrete cosine transformation (DCT) process and the baseline, sequential or progressive modes, decoding at most four components to eight bits per component

3.20
lossless
descriptive term for encoding and decoding processes and procedures in which the output of the decoding procedure(s) is identical to the input of the encoding procedure(s)

3.21
lossless coding
mode of operation which refers to any one of the coding processes defined in this part of ISO/IEC 18477 in which all of the procedures are lossless

3.22
lossy
descriptive term for encoding and decoding processes which are not lossless

3.23
low-dynamic range
image or image data comprised of data with no more than 8 bits per sample

3.24
marker
two-byte code in which the first byte is hexadecimal FF and the second byte is a value between 1 and hexadecimal FE

3.25
marker segment
marker together with its associated set of parameters

3.26
minimum coded unit
MCU
smallest group of data units that is coded

3.27
pixel
collection of sample values in the spatial image domain having all the same sample coordinates

Note 1 to entry: A pixel may consist of three samples describing its red, green and blue value.

3.28
precision
number of bits allocated to a particular sample or discrete cosine transformation (DCT) coefficient

3.29
procedure
set of steps which accomplishes one of the tasks which comprise an encoding or decoding process

3.30
residual codestream
codestream that contains an encoded (according to Rec. ITU-T T.81 | ISO/IEC 10918-1) residual image

3.31
residual data
data that contains luminance ratio and red, green, and blue (RGB) differential data

3.32
residual image
pseudo image that contains encoded luminance ratio as luminance and encoded chrominance data that is computed from red, green, and blue (RGB) differential data using Multiple Component Decorrelation Transformation defined in ISO/IEC 18477-1

3.33
red, green, and blue
RGB
additive colour model

3.34
luminance ratio
array of per pixel ratio of HDR image luminance and LDR image luminance

3.35
quantization value
integer value used in the quantization procedure

3.36
quantize
act of performing the quantization procedure for a value

3.37
upsampling
procedure by which the spatial resolution of a component is increased

4 Abbreviated terms and symbols

4.1 Abbreviated terms

ASCII	American Standard Code for Information Interchange
HDR	High Dynamic Range
LDR	Low Dynamic Range

4.2 Symbols

Nc	number of components in an image

5 Conventions

5.1 Conformance language

The keyword “reserved” indicates a provision that is not specified at this time, shall not be used, and may be specified in the future. The keyword “forbidden” indicates “reserved” and in addition indicates that the provision will never be specified in the future.

5.2 Operators

NOTE Many of the operators used in this document are similar to those used in the C programming language.

5.2.1 Arithmetic operators

+ addition

– subtraction (as a binary operator) or negation (as a unary prefix operator)

* multiplication

/ division without truncation or rounding

smod x smod a is the unique value y between $-\lceil (a-1)/2 \rceil$ and $\lceil (a-1)/2 \rceil$ for which y+Na = x with a suitable integer N

umod x mod a is the unique value y between 0 and a–1 for which y+Na = y with a suitable integer N

5.2.2 Logical operators

|| logical OR

&& logical AND

! logical NOT

∈ x ∈ {A, B} is defined as (x == A || x == B)

∉ x ∉ {A, B} is defined as (x != A && x != B)

5.2.3 Relational operators

> greater than

>= greater than or equal to

< less than

<= less than or equal to

== equal to

!= not equal to

5.2.4 Precedence order of operators

Operators are listed below in descending order of precedence. If several operators appear in the same line, they have equal precedence. When several operators of equal precedence appear at the same level in an expression, evaluation proceeds according to the associativity of the operator either from right to left or from left to right.

