INTERNATIONAL STANDARD

ISO/IEC 18477-8

Second edition
2020-05

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

Part 8: Lossless and near-lossless coding

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

Partie 8: Codage sans perte et quasi sans perte

Reference number
ISO/IEC 18477-8:2020(E)

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

Published in Switzerland

Contents

Page

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.

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) or the IEC list of patent declarations received (see http://patents.iec.ch).

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

For an explanation of the voluntary nature of standards, 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 www.iso.org/iso/foreword.html.

This document was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information technology*, Subcommittee SC 29, *Coding of audio, picture, multimedia and hypermedia information*.

This second edition cancels and replaces the first edition (ISO/IEC 18477-8:2016), which has been technically revised.

The main changes compared to the previous edition are as follows:

— Annex F.2 has been revised to adopt centred upsampling by default;

— minor editorial changes throughout.

A list of all parts in the ISO/IEC 18477 series can be found on the ISO website.

Any feedback or questions on this document should be directed to the user's national standards body. A complete listing of these bodies can be found at www.iso.org/members.html.

Introduction

This document specifies a coded codestream format for storage of continuous-tone high and low dynamic range photographic content. This is a scalable lossy to lossless image coding system supporting multiple component images consisting of integer samples between 8- and 16-bit resolution, or floating point samples of 16-bit resolution. It is by itself an extension of ISO/IEC 18477-6 and ISO/IEC 18477-7, which specify intermediate range and high-dynamic range image decoding algorithms. Both of these are based on the box-based file format specified in ISO/IEC 18477-3, which is again an extension of 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 lossy, low dynamic range, 8 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, providing 65 536 representable values to describe each channel within a pixel, resulting in over 2.8×10^{14} representable colour values. In some less common scenarios, even greater bit depths are used, requiring a floating-point sample representation.

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.

This document is primarily designed to encode intermediate or high dynamic image sample values **without loss**, or with a precisely controllable bounded loss using the tools defined in ISO/IEC 18477-1 and some minimal extensions of those tools. The goal is to provide a backwards-compatible coding specification that allows legacy applications and existing toolchains to continue to operate on codestreams conforming to this document.

JPEG XT has been designed to be backwards 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 intermediate and high dynamic range and wide colour gamut 8- to 16-bit integer or 16-bit floating point images while also enabling low-complexity encoder and decoder implementations.

This document is an extension of ISO/IEC 18477-1, a compression system for continuous tone digital still images which is backwards 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, though will possibly not be able to reconstruct such streams in full dynamic range, full quality or without loss.

This document is itself based on ISO/IEC 18477-3 that defines a box-based file format similar to other JPEG standards. It also contains elements of ISO/IEC 18477-6 and ISO/IEC 18477-7. The aim of this document is to provide a migration path for legacy applications to support lossless coding of intermediate and high dynamic range images, that is images that are either represented by sample values requiring 8- to 16-bit precision, or even using 16-bit floating point sample resolution. While Rec. ITU-T T.81 | ISO/IEC 10918-1 already defines a lossless mode for integer samples, images encoded in this mode cannot be decoded by applications only supporting the lossy 8-bit-mode; the coding engine for lossless coding in Rec. ITU-T T.81 | ISO/IEC 10918-1 is completely different from the lossy coding mode. Unlike the legacy standard, this document defines a lossless scalable coding engine supporting all bit depths between 8 and 16 bits per sample, including 16-bit floating point samples, while also staying compatible with legacy applications. Such applications will continue to work, but will only able

to reconstruct a lossy 8-bit standard low dynamic range (LDR) version of the full image contained in the codestream. The ISO/IEC 18477 series specifies a coded file format, referred to as JPEG XT, which is designed primarily for storage and interchange of continuous-tone photographic content.

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

Part 8: Lossless and near-lossless coding

1 Scope

This document specifies a coding format, referred to as JPEG XT, which is designed primarily for continuous-tone photographic content. This document defines extensions that allow lossless coding of such content while staying compatible with the core coding system specified in ISO/IEC 18477-1.

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 18477-1:2020, *Information technology — Scalable compression and coding of continuous-tone still images — Part 1: Scalable compression and coding of continuous-tone still images*

ISO/IEC 18477-3:2015, *Information technology — Scalable compression and coding of continuous-tone still images — Part 3: Box file format*

ISO/IEC 18477-6:2016, *Information technology — Scalable compression and coding of continuous-tone still images — Part 6: IDR Integer Coding*

ISO/IEC 18477-7:2017, *Information technology — Scalable compression and coding of continuous-tone still images — Part 7: HDR Floating-Point Coding*

ITU-T T.81 | ISO/IEC 10918-1, *Information technology — Digital compression and coding of continuous tone still images — Requirements and guidelines*

ITU-T BT.601, *Studio encoding parameters of digital television for standard 4:3 and wide screen 16:9 aspect ratios*

3 Terms, definitions, symbols and abbreviated terms

3.1 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

3.1.1
AC coefficient
any DCT coefficient for which the frequency is not zero in at least one dimension

3.1.2
ASCII encoding
encoding of text characters and text strings

Note 1 to entry: In accordance with ISO/IEC 10646.

3.1.3
base decoding path
process of decoding legacy codestream and refinement data to the base image, jointly with all further steps until residual data is added to the values obtained from the residual codestream

3.1.4
base image
collection of sample values obtained by entropy decoding the DCT coefficients of the legacy codestream and the refinement codestream, and inversely DCT transforming them jointly

3.1.5
box
structured collection of data describing the image or the image decoding process embedded into one or multiple APP_{11} marker segments

Note 1 to entry: See ISO/IEC 18477-3:2015, Annex B for the definition of boxes.

3.1.6
coding process
encoding process, a decoding process, or both

3.1.7
DC coefficient
DCT coefficient for which the frequency is zero in both dimensions

3.1.8
DCT coefficient
amplitude of a specific cosine basis function

Note 1 to entry: May refer to an original DCT coefficient, to a quantized DCT coefficient, or to a dequantized DCT coefficient.

3.1.9
decoder
embodiment of a decoding process

3.1.10
decoding process
process which takes as its input compressed image data and outputs a continuous-tone image

3.1.11
dequantization
inverse procedure to quantization by which the decoder recovers a representation of the DCT coefficients

3.1.12
encoder
embodiment of an encoding process

3.1.13
encoding process
process which takes as its input a continuous-tone image and outputs compressed image data

3.1.14
extension image
residual image
sample values as reconstructed by inverse quantization and inverse DCT transformation applied to the entropy-decoded coefficients described by the residual scan and residual refinement scans

[SOURCE: ISO/IEC 18477-6:2016, 3.1.54]

3.1.15
fixed point discrete cosine transformation
implementation of the discrete cosine transformation based on fixed point arithmetic

Note 1 to entry: As specified in Annex E.

3.1.16
forward DCT bypass
transformation that takes an 8×8 sample block and prepares it for entropy coding without applying a discrete cosine transformation

3.1.17
forward fixed point DCT
transformation of an 8×8 sample block from the spatial domain to the frequency domain using the fixed point arithmetic

Note 1 to entry: As specified in Annex E.

3.1.18
forward integer DCT
transformation of an 8×8 sample block from the spatial domain to the frequency domain using the integer approximation of the discrete cosine transformation

Note 1 to entry: As specified in Annex E.

3.1.19
inverse DCT bypass
transformation that takes an 8×8 sample block as generated by entropy decoding and level-shifts it without applying a discrete cosine transformation

3.1.20
inverse fixed point DCT
transformation of an 8×8 sample block from the frequency domain to the spatial domain using the fixed point arithmetic

Note 1 to entry: As specified in Annex E.

3.1.21
inverse integer DCT
the transformation of an 8×8 sample block from the frequency domain to the spatial domain using the integer approximation of the discrete cosine transformation

Note 1 to entry: As specified in Annex E.

3.1.22
frequency
two-dimensional index into the two-dimensional array of DCT coefficients

[SOURCE: ISO/IEC 10918-1:1994, 3.1.61]

3.1.23
high dynamic range
HDR
image or image data comprised of more than eight bits per sample

3.1.24
Huffman encoding
entropy encoding procedure which assigns a variable length code to each input symbol

3.1.25
intermediate dynamic range
image or image data comprised of more than eight bits per sample

3.1.26
legacy codestream
collection of markers and syntax elements

Note 1 to entry: The legacy codestream, as defined by Rec. ITU-T T.81 | ISO/IEC 10918-1 and any syntax elements defined by the ISO/IEC 18477 series, consists of the collection of all markers except those APP_{11} markers that describe JPEG XT boxes by the syntax defined in ISO/IEC 18477-3:2015, Annex A.

3.1.27
lossless
encoding and decoding processes and procedures in which the output of the decoding procedure(s) is identical to the input to the encoding procedure(s)

3.1.28
lossless coding
mode of operation which refers to any one of the coding processes in which all of the procedures are lossless

Note 1 to entry: Coding processes defined in ISO/IEC 18477-8.

3.1.29
lossy
encoding and decoding processes which are not lossless

3.1.30
low-dynamic range
LDR
image or image data comprised of data with no more than eight bits per sample

3.1.31
noise shaping
signal processing technique that removes quantization noise from the low frequency components and injects it into the high frequency domain where it can be removed by filtering

3.1.32
point transformation
application of a location independent global function to reconstructed sample values in the spatial domain

3.1.33
residual decoding path
collection of operations applied to the entropy coded data contained in the residual data box and residual refinement scan boxes up to the point where this data is merged with the legacy data to form the final output image

3.1.34
residual image
sample values as reconstructed by inverse quantization and inverse DCT transformation applied to the entropy-decoded coefficients described by the residual scan and residual refinement scans

3.1.35
residual scan
additional pass over the image data invisible to legacy decoders which provides additive and/or multiplicative correction data of the legacy scans to allow reproduction of high-dynamic range or wide colour gamut data

3.1.36
refinement scan
additional pass over the image data invisible to legacy decoders which provides additional least significant bits to extend the precision of the DCT transformed coefficients

Note 1 to entry: Refinement scans can be either applied in the legacy or residual decoding path.

3.1.37
superbox
box that carries other boxes as payload data

3.1.38
sub box
box that is contained as payload data within a superbox

3.1.39
uniform quantization
procedure by which DCT coefficients are linearly scaled in order to achieve compression

3.1.40
upsampling
procedure by which the spatial resolution of a component is increased

3.2 Symbols

X	width of the sample grid in positions
Y	height of the sample grid in positions
Nf	number of components in an image
$s_{i,x}$	subsampling factor of component i in horizontal direction
$s_{i,y}$	subsampling factor of component i in vertical direction
H_i	subsampling indicator of component i in the frame header
V_i	subsampling indicator of component i in the frame header
$v_{x,y}$	sample value at the sample grid position x,y
R_h	additional number of DCT coefficients bits represented by refinement scans in the base image, $8+R_h$ is the number of non-fractional bits (i.e. bits in front of the "binary dot") of the output of the inverse DCT process in the base image
R_r	additional number of DCT coefficients bits represented by refinement scans in the residual, $P+R_h$ is the number of non-fractional bits (i.e. bits in front of the "binary dot") of the output of the inverse DCT process in the residual image where P is the bit depth indicated in the frame header of the residual codestream
R_b	additional bits in the HDR image. $8+R_b$ is the sample precision of the reconstructed HDR image

3.3 Abbreviated terms

ASCII	American Standard Code for Information Interchange
LSB	least significant bit
MSB	most significant bit
TMO	tone mapping operator
DCT	discrete cosine transformation
FCT	fixed point multi-component transformation
ICT	irreversible multi-component transformation
RCT	reversible multi-component transformation
JPEG	joint photographic experts group

4 Conventions

4.1 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.

4.2 Operators

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

4.2.1 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 $\lfloor (a-1)/2 \rfloor$ for which $y+N\times a = x$ with a suitable integer N.
umod	x umod a is the unique value y between 0 and a−1 for which $y+N\times a = x$ with a suitable integer N

4.2.2 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)

4.2.3 Relational operators

>	greater than
>=	greater than or equal to
<	less than
<=	less than or equal to
==	equal to
!=	not equal to

4.2.4 Precedence order of operators

Operators are listed 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

4.2.5 Mathematical functions

⌈x⌉	ceil of x: returns the smallest integer that is greater than or equal to x
⌊x⌋	floor of x: returns the largest integer that is lesser than or equal to x
|x|	absolute value, is –x for x < 0, otherwise x
sign(x)	sign of x, 0 if x is zero, +1 if x is positive, –1 if x is negative
clamp(x,min,max)	clamps x to the range [min,max]: returns min if x < min, max if x > max or otherwise x
x^a	raises the value of x to the power of a: x is a non-negative real number, a is a real number; x^a is equal to exp(a×log(x)) where exp is the exponential function and log() the natural logarithm; if x is 0 and a is positive, x^a is defined to be 0

5 General

5.1 General definitions

This clause gives an informative overview of the elements specified in this document. It also introduces many of the terms which 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 *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.

5.2 Overview of this document

This document allows near-lossless and lossless coding of high and intermediate dynamic range of photographic images in a way that is backwards compatible to Rec. ITU-T T.81 | ISO/IEC 10918-1. Decoders compliant to Rec. ITU-T T.81 | ISO/IEC 10918-1 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, defined in Annex B and on. A short overview of these coding tools is given in this clause.

The syntax of an ISO/IEC 18477-8 compliant codestream is specified in ISO/IEC 18477-3, that is, this document uses a syntax element denoted as "box" to annotate its syntactical elements. The definition of the box syntax element is not repeated here (refer to ISO/IEC 18477-3). Additional boxes besides those already specified in ISO/IEC 18477-3 are defined in Annex B. In addition, this document also reuses boxes defined in ISO/IEC 18477-6:2016, Annex B and ISO/IEC 18477-7:2017, Annex B.

