INTERNATIONAL STANDARD

ISO/IEC 15444-10

Second edition
2011-12-15

Information technology — JPEG 2000 image coding system: Extensions for three-dimensional data

Technologies de l'information — Système de codage d'images JPEG 2000: Extensions pour données tridimensionnelles

Reference number
ISO/IEC 15444-10:2011(E)

ISO copyright office
Case postale 56 • CH-1211 Geneva 20
Tel. + 41 22 749 01 11
Fax + 41 22 749 09 47
E-mail copyright@iso.org
Web www.iso.org

Published by ISO in 2012

Published in Switzerland

CONTENTS

Foreword

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

International Standards are drafted in accordance with the rules given in the ISO/IEC Directives, Part 2.

The main task of the joint technical committee is to prepare International Standards. Draft International Standards adopted by the joint technical committee are circulated to national bodies for voting. Publication as an International Standard requires approval by at least 75 % of the national bodies casting a vote.

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.

ISO/IEC 15444-10 was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information technology*, Subcommittee SC 29, *Coding of audio, picture, multimedia and hypermedia information*, in collaboration with ITU-T. The identical text is published as ITU-T Rec. T.809.

This second edition cancels and replaces the first edition (ISO 15444-8:2008), which has been technically revised.

ISO/IEC 15444 consists of the following parts, under the general title *Information technology — JPEG 2000 image coding system*:

— *Part 1: Core coding system*

— *Part 2: Extensions*

— *Part 3: Motion JPEG 2000*

— *Part 4: Conformance testing*

— *Part 5: Reference software*

— *Part 6: Compound image file format*

— *Part 8: Secure JPEG 2000*

— *Part 9: Interactivity tools, APIs and protocols*

— *Part 10: Extensions for three-dimensional data*

— *Part 11: Wireless*

— *Part 12: ISO base media file format*

— *Part 13: An entry level JPEG 2000 encoder*

INTERNATIONAL STANDARD
RECOMMENDATION ITU-T

Information technology – JPEG 2000 image coding system: Extensions for three-dimensional data

1 Scope

This Recommendation | International Standard is a work item subdivision of ISO/IEC 15444 that provides extensions of Rec. ITU-T T.800 | ISO/IEC 15444-1 and Rec. ITU-T T.801 | ISO/IEC 15444-2 for logically cuboidal data sets. In particular, it respects all existing capabilities and syntax of Rec. ITU-T T.800 | ISO/IEC 15444-1 and part of the existing capabilities of Rec. ITU-T T.801 | ISO/IEC 15444-2 for multi-component images, while providing alternatives and extensions to some of those capabilities. Within these constraints, it provides an isotropic specification for three-dimensional data sets, i.e., the project provides identical processing capabilities in all three dimensions even though Rec. ITU-T T.800 | ISO/IEC 15444-1 and Rec. ITU-T T.801 | ISO/IEC 15444-2 codestream syntax differentiates between the two spatial axes and the cross-component axis. The context models currently used in this Recommendation | International Standard are as in Rec. ITU-T T.800 | ISO/IEC 15444-1 and Rec. ITU-T T.801 | ISO/IEC 15444-2. Improved context models will be introduced through an amendment.

2 Normative references

The following Recommendations and International Standards contain provisions which, through reference in this text, constitute provisions of this Recommendation | International Standard. At the time of publication, the editions indicated were valid. All Recommendations and Standards are subject to revision, and parties to agreements based on this Recommendation | International Standard are encouraged to investigate the possibility of applying the most recent edition of the Recommendations and Standards listed below. Members of IEC and ISO maintain registers of currently valid International Standards. The Telecommunication Standardization Bureau of the ITU maintains a list of currently valid ITU-T Recommendations.

– Recommendation ITU-T T.800 (2002) | ISO/IEC 15444-1:2004, *Information technology – JPEG 2000 image coding system: Core coding system.*

– Recommendation ITU-T T.801 (2002) | ISO/IEC 15444-2:2004, *Information technology – JPEG 2000 image coding system: Extensions.*

3 Terms and definitions

For the purposes of this Recommendation | International Standard, the following definitions apply:

3.1 3D bit-block: A three-dimensional array of bits. In this Recommendation | International Standard, a 3D bit-block refers to all the bits of the same magnitude in all coefficients or samples. This could refer to a 3D bit-block in a component, tile-component, 3D code-block, region of interest, or other.

3.2 3D code-block: A rectangular three-dimensional grouping of coefficients from the same sub-band of a tile-component.