Operators	Type of operation	Associativity
(), [ ], .	expression	left to right
–	unary negation	
*, /	multiplication	left to right

umod, smod	modulo (remainder)	left to right
+,−	addition and subtraction	left to right
<, >, <=, >=	relational	left to right

5.2.5 Mathematical functions

$\lceil x \rceil$	Ceiling of x. Returns the smallest integer that is greater than or equal to x.
$\lfloor x \rfloor$	Floor of x. Returns the largest integer that is lesser than or equal to x.
$\lvert x \rvert$	Absolute value, is −x for x < 0, otherwise x.
clamp(x,min,max)	Clamps x to the range [min,max]: returns min if x < min, max if x > max or otherwise x.

6 General

6.1 Elements specified

The purpose of this clause is to give an informative overview of the elements specified in this document. Another purpose is to introduce many of the terms that are defined in Clause 3. These terms are printed in *italics* upon first usage in this clause.

There are three elements specified in this document:

a) An *encoder* is an embodiment of an *encoding process*. An encoder takes as input *digital source image data* and *encoder specifications*, and by means of a specified set of *procedures* generates as output *a codestream*.

b) A *decoder* is an embodiment of a *decoding process*. A decoder takes as input a codestream, and by means of a specified set of procedures generates as output *digital reconstructed image data*.

c) The *codestream* is a compressed image data representation which includes all necessary data to allow a (full or approximate) reconstruction of the sample values of a digital image. Additional data might be required that define the interpretation of the sample data, such as colour space or the spatial dimensions of the samples.

6.2 High level overview of this document

This document allows backward compatible to Rec. ITU-T T.81 | ISO/IEC 10918-1 coding of high dynamic range images. Rec. ITU-T T.81 | ISO/IEC 10918-1 compliant decoders will be able to parse codestreams conforming to this document correctly, albeit in less precision, with a limited dynamic range, and potential loss in sample bit precision.

This document includes multiple tools to reach the above functionality. A short overview on these coding tools is given in this clause.

The high-level syntax of a 18477-2 codestream is identical to that defined in ISO/IEC 18477-1, which is a subset of the syntax defined in Rec. ITU-T T.81 | ISO/IEC 10918-1. Marker definitions and the syntax of the markers defined in the above recommendation remain in force and unchanged. However, this document defines the APP$_{11}$ marker, reserved in the legacy Recommendation | International Standard for encoding additional syntax elements. Legacy decoders will skip and ignore such marker elements, and hence will only be able to decode the image encoded by the legacy syntax elements. This document codestream will be denoted the *legacy codestream*.

This document extends the legacy standard by introducing extended syntax defined in Clause 8. The extended syntax uses the APP$_{11}$ marker to hide additional data from legacy applications.

High dynamic range coding of floating point samples is supported through *residual coding*. As shown in Figure 1 this coding method represents the high-dynamic range image by two codestreams, the *legacy codestream* and the *residual codestream*. The former represents a represents a low-dynamic range representation of the original image that is visible to legacy applications, the latter represents a residual image that is merged with the low-dynamic range legacy image in the spatial domain to obtain the high-dynamic range reconstructed output. The merging process first generates a *precursor image* of the high-dynamic range image by transforming the legacy codestream to a linear colourspace. The precision of the precursor image is then extended to full precision by the *residual image* which is decoded from the *residual codestream* packaged in the *residual codestream segments* defined in Clause 8.

The *residual codestream* is the entropy-coded representation of the residual image. This document defines multiple coding methods for encoding the residual image: The DCT baseline Huffman, extended Huffman or progressive coding method defined in Rec. ITU-T T.81 | ISO/IEC 10918-1.

