International Standard

ISO/IEC 18477-1

Third edition
2024-05

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

Part 1: Core coding system specification

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

Partie 1: Spécification du système de codage de noyau

Reference number
ISO/IEC 18477-1:2024(en)

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 or www.iec.ch/members_experts/refdocs).

ISO and IEC draw attention to the possibility that the implementation of this document may involve the use of (a) patent(s). ISO and IEC take no position concerning the evidence, validity or applicability of any claimed patent rights in respect thereof. As of the date of publication of this document, ISO and IEC had not received notice of (a) patent(s) which may be required to implement this document. However, implementers are cautioned that this may not represent the latest information, which may be obtained from the patent database available at www.iso.org/patents and https://patents.iec.ch. ISO and IEC shall not be held responsible for identifying any or all such patent rights.

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. In the IEC, see www.iec.ch/understanding-standards.

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 third edition cancels and replaces the second edition (ISO/IEC 18477-1:2020), which has been technically revised.

The main changes are as follows:

— the marker ID for the component decorrelation control marker was corrected.

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

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 and www.iec.ch/national-committees.

Introduction

This document specifies a coded codestream format for storage of continuous-tone photographic content. JPEG XT is a scalable image coding system that builds on the legacy Rec. ITU-T T.81 | ISO/IEC 10918-1 coding system, also known as JPEG, but extends it in a backwards compatible way. This document specifies the commonly deployed components of the JPEG coding system. Additional parts of the ISO/IEC 18477 series extend on this baseline.

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, Rec. ITU-T T.86 | ISO/IEC 10918-4 and Rec. ITU-T T.871 | ISO/IEC 10918-5 to extend the functionality of the legacy JPEG coding system. It is optimized for good image quality and compression efficiency while also enabling low-complexity encoding and decoding implementations.

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

Part 1:
Core coding system specification

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 the core coding system, which forms the basis for the entire ISO/IEC 18477 series.

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.

Rec. ITU-T T.81 | ISO/IEC 10918-1:1994, *Information technology — Digital compression and coding of continuous-tone still images — Part 1: Requirements and guidelines*

Rec. ITU-T T.86 | ISO/IEC 10918-4, *Information technology — Digital compression and coding of continuous-tone still images — Part 4: Registration of JPEG profiles, SPIFF profiles, SPIFF tags, SPIFF colour spaces, APPn markers, SPIFF compression types and Registration Authorities (REGAUT)*

Rec. ITU-T T.871 | ISO/IEC 10918-5, *Information technology — Digital compression and coding of continuous-tone still images — Part 5: JPEG File Interchange Format (JFIF)*

3 Terms and definitions

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

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

— ISO Online browsing platform: available at https://www.iso.org/obp

— IEC Electropedia: available at https://www.electropedia.org/

3.1
bitstream
partially encoded or decoded sequence of bits comprising an entropy-coded segment

3.2
block
8×8 array of samples or an 8×8 array of DCT coefficient values of one component

3.3
byte
group of 8 bits

3.4
coder
embodiment of a coding process

3.5
coding
encoding or decoding

3.6
compression
reduction in the number of bits used to represent source image data

3.7
component
two-dimensional array of samples having the same designation in the output or display device

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

3.8
continuous-tone image
image whose components have more than one bit per sample

3.9
discrete cosine transform
DCT
either the forward discrete cosine transform or the inverse discrete cosine transform

3.10
downsampling
procedure by which the spatial resolution of a component is reduced

3.11
entropy-coded data segment
independently decodable sequence of entropy encoded bytes of compressed image data

3.12
marker
two-byte code in which the first byte is hexadecimal FF and the second byte is a value between 1 and hexadecimal FE

3.13
marker segment
marker and associated set of parameters

3.14
precision
number of bits allocated to a particular sample or DCT coefficient

3.15
procedure
set of steps which accomplishes one of the tasks which comprise an encoding or decoding process

3.16
sample
one element in the two-dimensional array which comprises a component

3.17
sample grid
common coordinate system for all samples of an image with the samples at the top left edge of the image having the coordinates (0, 0), the first coordinate increases towards the right, the second to the bottom

3.18
scan
single pass through the data for one or more of the components in an image