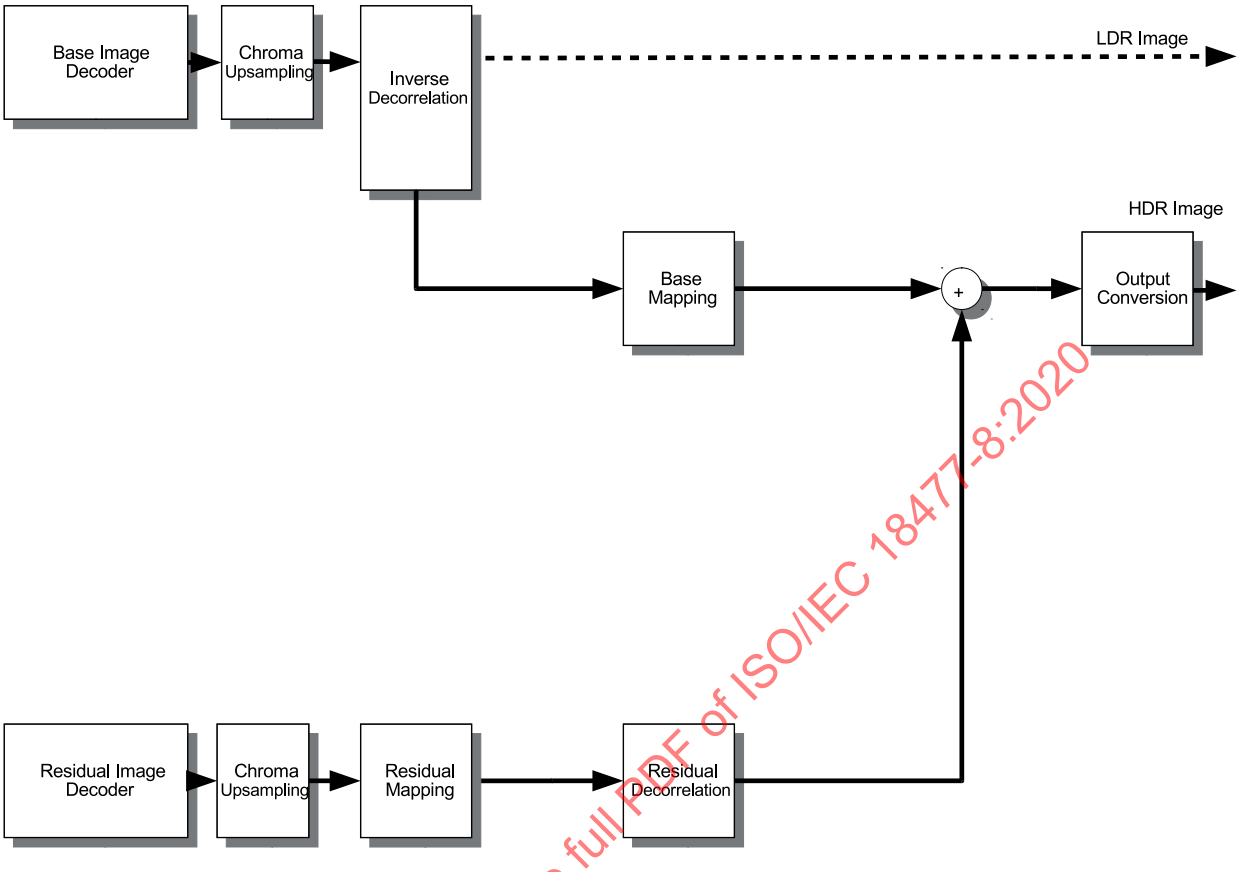

Figure 1 — Overview of the decoding process

To allow lossless and near-lossless coding, this document provides a stricter definition of two elements of the reconstruction process defined in Rec. ITU-T T.81 | ISO/IEC 10918-1 and ISO/IEC 18477-1. The DCT process, only loosely defined in an implementation-agnostic way in Rec. ITU-T T.81 | ISO/IEC 10918-1 is replaced by strictly defined algorithms (specified in Annex E) that a conforming decoder shall follow. Not following these steps will compromise lossless reconstruction. Annex C replaces the ICT transformation by a precise fixed-point implementation denoted as FCT operating entirely on integer samples and thus allowing a fully reproducible transformation conforming decoders shall follow as well. Again, deriving from the specifications of the FCT specified in Annex C will compromise lossless coding. The DCT operations in Annex E and the FCT in Annex C are fully backwards-compatible to the DCT in Rec. ITU-T T.81 | ISO/IEC 10918-1 and the ICT in ISO/IEC 18477-1 and approximate them within the error bounds of Rec. ITU-T T.83 | ISO/IEC 10918-2. Thus, a possible implementation choice for the ISO/IEC 18477 series is to **always** use the DCT and/or FCT as specified here, and not to provide a second implementation based on floating point or other technology.

Lossless coding can be achieved by two alternative mechanisms. First, by applying the integer DCT specified in Annex E, and replacing the base transformation by an identity transformation. This coding mode can only be applied to bit precisions of 8 bits per sample, or up to 12 bits per sample in the presence of **refinement scans** already defined in ISO/IEC 18477-6. Residual scans are not required if the integer DCT is deployed.

Second, by replacing the DCT by the fixed point DCT and by selecting the FCT, the scaled identity transformation or an integer-based free-form transformation as base transformation. The fixed point DCT is specified in Annex E. The FCT and modifications to free-form integer transformations are defined in Annex C. In this case, FCT and the fixed point DCT create an additional coding error that is precisely defined by the coding procedure. This coding error is then corrected by an additive residual scan. While residual scans were already defined in ISO/IEC 18477-6, the Residual DCT Specification box defined in Annex B allows users to bypass the DCT in the residual image completely

and thus avoids additional complexity not required for lossless coding. Residual data using the DCT bypass mode are entropy coded in a new scan type denoted as **residual scan** defined in Annex D. It is closely related to the regular (baseline, extended or progressive) Huffman scan types specified in Rec. ITU-T T.81 | ISO/IEC 10918-1. Since all coefficients are now error residuals in the spatial domain, the natural distinction between DC and AC coefficients no longer applies. This means that the special role the DC coefficient has in the Rec. ITU-T T.81 | ISO/IEC 10918-1 decoding procedure is no longer justified. The **residual scan type** thus extends the AC decoding procedure to the top-left coefficient of an 8×8 block while keeping everything else unchanged. It is thus only a very minor modification of the decoding process specified in Rec. ITU-T T.81 | ISO/IEC 10918-1 that improves the coding efficiency of DCT-bypassed coded error residuals.

If the DCT is bypassed, quantization in the residual domain may cause ringing and stair-casing artefacts. Such artefacts can be eliminated by the **noise shaping** algorithm specified in Annex G.

To decorrelate the components of the additive error residuals, Annex C specifies an additional lossless component decorrelation transformation denoted as the RCT. It is related, but not identical, to the RCT of Rec. ITU-T T.801 | ISO/IEC 15444-1.

The decoding procedure of this document is otherwise closely related to that of ISO/IEC 18477-6. Legacy codestream and refinement scans in the Refinement box form the DCT coefficients of the base image. The image is dequantized and then processed by either the fixed point DCT or the integer DCT. A multi-component decorrelation transformation, the identity, the scaled identity, the FCT or a free-form integer transformation follows. The output of this process is optionally processed by a non-linear point transformation selected by the base Non-linear Point Transformation box. The output of this process is called the **precursor image**.

Image reconstruction proceeds with the entropy decoding of the residual image, if present, encapsulated by the Residual Data box and the Residual Refinement box, both of which also specified in ISO/IEC 18477-6:2016, Annex B. Data is dequantized and either processed through the integer DCT, or the DCT bypass process specified in Annex E. The DCT bypass process requires residual data to be encoded in the **residual scan type** of Annex D. Output of this data is then linearly scaled to range, if required, and inversely decorrelated either by the RCT or an identity transformation. The error residuals are finally added to the precursor image, and modulo arithmetic is used in the addition to ensure a final reconstruction value in range. If the encoded data is floating point, the lossless conversion from floating point to integer specified in ISO/IEC 18477-7:2017, Annex D completes decoding, otherwise the sum of residual and precursor image is already the final output.

For the detailed specification of the decoding and merging process, see Annex A.

5.3 Profiles

The profiles define the implementation of a particular technology within the functional blocks of Figure 1. Profiles are defined in Annex H.

5.4 Encoder requirements

There is **no requirement** in this document that any encoder shall support all profiles. An encoder is only 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 the profile it conforms to. Profiles are defined in Annex H.

5.5 Decoder requirements

A decoding process converts compressed image data to reconstructed image data. It **may** follow the decoding operation specified in this document and ISO/IEC 18477-1 to generate an LDR image from the legacy codestream, and it **shall** follow the operations in **this** document to decode an IDR or an HDR image from the data in the full file. The Decoder shall parse the codestream syntax to extract the parameters, the residual image and the base image. The parameters shall be used to merge the residual image with the base image into the reconstructed IDR image.

In order to comply with this document, a decoder:

a) **may convert** a codestream conforming to this document **without considering the information in any box** into to a low dynamic range image;

b) **shall** convert a conforming codestream within the profile it claims to be conforming to into an intermediate dynamic range image **to exactly the same sample values** as the reference decoder specified in ISO/IEC 18477-5. Additional details on reference testing and allowable error bounds are specified in ISO/IEC 18477-4.

Annex A
(normative)

Encoding and decoding process

A.1 Decoding process

The decoding process relies on a layered approach to extend the capabilities of the Rec. ITU-T T.81 | ISO/IEC 10918-1 process. The encoder decomposes an IDR or HDR image into a base layer, which consists of a tone-mapped version of the IDR/HDR image, and a residual layer. In addition to the residual layer, the codestream includes a description of an approximate inverse tone mapping operation that allows the decoder to reconstruct from the LDR image an approximate IDR/HDR image denoted as precursor image. The errors of this approximation process, if any, are corrected by the residual codestream included in the residual data box and residual refinement box (see Annex B). Both the description of the tone mapping and the residual image are included in boxes invisible to legacy decoders. Such decoders will thus only see the tone mapped LDR image. While the base image complies to ISO/IEC 18477-1 and thus supports only the 8-bit extended or baseline, extended or progressive Huffman modes, the residual image may optionally be encoded in the 12-bit Huffman or progressive modes and may optionally use the residual scan type of Annex D bypassing the inverse DCT.

Figure A.1 illustrates the functionality of a compliant decoder.

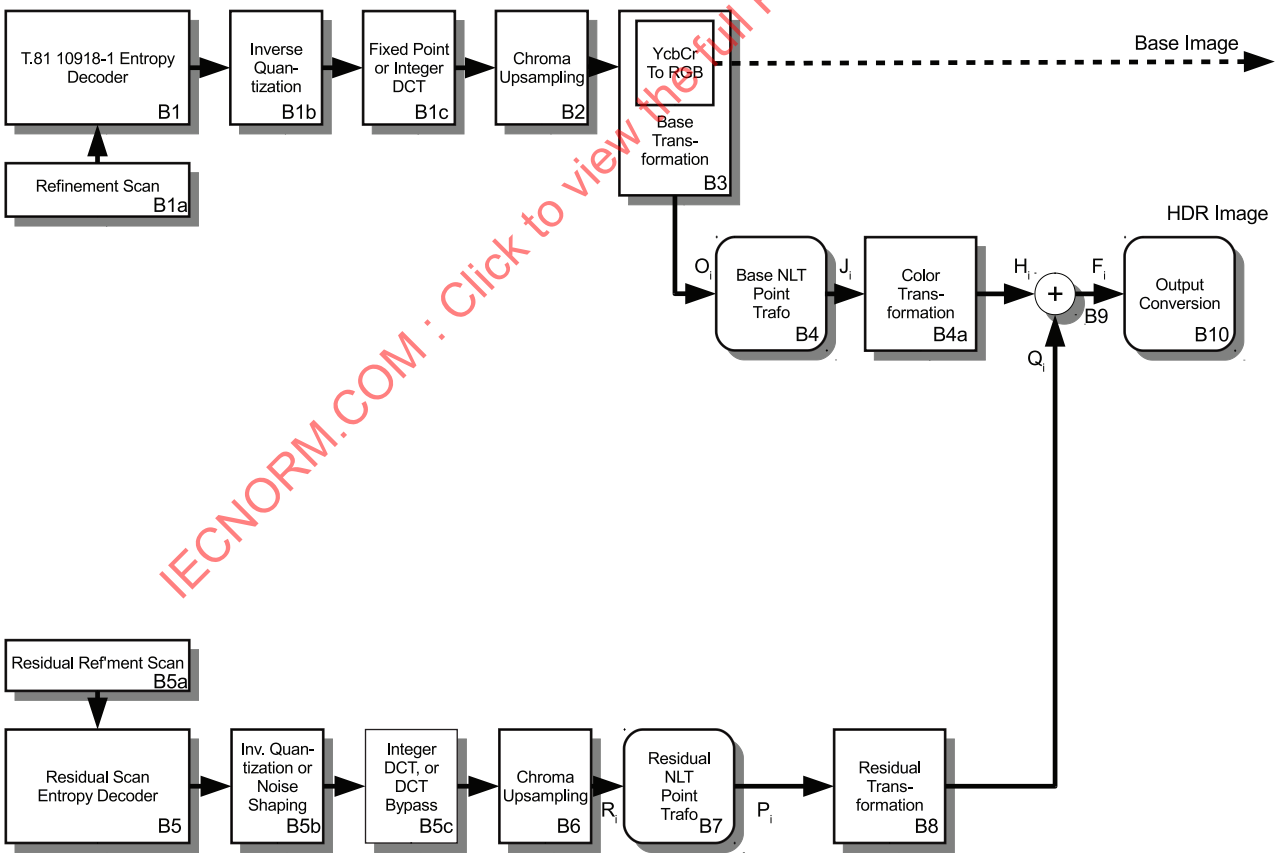

Figure A.1 — High-level overview of the decoding process of a compliant decoder

Bold lines carry three components (or one for greyscale). Round boxes implement point-transformations, square boxes (except B1, B1a, B5, B5a) multiplications by 3×3 matrices. Letters denote signal names.

The reconstruction process of an intermediate dynamic range image from a LDR image and a residual image decoders shall follow the following steps (see Figure A.1).

— In steps B1 and B1a, decode the base DCT coefficients from legacy codestream and the refinement codestream if a Refinement Data box is present. Refinement coding is specified in Annex D. Otherwise, the decoding process of Rec. ITU-T T.81 | ISO/IEC 10918-1 is used unaltered.

— In step B1b, apply inverse quantization as specified in Rec. ITU-T T.81 | ISO/IEC 10918-1.

— In step B1c, apply inverse discrete cosine transformation. The inverse DCT process is either the integer DCT or the fixed point DCT of Annex E, and the DCT process is selected by the **DCT Specification box** specified in Annex B.

— In step B2, the upsampling process specified in Annex F shall be followed to generate samples for all positions on the sample grid.

— In step B3 the linear transformation, as selected by the **Base Transformation box** defined in Annex B, shall be applied to inversely decorrelate the image components. Table B.1 defines which transformation to pick. The output of this block consists of either one or three samples per grid point O_i, depending on the number of components in the legacy codestream. The output of this transformation is rounded to integers and clipped to [0,255].