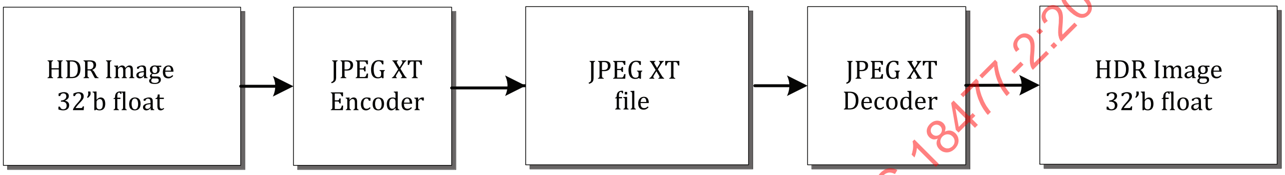

Figure 1 — Coding process

6.3 High level functional overview of decoding process

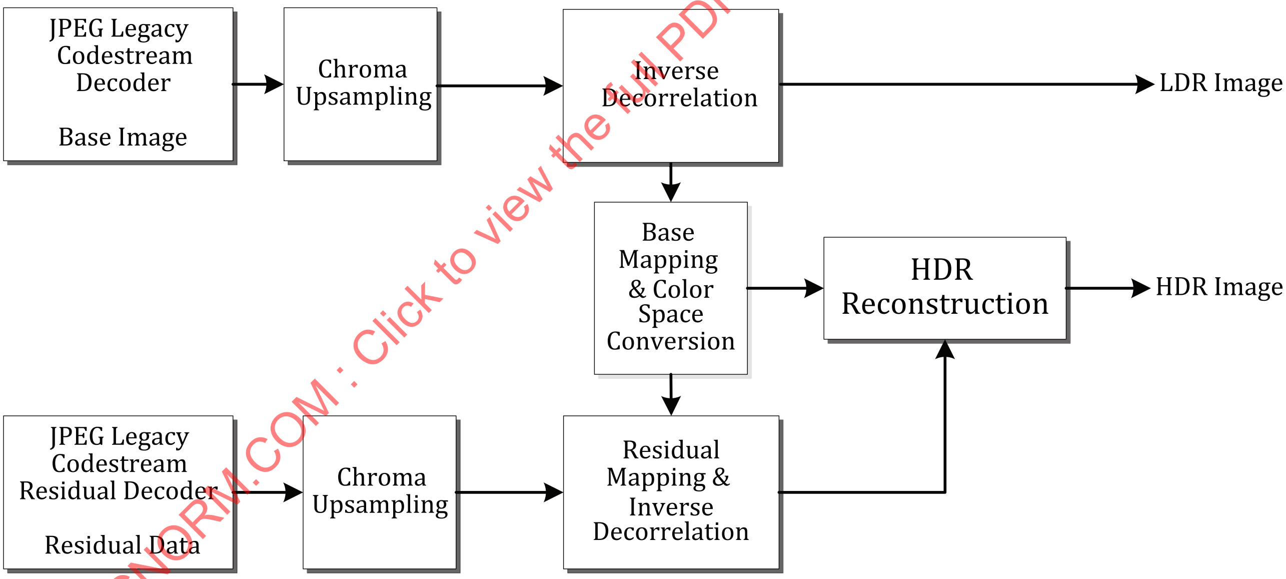

Figure 2 — Overview of the decoding process

Figure 2 depicts the decoding process. The process begins with the legacy decoder block which reconstructs the base image. This image is then optionally chroma upsampled followed by the Inverse Decorrelation block. The output of this transformation is a low-dynamic range image with 8 bits per sample in a RGB-type colour space. This upper path is the backward compatible part.

The low-dynamic range components are further mapped by the Base Mapping and Colour Space Conversion block to a floating point image which is called the *precursor image*. The precursor image is optionally converted to HDR colour space if the colour space of the precursor image is different and the luminance of the precursor image is calculated. The noise level may be used to avoid division by zero and to reduce the coding artefacts.

The residual decoder path uses the residual image stored in the residual codestream in the APP$_{11}$ markers. The residual image is decoded and optionally upsampled. The Residual Mapping and Inverse Decorrelation block maps the residual image to the residual data. The residual data consists of at least luminance ratio data and optional RGB differential data. This mapping uses the luminance computed by the Base Mapping and Colour Space Conversion block.

The HDR Reconstruction block uses the resulting residual data and the precursor image to calculate the reconstructed HDR Image.