3.19
scan header
marker segment that contains a start-of-scan marker and associated scan parameters that are coded at the beginning of a scan

3.20
upsampling
procedure by which the spatial resolution of a component is increased

3.21
vertical sampling factor
relative number of vertical data units of a particular component with respect to the number of vertical data units in the other components in the frame

4 Symbols and abbreviated terms

4.1 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

4.2 Abbreviated terms

ASCII	American Standard Code for Information Interchange
DC	lowpass
AC	highpass
LSB	least significant bit
MSB	most significant bit
DCT	discrete cosine transformation
JPEG	joint photographic experts group

5 Conventions

5.1 Conformance language

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

5.2 Operators

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

5.2.1 Arithmetic operators

\+ addition

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

× multiplication

/ division without truncation or rounding

5.2.2 Assignment operators

= assignment operator

5.2.3 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
+, −	addition and subtraction	left to right
<, >, <=, >=	relational	left to right

5.2.4 Mathematical functions

$\lceil x \rceil$ Ceiling of x. Returns the smallest integer that is greater than or equal to x.

$\lfloor x \rfloor$ Floor of x. Returns the largest integer that is lesser than or equal to x.

$|x|$ Absolute value, is −x for x < 0, otherwise x.

sign(x) Sign of x, zero 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.

6 General

6.1 General definitions

The purpose of this clause is to give an informative overview of the elements specified in this document.

There are three elements specified in this document:

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

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

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

6.2 Functional overview on the decoding process

The high-level algorithm for decoding is as follows: The samples are first reconstructed following the decoder specifications defined in Rec. ITU-T T.81 | ISO/IEC 10918-1. If the resulting component arrays are subsampled, they are upsampled on a common sample grid following the specifications in Annex A. Following that, the output data is processed by an inverse decorrelation transformation. If the data is already in an RGB type colour space, e.g. RGB with ITU-R Rec. BT.601 primaries, this transformation will be the identity transformation. Otherwise, the ICT is used to transform the data into RGB. The inverse decorrelation transformation is defined in Annex C, and the markers that are required to select the transformation are defined in Annex B.

6.3 Encoder requirements

An encoding process converts source image data to compressed image data. This includes first obtaining a low dynamic range image, and representing it by a coding process specified in Rec. ITU-T T.81 | ISO/IEC 10918-1:1994, Annex F or Annex G.

In order to comply with this document, an encoder shall satisfy at least one of the following two requirements. An encoder shall with appropriate accuracy, convert source image data to compressed image data which comply with the codestream format syntax specified in Annex B for the encoding process(es) embodied by the encoder. A limited accuracy sufficient to match the error bounds specified in the compliance tests is acceptable.

There is no requirement in this document that any encoder which embodies one of the encoding processes specified here shall be able to operate for all ranges of the parameters which are allowed for that process. An encoder is only required to meet the compliance tests and to generate the compressed data format according to Annex B for those parameter values which it does use.

6.4 Decoder requirements

A decoding process converts compressed image data to reconstructed image data. For that, it has to follow the decoding operation specified in Rec. ITU-T T.81 | ISO/IEC 10918-1 with sufficient accuracy, using either the baseline, sequential or progressive scan process defined in Rec. ITU-T T.81 | ISO/IEC 10918-1:1994, Annex F or Annex G. This process generates sample values on a sample grid, which are then converted into a digital image by following the upsampling specifications in Annex B of this document and the multi-component decorrelation (ICT) process in Annex C of this document.

Annex A
(normative)

Component subsampling and expansion of subsampling

A.1 General

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

A.2 Component dimensions and subsampling factors

An image is defined to consist of Nf components, each of which is identified by a unique identifier C_i defined in the frame header of the codestream format specified in Annex B. The number of components Nf shall be either 1 or 3. A component consists of a rectangular array of samples x_i wide and y_i samples high. The component dimensions are derived from the image dimensions X and Y, also parameters recorded in the frame header. These two parameters define a sample grid of X grid points wide and Y grid points high, where the left topmost grid coordinate is (0, 0) and coordinates increase from left to right and from top to bottom. However, the dimensions of the component do not need to coincide with the dimensions of the image. For each component, two subsampling factors $s_{i,x}$ and $s_{i,y}$ define the spacing between sample points of component i relative to the sample grid and the size of the component array. If X and Y are the dimensions of the sample grid, the size of component i with subsampling factors $s_{i,x}$ and $s_{i,y}$ is

$$\lceil X/s_{i,x} \rceil \text{ and } \lceil Y/s_{i,y} \rceil$$

Upsampling by interpolation from surrounding samples as specified in Annex A generates then sample values on all grid points of the sample grid.