— In step B4, a non-linear point transformation shall be applied to each of the output components O_i. This process is selected according to the Base non-linear Point Transformation subbox of the Merging Specification box, implementing L_i of Annex C and following the specifications of ISO/IEC 18477-3:2015, Annex C. The outputs of this process are the predicted high dynamic range samples J_i. As above, i=1..Nf.

— Also in step B4, an integer colour transformation is applied to the input values J_i resulting in the output pixel values H_i. The transformation is selected by the Colour Transformation subbox of the Merging Specification box, which selects one of the transformations defined in Annex C. If Nf equals 1, no transformation is performed.

— In steps B5 and B5a, the residual image shall be reconstructed from the data contained in the Residual Codestream box and the Residual Refinement box. The codestream contained in this box either follows the specifications defined in Rec. ITU-T T.81 | ISO/IEC 10918-1 or is encoded in the **residual scan** specified in Annex D. If a **Residual Refinement box** is present, the precision of the samples of the residual codestream shall be extended by refinement coding as specified in Annex D of this document and ISO/IEC 18477-6:2016, Annex D. The number of components of the residual image shall be equal to the number of components signalled in the base image.

— In step B5b, inverse quantization is performed. If the DCT bypass operation is selected, an optional Noise Shaping process is applied to the reconstructed coefficients. Noise shaping is specified in Annex C. Otherwise, the inverse quantization procedure of Rec. ITU-T T.81 | ISO/IEC 10918-1 applies unaltered.

— In step B5c apply either the **Integer DCT** or the **DCT bypass** as specified in Annex E. The DCT operation is selected by the **Residual DCT Specification box** specified in Annex B and shall fit to the scan type of the residual image.

— In step B6, residual data is upsampled to the common sample grid following the specification of Annex F.

— In step B7, apply a non-linear point transformation that is defined by the Residual non-linear Point Transformation subbox of the Merging Specification box. The outputs of this operation are one or three residual sample values per pixel denoted by P_i.

— The outputs of the residual decoding process are one or three integer sample values P_i per sample grid point.

— In step B8, an inverse colour decorrelation shall be applied to the P_i data. This is either the identity transformation **or the Reversible Colour Transformation** (RCT), selected by the Residual Transformation box of Annex B. The RCT is specified in Annex C. The outputs of this process are the inversely decorrelated prediction errors Q_i.

— In step B9, the intermediate dynamic range output F_i, i.e. the output of the decoding process, is reconstructed from H_i, the predicted high dynamic range signal, and the inversely decorrelation prediction errors Q_i. The output bit precision R_b is taken from the Output Conversion subbox of the Merging Specification box defined in Annex B. For this, compute

$$F_i = H_i + Q_i - 2^{Rb+8-1} \quad \text{umod} \quad 2^{Rb+8}$$

— If the Oc flag of the Output Conversion box is set, the values F_i shall be converted to floating point by the algorithm specified in ISO/IEC 18477-7:2017, Annex D. Otherwise, the F_i are already the final output.

A.2 Encoding process

This document does not define a normative encoding process. Any encoding process that generates a file format that is compliant to this document is acceptable as long as this file format reconstructs to the input image feed into the encoder without loss for lossless coding, or with a loss that is within an acceptable error bound defined by ISO/IEC 18477-4.

However, for the sake of clarity, a possible encoder mechanism will be described in this section. The encoder depicted in Figure A.2 first estimates a suitable inverse tonemapping from the input image and suitable image parameters, or two input images if desired. This inverse tonemapping is encoded in an Integer Table Lookup box, and the Base non-linear Point Transformation box of the Merging Specification box points to the Table Lookup box. The Colour Transformation box and Base Transformation box select a suitable transformation from the IDR colour space and the colour space of legacy JPEG applications, which is given by Rec. ITU-T BT.601. Depending on the desired colour spaces and compression performance, the Colour Transformation box may be dropped signalling an identity transformation in the IDR colourspace.

Processing of data is now performed as follows: Input IDR data is first transformed into an alternative colourspace by the inverse of the linear decorrelation given by the Colour Transformation box, if present. Computation of the inverse matrix is up to the encoder. Data is then clamped to range if necessary, and an inverse of the nonlinear point transformation lookup process, given by ISO/IEC 18477-3:2015, Annex C is applied. As above, computing an approximate inverse of the table lookup or parametric curve is up to the encoder and not specified by this document.

The output of this process is decorrelated by the inverse of the base correlation matrix, computing samples in the YCbCr colourspace. The inverse of the Base Transformation is again computed by the encoder and not signalled; only the transformation matrix for the decoder process is included in the codestream if different from the standard YCbCr to RGB transformation. The outputs of this decorrelation process are first clamped to range, then subsampled if desired and transformed by the Fixed Point DCT, quantized and encoded.

To compute the residual image the DCT coefficients computed for the base image are then decoded to the **precursor image** as defined in A.1. That is, the encoding process includes a partial image decoder implementation that mimics the decoder except for the lossless entropy coding stage. The outputs of this partial decoding process are the sample values of the precursor image. The residual image is then computed by subtracting the precursor sample values from the IDR input image sample values modulo 2^{Rb+8} and an offset of 2^{Rb+8-1} is added, again modulo 2^{Rb+8}.

The residual image values are then inversely decorrelated by the inverse of the Residual Transformation and mapped into the residual coding domain by the inverse of the Residual Point transformation. The only possible choices are here the identity transformation or the RCT. If lossless coding is desired, the non-linear point transformation is constrained to be the identity; other choices are available for near-lossless coding. Computation of suitable inverses for near-lossless coding is again up to the encoder; only

the decoding matrix and the decoding non-linear point transformations are signalled in the codestream. The outputs of these processes are then subsampled, where subsampling is only applicable for near-lossless coding. If lossless coding is desired, subsampling cannot be applied to residuals. The resulting samples are up to 16 bits in size; if lossless coding is desired, the entropy coding process can be either a residual scan over non-transformed coefficients using a full 16-bit range, or an integer DCT followed by a 12-bit extended Huffman scan plus a 4-bit residual scan. For near-lossless coding, prediction errors are quantized and the bit depth is determined by the residual non-linear point transformation.

Because the encoder predicts the image a reference decoder will reconstruct from the base layer input codestream, implementation freedom exists in the choice of the forwards DCT and forwards colour transformation. Selecting a DCT or decorrelation transformation that deviates from the description in this standard will still enable a standard conforming decoder to reconstruct the image without loss. Deviating from the informative encoder specifications will, at worst, enlarge the error residual and thus lower the coding efficiency, though will not impact the faithful reconstruction of the image.

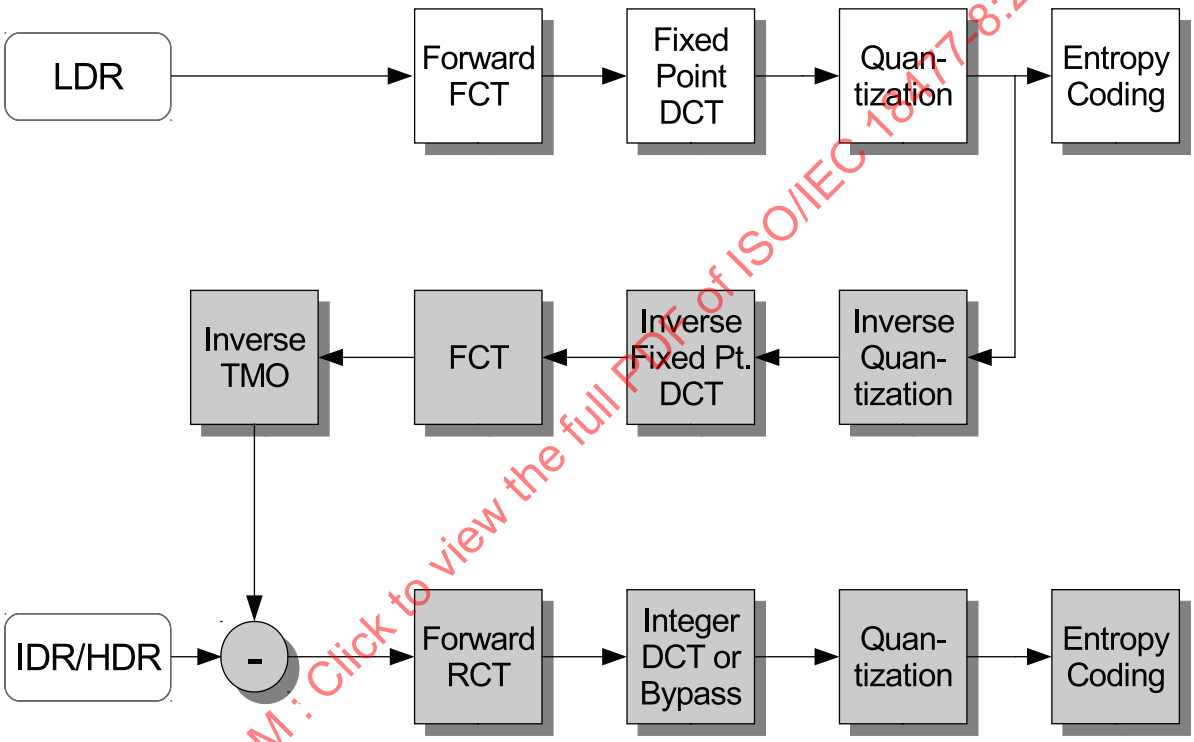

Figure A.2 — High level overview of the closed loop encoder

Operations in grey boxes have to be implemented as specified in this document to allow lossless reconstruction, implementation freedom exists for white boxes.

An entry-level lossless encoder for 8- to 12-bit samples without residual scans is also possible (see Figure A.3). In this case, the Colour Transformation is absent (i.e. an identity) and the Base transformation is the identity as well. That is, both the Colour Transformation boxes as well as the Base transformation box are absent, and a Component Decorrelation Marker with a cc value of 0 is inserted into the codestream. The Base non-linear Point Transformation box shall be absent as well, signalling an identity transformation. The DCT transformation in the base image coding pass is the Integer DCT, allowing a precise inverse, and the overall quantization matrix contains only ones. Entropy coding consists of the regular, extended or progressive Huffman scan with 8-bit precision defined in Rec. ITU-T T.81 | ISO/IEC 10918-1, where additional refinement scans are added as needed to extend the bit depths of the codestream to the required sample precision.

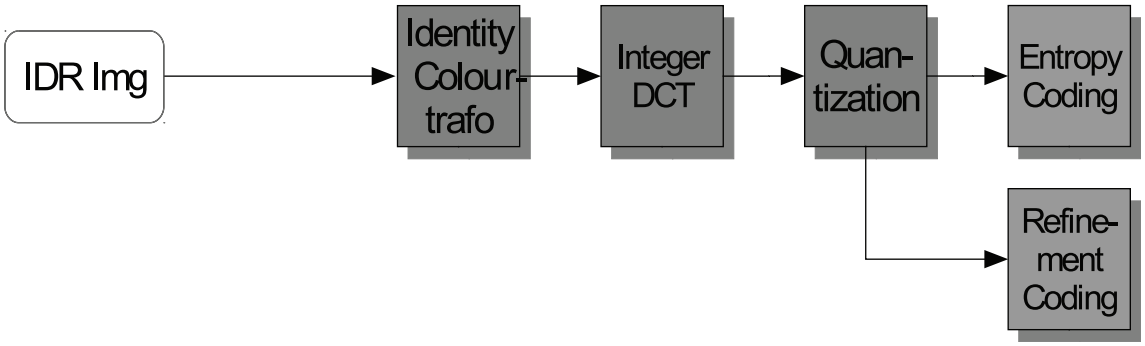

Figure A.3 — High level overview of the entry level encoder

NOTE All operations have to be implemented as given for lossless decoding, no implementation freedom exists.

Annex B
(normative)

Boxes

B.1 General

Annex B selects and refines a subset of the boxes defined in ISO/IEC 18477-3 for the purpose of lossless and near-lossless coding of intermediate and high dynamic range images. It lists those boxes of ISO/IEC 18477-3 that are required for this document. All other boxes are optional and its interpretation is outside the scope of this document. Other parts of ISO/IEC 18477 or other standards may define their meaning, and decoders conforming to this document may ignore them.

Table B.1 lists the boxes required in this document that are paired with ISO/IEC 18477-3. Some of the boxes require additional specifications that are listed in subsequent clauses of Annex B.

Table B.1 — Boxes within ISO/IEC 18477-8

Box name	Box type	Subclause for further definitions in this document
File Type box	0x66747970 ("ftyp")	
Legacy Data Checksum box	0x4C43484B ("LCHK")	
Residual Data box	0x52455349 ("RESI")	
Residual Refinement box	0x5246494e ("RFIN")	
Refinement Data box	0x46494e45 ("FINE")	
Merging Specification box	0x53504543 ("SPEC")	
Parametric Curve box	0x43555256 ("CURV")	
Integer Table Lookup box	0x544f4e45 ("TONE")	
Fix-point Linear Transformation box	0x4D545258 ("MTRX")	
Output Conversion box	0x4F434F4E ("OCON")	B.2
Refinement Specification box	0x52535043 ("RSPC")	
Base Non-linear Point Transformation Specification box	0x4C505453 ("LPTS")	B.3
Residual Non-linear Point Transformation Specification box	0x5152505453 ("QPTS")	B.4
Base Transformation box	0x4C545246 ("LTRF")	B.5
Residual Transformation box	0x52545246 ("RTRF")	B.6
Colour Transformation box	0x43545246 ("CTRF")	B.7
Base DCT Specification box	0x4C444354 ("LDCT")	B.8
Residual DCT Specification box	0x52444354 ("RDCT")	B.9

B.2 Output Conversion box

This mandatory box defines the conversion process from the result of the base image/residual image merging process to the final samples. It describes the final merging process and by that step B10 of the algorithm described in A.1. This box is already defined in ISO/IEC 18477-3:2015, Annex B, though its application to this document further constraints the value of its fields.

This box shall never appear top level in the file, but it shall be a subbox of the Merging Specification box defined in ISO/IEC 18477-3:2015, Annex B. Exactly one Output Conversion box shall appear in the Merging Specification box if a Merging Specification box exists.

The detailed list of operations to be performed by the output conversion is specified in ISO/IEC 18477-3:2015, Annex B.

Table B.2 constraints the parameters of the Output Conversion box as applied in this document.