6.4 Encoder requirements

An encoder is required to meet the compliance tests and to generate the codestream according to the syntax and to limit the coding parameters to those valid within this document.

6.5 Decoder requirements

A decoding process converts compressed image data to reconstructed image data. The process shall follow the decoding operation specified in this document and ISO/IEC 18477-1 for the residual and base image codestreams. The decoder shall parse the codestream syntax to extract the parameters, the residual data and the precursor image. The parameters shall be used to merge the residual data and precursor image into the reconstructed HDR Image.

In order to comply with this document, a decoder:

a) shall convert a codestream conforming to this document *without considering the residual codestream* into a low dynamic range image.

b) *may additionally*, convert a conforming codestream including the residual codestream according to the codestream syntax, into a high dynamic range continuous tone image.

c) shall implement *at least* all the functional blocks of the JPEG XT decoding process defined in this document.

7 Decoder definition

7.1 Decoder functionality overview

The decoder relies on a layered approach to extend capabilities of Rec. ITU-T T.81 | ISO/IEC 10918-1 compliant codestream. An encoder compliant to this document decomposes an HDR image into a base layer and an HDR residual layer. The base layer is a tone mapped image obtained from the original floating point HDR with either a local or global tonemapping or/and gamut mapping operator. The base layer codestream constitutes the backward compatible part and is accessible by all legacy decoders. The base layer codestream is accessible in exactly the same way a legacy JPEG image would be coded.

The residual codestream is stored in the Application layer of the legacy codestream and contains HDR quantized encoded luminance ratio and a CbCr difference component. The encoded luminance ratio and the encoded CbCr difference component are combined into a residual image. The residual image is encoded into residual codestream by a Rec. ITU-T T.81 | ISO/IEC 10918-1 compliant encoder. To extract a residual image, the JPEG XT decoder uses a Rec. ITU-T T.81 | ISO/IEC 10918-1 compliant or legacy decoder once extracted from the Application layer.

A codestream compliant with this document contains residual codestream stored in the APP$_{11}$ markers. The legacy decoders ignore the APP$_{11}$ segments and decode only the base image codestream. Thereby the codestream compliant with this document is backward compatible to Rec. ITU-T T.81 | ISO/IEC 10918-1. JPEG XT compliant decoders will combine the base image codestream and the residual codestream to recover the HDR image as described below.

Figure 3 illustrates the functionality of a Main Profile decoder.