3.3 3D code-block scan: The order in which the coefficients within a 3D code-block are visited during a coding pass. The 3D code-block is processed in stripes, each consisting of four rows (or all remaining rows if less than four) and spanning the width of the 3D code-block. Each stripe is processed column by column from top to bottom and from left to right. The complete 3D code-block is consequently scanned slice by slice. Within a slice, Rec. ITU-T T.800 | ISO/IEC 15444-1 is followed.

3.4 component (update of Rec. ITU-T T.801 | ISO/IEC 15444-2): Compressed data from the codestream representing a single set of two- or three-dimensional data.

3.5 conforming reader (update of Rec. ITU-T T.800 | ISO/IEC 15444-1): An application that reads and interprets a JP3D file correctly.

3.6 decomposition level (update of Rec. ITU-T T.801 | ISO/IEC 15444-2): A collection of sub-bands where each coefficient has the same spatial impact or span with respect to the original samples. These include the [H|L|X][H|L|X][H|L|X] sub-band (e.g., LLL, LXL, XXH, …, exclusive XXX) split out of the three-dimensional decomposition sublevels.

3.7 [H|L|X][H|L|X][H|L|X] sub-band: H refers to high-pass filtering and L to low-pass filtering, while X refers to no filtering. The filter specified first refers to the horizontal filtering, the second to the vertical filtering and the third to the axial filtering (i.e., respectively along X-, Y- and Z-axes). The filter ordering for this sub-band should always be respected. The reconstruction will follow the inverse filtering order.

NOTE – The XXX sub-band does not exist (as defined in 3.6).

3.8 image (update of Rec. ITU-T T.800 | ISO/IEC 15444-1): The set of all components, which can have either two- or three-spatial dimensions.

3.9 image area offset (update of Rec. ITU-T T.800 | ISO/IEC 15444-1): The number of reference grid points down, to the right (and to an increased axial position) of the reference grid origin.

3.10 intermediate component (update of Rec. ITU-T T.801 | ISO/IEC 15444-2): A single two- or three-dimensional array of data involved in a stage of a multiple component transformation.

3.11 raster order (update of Rec. ITU-T T.800 | ISO/IEC 15444-1): A particular sequential order of data of any type within an array. The raster order starts with the top left data point of the first slice and moves to the data point immediately to the right, and so on to the end of the row. After the end of the row is reached, the next data point in the sequence is the left-most data point immediately below the current row. This order is continued to the end of the slice. Thereafter the next slice is processed in case of a three-dimensional array. This order is continued to the end of the array.

3.12 resolution (update of Rec. ITU-T T.801 | ISO/IEC 15444-2): The spatial relation of samples to a physical space. In this Recommendation | International Standard, the decomposition levels of the wavelet transform create resolutions that differ by powers of two in the horizontal, the vertical, or – in the three-dimensional case – the axial direction, or any possible combination of directions. The last (highest) decomposition level includes an [L|X][L|X][L|X] sub-band (note that XXX is non-existing), which is considered to be a lower resolution. Therefore, there is one more resolution level than decomposition levels.

3.13 resolution level (update of Rec. ITU-T T.800 | ISO/IEC 15444-1): Equivalent to decomposition level with the exception that the [L|X][L|X][L|X] sub-band is also a separate resolution level.

3.14 sample (update of Rec. ITU-T T.800 | ISO/IEC 15444-1): One element in the two-dimensional or three-dimensional array that comprises a component.

3.15 slice: A slice is a two-dimensional pixel subset of a volumetric entity, a volumetric code-block or a volumetric image. A slice is positioned perpendicular to the axial or z-axis.

3.16 spatial coordinates: Spatial coordinates are indicated by x, y and z. Generally, the term axial will be used to address the Z dimension.

3.17 sub-band (update of Rec. ITU-T T.800 | ISO/IEC 15444-1): A group of transform coefficients resulting from the same sequence of low-pass and high-pass filtering operations.

3.18 sub-band order: Within one resolution level, sub-bands are processed and signalled as defined in Rec. ITU-T T.800 | ISO/IEC 15444-1 and Rec. ITU-T T.801 | ISO/IEC 15444-2 for two-dimensional filtering, following a Morton scanning order [1]. The specification is extended to the three-dimensional case by deploying consequently a three-dimensional Morton scanning order.

3.19 tile (update of Rec. ITU-T T.800 | ISO/IEC 15444-1): A cuboidal array of points on the reference grid, registered with an offset from the reference grid origin and defined by a width (x dimension), a height (y dimension) and a depth (z dimension). The tiles that overlap are used to define tile-components.