Table B.2 — Parameter constraints for the Output Conversion box

Parameter	Constraints within this document	Meaning
R_b	0..8	Number of additional bits available for high or intermediate dynamic range data. The bit precision of the reconstructed image shall be computed as $8+R_b$. The parameter of this field shall be 8 if Oc is 1.
Lf	1	This field shall be 1, indicating lossless and near-lossless coding. This document only specifies lossless and near lossless coding.
Oc	0..1	Pseudo-logarithmic output-enable flag. If this flag is set, the output of the merging step shall be the half-logarithmic map defined in ISO/IEC 18477-7:2017, Annex D. If this flag is set, the value of R_b shall be 8.
Ce	0	This field shall be 0. This field indicates whether the output shall be clipped to range $[0,2^{Rb}-1]$ if O_c is 0, or clipped to [−0x7bff,0x7bff] if O_c is 1, before processing the data further.
Ol	0	This field shall be 0. This field indicates whether an output lookup or point transformation is required.
to_0	0	Unused, shall be 0.
to_1	0	Unused, shall be 0.
to_2	0	Unused, shall be 0.
to_3	0	Unused, shall be 0.

B.3 Base Non-linear Point Transformation Specification box

This box defines the non-linear point transformation between the output of the base transformation and the colour transformation. It thus defines step B4 in the decoder description in Annex A. The box layout and box structure is given by the Non-Linear Point Transformation Specification box, defined in ISO/IEC 18477-3:2015, Annex B.

Additional constraints apply, however. The td_i values of the box shall only reference Integer Table Lookup boxes. References to Floating point Table Lookup boxes or Parametric Curve boxes shall not be used. The non-linear point transformation itself is given by the process specified in ISO/IEC 18477-3:2015, Annex C. It requires four additional parameters, the input range R_w, R_e and the output range R_t, R_f. The two value pairs shall be given as follows:

$$R_w = 8 + R_h \quad R_e = 0$$

$$R_t = 8 + R_b \quad R_f = 0$$

The value R_h is the number of refinement scans in the base decoding path and is found in the Refinement Specification box defined in ISO/IEC 18477-3:2015, Annex B. If the Refinement specification box is

absent, the inferred value of R_h is 0. The value R_b is found in the Output Conversion box, where R_b+8 defines the total output precision of the image.

If this box is not present, input v shall values shall be scaled to output values w by:

$$w = v \times 2^{Rb-Rh} \quad \text{if} \quad R_h \leq R_b$$

$$w = \lfloor v/2^{Rh-Rb} \rfloor \quad \text{if} \quad R_h > R_b$$

The type of this box shall be 0x4C52505453, ASCII encoding of "LPTS". The box structure and layout does not deviate from that in ISO/IEC 18477-3:2015, Annex B.

NOTE The constraints for the Base Non-linear Point Transformation Specification box in this document differ slightly from the constraints and definitions in ISO/IEC 18477-6 and ISO/IEC 18477-7. Implementations that want to implement multiple standards may use the Lf flag of the output conversion box to distinguish between the constraints here and those that apply to other parts of the ISO/IEC 18477 series.

B.4 Residual Non-linear Point Transformation Specification box

This box defines the non-linear point transformation between the output of the residual DCT process and the input of the residual transformation. It implements step B7 of Figure A.1 in Annex A. The box structure and layout is already defined in ISO/IEC 18477-3:2015, Annex B, though its purpose is refined here and additional constraints apply. At most one Residual non-linear Point Transformation Specification box shall exist as a sub-box of the Merging Specification box. It shall not appear at top-level of the file. This box shall only be present if the Residual Data box is present.

The td_i values shall only reference Integer Table Lookup boxes. References to Floating point Table Lookup boxes or Parametric Curve boxes shall not be used. If this box is not present, input v values shall be scaled to output values w by:

$$w = v \times 2^{Rb+8+Rf-Rr-P} \quad \text{if} \quad R_r + P \leq R_b + 8 + R_f$$

$$w = \lfloor v/2^{Rr+P-Rb-8-Rf} \rfloor \quad \text{if} \quad R_r + P > R_b + 8 + R_f$$

where R_r is the number of refinement scans in the residual decoding path and is found in the Refinement Specification box defined in ISO/IEC 18477-3:2015, Annex B and R_b is the number of excess integer bits defined by the Output Conversion box specified in B.2. If the Refinement Specification box is absent, the inferred value of R_r is 0. P is the frame precision of the codestream, as recorded in the frame header of the codestream in the Residual Codestream box.

The non-linear point transformation as specified in ISO/IEC 18477-3:2015, Annex C requires two additional input parameter pairs, namely R_w, R_e and R_t, R_f. They shall be computed as follows:

$$R_w = P + R_r - R_e \text{ and } R_e \text{ as selected from Table B.2}$$

$$R_t = 8 + R_b$$

Parameters P, R_r, R_b are as above, the values of R_e and R_f depend on the Residual Colour Transformation, namely the value Xt. Values for R_e are specified in Table B.3, values for R_f in Table B.4.

NOTE The constraints and R_t, R_e and R_w, R_f parameters of the Residual Non-linear Point Transformation Specification box in this document differ slightly from the constraints and definitions in ISO/IEC 18477-6 and ISO/IEC 18477-7. Implementations that want to implement multiple standards may use the Lf flag of the Output Conversion box to distinguish between the constraints here and those that apply to other parts of the ISO/IEC 18477 series.

Table B.3 — Values of R_e depending on Xt

Value of Xt of the Residual Transformation box (see B.6)	Value of P+R_r	Value of R_e
4 (RCT)	<16	1
4 (RCT)	=16	1 (see NOTE)
1 (Identity)	ignore	0
All other values		invalid

Table B.4 — Values of R_f depending on Xt

Value of Xt of the Residual Transformation box (see B.6)	Value of P+R_r	Value of R_f
4 (RCT)	<16	1
4 (RCT)	=16	1 (see NOTE)
1 (Identity)	ignore	0
All other values		invalid

NOTE In case Xt is 4 selecting the RCT as residual transformation and P+R_r is 16, the total number of bits required for a table lookup process is be 17, requiring 32-bit entries in the Integer Table Lookup box, see ISO/IEC 18477-3:2015, Annex B.

B.5 Base Transformation box

This box defines the linear transformation between the output of the base entropy decoding process and the input of the base image non-linear point transformation. It defines the transformation in step B3 of the decoding process specified in Annex A. The box structure and layout is already defined in ISO/IEC 18477-3:2015, Annex B though its purpose is refined here and additional constraints apply to the parameters of the box.

There shall be exactly one Base Transformation box as subbox of the Merging Specification box. and the component transformation shall be consistent with the transformation indicated by the Component Decorrelation Marker specified in ISO/IEC 18477-1. That is, if the cc value of the Component Decorrelation Marker is 0, Xt shall be 1 or larger than 5. If the Component Decorrelation marker is absent or its cc value is nonzero, Xt shall be 3 or larger than 5. If the base file contains only a single component, Xt shall be 1.

The linear transformations specified in Annex C require an additional level shift parameter R_s and a scale value R_e. The value of R_s for the Base transformation shall be given as

$$R_s = 8 + R_h - 1 + R_e$$

where R_h is the number of refinement scans contained in the base decoding path. The value of R_h is found in the Refinement Specification subbox of the Merging Parameter box. It shall be 0 if no Refinement Specification box is present. R_e depends on the choice of the DCT in the base decoding path and hence on the entry of the DCT Specification box. Its value is selected according to Table B.5.

The type of the Base Transformation box shall be 0x4C545246, ASCII encoding of "LTRF".

Table B.5 constraints the parameters and parameter sizes of this box, Table B.6 defines the encoding of the Xt parameter and Table B.7 the value of R_e.

Table B.5 — Parameter constraints of the Base Transformation box

Parameter	Constraints within this document	Meaning
Xt	1, 3, 5..15	Defines the linear transformation to be used as base transformation, see Table B.5 for the encoding.
Re	0	Shall be 0.

Table B.6 — Encoding of the LXt Parameter

Value	Transformation to be used
0	Reserved for ITU | ISO/IEC purposes.
1	The identity transformation shall be used.
2	Reserved for ITU | ISO/IEC purposes.
3	The FCT Transformation as specified in Annex C shall be used.
4	Reserved for ITU | ISO/IEC purposes.
5..15	The free form transformation with offset shift defined by the Integer or Floating Point Linear Transformation box whose M value matches the value of Xt shall be used. The Integer or Floating Point Linear Transformation boxes are specified in ISO/IEC 18477-3:2015, Annex B and their application and implementation are specified in Annex C.

Table B.7 — Selection of R_e for the Base Transformation

Value of dct in the base DCT Specification box (see B.8)	Value of R_e
0 (fixed point DCT)	4
2 (integer DCT)	0
All other values	invalid

B.6 Residual Transformation box

This box defines the linear transformation between the output of the non-linear point transformation in the residual decoding path and the addition of the residual to the output of the colour transformation in the base decoding path. It defines the linear transformation in step B8 of the decoding process specified in Annex A of this document. The box structure and layout is already defined in ISO/IEC 18477-3:2015, Annex B though its purpose and its parameters are refined here.

This box shall only exist as a subbox of the Merging Specification box specified in ISO/IEC 18477-3:2015, Annex B. It shall not appear top level. This box shall exist if and only if a Residual Data box is present at the top level of the file.

The linear transformations specified in Annex C require an additional level shift parameter R_s and a scale value R_e. The value of R_s for the residual transformation shall be given as:

$$R_s = 8 + R_b - 1$$

where R_b+8 is the sample precision of the reconstructed IDR output image. The value of R_b can be found in the Output Conversion box, see B.2. The scale value R_e shall be 0.

The type of this box shall be 0x52545246, ASCII encoding of "RTRF".

Table B.8 constraints the parameters and parameter sizes, Table B.9 describes the encoding of the Xt parameter.

Table B.8 — Parameter constraints of the Residual Transformation box

Parameter	Constraints within this document	Meaning
Xt	1, 4..15	Defines the linear transformation to be used as residual transformation, see Table B.8 for the encoding.
Re	0	Shall be 0.

Table B.9 — Encoding of the Xt parameter

Value	Transformation to be used
0	Reserved for ITU | ISO/IEC purposes.
1	The identity transformation shall be used.
2..3	Reserved for ITU | ISO/IEC purposes.
4	The RCT Transformation as specified in Annex C shall be used.
5..15	Reserved for ITU | ISO/IEC purposes.

B.7 Colour Transformation box

This box defines the linear transformation between the output of the non-linear point transformation in the base domain and the addition of the inversely decorrelated transformed residual. It defines the linear transformation in step B4a of the decoding process specified in Annex A. The box structure and layout is already defined in ISO/IEC 18477-3:2015, Annex B, though its purpose is refined here.

This box shall only exist as a subbox of the Merging Specification box specified in ISO/IEC 18477-3:2015, Annex B and it may only exist if the number of components in the image Nf equals 3. It shall not appear top level. If this box does not exist, the Colour Transformation shall be the identity transformation; otherwise the Xt parameter of the Colour Transformation box specifies the transformation matrix to pick. If a free form transformation is selected, i.e. the Xt parameter is greater than 4, it shall only reference Fix Point Linear Transformation boxes.

The linear transformations specified in Annex C require an additional level shift parameter R_s. The value of R_s for the colour transformation shall be given as:

$$R_s = -\infty \quad \text{(i.e. no level shift)}$$

The type of this box shall be 0x43545246, ASCII encoding of "CTRF".

Table B.10 constraints the parameters from that defined in ISO/IEC 18477-3.

Table B.10 — Parameters of the Residual Transformation box

Parameter	Constraints within this document	Meaning
Xt	1 or 5..15	Defines the linear transformation to be used as base transformation, see Table B.11 for the encoding.
Re	0	Shall be 0.

Table B.11 — Encoding of the Xt parameter of the Colour Transformation box

Value	Transformation to be used
0	Reserved for ITU | ISO/IEC purposes.
1	The identity transformation shall be used.
2..4	Reserved for ITU | ISO/IEC purposes.

Table B.11 *(continued)*

Value	Transformation to be used
5..15	The free form transformation without offset shift defined by the Integer or Floating Point Linear Transformation box whose M value matches the value of Xt shall be used. The Integer or Floating Point Linear Transformation boxes are specified in ISO/IEC 18477-3:2015, Annex B and their application and implementation are specified in Annex C.

B.8 Base DCT Specification box

This mandatory box defines the DCT operation in the base decoding path. Lossless and near-lossless decoding requires a fully-specified, bit-precise DCT, two of which are specified in Annex E. It defines the operation of the B1c box in the functional diagram of Annex A. This box uses the layout and structure of the DCT Specification box defined in ISO/IEC 18477-3:2015, Annex B, but refines its parameters.

The box selects between two possible DCT implementations: The **Integer DCT** is a fully invertible integer to integer transformation of a relatively high implementation complexity. It allows lossless coding even without a residual codestream. The **Fixed Point DCT** is a fixed point approximation of the DCT that is only invertible up to a small error. However, since the **Fixed Point DCT** is fully specified, the error at decoder side can be predicted precisely and can be corrected by an additional residual scan over the data. The implementation complexity of the **Fixed Point DCT** is much lower than that of the **Integer DCT**.

The type of the Base DCT Specification box shall be 0x4C444354, ASCII encoding of "LDCT".

Table B.12 constraints the parameters and parameter sizes; Table B.13 specifies the encoding of the DCT parameter.

Table B.12 — Parameter constraints for the Base DCT Specification box

Parameter	Constraints within this document	Meaning
dct	0, 2	Selects the DCT that shall be used to reconstruct the base image. See Table B.13 for the encoding.
Ns	0	Reserved for ITU | ISO/IEC purposes

Table B.13 — Encoding of the DCT parameter of the Base DCT Specification box

Value	Transformation to be used
0	The fixed point DCT shall be used
2	The integer DCT shall be used
All other values	Reserved for ITU | ISO/IEC purposes

B.9 Residual DCT Specification box

This box defines the DCT operation and the noise shaping in the residual decoding path and thus selects the DCT transformation to be used for step B5c of the functional diagram in Annex A. Its structure and layout is defined by the DCT Specification box, specified in ISO/IEC 18477-3:2015, Annex B, though the purpose of the box and its parameters are refined.