The subsampling factors $s_{i,x}$ and $s_{i,y}$ are not directly represented in the binary codestream or any of its markers, but shall be derived from the parameters H_i and V_i recorded in the frame header. If Nf equals 1, i.e. the image consists of a single component, H_1 and V_1 shall be 1, and $s_{1,x}$ and $s_{1,y}$ are both 1. If Nf equals 3, Table A.1 defines the relation between H_i, V_i and $s_{i,x}$ and $s_{i,y}$. No other combinations of H_i and V_i than those listed in Table A.1 shall be used.

Table A.1 — Sampling values

H_1	V_1	H_2	V_2	H_3	V_3	$s_{1,x}$	$s_{1,y}$	$s_{2,x}$	$s_{2,y}$	$s_{3,x}$	$s_{3,y}$
1	1	1	1	1	1	1	1	1	1	1	1
2	2	2	1	2	1	1	1	1	2	1	2
2	2	1	2	1	2	1	1	2	1	2	1
2	2	1	1	1	1	1	1	2	2	2	2
All other values reserved for ITU/ISO purposes.											

NOTE Rec. ITU-T T.81 | ISO/IEC 10918-1 allowed other component arrangements and relations between grid positions and sample positions that are not valid in this document. However, the definitions given here are special cases of the more general relations provided in Rec. ITU-T T.81 | ISO/IEC 10918-1 and both definitions agree whenever both are defined.

A.3 Expansion of subsampled components

Whenever the subsampling factors $s_{i,x}$ and $s_{i,y}$ are not both 1, interpolation is used to populate all grid positions of the image sample grid. The following bilinear interpolation algorithm can be used to provide sample values at all sampling grid positions. Readers should be aware that the algorithm described here will also change the sample values at sampling grid positions whose values are represented in the codestream. This may have the effect of a continuous loss of precision of the subsampled components over multiple compression-decompression cycles.

A.4 Bilinear expansion of subsampled components

Upsampling is performed in two steps. First, upsampling in the vertical direction if $s_{i,y}$ is 2, generating an intermediate image. Second, upsampling in the horizontal direction if $s_{i,x}$ is 2, generating the final output image from the intermediate image.

In a first step, check for each component i whether $s_{i,y}$ is 2 or 1. If $s_{i,y}$ is 1, copy the reconstructed samples to the intermediate image $v^{(up,y)}$ directly. Otherwise, compute the intermediate image $v^{(up,y)}$ from the reconstructed samples v by first setting $v_{x,-1}$ to $v_{x,0}$ and $v_{x,\lceil Y/2\rceil}$ to $v_{x,\lceil Y/2\rceil -1}$, and then set for all x such that $0 \le x < X$ and all y such such that $0 \le y < \lceil Y/2\rceil$:

$$v^{(up,y)}_{x,2y} = \lfloor (v_{x,y-1}+3\times v_{x,y}+1 + (x \bmod 2))/4 \rfloor$$

$$v^{(up,y)}_{x,2y+1} = \lfloor (v_{x,y+1}+3\times v_{x,y}+2 - (x \bmod 2))/4 \rfloor$$

The outputs $v^{(up,y)}_{x,2\lceil Y/2\rceil +1}$ are discarded if the image height Y is odd.