4 Abbreviations

For the purposes of this Recommendation | International Standard, the abbreviations defined in clause 4 of Rec. ITU-T T.800 | ISO/IEC 15444-1, and clause 4 of Rec. ITU-T T.801 | ISO/IEC 15444-2 also apply to this Recommendation | International Standard.

5 Symbols (and abbreviated terms)

For the purposes of this Recommendation | International Standard, the symbols defined in clause 4 of Rec. ITU-T T.800 | ISO/IEC 15444-1, and clause 5 of Rec. ITU-T T.801 | ISO/IEC 15444-2 also apply to this Recommendation | International Standard.

6 General description

This Recommendation | International Standard defines a set of lossless (bit-preserving) and lossy compression methods for coding continuous-tone, bi-level, grey-scale, colour digital volumetric images, or multi-component volumetric images. This set of methods (see Annex A) extends the elements in the core coding system described in Rec. ITU-T T.800 | ISO/IEC 15444-1 and Rec. ITU-T T.801 | ISO/IEC 15444-2. Extensions which pertain to encoding and decoding are defined as procedures which may be used in combination with the encoding and decoding processes described in Rec. ITU-T T.800 | ISO/IEC 15444-1 and Rec. ITU-T T.801 | ISO/IEC 15444-2. Each encoding or decoding extension shall be used only in combination with particular coding processes and only in accordance with the requirements set forth herein. This Recommendation | International Standard also defines extensions to the compressed data format, i.e., interchange format and the abbreviated formats.

In particular, for Rec. ITU-T T.801 | ISO/IEC 15444-2, the following extensions are supported by this Recommendation | International Standard:

1) variable DC offset;
2) arbitrary wavelet transform kernels;
3) multi-component transformations;
4) non-linear transformations;
5) region-of-interest.

Annex A

Codestream syntax, extension

(This annex forms an integral part of this Recommendation | International Standard.)

A.1 Extended capabilities

The syntax in this annex supports the extensions in this Recommendation | International Standard. These marker segments conform to the same rules as the syntax Annex A of Rec. ITU-T T.800 | ISO/IEC 15444-1. The addition of parameter values to some marker segments in Rec. ITU-T T.800 | ISO/IEC 15444-1 and Rec. ITU-T T.801 | ISO/IEC 15444-2, and the addition of new marker segments signal the information specific to the extensions in this Recommendation | International Standard.

In every marker segment, the first two bytes after the marker shall be an unsigned value that denotes the length in bytes of the marker segment parameters (including the two bytes of this length parameter but not the two bytes of the marker itself).

When a marker segment that is not specified in this Recommendation | International Standard or in Rec. ITU-T T.800 | ISO/IEC 15444-1 and Rec. ITU-T T.801 | ISO/IEC 15444-2 is encountered in a codestream, the decoder shall use the length parameter to discard the marker segment. Table A.1 shows the marker segment usage specified for this Recommendation | International Standard.

Table A.1 – List of markers and marker segments

	Symbol	Code	Main header	Tile-part header	Rec. ITU-T T.80x | ISO/IEC 15444-x Heritage/ Extended
Delimiting markers and marker segments					
Start of codestream	SOC	0xFF4F	required	not allowed	Rec. ITU-T T.800 (2002) | ISO/IEC 15444-1:2004
Start of tile-part	SOT	0xFF90	not allowed	required	Rec. ITU-T T.801 (2002) | ISO/IEC 15444-2:2004
Start of data	SOD	0xFF93	not allowed	last marker	Rec. ITU-T T.800 (2002) | ISO/IEC 15444-1:2004
End of codestream	EOC	0xFFD9	not allowed	not allowed	Rec. ITU-T T.800 (2002) | ISO/IEC 15444-1:2004
Fixed information marker segments					
Image and tile size	SIZ	0xFF51	required	not allowed	Rec. ITU-T T.800 (2002) | ISO/IEC 15444-1:2004
Additional dimension image and tile size	NSI	0xFF54	required	not allowed	
Extended capabilities	CAP	0xFF50	required	not allowed	Rec. ITU-T T.801 (2002)/ Amd.2 (2005) | ISO/IEC 15444-2:2004/ Amd.2:2006
Functional marker segments					
Coding style default	COD	0xFF52	required	optional	Rec. ITU-T T.800 (2002) | ISO/IEC 15444-1:2004, Extended
Coding style component	COC	0xFF53	optional	optional	Rec. ITU-T T.800 (2002) | ISO/IEC 15444-1:2004, Extended
Region-of-interest	RGN	0xFF5E	optional	optional	Rec. ITU-T T.801 (2002) | ISO/IEC 15444-2:2004, Extended
Quantization default	QCD	0xFF5C	required	optional	Rec. ITU-T T.800 (2002) | ISO/IEC 15444-1:2004, Extended