To enable lossless coding, the residual coding pass corrects for residual errors of the base decoding pass and is thus fully invertible. Two choices exist for the DCT: Either the invertible **Integer DCT** or the **DCT bypass** process which replaces the DCT by a simple level shift. The latter shift uses a slightly modified entropy coding defined in Annex D. For near lossless coding, the DCT or DCT-bypassed signal is quantized, causing staircasing artifacts if the DCT is bypassed and image residuals are quantized

directly in the spatial domain. An optional **Noise Shaping** process specified in Annex G avoids this problem.

The type of the Residual DCT Specification box shall be 0x52444354, ASCII encoding of "RDCT".

Table B.14 refines the definition of the parameters and parameter sizes; Table B.15 describes the encoding of the DCT parameter.

Table B.14 — Parameter constraints for the Residual DCT Specification box

Parameter	Parameter constraints within this document	Meaning
DCT	2, 3	Selects the DCT that shall be used to reconstruct the base image. See Table B.15 for the encoding.
RNs	0, 1	If 0, noise shaping is disabled. If 1, noise shaping as specified in Annex G shall be enabled. The value of this parameter shall be 0 if DCT is 2. All other values are reserved for ITU | ISO/IEC purposes.

Table B.15 — Encoding of the DCT parameter of the Residual DCT Specification box

Value	Transformation to be used
2	The Integer DCT shall be used
3	The DCT bypass process shall be used.
All other values	Reserved for ITU | ISO/IEC purposes

Annex C
(normative)

Multi-component decorrelation transformation

C.1 General

NOTE In this annex, the flowcharts and tables are normative only in the sense that they are defining an output that alternative implementations shall duplicate.

This annex defines the multiple component decorrelation transformations available as base, residual and colour transformations of the decoding process (see Figure C.1). Since lossless and near-lossless coding requires predictable errors, the transformations in ISO/IEC 18477-6:2016, Annex C and ISO/IEC 18477-7:2017, Annex C cannot be applied without change. This annex specifies a fixed-point approximation of the ICT specified in ISO/IEC 18477-6 denoted as the **FCT**. Since the FCT approximates the ICT within the error bounds of ISO/IEC 18477-4, implementations may choose to always substitute the ICT by the FCT with proper scaling.

This annex also introduces an additional invertible integer to integer transformation denoted as **RCT**. It is a fully invertible integer approximation of the FCT that can be used in the residual coding pass to improve the coding efficiency of colour images.

In addition to these processes, an identity transformation is also available that shall be used for grey-scale images and if the colour transformation is disabled by the Base or Residual Transformation boxes. Since the **Fixed Point DCT** is a fixed point transformation, an additional scaling factor is included in the Identity transformation to compensate for the preshifted integer bits.

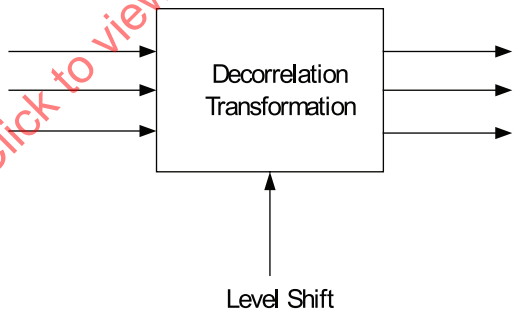

Figure C.1 — Input and output of a decorrelation transformation: Input components, output components and the DC level shift

C.2 Fixed point multi-component transformation (FCT)

This transformation can be applied in the lossless process for reconstructing a low-dynamic range version of the image from the data encoded in the entropy coded data segment following SOS markers. It is a fully specified version of the YCbCr to RGB transformation based on fixed points arithmetic.

In the following, set I_0, I_1 and I_2 be the sample values reconstructed from the upsampled components 0, 1 and 2, and let L_0, L_1 and L_2 be the sample values of the reconstructed data, let R_s be the number of level shift scale bits defined in B.5 and let R_e be the number of integer scale bits defined by Table B.6. Then:

$$L_0 = \lfloor (2^{13} \times I_0 + 11485 \times (I_2 - 2^{Rs}) \quad +2^{12+Re})/2^{13+Re} \rfloor$$

$$L_1 = \lfloor (2^{13} \times I_0 - 5850 \times (I_2 - 2^{Rs}) - 2819 \times (I_1 - 2^{Rs}) \quad +2^{12+Re})/2^{13+Re} \rfloor$$

$$L_2 = \lfloor (2^{13} \times I_0 + 14516 \times (I_1 - 2^{Rs}) + 2^{12+Re})/2^{13+Re} \rfloor$$

NOTE This transformation approximates the ICT of ISO/IEC 18477-6.

C.3 Fixed point forward multi component transformation (FCT)

The purpose of the FCT is to provide a fully specified component decorrelation transformation allowing lossless coding that is compatible with the RGB to YCbCr transformation deployed in legacy implementations.

Denote by L_0, L_1 and L_2 the input sample values of the high-dynamic range source image. Denote by I_0, I_1 and I_2 the output components, let R_s be the number of level shift bits defined in B.5. Then:

$$I_0 = \lfloor (2449 \times L_0 + 4809 \times L_1 + 934 \times L_2 + 2^{12-Re})/2^{13-Re} \rfloor$$

$$I_1 = \lfloor (-1382 \times L_0 - 2714 \times L_1 + 4096 \times L_2 + 2^{13-Re} \times 2^{Rs} + 2^{12-Re})/2^{13-Re} \rfloor$$

$$I_2 = \lfloor (4096 \times L_0 - 3430 \times L_1 - 666 \times L_2 + 2^{13-Re} \times 2^{Rs} + 2^{12-Re})/2^{13-Re} \rfloor$$

C.4 Free-form multi-component transformation

This transformation offers a generic linear transformation that can be applied as Colour Transformation. Let I_0, I_1 and I_2 be the inputs to this transformation and O_0, O_1 and O_2 its output.

$$O_0 = \lfloor (a_{11} \times I_0 + a_{12} \times (I_1 - 2^{Rs}) + a_{13} \times (I_2 - 2^{Rs}) + 2^{12+Re})/2^{13+Re} \rfloor$$

$$O_1 = \lfloor (a_{21} \times I_0 + a_{22} \times (I_1 - 2^{Rs}) + a_{23} \times (I_2 - 2^{Rs}) + 2^{12+Re})/2^{13+Re} \rfloor$$

$$O_2 = \lfloor (a_{31} \times I_0 + a_{32} \times (I_1 - 2^{Rs}) + a_{33} \times (I_2 - 2^{Rs}) + 2^{12+Re})/2^{13+Re} \rfloor$$

NOTE It is up to the encoder to determine a suitable inverse of the above transformation.

C.5 Inverse identity transformation

The inverse identity transformation maps its input values to the output without applying an inverse linear decorrelation step. Denote the input value of the identity transformation with I_0 and the output value by L_0. R_e is the number of level shift bits defined by Table B.6 if the transformation is applied in the legacy decoding pass, R_e is 0 if the transformation is applied in the residual decoding pass. The inverse identity transformation then sets:

$$L_0 = \lfloor (2^{13} \times I_0 + 2^{12+Re})/2^{13+Re} \rfloor$$

If more than one component has to be transformed, the above equation(s) apply identically to all components.

C.6 Forward identity transformation

This transformation is used to generate the input of the base coding process when the number of components in the frame is one, or the component decorrelation transformation is disabled. Let R_s be the number of level shift bits defined in B.5. If L_0 is the input value, then this transformation sets the output value I_0 to:

$$I_0 = \lfloor (2^{13} \times L_0 + 2^{12-Re})/2^{13-Re} \rfloor \text{ umod } 2^{Rs+1}$$

If more than one component is to be transformed, then the above applies to all three components identically.

C.7 Reversible multi component transformation (RCT)

The RCT is a lossless integer to integer transformation that is a coarse approximation of the FCT, but unlike the above, an exact inverse transformation exists.

In the following, let I_0, I_1 and I_2 be the reconstructed data serving as input to the transformation, and R_0, R_1 and R_2 generated output sample values. Let R_s be the number of scale bits defined in B.6. Then set:

$$T_0 = \lfloor I_0/2 \rfloor$$

$$T_1 = I_1 - 2^{Rs+1}$$

$$T_2 = I_2 - 2^{Rs+1}$$

$$R_1 = (T_0 - \lfloor (T_1+T_2)/4 \rfloor) \bmod 2^{Rs+1}$$

$$R_0 = (R_1 + T_2) \bmod 2^{Rs+1}$$

$$R_2 = (R_1 + T_1) \bmod 2^{Rs+1}$$

C.8 Reversible forward multi component transformation (RCT)

The transformation supplied by this subclause decorrelates the additive error residuals of the lossless and lossy additive coding process and is the exact inverse of the RCT defined in C.7. Let I_0, I_1 and I_2 be the (integer) error residuals of the additive coding process, and R_0, R_1 to R_2 the output of the transformation. Let R_s be the number of level shift bits, defined in B.6. Then set:

$$R_1 = (((I_2 \bmod 2^{Rs+1}) - (I_1 \bmod 2^{Rs+1})) \operatorname{smod} 2^{Rs+1}) + 2^{Rs+1}$$

$$R_2 = (((I_0 \bmod 2^{Rs+1}) - (I_1 \bmod 2^{Rs+1})) \operatorname{smod} 2^{Rs+1}) + 2^{Rs+1}$$

$$R_0 = 2 \times ((I_1 + \lfloor (R_1+R_2) / 4 \rfloor) \bmod 2^{Rs+1})$$

NOTE This transformation expands the range of the components by one bit, and quantization parameters would be configured accordingly. Lossless coding requires a quantization interval of size 1 for components 1 and 2 (chroma), and a quantization value of either 1 or 2 for component 0 (luma) if the DCT is bypassed. It is important to implement the operations, especially the modulo operations, exactly as stated to ensure lossless reconstruction because wrap-around may take place.

Annex D
(normative)

Entropy coding of residual data in the DCT-bypass and large range mode

D.1 General

NOTE In this annex the flowcharts and tables are normative only in the sense that they are defining an output that alternative implementations shall duplicate.

This annex describes extensions of the entropy coding/decoding process for coding of residual data. The first class of processes is used when residual coding bypasses the DCT, i.e. the Rdct parameter of the Residual DCT Specification box is set to 3. In such a case, the residual scan types specified in this annex replace the regular scan types specified in Rec. ITU-T T.81 | ISO/IEC 10918-1. The second class of processes is used when the DCT remains enabled, though the DCT may generate coefficients that are too large to be representable by the regular or extended Huffman coding mode defined in Rec. ITU-T T.81 | ISO/IEC 10918-1.

NOTE Unlike refinement coding, the residual codestream is always contained in a single box which can, however, spread over several JPEG XT extension marker segments.

The syntax of the residual codestream follows the syntax of Rec. ITU-T T.81 | ISO/IEC 10918-1 and consists of a frame header, table definitions, followed by at least one scan over the residual data. The frame header defines the entropy coding process; for the purpose of residual coding, the type of the SOF marker introducing the frame header shall be either SOF_{r1} SOF_{r2} indicating sequential or progressive residual scan types or SOF_{e1} for the large range sequential scan type.

The residual sequential and progressive scan types are defined in D.2.2 and D.2.3, the large range sequential scan in D.2.4. A large range progressive scan type is not defined, though implementation notes for encoding DCT coefficients in a progressive mode overrunning the limitations of Rec. ITU-T T.81 | ISO/IEC 10918-1 are given in D.3.

The image dimensions and the number of components indicated in the frame header of the residual codestream shall be identical to those of the legacy codestream. Subsampling factors of the residual codestream shall be 1, i.e. no subsampling shall take place.

Similar to regular scan types defined in Rec. ITU-T T.81 | ISO/IEC 10918-1, residual DCT-bypass scan types can also be extended by Refinement Scans (see ISO/IEC 18477-6:2016, Annex D). The entropy coding for residual refinement scans with the DCT bypassed differs slightly from the regular refinement scan. The necessary modifications for the residual refinement scan are specified in D.4.

D.2 Modified decoding process of residual error signals

D.2.1 Frame Markers of DCT bypass and Large Range entropy coding modes

The Residual Data box shall be decoded using the DCT-bypass entropy coding algorithm if and only if the DCT in the residual coding path is disabled by setting the Rdct parameter of the Residual DCT box to 3.

The syntax of the codestream in the Residual Data box shall follow the syntax specified in Rec. ITU-T T.81 | ISO/IEC 10918-1, except that the new start of frame markers listed in Table D.1 indicate the presence of one of two possible DCT bypass entropy coding algorithms. Decoding of data in the entropy coded segments ECS_0 to ECS_{last} shall follow these new algorithms.

Table D.1 — Start-of-frame markers for DCT-bypass entropy coding

Start of frame marker type	Value	Entropy coding algorithm
SOF_{r1}	0xFFB1	Sequential (extended Huffman) DCT-bypass entropy coding as specified in D.2.2
SOF_{r2}	0xFFB2	Progressive DCT-bypass entropy coding as specified in D.2.3
SOF_{e1}	0xFFB3	Large range sequential entropy coding as specified in D.2.4
SOF_{r3}	0xFFB9	Reserved for ITU | ISO/IEC purposes.
SOF_{r4}	0xFFBA	Reserved for ITU | ISO/IEC purposes.
SOF_{e2}	0xFFBB	Reserved for ITU | ISO/IEC purposes.

The bit precision of the DCT-bypass or large range encoded samples is indicated by the P parameter of the frame header, as for regular scan types, see Rec. ITU-T T.81 | ISO/IEC 10918-1:1994, Table B.2. However, acceptable values for P shall be between 8 and 17, i.e. the upper limit for P is 17, and it is not limited to 8 or 12. The syntax of the frame header of the sequential and progressive DCT-bypass and large range sequential entropy coding is specified in Table D.2. It is similar to the frame header defined in Rec. ITU-T T.81 | ISO/IEC 10918-1:1994, Table B.2.

Table D.2 — Frame header of the DCT-bypass entropy coding scans

Parameter	Size (in Bits)	Values	Meaning
Lf	16	8+3×Nf	Size of the marker segment, not including the marker
P	8	8..17	Precision of the residual signal in bits (see D.6 for recommendations)
Y	16	1..65535	Height of the residual image in pixels. This value shall be identical to the value of Y of the legacy frame header
X	16	1..65535	Width of the residual image in pixels. This value shall be identical to the value of X of the legacy frame header
Nf	8	1 or 3	Number of components of the residual frame. This value shall be identical to the value of Nf of the legacy frame header
C_i	8 (per entry)	0..255	Component identifier. This value assigns a unique label to the i-th component in the sequence of frame component specification parameters; the component identifiers are referenced in the scan header of the residual scans.
H_i	4	1..2	Horizontal subsampling identifier for component i. The pair (H_i,V_i) of component i identifies the subsampling factors according to ISO/IEC 18477-1:2020, Annex A.
V_i	4	1..2	Vertical subsampling identifier for component i. The pair (H_i,V_i) of component i identifies the subsampling factors according to ISO/IEC 18477-1:2020, Annex A.
Tq_i	8	0..3	Quantization table selector; selects one of four possible quantization tables to use for dequantization of the decoded samples of component i. Only entry #63 of each quantization table is used.