In a second step, check for each component i whether $s_{i,x}$ is 2 or 1. If $s_{i,x}$ is 1, copy the intermediate image to the output image directly. Otherwise, compute the output image $v^{(up,x,y)}$ from the intermediate image $v^{(up,y)}$ by first setting $v^{(up,y)}_{-1,y}$ to $v^{(up,y)}_{0,y}$ and $v^{(up,y)}_{\lceil X/2\rceil,y}$ to $v^{(up,y)}_{\lceil X/2\rceil -1,y}$ and then set for all y such that $0 \le y < Y$ and all x such such that $0 \le x < \lceil X/2\rceil$:

$$v^{(up,x,y)}_{2x,y} = \lfloor (v^{(up,y)}_{x-1,y}+3\times v^{(up,y)}_{x,y}+2)/4 \rfloor$$

$$v^{(up,x,y)}_{2x+1,y} = \lfloor (v^{(up,y)}_{x+1,y}+3\times v^{(up,y)}_{x,y}+1)/4 \rfloor$$

The outputs $v^{(up,x,y)}_{2\lceil X/2\rceil +1,y}$ are discarded if the image width X is odd.

A.5 Downsampling of components

This document does not define a normative procedure by which the resolution of components whose si, x and si, y factors are not both one shall be reduced. Any procedure that generates components of the size $\lceil X/si,x\rceil$ and $\lceil Y/si,y\rceil$ is acceptable as long as it is compatible with the upsampling procedure defined above. A very simple downsampling filter is given in the next subclause.

A.6 Downsampling by a box filter

The box filter is the simplest possible downsampling filter and provides only poor quality. Even though better alternatives exist, the box filter is nevertheless presented here as an example. The input of the box filter is a X × Y component array of samples, where the sample value at position x, y is denoted by $v_{x,\,y}$.

$$x_{min}:=s_{i,\,x}\times x_s \ x_{max}:=\min(s_{i,\,x}\times x_s+s_{i,\,x}-1, X-1) \ y_{min}:= s_{i,\,y}\times y \ y_{max}:=\min(s_{i,\,y}\times y_s+s_{i,\,y}-1, Y-1)$$

The output of the box filter at position x_s, y_s is then defined as:

$$v^s_{x,\,y}:=(\Sigma_{x=xmin..xmax}\ \Sigma_{y=xmin..ymax} v_{x,\,y})/((x_{max}-x_{min}-1) \times (y_{max}-y_{min}-1))$$

i.e. the average over the box x_{min}, y_{min} to x_{max}, y_{max}. The array of downsampled sample values $v^s_{x,\,y}$ is then subject to further processing, e.g. DCT transformation and entropy coding.

Annex B
(normative)

Codestream syntax

B.1 General

This annex defines the compressed bitstream syntax which, structurally, consists of an ordered collection of marker segments and entropy coded data segments. Marker segments specify parameters necessary to reconstruct the sample values from the entropy coded data segments. Because all of these constituent parts are represented with byte-aligned codes, each compressed data format consists of an ordered sequence of 8-bit bytes. For each byte, a most significant bit (MSB) and a least significant bit (LSB) are defined.

NOTE The codestream syntax defined here agrees mostly with the "interchange format" defined in Rec. ITU-T T.81 | ISO/IEC 10918-1, with some additional constraints on the parameters in the marker segments and some additional markers carrying information that is irrelevant for the older standard.

B.2 Parameters

Parameters are unsigned integers, with values specific to the encoding process, source image characteristics, and other features selectable by the application. Parameters are assigned either 4-bit, 1-byte, 2-byte or 4-byte codes. Except for certain optional groups of parameters, parameters encode critical information without which the decoding process cannot properly reconstruct the image. The code assignment for a parameter shall be an unsigned integer of the specified length in bits with the particular value of the parameter.

For parameters which are 2 bytes (16 bits) in length, the most significant byte shall come first in the compressed data's ordered sequence of bytes. The same holds for parameters that are 4 bytes (32 bit) in length, where bits are ordered in the codestream in decreasing significance. Parameters which are 4 bits in length always come in pairs, and the pair shall always be encoded in a single byte. The first 4-bit parameter of the pair shall occupy the most significant 4 bits of the byte. Within any 32-, 16-, 8-, or 4-bit parameter, the MSB shall come first and LSB shall come last. This encoding is commonly known as "big endian" representation of unsigned integers.

B.3 Markers

Markers serve to identify the various structural parts of the compressed data formats. Most markers start marker segments containing a related group of parameters; some markers stand alone. All markers are assigned two-byte codes: an 0xFF byte followed by a byte which is not equal to 0x00 or 0xFF. Any marker may optionally be preceded by any number of fill bytes, which are bytes of the value 0xFF.