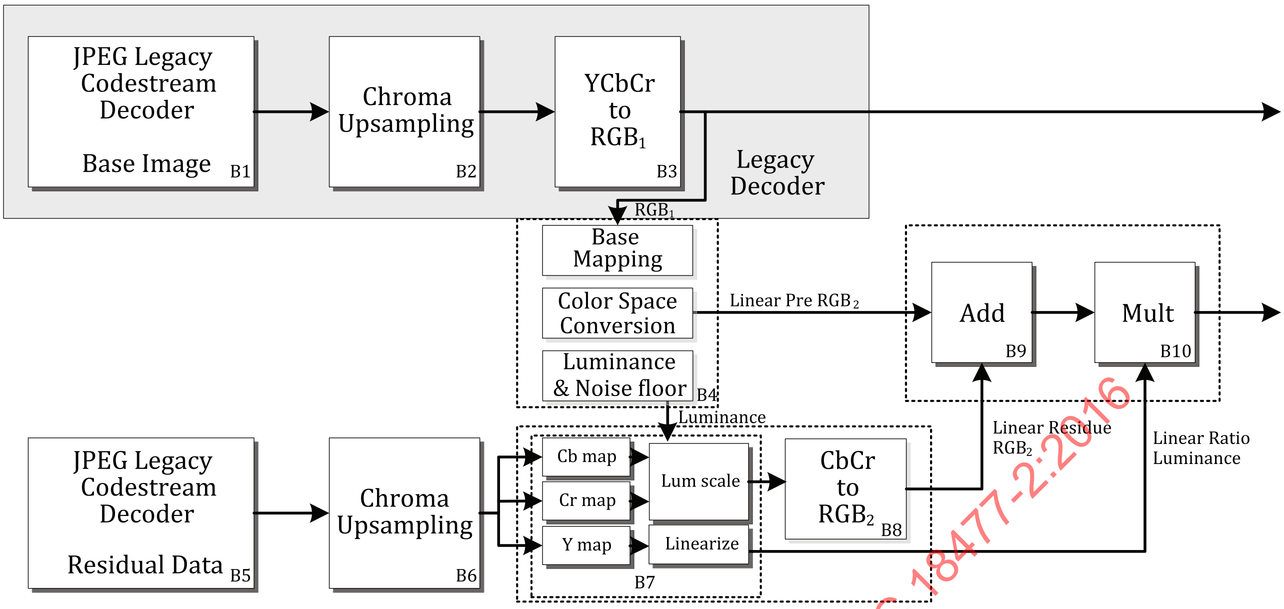

Figure 3 — JPEG XT decoder functionality

A compliant codestream shall use the codestream syntax defined in Clause 8.

As shown in Figure 3, both the base image and residual image of a compliant codestream shall be encoded using the syntax and semantics of ISO/IEC 10918-1, where both are constraint to the 8 bit baseline Huffman, extended Huffman or progressive coding process.

In Figure 3 the upper path which comprises blocks B1, B2, and B3 will be the standard flow of a legacy decoder and outputs a backward compatible LDR image in sRGB space (IEC 61966-2-1:1999) typically. This decoder uses the decorrelation, which is either a YCbCr to RGB conversion as defined by Multiple Component Decorrelation Transformation of ISO/IEC 18477-1 or an identity transform depending on the presence of the Component Decorrelation Control marker specified in ISO/IEC 18477-1.

This base image data are then mapped into linear HDR space and processed by the colour space conversion operation in B4. This block converts the LDR image into the colour space of the original HDR image, and it also maps the image to floating point value and is called Linear Pre RGB2, it can also be referred to as LP_RGB2. A noise floor value specified in the parameter codestream is added to the luminance component of the LP_RGB2 to avoid divide by 0 and to avoid amplifying any noise that could occur due to operations downstream from this block for small values. This noise floor offset luminance value will be referred to as lum below.

7.2 Legacy Inverse Decorrelation Block B3

Block B3 is the decorrelation transformation defined by Multiple Component Decorrelation Transformation of ISO/IEC 18477-1 that converts upsampled data from YCbCr space to RGB space if appropriate.

If the number of components of the legacy image Nc equals three, denote the reconstructed sample values from block B3 by inputR, inputG, and inputB, respectively. If the number of components of the legacy image is one, the reconstructed sample value is denoted by inputY below.

7.3 Base Mapping and Colour space conversion Block B4

The Block B4 is described by the following:

For base mapping in B4:

```
    // basemapping RGB$_1$ see Figure 3
    If (Nc = = 3) {
R1 = inputR;
G1 = inputG;
B1 = inputB,
    } else if (Nc = = 1) {
R1 = inputY;
G1 = inputY;
B1 = inputY;
    }
    If(header_param_bga present) {
basemappingLUT[] = bga;
    } else {
basemappingLUT[] = IEC 61966-2-1:1999_inverse_sRGB_curve;
    }
    dgR1 = basemappingLUT[R1];
    dgG1 = basemappingLUT[G1];
    dgB1 = basemappingLUT[B1];
```

The basemappingLUT is loaded with the data in bga parameter in the header specified in 8.1. if bga parameter is present. Otherwise, basemappingLUT shall be initialized by the inverse sRGB colour transform function specified in IEC 61966-2-1:1999.