Table A.1 – List of markers and marker segments

	Symbol	Code	Main header	Tile-part header	Rec. ITU-T T.80x | ISO/IEC 15444-x Heritage/ Extended
Quantization component	QCC	0xFF5D	optional	optional	Rec. ITU-T T.800 (2002) | ISO/IEC 15444-1:2004, Extended
Arbitrary transformation kernels	ATK	0xFF79	optional	optional	Rec. ITU-T T.801 (2002) | ISO/IEC 15444-2:2004
Component bit depth definition	CBD	0xFF78	optional	optional	Rec. ITU-T T.801 (2002) | ISO/IEC 15444-2:2004
Multiple component transformation definition	MCT	0xFF74	optional	optional	Rec. ITU-T T.801 (2002) | ISO/IEC 15444-2:2004
Multiple component transform collection	MCC	0xFF75	optional	optional	Rec. ITU-T T.801 (2002) | ISO/IEC 15444-2:2004
Multiple component transform ordering	MCO	0XFF77	optional	optional	Rec. ITU-T T.801 (2002) | ISO/IEC 15444-2:2004
Non-linearity point transformation	NLT	0xFF76	optional	optional	Rec. ITU-T T.801 (2002) | ISO/IEC 15444-2:2004
Variable DC offset	DCO	0XFF70	optional	optional	Rec. ITU-T T.801 (2002) | ISO/IEC 15444-2:2004
Pointer marker segments					
Tile-part lengths	TLM	0xFF55	optional	not allowed	Rec. ITU-T T.800 (2002) | ISO/IEC 15444-1:2004
Packet length, main header	PLM	0xFF57	optional	not allowed	Rec. ITU-T T.800 (2002) | ISO/IEC 15444-1:2004
Packet length, tile-part header	PLT	0xFF58	not allowed	optional	Rec. ITU-T T.800 (2002) | ISO/IEC 15444-1:2004
Packed packet headers, main header	PPM	0xFF60	optional	not allowed	Rec. ITU-T T.800 (2002) | ISO/IEC 15444-1:2004
Packed packet headers, tile-part header	PPT	0xFF61	not allowed	optional	Rec. ITU-T T.800 (2002) | ISO/IEC 15444-1:2004
In bit stream markers and marker segments					
Start of packet	SOP	0xFF91	not allowed	not allowed in tile-part header, optional in bit stream	Rec. ITU-T T.800 (2002) | ISO/IEC 15444-1:2004
End of packet header	EPH	0xFF92	optional inside PPM marker segment	optional inside PPT marker segment or in bit stream	Rec. ITU-T T.800 (2002) | ISO/IEC 15444-1:2004
Informational marker segments					
Component registration	CRG	0xFF63	optional	not allowed	Rec. ITU-T T.800 (2002) | ISO/IEC 15444-1:2004, Extended
Comment	COM	0xFF64	optional	optional	Rec. ITU-T T.800 (2002) | ISO/IEC 15444-1:2004

[a] Required means the marker or marker segment shall be in this header, optional means it may be used.

A.2 Extensions to Rec. ITU-T T.800 | ISO/IEC 15444-1 and Rec. ITU-T T.801 | ISO/IEC 15444-2 marker segment parameters

A.2.1 Additional dimension image and tile size (NSI)

Function: Provides information about the uncompressed image such as the depth of the reference grid, the depth of the tiles, and the separation of component samples with respect to the reference grid.

Usage: Main header. There shall be one and only one in the main header.

Length: Variable depending on the number of components.

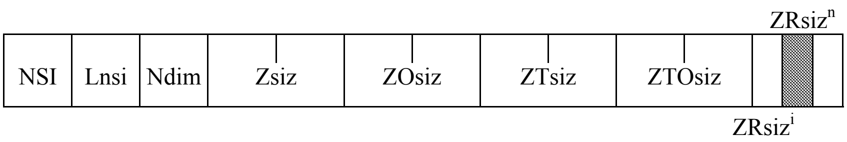


Figure A.1 – Additional dimension image and tile size syntax (extended)

NSI: Marker code. Table A.2 shows the size and parameter values of the symbol and parameters for additional dimension image and tile size marker segment.

Lnsi: Length of marker segment in bytes (not including the marker). The value of this parameter is determined by the following equation:

$$Lnsi = 19 + Csiz \tag{A-1}$$

Ndim: Defines the dimensionality of the dataset (disregarding the component dimension). This value is set to 3 by default.