NOTE Because of this special code-assignment structure, markers make it possible for a decoder to parse the compressed data and locate its various parts without having to decode other segments of image data.

B.4 Marker assignments

All markers are assigned two-byte codes. The byte codes are specified in Table B.1 for each defined marker. Most of the marker segments used by this document are defined in Rec. ITU-T T.81 | ISO/IEC 10918-1, though some marker segments defined there shall not be used in this document. For completeness, these markers are also included in Table B.1. Furthermore, care must be taken that the parameters of some markers are more constrained than in Rec. ITU-T T.81 | ISO/IEC 10918-1, and the meaning of several markers changed slightly. Such changes are also indicated in Table B.1, and a full specification of the changes follows.

Table B.1 — Markers and marker segments

Code assignment	Symbol	Description	Defined in
Start of frame markers valid in this document			
0xFFC0	SOF_0	Baseline DCT process	This document
0xFFC1	SOF_1	Extended sequential DCT process limited to 8 bit sample precision (extended sequential, Huffman coding, 8-bit sample precision)	
0xFFC2	SOF_2	Progressive DCT process limited to 8 bit sample precision (spectral selection only, Huffman coding, 8-bit sample precision or full progression, Huffman coding, 8-bit sample precision)	
Start of frame markers defined in Rec. ITU-T T.81 | ISO/IEC 10918-1 that shall not be used in this document			
0xFFC3	SOF_3	Additional start of frame markers, shall not be used.	Rec. ITU-T T.81 | ISO/IEC 10918-1
0xFFC5	SOF_5		
0xFFC6	SOF_6		
0xFFC7	SOF_7		
0xFFC9	SOF_9		
0xFFCA	SOF_{10}		
0xFFCB	SOF_{11}		
0xFFCD	SOF_{12}		
0xFFCE	SOF_{14}		
0xFFCF	SOF_{15}		
Table specifications valid in this document			
0xFFC4	DHT	Define Huffman tables	Rec. ITU-T T.81 | ISO/IEC 10918-1
0xFFDB	DQT	Define quantization tables	Rec. ITU-T T.81 | ISO/IEC 10918-1
Restart interval termination valid in this document			
0xFFD0 through 0xFFD7	RST_m^*	Restart modulo 8 count 'm'	Rec. ITU-T T.81 | ISO/IEC 10918-1
0xFFDD	DRI	Define restart interval	Rec. ITU-T T.81 | ISO/IEC 10918-1
Other markers valid in this document			
0xFFD8	SOI*	Start of image	Rec. ITU-T T.81 | ISO/IEC 10918-1
0xFFD9	EOI*	End of image	
0xFFDA	SOS	Start of scan	
0xFFFE	COM	Comment	
JPEG extension markers valid in this document			
0xFFE0	APP_0	JFIF application marker	Rec. ITU-T T.86 | ISO/IEC 10918-4 and Rec. ITU-T T.871 | ISO/IEC 10918-5
0xFFE1	APP_1	EXIF application marker	Rec. ITU-T T.86 | ISO/IEC 10918-4

Table B.1 *(continued)*

Code assignment	Symbol	Description	Defined in
0xFFE2	APP_2	ICC profile marker	Rec. ITU-T T.86 | ISO/IEC 10918-4 and Rec. ITU-T T.871 | ISO/IEC 10918-5
0xFFE3 through 0xFFE8	APP_3 through APP_{10}	Application markers 3 through 10	Rec. ITU-T T.86 | ISO/IEC 10918-4
0xFFEB	APP_{11}	JPEG XT marker	This document
0xFFEC through 0xFFED	APP_{12} through APP_{13}	Application markers 12 and 13	Rec. ITU-T T.86 | ISO/IEC 10918-4
0xFFEE	APP_{14}	Component decorrelation control marker	This document
0xFFEF	APP_{15}	Application marker 15	Rec. ITU-T T.86 | ISO/IEC 10918-4
Markers defined in other specifications that shall not be used in this document			
0xFFC8	JPG	Reserved for JPEG extensions	Rec. ITU-T T.81 | ISO/IEC 10918-1
0xFFCC	DAC	Define arithmetic coding conditions	
0xFFDC	DNL	Define number of lines	
0xFFDE	DHP	Define hierarchical progression	
0xFFDF	EXP	Expand reference components	
All other values		Reserved for ITU/ISO purposes, shall not be used in this document	
An asterisk (*) indicates a marker which stands alone, that is, which is not the start of a marker segment.			