For the colour space conversion in B4:

```
    // colour space conversion from RGB$_2$ to HDR (see Figure 3)
    if(header_param_csb present) {
csb[0:8] = header_param_csr[0:8];
    } else {
csb[0] = 1; csb[1] = 0; csb[2] = 0;
csb[3] = 0; csb[4] = 1; csb[5] = 0;
csb[6] = 0; csb[7] = 0; csb[8] = 1;)
    }
    LP_R2 = csb[0]*dgR1 + csb[1]*dgG1 + csb[2]*dgB1;
    LP_G2 = csb[3]*dgR1 + csb[4]*dgG1 + csb[5]*dgB1;

    LP_B2 = csb[6]*dgR1 + csb[7]*dgG1 + csb[8]*dgB1;
```

For luminance and noise floor in B4:

```
    // calculate luminance and add noise floor

    if (header_param_cst_present) {

inv_csr = inverse(csr);

lum = (inv_csr[0]*LP_R2 + inv_csr[1]*LP_G2 + inv_csr[2]*LP_B2) + header_param_nf ;

    } else {

lum = (0.299*LP_R2 + 0.587*LP_G2 + 0.114*LP_B2) + header_param_nf ;

    }
```

The basemappingLUT specified in 8.1 is a 256-entry table loaded by a default sRGB table which is a combination of a linear function and a power function of 2.4. If the base image is encoded in an alternate colour space, the look up table will be sent in the header. The basemappingLUT is typically the inverse gamma table but could also be any table transform that facilitates the decoding process. Combing the two reduces the amount of data that is stored in the residual codestream.

The Colour Space Conversion block defaults to identity transform when the base image and the HDR target colour space are the same. If they are different, this matrix is sent in the header as csb parameters. These matrix parameters represent the product of the base colour space primaries and the inverse of the HDR target colour space matrix. The target colour space for the HDR can be found in the header as the cst parameters.

7.4 Residual decode Blocks B5, B6

In Figure 3 the lower path starting from B5 begins with the residual codestream that represents the residual data of the high-dynamic range image, and is represented by the ISO/IEC 10918-1 codestream format. This codestream is embedded in the APP_{11} marker as a residual image segment described in Clause 8. Once the residual codestream is extracted from the header it is decoded by the Rec. ITU-T T.81 | ISO/IEC 10918-1 Decoder. The chroma upsampling step is performed by B6 to bring all components to full resolution, i.e. 4:4:4. Chroma upsampling is identical to the upsampling process defined in Annex A of ISO/IEC 18477-1.

The residual data are then separated by B7 into floating-point linear luminance ratio values and the linear residual colour difference values. If present the riq lookup table parameter defined in 8.1 maps residual image luminance (Ymap) to linear linear residual luminance. If the riq parameter is not present the residual image luminance (Ymap) is first scaled using ln0, ln1 parameters from the parameters segment with following application of exponent as described in the pseudo code below. The residual image chroma values are inverse quantized according to the minimum and maximum parameters stored in the parameter segment of the codestream as cb0,cb1 and cr0,cr1.

For the following, rY is the first, and rCb and rCr are the output of block B6, the chroma-upsampled image reconstructed from the residual codestream. If the residual codestream contains only one component, rCb and rCr shall be set to zero.

7.5 Residual Mapping and Inverse Decorrelation Blocks B7, B8

The following Residual mapping is described in the function of block B7:

```
    // rY,rCb,rCr are the inputs from the decoded residual data
    // dequant and Linearize Luminance Ratio
    If (header_paramater_riq present) {
Dqlut[] = header_parameter_riq;
LR_lum = dqlut[rY];
    } else {
val = rY * (header_paramater_ln1 − header_paramater_ln0) + header_paramater_ln0;
LR_lum = exp2(val);
    }
     // Dequantize and Linearize Chroma Residual
    val = rCb * (header_paramater_cb1 − header_paramater_cb0) + header_paramater_cb0;
    LR_Cb = val * lum;
    val = rCr * (header_paramater_cr1 − header_paramater_cr0) + header_paramater_cr0;
    LR_Cr = val * lum;
```