Zsiz: Depth of the reference grid.

ZOsiz: Depth offset from the origin of the reference grid to the front left upper corner of the image volume.

ZTsiz: Depth of one reference tile with respect to the reference grid.

ZTOsiz: Vertical offset from the origin of the reference grid to the front left upper corner of the first tile.

ZRsiz^i: Depth separation of a sample of the *i*th component with respect to the reference grid. There is one occurrence of this parameter for each component, in order.

Table A.2 – Additional dimension image and tile size parameter values (extended)

Parameter	Size (bits)	Values
NSI	16	0xFF54
Lnsi	16	20-16403
Ndim	8	3
Zsiz	32	1-(2^{32} – 1)
ZOsiz	32	0-(2^{32} – 1)
ZTsiz	32	1-(2^{32} – 1)
ZTOsiz	32	1-(2^{32} – 2)
ZRsiz^i	8	1-255

A.2.2 Coding style default (COD), Rec. ITU-T T.800 | ISO/IEC 15444-1 extended

Function: Describes the coding style, number of decomposition levels and layering that is the default used for compressing all components of an image (if in the main header) or a tile (if in the tile-part header). The parameter values can be overridden for an individual component by a COC marker segment in either the main or tile-part header.

Usage: Main and first tile-part header of a given tile. There shall be one and only one in the main header. *Additionally, there may be at most one for each tile. If there are multiple tile-parts in a tile, and this* marker segment is present, it shall be found only in the first tile-part (*TPsot* = 0).

When used in the main header, the COD marker segment parameter values are used for all tile-components that do not have a corresponding COC marker segment in either the main or tile-part header. When used in the tile-part header, it overrides the main header COD and COCs and is used for all components in that tile without a corresponding COC marker segment in the tile-part. Thus, the order of precedence is the following:

Tile-part COC > Tile-part COD > Main COC > Main COD

where the "greater than" sign, >, means that the greater overrides the lesser marker segment.

Length: Variable depending on the value of Scod (see Lcod parameter).

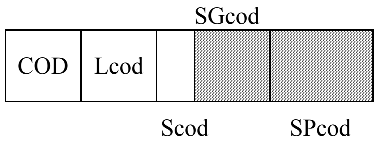


Figure A.2 – Coding style default syntax

COD: Marker code. Table A.3 shows the size and values of the symbol and parameters for coding style default marker segment.

Lcod: Length of marker segment in bytes (not including the marker). The value of this parameter is determined by the following equation:

$$Lcoc = \begin{cases} 17 & \text{maximum\_precincts} \\ 17 + 2 \cdot \text{number\_of\_resolution\_levels} & \text{user\_defined\_precincts} \end{cases} \quad \text{(A-2)}$$

where *maximum_precincts* and *user_defined_precincts* are indicated in the *Scod* parameter and *number_of_resolution_levels* is calculated by use of the number of decomposition level parameters for each of the three dimensions, X, Y and Z, as indicated in the *SPcod* parameter. The actual equation for calculating the number of resolution levels is given in B.5.

Scod: Coding style for all components. Table A.4 shows the value for the *Scod* parameter.

SGcod: Parameters for coding style designated in *Scod*. The parameters are independent of components and are designated, in order from top to bottom, in Table A.5. The coding style parameters within the *SGcod* field appear in the sequence shown in Figure A.3.

SPcod: Parameters for coding style designated in *Scod*. The parameters relate to all components and are designated, in order from top to bottom, in Table A.6. The coding style parameters within the *SPcod* field appear in the sequence shown in Figure A.3.

Table A.3 – Coding style default parameters values, extended

Parameter	Size (bits)	Values
COD	16	0xFF52
Lcod	16	17-83
Scod	8	Table A.4
SGcod	32	Table A.5
SPcod	variable	Table A.6