B.5 Marker segments

A marker segment consists of a marker followed by a sequence of related parameters. The first parameter in a marker segment is the two-byte length parameter. This length parameter encodes the number of bytes in the marker segment, including the length parameter and excluding the two-byte marker. The marker segments identified by the SOF and SOS marker codes are referred to as headers: the frame header and the scan header respectively.

B.6 Entropy-coded data segments

An entropy-coded data segment contains the output of an entropy-coding procedure. It consists of an integer number of bytes created by the Huffman coding procedure. An entropy coded data segment stands for itself and is started by an SOS marker.

Entropy coded segments shall be always a multiple of 8 bits long and shall be padded by zero or more bits of value 1 at the end of the segment to fill an entire byte. To ensure that a marker does not occur within an entropy-coded segment, the encoder shall insert a 0 byte into the output bitstream following any 0xff byte generated by either the Huffman encoder or by the above termination algorithm. Decoders shall remove such "stuffed" zero bytes before feeding the resulting bitstream into the Huffman decoder.

B.7 High-level syntax

The high-level syntax of the codestream defined in this document shall follow the syntax of the interchange format specified in Rec. ITU-T T.81 | ISO/IEC 10918-1:1994, B.2 and B.3 where only the subset of the markers specified in Table B.1 shall be used.

Specifically, conforming codestreams shall start with an SOI marker, followed by a single frame, followed by a single EOI marker. The high level syntax is depicted in Table B.2.

Table B.2 — High-level syntax of the codestream

SOI Marker	Frame	EOI

NOTE Rec. ITU-T T.81 | ISO/IEC 10918-1 allowed multiple frames for the hierarchical coding mode, though this mode is not applicable in this document.

A frame consists of zero or more table definitions containing the coding parameters of the image data, followed by the frame header, followed by one or more scans over the image. The scan type is indicated by the type of the SOF (start of frame) marker starting the frame header, see Table B.1. The frame header syntax is defined in B.2. The syntax of a frame is indicated in Table B.3.

Table B.3 — Syntax of a frame

[Tables/JPEG extensions markers/misc]	Frame header	Scan_1	...	$\text{Scan}_{\text{Last}}$

The tables consist of the following marker segments:

— zero or more DHT marker segments defining the Huffman code required for decoding the entropy coded segment,

— zero or more DQT marker segments defining the quantization matrix required to reconstruct the DCT coefficients,

— zero or one DRI marker segments defining the restart interval,

— zero or one component decorrelation control markers, defining the inverse multi-component decorrelation transformation for the bitstream,

— zero or more application specific data encoded in APP_n markers, defined in other parts of the ISO/IEC 18477 series.

— zero or more COM markers including additional codestream comments.

Unlike Rec. ITU-T T.81 | ISO/IEC 10918-1, DNL markers shall not be used, and are not valid in this document. The size of the image is thus to be completely defined by the frame header.

A scan consists of zero or more table definitions including coding parameters, followed by a scan header starting with an SOS marker, and one or more entropy coded segments. If restart markers are disabled, a scan shall only contain a single entropy coded segment. Otherwise, the scan may contain multiple entropy coded segments separated by restart markers. The syntax of the restart markers shall follow the definitions found in Rec. ITU-T T.81 | ISO/IEC 10918-1. The syntax of a scan is indicated in Table B.4.

There shall be at most one DQT table for each table destination in the codestream, replacing an already defined quantization matrix by inserting a DQT marker segment with a table destination already defined earlier in the codestream is invalid. Furthermore, the codestream shall include sufficient information to be self-contained, i.e. to allow reconstruction of the sample values with information contained in the codestream only.

Additional requirements may apply to other marker segments found in the tables/extensions/misc segment–they are defined together with the corresponding marker segment in this Annex.