D.2.2 Modified sequential decoding process of DCT-bypassed error signals

Sequential DCT-bypass coding is indicated by a frame marker of type SOF_{r1} in the Residual Codestream box. It shall only be signalled in the codestream if Rdct parameter of the Residual DCT Specification box is set to 3. Residual data is in this case encoded by a modified extended Huffman process. The coding

procedure is identical to that specified in Rec. ITU-T T.81 | ISO/IEC 10918-1:1994, F.2 except for the following modifications:

— The DC decoding steps, Rec. ITU-T T.81 | ISO/IEC 10918-1:1994, F.2.1.3.1 and F.2.2.1, shall be skipped and shall be not effective.

— Decoding starts with the AC coding, Rec. ITU-T T.81 | ISO/IEC 10918-1:1994, F.2.2.2, the decoding procedure starts with the coefficient in the top left corner, i.e. in Rec. ITU-T T.81 | ISO/IEC 10918-1:1994, F.2.2.2, Figure F.13 the value of K shall be initialized with 0, and not with 1; furthermore, all coefficients shall be set to zero, not only the AC coefficients, i.e. ZZ(0,...,63)=0.

— Only the AC Huffman tables shall be used for decoding the entropy coded data segment. DC Huffman table indices in the scan header shall be ignored. The corresponding Huffman table definitions shall be found in the tables/miscellaneous section of the residual codestream.

— An extended Huffman alphabet, as depicted in Figure D.1 shall be used to encode the size category of the residual samples. The value of the size category symbol SSSS for AC coding and the size ranges of the error residuals are specified in Table D.3. Depending on the value of P in the residual frame header, only a subset of the entries in the table are required, i.e. only all entries up to category P-1. Samples in magnitude category SSSS=16 do not use the combined magnitude/runlength Huffman coding employed for all other magnitude categories. Instead, a sample of this size is encoded by the Huffman symbol 0x10 (16) which is otherwise unused in the sequential scan pattern, followed by the run-length, which is encoded in four bits that are directly appended to the output stream bypassing the Huffman coder. Low order bits of the coefficient are **not** appended to the codestream.

Table D.3 — Magnitude categories and corresponding error residual ranges

SSSS (Symbol for Huffman coding of the magnitude category)	Error residual range
1	–1, 1
2	–3, –2, 2, 3
3	–7..–4, 4..7
4	–15..–8, 8..15
5	–31..–16, 16..31
6	–63..–32, 32..63
7	–127..–64, 64..127
8	–255..–128, 128..255
9	–511..–256, 256..511
10	–1023..–512, 512..1023
11	–2047..–1024, 1024..2047
12	–4095..–2048, 2048..4095
13	–8191..–4096, 4096..8192
14	–16383..–8192, 8192..16383
15	–32767..–16384, 16384..32767
16	–32768, see text and note below

NOTE The magnitude category SSSS=16 can only appear in DCT-bypass coding for which P=17. If implementations want to avoid this case, refinement scans in the residual domain are one alternative, or progressive residual scans are another. Note further that the magnitude category SSSS=16 includes only a single sample value, thus low-order bits for this sample value are not included.

	SSSS=0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
R=0	EOB															
1	SSSS=16															
2																
3																
4																
5																
6																
7																
8																
9																
10																
11																
12																
13																
14																
15	ZRL															

Figure D.1 — Huffman symbol allocation for the residual data coding in the DCT-bypass mode

NOTE Vertical: Size of the run, horizontal the magnitude category. Table cells that are crossed out are not in use.

D.2.3 Modified progressive decoding process of DCT-bypassed error signals

Progressive DCT-bypass coding is indicated by a frame marker of type SOF_{r2} in the Residual Codestream box. It shall only be signalled in the codestream if Rdct parameter of the Residual DCT Specification box is set to 3. Decoding shall be performed according to Rec. ITU-T T.81 | ISO/IEC 10918-1:1994, G.2 though Rec. ITU-T T.81 | ISO/IEC 10918-1:1994, Figure G.3 also applies if Ss=0. This causes the following modifications in the decoding operation:

— The DC decoding step will be skipped, i.e. Rec. ITU-T T.81 | ISO/IEC 10918-1:1994, G.1.2.2 is not effective.

— The AC decoding algorithms defined in Rec. ITU-T T.81 | ISO/IEC 10918-1:1994, G.1.2.2 and G.1.2.3, specifically Rec. ITU-T T.81 | ISO/IEC 10918-1:1994, Figures G.3 and G.7 also apply if SS = 0, i.e. if the start of the spectral selection is zero.

  NOTE For example, if SS = 0, the flowcharts in Rec. ITU-T T.81 | ISO/IEC 10918-1:1994, G.3 and G.7 start with K=–1 and the first coefficient decoded is ZZ(0), the top left block edge.

— Only the AC Huffman tables will be used for decoding the entropy coded data segment. DC Huffman table indices in the scan header will be ignored.

The extended Huffman table of Table D.3 shall be used to encode the size category of the residual samples for the first scan of a progressive scan pattern. The magnitude category SSSS=16 is not used for progressive DCT-bypass decoding.

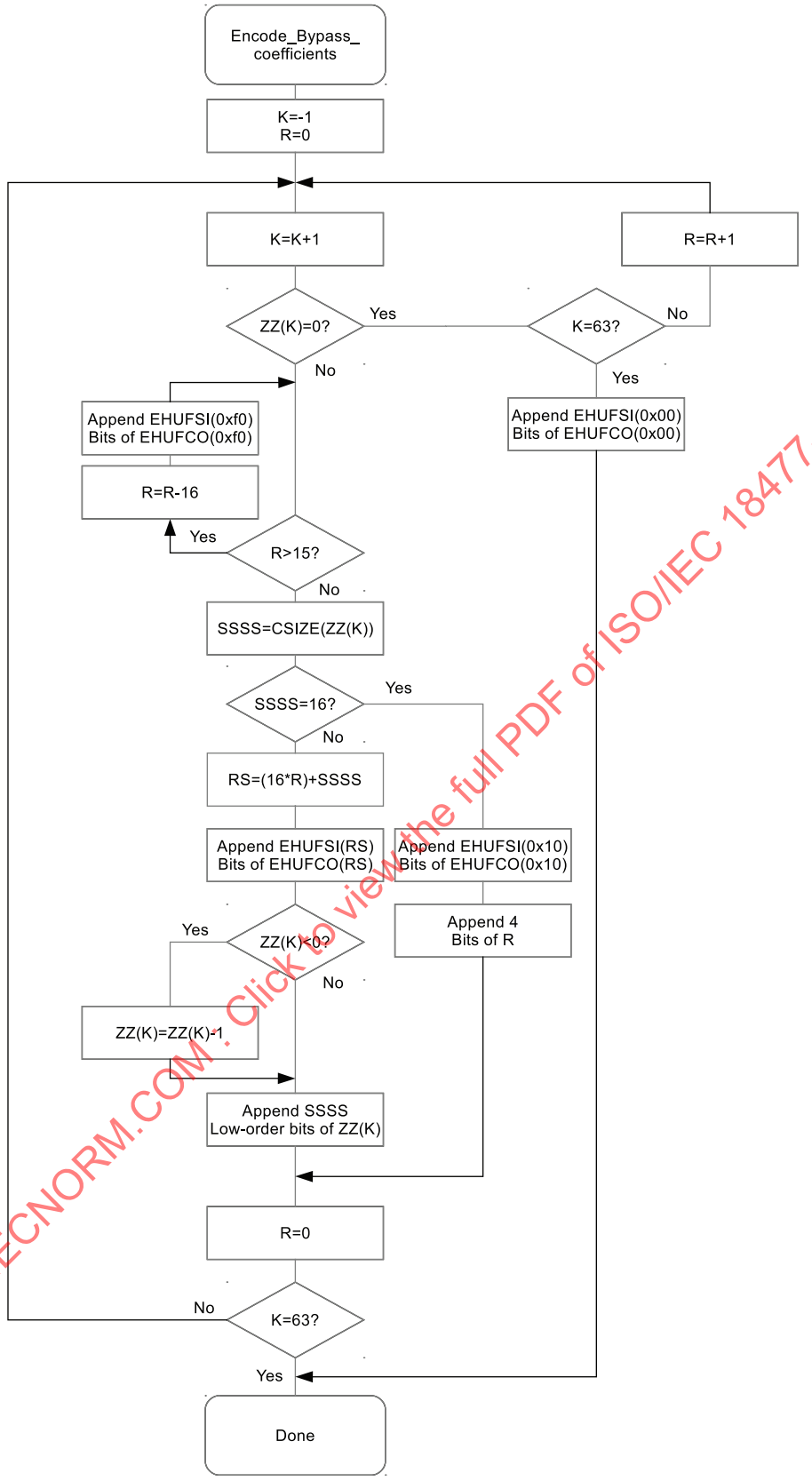

Figure D.2 — Encoding procedure for sequential DCT-bypass coding

D.2.4 Modified decoding process of large range error signals

Sequential large range DCT coding is indicated by a frame marker of type SOF_{e1} in the Residual Codestream box. It shall only be signalled in the codestream if Rdct parameter of the Residual DCT Specification box is set to 2, indicating the presence of the Integer DCT process. Since error residuals may be up to 16 bits large, DCT coefficients may have a range of up to 21 bits, requiring an extended encoding and decoding mechanism that is specified in the following as a modification of the decoding process in Rec. ITU-T T.81 | ISO/IEC 10918-1:1994, F.2.2.1 (DC coding) and F.2.2.2 (AC coding).

— The DC decoding step, as indicated in Rec. ITU-T T.81 | ISO/IEC 10918-1:1994, F.2.2.1 of the legacy standard shall be applied unaltered, though the Huffman alphabet for DC coding, required to decode the difference magnitude category, is now up to 20 entries large. The extended difference magnitude table is depicted in Table D.4. The number of required entries depends on the frame precision P.

— Decoding continues with the AC coding, Rec. ITU-T T.81 | ISO/IEC 10918-1:1994, F.2.2.2, though a modified decoding procedure shall be applied that uses an enlarged Huffman alphabet that is able to express additional magnitude categories. The Huffman alphabet for large range AC coding is shown in Figure D.3. The corresponding encoder procedure that shall be matched by the decoder is shown in Figure D.4.

  Samples in the magnitude categories SSSS=16 and above do not use the combined magnitude/runlength Huffman coding employed for all other magnitude categories. Instead, a sample of this size is encoded by Huffman symbols 0x10 (16) to 0x50 (80), followed by four bits encoding the run, followed by SSSS low-order bits of the symbol to encode. The additional entries in the Huffman alphabet, see Figure D.3, are unused in the Huffman table specified in Rec. ITU-T T.81 | ISO/IEC 10918-1.

NOTE DCT-bypass coding of residual samples does **not** encode the low-order bits of the sample values for SSSS=16 since the sample is already unambiguous. This is different for large range DCT coding where low-order bits are always coded, regardless of the magnitude category.

Table D.4 — Magnitude categories and corresponding error residual ranges. For DC coding, SSSS is the Huffman symbol itself, for AC-coding it forms part of the Huffman symbol

SSSS (Symbol for Huffman coding of the magnitude category)	Error residual range
1	–1, 1
2	–3, –2, 2, 3
3	–7..–4, 4..7
4	–15..–8, 8..15
5	–31..–16, 16..31
6	–63..–32, 32..63
7	–127..–64, 64..127
8	–255..–128, 128..255
9	–511..–256, 256..511
10	–1023..–512, 512..1023
11	–2047..–1024, 1024..2047
12	–4095..–2048, 2048..4095
13	–8191..–4096, 4096..8192
14	–16383..–8192, 8192..16383
15	–32767..–16384, 16384..32767
16	–65535..–32768, 32768..65535
17	–131071..–65536, 65536..131071
18	–262143..–131072, 131072..262143

Table D.4 *(continued)*

SSSS (Symbol for Huffman coding of the magnitude category)	Error residual range
19	–524287..–262144, 262144..524287
20	–1048575..–524288, 524288..1048575
21	–1048576