The Chroma values are then processed by B8, the YCbCr to RGB2 block and will convert the Linear Dequantized YCbCr to a linear residual RGB2 in the HDR colour space, alternatively referred to as LR_RGB2.

```
// YCbCr to RGB2 (see Figure 3)
    if(header_param_csr present) {
csr[0:8] = header_param_csr[0:8];
    } else {
csr[0] = 1; csr[1] = 0;      csr[2] =   1.402;
csr[3] = 1; csr[4] = −0.34414; csr[5] = −0.71414;
csr[6] = 1; csr[7] = 1.772;      csr[8] = 0;
    }
    LR_R2 =  csr[1]*LR_Cb + csr[2]*LR_Cr;
    LR_G2 =  csr[4]*LR_Cb + csr[5]*LR_Cr;
    LR_B2 = csr[7]*LR_Cb + csr[8]*LR_Cr;
```

7.6 HDR Reconstruction Blocks B9, B10

The final blocks B9 and B10 constructs an HDR image by first adding the Linear Pre RGB2 to the linear residue RGB2 in B9 and then multiplying the result by the linear luminance ratio in B10.

Blocks 9 and 10 have the following function:

```
hdrR2 = LP_R2 + LR_R2;

hdrG2 = LP_G2 + LR_G2;

hdrB2 = LP_B2 + LR_B2;

/* Mult */

hdrR = LR_lum * hdrR2;

hdrG = LR_lum * hdrG2;

hdrB = LR_lum * hdrB2;
```

If Nc equals three, the final floating-point output values shall be hdrR, hdrG, hdrB and represents the final HDR image. Otherwise, if Nc equals one, the final floating-point output value shall be hdrG.

8 Codestream syntax for Main Profile

8.1 Main Profile Header Structure

The codestream syntax for Main Profile defines parameters and codestream data required for HDR image reconstruction. The syntax of Main Profile uses the Rec. ITU-T T.81 | ISO/IEC 10918-1 and ISO/IEC 10918-6 APP$_{11}$ marker clause. The rest of the codestream remains as specified by Rec. ITU-T T.81 | ISO/IEC 10918-1 for the backward compatible base image section. The APP$_{11}$ marker is broken into a parameter data segment and a residual codestream segment. The parameter segment has two types of segments, an ASCII based parameter type and a binary parameter type.

Figure 4: Main Profile Header Structure identifies the data that is contained in the APP11 header segment.