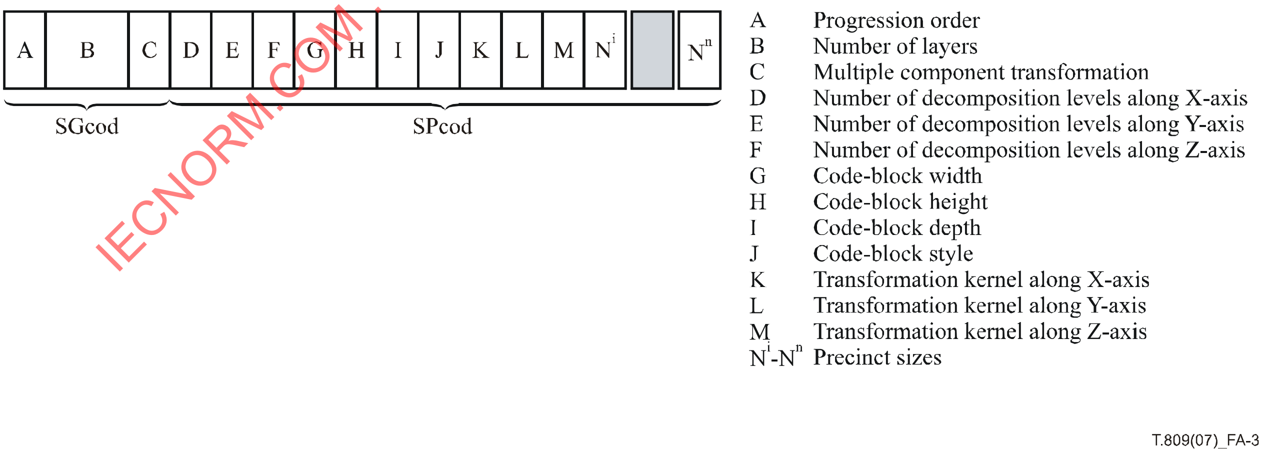


Figure A.3 – Coding style parameter diagram of the *SGcod* and *SPcod* parameters

Table A.4 – Coding style parameter values for the Scod parameter

Values (bits) MSB LSB	Coding style
xxxx xxx0	Entropy coder, precincts with $PPx = 15$, $PPy = 15$ and $PPz = 15$
xxxx xxx1	Entropy coder with custom precincts defined below
xxxx xx0x	No SOP marker segments used
xxxx xx1x	SOP marker segments may be used
xxxx x0xx	No EPH marker used
xxxx x1xx	EPH marker shall be used
	All other values reserved

Table A.5 – Coding style parameter values for the SGcod parameter

Parameters (in order)	Size (bits)	Values	Meaning of SGcod values
Progression order	8	Rec. ITU-T T.800 | ISO/IEC 15444-1 Table A.16	Progression order
Number of layers	16	1-65535	Number of layers
Multiple component transform	8	Rec. ITU-T T.801 | ISO/IEC 15444-2 Table A.8	Multiple component transform usage

Table A.6 – Coding style parameter values of the SPcod and SPcoc paramaters, extended

Parameters (in order)	Size (bits)	Values	Meaning of SPcod values
Number of decomposition levels along X-axis	8	0-32	Number of decomposition levels along X-axis, N_{LX}, zero implies no transformation
Number of decomposition levels along Y-axis	8	0-32	Number of decomposition levels along Y-axis, N_{LY}, zero implies no transformation
Number of decomposition levels along Z-axis	8	0-32	Number of decomposition levels along Z-axis, N_{LZ}, zero implies no transformation
3D code-block width	8	Table A.7	Code-block width exponent offset value, *xcb*
3D code-block height	8	Table A.7	Code-block height exponent offset value, *ycb*
3D code-block depth	8	Table A.7	Code-block depth exponent offset value, *zcb*
3D code-block style	8	Table A.8	Style of the 3D code-block coding passes
Transformation kernel along X-axis	8	Rec. ITU-T T.801 | ISO/IEC 15444-2 Table A.10	Wavelet transformation used along X-axis
Transformation kernel along Y-axis	8	Rec. ITU-T T.801 | ISO/IEC 15444-2 Table A.10	Wavelet transformation used along Y-axis
Transformation kernel along Z-axis	8	Rec. ITU-T T.801 | ISO/IEC 15444-2 Table A.10	Wavelet transformation used along Z-axis
Precinct size	variable	Table A.9	If *Scod* or *Scoc* = xxxx xxx0, this parameter is not present; otherwise, this indicates precinct width, height and depth. The first parameter (16 bits) corresponds to the N_LLLL sub-band. Each successive parameter corresponds to each successive resolution level in order.

Table A.7 – Width, height or depth exponent of the 3D code-blocks for the SPcod and SPcoc parameters

Values (bits) MSB LSB	3D code-block width, height and depth
xxxx 0000 – xxxx 1011	3D code-block width, height and depth exponent values *xcb* = *value*, *ycb* = *value* or *zcb* = *value*. NOTE – This redefines Rec. ITU-T T.800 | ISO/IEC 15444-1 significantly! The 3D code-block width, height and depth are limited to powers of two with the minimum size being 2^0 and the maximum being 2^{10}. Further, the 3D code-block size is restricted so that $4 \le xcb+ycb+zcb \le 18$.
	All other values reserved