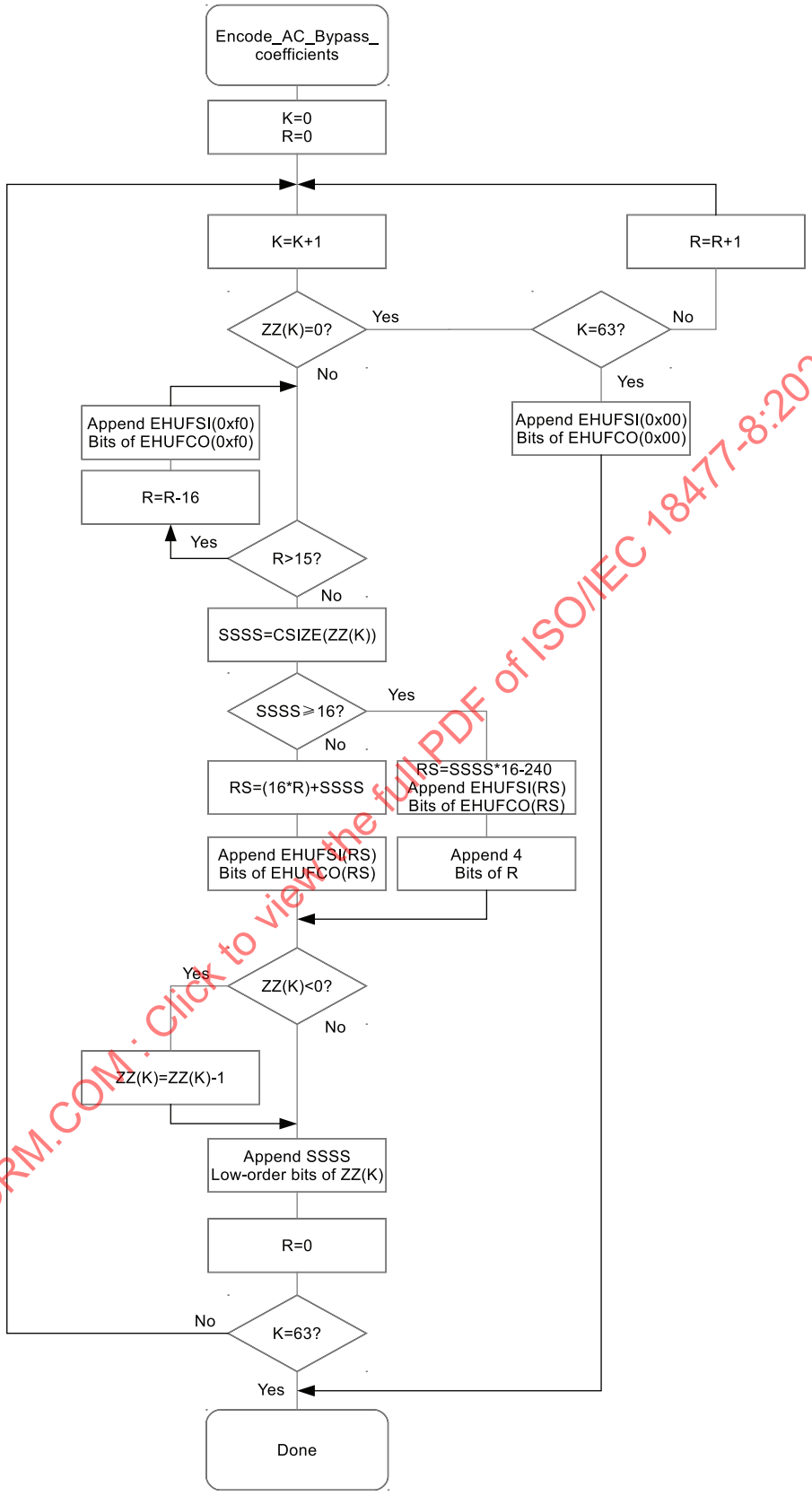

Figure D.3 — Encoding procedure for large-range coding

	SSSS =0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
R=0	EOB															
1	SSSS =16															
2	SSSS =17															
3	SSSS =18															
4	SSSS =19															
5	SSSS =20															
6	SSSS =21															
7																
8																
9																
10																
11																
12																
13																
14																
15	ZRL															

Figure D.4 — Huffman symbol allocation for the residual data coding in the large range DCT mode

NOTE Vertical: Size of the run, horizontal the magnitude category. Table cells that are crossed out are not in use.

D.3 Coding of residual error signals by the DCT-bypass scan

Entropy coding by the sequential DCT-bypass scan follows the modifications made in D.2.1 and the flowchart indicated in Figure D.2. This is, the DC coding steps in ITU-T T.81 | ISO/IEC 10918-1:1994, F.1.2.1 will be skipped. The first coefficient to be coded by the AC scan is the left-top coefficient in the block which is included in the AC coding procedure of Figure D.2. Additionally, the size category SSSS=16 will be encoded by the Huffman symbol 16, followed by direct encoding of the run-length in four bits.

Entropy coding by the progressive DCT-bypass scan follows exactly the unaltered flowcharts of Rec. ITU-T T.81 | ISO/IEC 10918-1:1994, G.3 and G.7, though the flowcharts also apply if Ss=0, i.e. the start of scan includes the DC coefficient. No specific DC coding procedure is used, and DC coding steps as indicated in ITU-T T.81 | ISO/IEC 10918-1:1994, G.1.1.2.1 and G.1.2.1 are not effective.

Entropy coding of DC coefficients in the large-range DCT process uses a larger Huffman alphabet of up to 20 symbols, but carries otherwise over from Rec. ITU-T T.81 | ISO/IEC 10918-1. Entropy coding of AC coefficients in the large-range DCT scan pattern follows Figure D.3 and applies a modified encoding procedure for coefficients of magnitude category SSSS=16 and above. The magnitude category is

determined here according to Table D.3. The enlarged Huffman alphabet for AC coding for large-range DCT coding is depicted in Figure D.4.

NOTE Progressive DCT-bypass coding is not able to encode coefficients of magnitude category SSSS=16 that can appear if the precision P of the residual frame equals 17. Similarly, regular progressive mode coding is not able to represent samples of magnitude category SSSS=16 or above. This is because the Huffman symbol selection according to Table D.3 in the legacy standard does not include the case SSSS=16 for the progressive DCT-bypass mode, or alternatively, no free space is available in the Huffman alphabet that could encode such magnitudes. The problem of having to code residuals of magnitude category SSSS=16 or above can be addressed, however, by using one additional subsequent approximation scan, limiting the magnitude category to at most SSSS=15. Alternatively, residual refinement scans can lower the required value of P and by that limit the maximum applicable category, see also D.2 and D.6. Specifically, refinement scans can entirely eliminate the need for large-range DCT scans.

D.4 Decoding of refinement scans in the DCT-bypass mode

Refinement decoding of DCT-bypassed error residual coefficients follows the algorithm as in ISO/IEC 18477-6:2016, Annex D and thus closely resembles the procedure for a subsequent scan in successive approximation coding defined in Rec. ITU-T T.81 | ISO/IEC 10918-1. However, the following modifications shall apply:

— The DC decoding step of the entire entropy decoding procedure specified in ISO/IEC 18477-6:2016, Annex D shall be skipped. That is, Rec. ITU-T T.81 | ISO/IEC 10918-1:1994, G.1.2.1 is not effective.

— The AC decoding algorithms defined in Rec. ITU-T T.81 | ISO/IEC 10918-1:1994, G.1.2.2 and G.1.2.3, specifically Figures G.3 and G.7 also apply if SS = 0, i.e. if the start of the spectral selection is zero. Otherwise, decoding proceeds with the same modifications to subsequent successive approximation coding as in ISO/IEC 18477-6:2016, Annex D.

D.5 Encoding of refinement scans in the DCT-bypass mode

Refinement encoding of DCT-bypassed error residual coefficients is defined by the flowchart in Rec. ITU-T T.81 | ISO/IEC 10918-1:1994, Figure G.7 and follows the same algorithm as defined in Rec. ITU-T T.81 | ISO/IEC 10918-1:1994, G.1.2.3. This flowchart, however, is also applicable if SS = 0, i.e. decoding includes the coefficient in the left top corner, thus the same modifications due to ISO/IEC 18477-6:2016, Annex D are in place. In addition, the following modifications apply to the encoding process:

— The DC encoding step of the entire entropy decoding procedure specified in ISO/IEC 18477-6:2016, Annex D is skipped. That is, Rec. ITU-T T.81 | ISO/IEC 10918-1:1994, G.1.2.1 is not effective.

— The AC decoding algorithms defined in Rec. ITU-T T.81 | ISO/IEC 10918-1:1994, G.1.2.2 and G.1.2.3, specifically Figures G.3 and G.7 also apply if Ss = 0, i.e. if the start of the spectral selection is zero.

D.6 Selection of the frame precision P for residual coding

This document does not require encoders to select a specific frame precision P for residual coding, provided all error residuals to be coded can be expressed with the bitrange indicated in the frame header. However, it is generally advisable to select P according to the final output bit depths of the image as identified by $8+R_b$, the number of residual refinement scans R_r and the selection of the residual transformation as specified by the Xt parameter of the Residual Transformation box. The value of R_b is defined by the Output Conversion box specified in B.2 and the value of R_r is defined by the Residual Refinement box. Table D.4 lists recommended choices.

NOTE The level shift of the DCT bypass specified in E.2 also depends on P. Selecting a P different from that recommended in Table D.5 is not advisable since it results in a DC-offset that impacts coding performance.

Table D.5 — Selection of the frame precision P for DCT-bypass frames

Value of Xt (Residual transformation box, see Table B.7)	Recommended value for P of the DCT bypass frame header
4 (RCT)	R_b+8-R_r+1 (Bit range is extended by one bit)
1 (Identity)	R_b+8-R_r
all other values reserved for ITU | ISO/IEC	

NOTE Even though the RCT extends the range of the error residuals by one bit, the sixteen magnitude categories defined by Table D.3 are sufficient for DCT bypass coding since the RCT operates with modulo arithmetic.

Annex E
(normative)

Discrete cosine transformation

E.1 General

NOTE In this annex the flowcharts and tables are normative only in the sense that they are defining an output that alternative implementations shall duplicate.

This annex defines a fixed point, an integer and a bypass version of the inverse DCT transformation. To enable lossless reconstruction of samples, the output of these processes shall be replicated exactly. The fixed point DCT is a fixed-point implementation of the DCT process defined in Rec. ITU-T T.81 | ISO/IEC 10918-1, and is within its implementation precision, identical to it. It pre-shifts its output by four bits, i.e. the output is 16 times as large as that of the DCT as specified by Rec. ITU-T T.81 | ISO/IEC 10918-1, though represented by integer values. This coefficient scaling is un-done by the multiple-component transformations specified in Annex C through the bit shift parameter R_e.

The Integer DCT is an integer-only implementation of the DCT and again identical to its legacy version within its implementation precision. Even though the fixed point DCT is not exactly invertible, it causes a predictable coding error that can be compensated for by a residual codestream. The Integer DCT is exactly invertible, but has a higher implementation complexity. That is, the integer DCT allows lossless encoding even without a residual codestream. Unlike the fixed point DCT, the Integer DCT does not scale its output.

The DCT-bypass process disables the DCT completely and only performs a level shift on all 64 inputs that is identical to the level shift of the Integer DCT. Disabling the DCT is only available as an option in the residual codestream, and there controlled by the Rdct parameter of the Residual DCT Specification box, see B.9. Samples created by the DCT bypass operation shall be coded by one of the DCT bypass scan types specified in Annex D. Similar to the Integer DCT, the DCT bypass does not scale its output.

The algorithms specified in this annex replace the level shift, DCT and quantization procedures of Rec. ITU-T T.81 | ISO/IEC 10918-1:1994, A.3.1 and F.2.1.5, A.3.3 and A.3.4 respectively, and operations specified there shall be replaced by the following algorithms and specifications.

E.2 The DCT bypass process

The DCT bypass process takes an 8×8 input block of decoded data produced by a decoding algorithm specified in Annex D $S_{i,k}$ and generates from them output samples $Y_{i,k}$.

— First dequantize the coefficients $S_{j,k}$. If the RNs flag of the Residual DCT Specification box is not set, multiply $S_{j,k}$ by $Q_{7,7}$, the bottom-right entry of the quantization matrix Q defined in a DQT marker segment:

$$R_{j,k} = S_{j,k} \times Q_{7,7}$$

— If the RNs flag of the Residual DCT Specification box (see B.9) is set, follow the noise-shaping algorithm specified in Annex G to compute $R_{j,k}$ from $S_{j,k}$.

— Second, level shift the value $R_{j,k}$ by S_{dc} to reconstruct the output value $Y_{j,k}$ of the DCT bypass process.

$$Y_{i,k} = R_{i,k} + S_{dc}$$

The level shift value S_{dc} is defined by:

$S_{dc} = 2^{P+Rr-1}$

where R_r is the number of refinement scans in the residual decoding path and is found in the Refinement Specification box and P is the bit precision of the residual codestream and defined in its start-of-frame marker, see Rec. ITU-T T.81 | ISO/IEC 10918-1:1994, Table B.2.

NOTE The DCT bypass process is never applied in the base coding path.

E.3 Fix point DCT

E.3.1 Two dimensional inverse fixed point DCT

The fixed point DCT (Figure E.1) takes integer samples $Q_{i,j}$ in the form of an 8×8 block, and generates from that integer samples $Y_{i,j}$ also organized in an 8×8 block. The input samples $Q_{i,j}$ are integers generated by dequantization and the entropy decoding procedure described in Rec. ITU-T T.81 | ISO/IEC 10918-1, the output samples $Y_{i,j}$ are integer values as well. The DC level shift specified in Rec. ITU-T T.81 | ISO/IEC 10918-1:1994, A.3.1, F.1.1.3 and F.2.1.5 are included in the DCT algorithm specified in E.3.2, thus additional level shifts as specified by Rec. ITU-T T.81 | ISO/IEC 10918-1 shall not be performed.

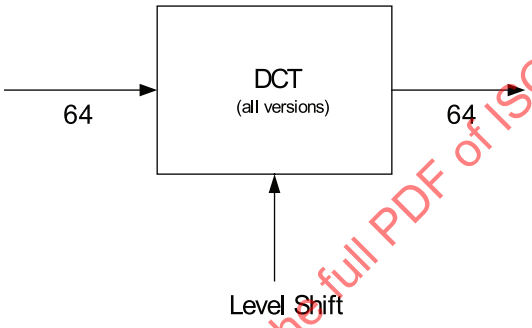

Figure E.1 — Fixed point DCT inputs and outputs

The steps of the algorithms are as follows:

— Level-shift the entropy decoded DC coefficient $Q_{0,0}$ and scale all coefficients to the precision of the decorrelation transformation; this step implements the inverse DC level shift of Rec. ITU-T T.81 | ISO/IEC 10918-1:1994, A.3.1 and F.2.1.5 as part of the DCT transformation. For that, set:

$X_{0,0} = 2^4 \times (Q_{0,0} + S_{dc} \times 2^3)$;

$X_{i,j} = 2^4 \times Q_{i,j}$ for $(i,j) \neq (0,0)$.

The values of S_{dc} shall be given by:

$S_{dc} = 2^{8+Rh--1}$

where R_h is the number of refinement scans in the base image as defined by the Refinement Specification box. If this box is not present, the inferred value of R_h shall be 0.

— Run a one dimensional inverse fixed point DCT process over all rows of the 8×8 block:

  — by setting $A_k = X_{l,k}$ where k is the column and l is the row index;

  — following the procedure of the one-dimensional inverse Fixed Point DCT process specified in E.3.2 taking the vector A as input and generating the vector B as output;

  — backwards scaling B and generating one row of the 8×8 block Z from B:

  $Z_{l,k} = \lfloor (B_k + 2^8)/2^9 \rfloor$

— repeating the above steps for all rows l∈[0,7].

— Run a one-dimensional inverse Fixed Point DCT process over all columns of the 8×8 block:

— by setting $A_k=Z_{k,j}$ where j is the column and k is the row index;

— following the procedure of the one-dimensional DCT specified in E.3.2 process taking the vector A as input and generating the vector B as output;

— backwards scaling the output B and inserting this as one column into the output matrix Y:

$$Y_{k,j} = \lfloor (B_k + 2^{8+3}) / 2^{9+3} \rfloor$$

— repeating the above steps for all columns j∈[0,7].