Table B.4 — Syntax of a scan

[Tables/JPEG extensions markers/misc]	Scan header	ECS_0	RST_0	...	ECS_{Last-1}	RST_{Last}	ECS_{Last}

B.8 Frame header syntax

Table B.5 specifies the frame header which shall be present at the start of a frame. Frame headers always start with a SOF_n marker, where n defines the type of the scans contained in the frame. See Table B.1 for scan types supported by this document.

The frame header specifies the dimensions of the image, the components in the frame, and the sampling factors for each component, and specifies the destinations from which the quantized tables to be used with each component are retrieved.

Table B.5 — Frame header parameters and values

Parameter	Size (in bits)	Value	Meaning
SOF_n	16	0xFFC0 through 0xFFC2	Start of frame marker, indicating this marker segment and the type of the scans
Lf	16	8+3*Nf	Size of the marker segment not including the marker
P	8	8	Precision of the low dynamic range representation of the image contained in the codestream
Y	16	1-65535	Height of the sample grid in lines
X	16	1-65535	Width of the sample grid in lines
Nf	8	1 or 3	Number of visible components in the image
C_i	8	0-255	Component identifiers
H_i	4	1 or 2	Specification of the component sampling factors, see Table A.1 for allowable values and the relation between H_i V_i and $s_{i,x}$ and $s_{i,y}$
V_i	4	1 or 2	
Tq_i	8	0-3	Quantization table destination selector

NOTE The frame header syntax defined here is identical to the syntax defined in Rec. ITU-T T.81 | ISO/IEC 10918-1 except that some parameters are restricted. Specifically, the bit precision of the components is always 8, the height is greater than 0, indicating that the number of lines of the image is known, the number of components is either 1 or 3 and the horizontal and vertical sampling factors are constrained to the values listed in Table A.1.

B.9 Scan header syntax

The scan header defines the parameters necessary to decode a single scan over the image. For the baseline or sequential frame syntax, this is the only scan that represents the image. Multiple scans are possible in the progressive mode.

The syntax of the scan header follows the definition for the same marker in Rec. ITU-T T.81 | ISO/IEC 10918-1, though some of the parameters are here more constrained than in Rec. ITU-T T.81 | ISO/IEC 10918-1. The parameters and sizes of the scan header are defined in Table B.6.

Table B.6 — Scan header parameters and values

Parameter	Size (in Bits)	Values			Meaning
		Baseline	Extended Sequential	Progressive	
SOS	16	0xFFDA			SOS marker, identifies the start of the scan
Ls	16	6+2×Ns			Size of the marker segment (not including the marker)
Ns	8	1 or 3		1 or 3 (if the progressive scan is a subsequent scan and includes AC components, only 1 is allowed here)	Number of components in the scan
Cs_j	8	0-255 (shall be a subset of the component identifiers defined in the frame header)			Component identifiers of the components defined by the scan
Td_j	4	0-1	0-3		Huffman DC table specification
Ta_j	4	0-1	0-3		Huffman AC table specification
Ss	8	0		0-63	Start of spectral selection
Se	8	63		Ss-63	End of spectral selection
Ah	4	0		0-9 If nonzero, must be equal to Al+1	Successive approximation high bit position
Al	4	0		0-8	Successive approximation low bit position

NOTE The scan header syntax defined here is identical to the syntax defined in Rec. ITU-T T.81 | ISO/IEC 10918-1, though a scan contains at most three components.

B.10 Component decorrelation control marker

The APP_{14} marker controls whether the multi-component decorrelation transformation described in Annex C is applied to the components before subsampling and entropy coding. This transformation typically improves the compression performance by representing data in an RGB type colourspace, e.g. RGB with primaries defined by ITU-R Rec. BT.601 in a luma/chroma format that is close to YCbCr, e.g. ITU-R Rec. BT.601. This marker shall only be inserted into the codestream as parts of the table/misc segment if the number of components in the frame (Nf) is 3, and there shall be at most 1 such marker in the codestream. The marker parameters and sizes are defined in Table B.8, the organization of the marker is shown in Table B.7.

NOTE This marker is also listed in Rec. ITU-T T.86 | ISO/IEC 10918-4, though the possible values of the parameters inside the marker segment are more restricted.

Table B.7 — Organization of the component decorrelation control marker

0xFFEE	La	ACid	ver	f1	f2	cc