Table A.8 – 3D code-block style for the SPcod and SPcoc paramaters, extended

Values (bits) MSB LSB	3D code-block style
xxxx xxx0	No selective arithmetic coding bypass
xxxx xxx1	Selective arithmetic coding bypass
xxxx xx0x	No reset of context probabilities on coding pass boundaries
xxxx xx1x	Reset context probabilities on coding pass boundaries
xxxx x0xx	No termination on each coding pass
xxxx x1xx	Termination on each coding pass
xxxx 0xxx	No causal contexts
xxxx 1xxx	Causal contexts
xxx0 xxxx	No predictable termination
xxx1 xxxx	Predictable termination
xx0x xxxx	No segmentation symbols are used
xx1x xxxx	Segmentation symbols are used
	All other values reserved

Table A.9 – Precinct width, height and depth for the SPcod and SPcoc parameters, extended

Values (bits) MSB LSB	Precinct size
xxxx xxxx xxxx 0000 – xxxx xxxx xxxx 1111	4 LSBs are the precinct width exponent *PPx* = *value*. This value may only equal zero at the resolution level corresponding to the N_LLLL band.
xxxx xxxx 0000 xxxx – xxxx xxxx 1111 xxxx	Next 4 bits are the precinct height exponent *PPy* = *value*. This value may only equal zero at the resolution level corresponding to the N_LLLL band.
xxxx 0000 xxxx xxxx – xxxx 1111 xxxx xxxx	Next 4 bits are the precinct depth exponent *PPz* = *value*. This value may only equal zero at the resolution level corresponding to the N_LLLL band.
	All other values reserved

A.2.3 Coding style component (COC), Rec. ITU-T T.800 | ISO/IEC 15444-1 extended

Function: Describes the coding style and number of decomposition levels used for compressing a particular component.

Usage: Main and first tile-part header of a given tile. The usage is optional in both the main and tile-part headers. No more than one per any given component may be present in either the main or tile-part headers. If there are multiple tile-parts in a tile, and this marker segment is present, it shall be found only in the first tile-part (*TPsot* = 0).

When used in the main header, it overrides the main COD marker segment for the specific component. When used in the tile-part header, it overrides the main header COD, main COC and tile COD for the specific component. Thus, the order of precedence is the following:

Tile-part COC > Tile-part COD > Main COC > Main COD

where the "greater than" sign, >, means that the greater overrides the lesser marker segment.

Length: Variable depending on the value of Scoc (see Lcoc parameter).

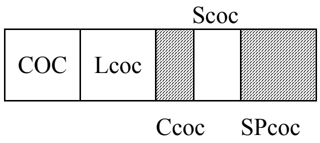


Figure A.4 – Coding style component syntax

COC: Marker code. Table A.10 shows the size and values of the symbol and parameters for the coding style component marker segment.

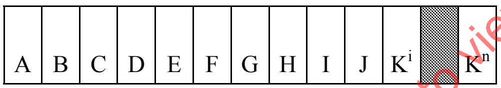
Lcoc: Length of marker segment in bytes (not including the marker). The value of this parameter is determined by the following equation:

$$Lcoc = \begin{cases} 14 & \text{maximum\_precincts AND } Csiz < 257 \\ 15 & \text{maximum\_precincts AND } Csiz \geq 257 \\ 14 + 2 \cdot \text{number\_of\_resolution\_levels} & \text{user\_defined\_precincts AND } Csiz < 257 \\ 15 + 2 \cdot \text{number\_of\_resolution\_levels} & \text{user\_defined\_precincts AND } Csiz \geq 257 \end{cases} \quad \text{(A-3)}$$

Ccoc: The index of the component to which this marker segment relates. The components are indexed 0, 1, 2, etc.

Scoc: Coding style for this component. Table A.11 shows the values for each *Scoc* parameter.

SPcoc: Parameters for coding style designated in *Scoc*. The coding style parameters within the *SPcoc* field appear in the order sequence shown in Figure A.5.



A	Number of decomposition levels along X-axis
B	Number of decomposition levels along Y-axis
C	Number of decomposition levels along Z-axis
D	Code-block width
E	Code-block height
F	Code-block depth
G	Code-block style
H	Transformation kernel along X-axis
I	Transformation kernel along Y-axis
J	Transformation kernel along Z-axis
K^i-K^n	Precinct sizes

Figure A.5 – Coding style parameter diagram of the SPcoc parameter

Table A.10 – Coding style component parameter values, extended

Parameter	Size (bits)	Values
COC	16	0xFF53
Lcoc	16	4-102
Ccoc	8 16	0-255; if Csiz < 257 0-16383; if Csiz ≥ 257
Scoc	8	Table A.11
$SPcoc^i$	variable	Table A.6

Table A.11 – Coding style parameter values for the Scoc parameter, extended

Values (bits) MSB LSB	Coding style
xxxx xxx0	Entropy coder with maximum precinct values $PPx = PPy = PPz = 15$
xxxx xxx1	Entropy coder with precinct values defined in *SPcoc*
	All other values reserved

A.2.4 Region-of-interest (RGN), Rec. ITU-T T.801 | ISO/IEC 15444-2 extended

Function: Signals the presence of a region-of-interest (ROI) in the codestream.