NOTE The DCT process defined in E.3.1 is designed such that 32-bit wide variables are sufficient to implement all steps for h ≤ 4, i.e. for at most 12-bit sample precision without causing overflows.

E.3.2 One dimensional inverse fixed point DCT

A one-dimensional inverse fixed point DCT process is a building block of the two-dimensional fixed point DCT defined in E.3.1. It takes an eight-dimensional vector A_k of integer numbers as input, and generates an eight-dimensional vector B_k of integer numbers as output.

— Set $Z_1 = (A_2 + A_6) \times 277$

— Set $T_2 = Z_1 - A_6 \times 946$

— Set $T_3 = Z_1 + A_2 \times 392$

— Set $T_0 = (A_0 + A_4) \times 512$

— Set $T_1 = (A_0 - A_4) \times 512$

— Set $T_{10} = T_0 + T_3$

— Set $T_{13} = T_0 - T_3$

— Set $T_{11} = T_1 + T_2$

— Set $T_{12} = T_1 - T_2$

— Set $Z_4 = A_7 + A_3$

— Set $Z_5 = A_5 + A_1$

— Set $Z_6 = (Z_4 + Z_5) \times 602$

— Set $Z_7 = (A_7 + A_1) \times -461$

— Set $Z_8 = (A_5 + A_3) \times -1312$

— Set $Z_9 = Z_4 \times -1004 + Z_6$

— Set $Z_{10} = Z_5 \times -200 + Z_6$

— Set $T_{30} = A_7 \times 153 + Z_7 + Z_9$

— Set $T_{31} = A_5 \times 1051 + Z_8 + Z_{10}$

— Set $T_{32} = A_3 \times 1573 + Z_8 + Z_9$

— Set $T_{33} = A_1 \times 769 + Z_7 + Z_{10}$

— Set $B_0 = T_{10} + T_{33}$

— Set $B_7 = T_{10} - T_{33}$

— Set $B_1 = T_{11} + T_{32}$

— Set $B_6 = T_{11} - T_{32}$

— Set $B_2 = T_{12} + T_{31}$

— Set $B_5 = T_{12} - T_{31}$

— Set $B_3 = T_{13} + T_{30}$

— Set $B_4 = T_{13} - T_{30}$

NOTE This algorithm implements an (unscaled) one-dimensional DCT transformation that is, by itself, not exactly invertible. The lossless process will correct errors from the forwards transformation by encoding residuals between this DCT and the ideal transformation in a residual scan.

E.4 Integer DCT

E.4.1 Two dimensional inverse integer DCT

The integer DCT is an integer to integer DCT that is, unlike the fixed point DCT, exactly invertible. If the integer DCT is combined with the inverse identity transformation as decorrelation transformation, the result is a completely lossless image coding compatible to Rec. ITU-T T.81 | ISO/IEC 10918-1 that does not require a residual coding pass. However, its compression performance is, due to a lack of a component decorrelation transformation, worse than that of fixed point DCT plus residual coding with the RCT.

The steps of the algorithms are as follows:

— Dequantize the entropy-decoded coefficients S_{uv} by multiplying them with the quantization matrix Q_{uv} from the DQT marker segment generating the dequantized DCT coefficients R_{uv}. This step implements Rec. ITU-T T.81 | ISO/IEC 10918-1:1994, A.3.4.

$$R_{uv} = S_{uv} \times Q_{uv}$$

— Level-shift the DC coefficient $Q_{0,0}$, this step implements the inverse DC level shift of Rec. ITU-T T.81 | ISO/IEC 10918-1:1994, A.3.1 and F.2.1.5 as part of the DCT transformation. For that, set:

$$X_{0,0} = Q_{0,0} + S_{dc} \times 2^3$$

$$X_{i,j} = Q_{i,j} \quad \text{for } (i,j) \neq (0,0)$$

where the value of S_{dc} is specified by Table E.1 and depends on whether the DCT is applied in the base or residual coding path.

— Run the one-dimensional Integer DCT specified in E.4.2 over all rows of the input vector, that is:

  — set $A_k = X_{l,k}$ where l is the row and k is the column index;

  — following the procedure of the one-dimensional IDCT process as defined in E.4.2 taking the vector A as input and generating the vector B as output;

  — place the output into row l of the intermediate output matrix Z:

$$Z_{l,k} = B_k \quad (k=0..7);$$

  — repeat the above steps for all rows $l \in [0,7]$.

— Run a one-dimensional Integer DCT of E.4.2 over all columns of the 8×8 block:

  — by setting $A_k = Z_{k,j}$ where k is the row and j is the column index;

- following the procedure of the one-dimensional integer DCT process of E.4.1 taking the vector A as input and generating the vector B as output;
- place the output vector B of this process unscaled into the output matrix Y:

  $Y_{k,j} = B_k$ (k=0..7);
- Repeating the above steps for all columns j∈[0,7].

— The result of this step is the inversely transformed signal.

Table E.1 — Level shift value for the IDCT

Integer DCT applied in	Value of S_{dc}
Legacy coding path	2^{8+Rh-1}
Residual coding path	2^{P+Rr-1}

In Table E.1, R_h is the number of refinement coding passes indicated in the Refinement Specification box, R_r is the number of residual refinement coding passes defined in ISO/IEC 18477-3:2015, Annex B and P is the bit precision indicated in the frame header of the residual codestream. The frame header is specified in Rec. ITU-T T.81 | ISO/IEC 10918-1:1994, Table B.2. If the Refinement Specification box is not present and hence refinement scans are not used, the inferred values of R_r and R_h shall be 0.

E.4.2 One dimensional inverse integer DCT

The one-dimensional inverse integer DCT takes an input vector A of eight integers and transforms this to an output vector B of 8 integers. The transformation requires additional multiplication steps $pmul\_tan_1$ through $pmul\_tan_4$ and $pmul\_sin_1$ through $pmul\_sin_4$ that are specified in E.4.3. The following steps shall be performed unaltered to ensure lossless reconstruction:

— Set $Z_{20} = A_0$
— Set $Z_{c0} = A_1$
— Set $Z_{21} = A_2$
— Set $Z_1 = -A_3$
— Set $Z_{10} = -A_4$
— Set $Z_{11} = -A_6$
— Set $Z_{c2} = A_7$
— Set $Z_0 = A_5 - pmul\_tan_4(Z_1)$
— Set $Z_{c3} = Z_1 + pmul\_sin_4(Z_0)$
— Set $Z_{c1} = Z_0 - pmul\_tan_4(Z_{c3})$
— Set $Z_{00} = Z_{20} - pmul\_tan_4(Z_{10})$
— Set $Z_{01} = Z_{21} - pmul\_tan_2(Z_{11})$
— Set $Z_{b1} = Z_{10} + pmul\_sin_4(Z_{00})$
— Set $Z_{b2} = Z_{11} + pmul\_sin_2(Z_{01})$
— Set $Z_{b0} = Z_{00} - pmul\_tan_4(Z_{b1})$
— Set $Z_{b2} = Z_{01} - pmul\_tan_2(Z_{b3})$
— Set $Z_{c1} = -Z_{c1}$

— Set $Z_{c0} = Z_{c0} - pmul\_tan_4(Z_{c1})$
— Set $Z_{21} = Z_{c1} + pmul\_sin_4(Z_{c0})$
— Set $Z_{20} = Z_{c0} - pmul\_tan_4(Z_{21})$
— Set $Z_{c2} = -Z_{c2}$
— Set $Z_{c3} = Z_{c3} - pmul\_tan_4(Z_{c2})$
— Set $Z_{10} = Z_{c2} + pmul\_sin_4(Z_{c3})$
— Set $Z_{11} = Z_{c3} - pmul\_tan_4(Z_{10})$
— Set $Z_{00} = Z_{20} - pmul\_tan_1(Z_{10})$
— Set $Z_{01} = Z_{21} - pmul\_tan_3(Z_{11})$
— Set $X_z = Z_{10} + pmul\_sin_1(Z_{00})$
— Set $X_6 = Z_{11} + pmul\_sin_3(Z_{01})$
— Set $X_4 = Z_{00} - pmul\_tan_1(X_7)$
— Set $X_5 = Z_{01} - pmul\_tan_3(X_6)$
— Set $Z_{b2} = -Z_{b2}$
— Set $Z_{b0} = Z_{b0} - pmul\_tan_4(Z_{b2})$
— Set $X_3 = Z_{b2} + pmul\_sin_4(Z_{b0})$
— Set $X_0 = Z_{b0} - pmul\_tan_4(X_3)$
— Set $Z_{b3} = -Z_{b3}$
— Set $Z_{b1} = Z_{b1} - pmul\_tan_4(Z_{b3})$
— Set $X_2 = Z_{b3} + pmul\_sin_4(Z_{b1})$
— Set $X_1 = Z_{b1} - pmul\_tan_4(X_2)$
— Set $X_4 = -X_4$
— Set $X_0 = X_0 - pmul\_tan_4(X_4)$
— Set $B_7 = X_4 + pmul\_sin_4(X_0)$
— Set $B_0 = X_0 + pmul\_tan_4(B_7)$
— Set $X_5 = -X_5$
— Set $X_1 = X_1 - pmul\_tan_4(X_5)$
— Set $B_6 = X_5 + pmul\_sin_4(X_1)$
— Set $B_1 = X_1 - pmul\_tan_4(B_6)$
— Set $X_6 = -X_6$
— Set $X_2 = X_2 - pmul\_tan_4(X_6)$
— Set $B_5 = X_6 + pmul\_sin_4(X_2)$
— Set $B_2 = X_2 - pmul\_tan_4(B_5)$

— Set $X_7 = -X_7$

— Set $X_3 = X_3 - \text{pmul\_tan}_4(X_7)$

— Set $B_4 = X_7 + \text{pmul\_sin}_4(X_3)$

— Set $B_3 = X_3 - \text{pmul\_tan}_4(B_4)$

E.4.3 Multiplication steps of the integer DCT

E.4.3.1 General

Eight multiplication steps are required by the IDCT algorithm specified in E.4.2. Each of the multiplication steps implements an approximation of a multiplication of its input value X by a trigonometric constant, returning an output value Y. Implementations shall follow the specified steps exactly to allow lossless reconstruction of the image. Note that the division by powers of 2 plus round-down can be implemented by right-shifts, and multiplications by powers of 2 can be implemented by left-shifts.

E.4.3.2 Specification of pmul_tan_1

An approximation of the multiplication of its input value X by $\tan(\pi/32)$, returning an output value Y is specified as:

— Set $T = X + 2 \times X$

— Set $Y = \lfloor (T + 2^4 \times X + 2^7 \times T + 2^{11})/2^{12} \rfloor$

E.4.3.3 Specification of pmul_tan_2

An approximation of the multiplication of its input value X by $\tan(\pi/16)$, returning an output value Y is specified as:

— Set $Y = \lfloor (2^6 \times X - 2^4 \times X - X + 2^8 \times X + 2^9 \times X + 2^{11})/2^{12} \rfloor$

E.4.3.4 Specification of pmul_tan_3

An approximation of the multiplication of its input value X by $\tan(3\pi/32)$, returning an output value Y is specified as:

— Set $T = X + 2 \times X$

— Set $Y = \lfloor (T + 2^3 \times T + 2^6 \times T + 2^{10} \times X + 2^{11})/2^{12} \rfloor$

E.4.3.5 Specification of pmul_tan_4

An approximation of the multiplication of its input value X by $\tan(\pi/8)$, returning an output value Y is specified as:

— Set $Y = \lfloor (2^5 \times X - X + 2^8 \times X + 2^9 \times X + 2^{11})/2^{12} \rfloor$

E.4.3.6 Specification of pmul_sin_1

An approximation of the multiplication of its input value X by $\sin(\pi/16)$, returning an output value Y is specified as:

— Set $Y = \lfloor (2^5 \times X - X + 2^8 \times X + 2^9 \times X + 2^{11})/2^{12} \rfloor$

E.4.3.7 Specification of pmul_sin$_2$

An approximation of the multiplication of its input value X by sin(π/8), returning an output value Y is specified as:

— Set Y = $\lfloor(2^5 \times X - X + 2^9 \times X + 2^{10} \times X + 2^{11})/2^{12}\rfloor$

E.4.3.8 Specification of pmul_sin$_3$

An approximation of the multiplication of its input value X by sin(3π/16), returning an output value Y is specified as:

— Set Y = $\lfloor(2^8 \times X - 2^5 \times X + 2^2 \times X + 2^{11} \times X + 2^{11})/2^{12}\rfloor$

E.4.3.9 Specification of pmul_sin$_4$

An approximation of the multiplication of its input value X by sin(π/4), returning an output value Y is specified as:

— Set T = $X + 2^2 \times X$

— Set Y = $\lfloor(2^4 \times X + 2^6 \times T + 2^9 \times T + 2^{11})/2^{12}\rfloor$

E.5 The forward DCT bypass process

The forward DCT bypass process takes an 8×8 input block of source signals $Y_{i,k}$ and generates from it output samples $S_{i,k}$ that are level-shifted by S_{dc} and quantized. The generated values then form the input of the entropy coding algorithm specified in Annex D. The DC shift is as in E.2.

— First level-shift the data by S_{dc}, where S_{dc} is specified in E.2:

$$R_{i,k} = Y_{i,k} - S_{dc}$$

— Second, quantize the data $R_{j,k}$. If the RNs flag of the Residual DCT box is not set, quantization is performed by division of $R_{j,k}$ by $Q_{7,7}$, where Q is the quantization matrix defined by a DQT marker segment:

$$S_{j,k} = \lfloor R_{j,k}/Q_{7,7} + 0.5\rfloor$$

— If the RNs flag of the Residual DCT box is set, follow the noise shaping procedure of Annex G to compute the coefficients $S_{j,k}$

E.6 Forward fixed point DCT

E.6.1 General

An overview and implementation guideline for a forwards reversible fixed point DCT is provided in E.6. Many other implementation choices are possible, even for the lossless profile, provided that the residual is computed according to inverse fixed point DCT specified above.

The forwards fixed point DCT transform takes an 8×8 block of sample values $Y_{i,j}$ as input and generates from that a DCT transformed signal $X_{i,j}$ which is also organized in an 8×8 block. Level shifting as specified in Rec. ITU-T T.81 | ISO/IEC 10918-1:1994, A.3.1 and F.1.1.3 is included in the algorithm.

The forwards fixed point DCT consists of the following steps:

— Extract column k from the source signal $Y_{k,l}$ and denote this column-vector by B_l:

$$B_k = Y_{k,l}$$