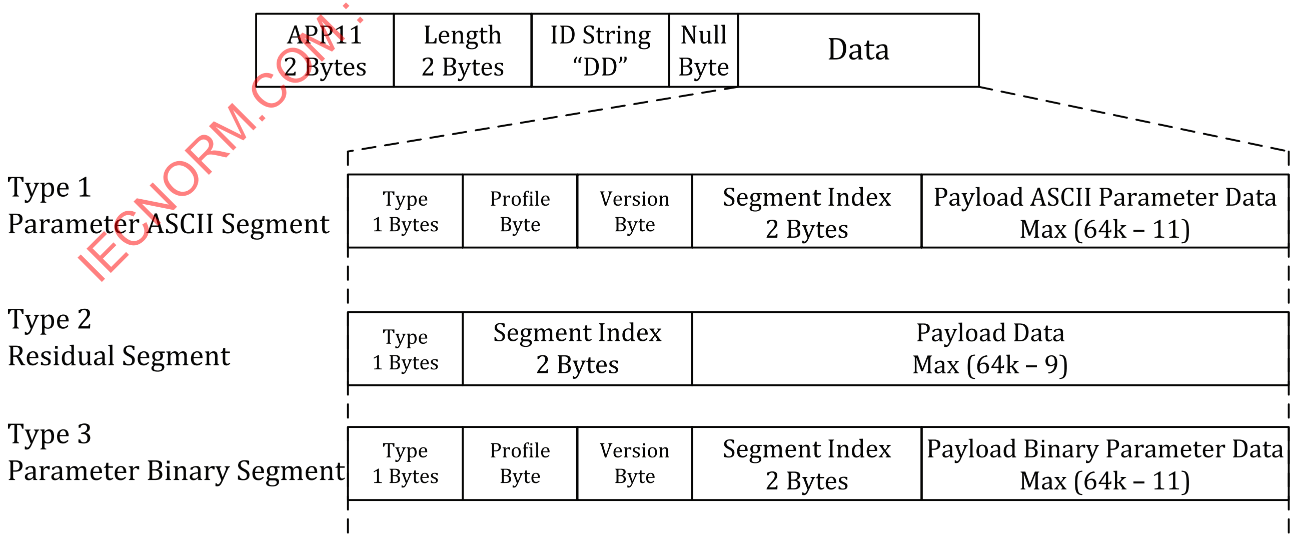

Figure 4 — Main Profile Header Structure

8.2 Parameter ASCII segment

The parameter ASCII segment is defined as Type 1. The parameter ASCII segment shall contain only one set of parameter data. The following list of parameters encoded as ASCII text as payload data.

Table 1 — Main Profile parameter segment

Parameter	Description	Mandatory/ Optional
< ln0 = %e >	log-r minimum	Mandatory
< ln1 = %e >	log-r maximum	Mandatory
< s2*n* = %e >	Multiplier that converts HDR values into luminance in nits	Optional
< cb0 = %e >	Cb minimum	Mandatory
< cb1 = %e >	Cb maximum	Mandatory
< cr0 = %e >	Cr minimum	Mandatory
< cr1 = %e >	Cr maximum	Mandatory
< nf = %e >	Noise floor normalized from 0 to 1.0	Mandatory
< riq %e %e ... %e >	256-point LUT to decode ratio image	Optional
< swb %e %e %e >	White balance multiplier in camera sensor RGB space This is additional metadata a decoder may take advantage of to improve the visual appearance of an image in the presence of clipping. Facilitate highlight rendering for clipped or almost clipped rendering. See also spr tag.	Optional
< bga %e %e ...%e >	256-point LUT to inverse map base image	Optional
< spr %e %e ... %e >	Colour space conversion 3x3 matrix that converts data from camera sensor colour space to target hdr colourspace. The tag is used in combination with the swb tag. This is additional metadata a decoder may take advantage of to improve the visual appearance of an image in the presence of clipping.	Optional
< csb %e %e ... %e >	Colour space conversion 3x3 matrix for base layer (9 coefficients)	Optional
< csr %e %e ... %e >	Colour space conversion 3x3 matrix for residual layer (9 coefficients)	Optional
< cst %e %e ... %e >	Colour space 3x3 matrix for the target image (9 coefficients) informational only	Optional
< tmt %s >	Tonemapper type and Parameters This is a string that identifies the tonemapper that was used to generate the LDR image. This Recommendation | International Standard does not enforce a specific encoding and it serves only for informational purposes.	Optional
< enc %s >	Encryption parameters as segment index, exchange, segment ID, key string; if multiple segments are encoded there can be repeated enc parameters	Optional
< sdsc %s >	Segment index, segment image height, width, xy start location if there are multiple segments embedded sdsc will be repeated	Optional
< ckb = %x >	Fletcher 16 Checksum of base layer codestream based on all bytes in base layer codestream. The computation of the checksum is specified in Annex A.	Mandatory

Each parameter is coded as an ASCII string and enclosed in angle brackets (< >). For single-value parameters, no extra space should be inserted between the paired angle brackets. %e denotes ASCII encoding of floating point value in mantissa / exponent format. The format consists of optional value sign, one mantissa digit, a decimal point, no more than 5 mantissa digits, a letter "e" or "E", an optional