Usage: Main and first tile-part header of a given tile. If there is RGN marker segment in the main header with a Srgn = 0, there shall not be any RGN marker segment anywhere in the codestream with a non-zero Srgn value for the component given by the corresponding Crgn value. Likewise, if there is RGN marker segment in the main header with a non-zero Srgn value, there shall not be any RGN marker segment anywhere in the codestream with a Srgn = 0 for the component given by the corresponding Crgn value.

When used in both the main header and the first tile-part header, the RGN in the first tile-part header overrides the main RGN for that tile. Also, an RGN specifying a single component (Crgn ≠ 65 535) overrides on specifying all components (Crgn = 65 535). Thus the order of precedence is the following:

Tile-part RGN (Crgn ≠ 65 535) > Tile-part RGN (Crgn = 65 535) > Main RGN (Crgn ≠ 65 535) > Main RGN (Crgn = 65 535)

where the "greater than" sign, >, means that the greater overrides the lesser marker segment.

Length: Variable.

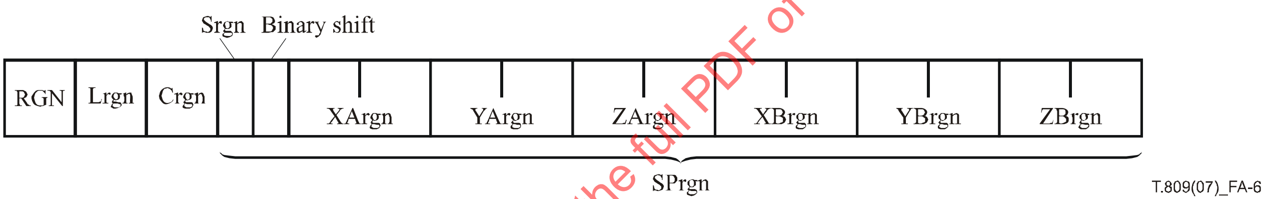


Figure A.6 – Region-of-interest syntax

RGN: Marker code. Table A.12 shows the size and values of the symbol and parameters for the region of interest marker segment.

Lrgn: Length of the marker segment in bytes (not including the marker).

Crgn: The index of the component to which this marker segment relates. The components are indexed 0, 1, 2, etc.

Srgn: ROI style for the current ROI. Table A.16 of Rec. ITU-T T.801 | ISO/IEC 15444-2 shows the value for the Srgn parameter.

SPrgn: Parameter for ROI style designated in *Srgn*. *SPrgn* is only signalled for *Srgn* = 1 or *Srgn* = 2.

Table A.12 – Region-of-interest parameter values, extended

Parameter	Size (bits)	Values
RGN	16	0xFF5E
Lrgn	16	5-30
Crgn	16	Rec. ITU-T T.801 | ISO/IEC 15444-2 Table A.17
Srgn	8	Rec. ITU-T T.801 | ISO/IEC 15444-2 Table A.16
SPrgn	variable	Table A.13

Table A.13 – Region of interest values from *SPrgn* parameter (*Srgn* = 1 or *Srgn* = 2), extended

Parameters (in order)	Size (bits)	Values	Meaning of SPrgn values
Binary shift	8	0-255	Binary shifting of coefficients in the region of interest above the background.
XArgn (left)	32	$0\text{-}(2^{32} - 1)$	Horizontal reference grid point from the origin of the first point. (In the case of the ellipse, *Srgn* = 2, this value shall not exceed the width of the image.)
YArgn (top)	32	$0\text{-}(2^{32} - 1)$	Vertical reference grid point from the origin of the first point. (In the case of the ellipse, *Srgn* = 2, this value shall not exceed the height of the image.)
ZArgn (front)	32	$0\text{-}(2^{32} - 1)$	Axial reference grid point from the origin of the first point. (In the case of the ellipse, *Srgn* = 2, this value shall not exceed the depth of the image.)
XBrgn (right)	32	$0\text{-}(2^{32} - 1)$	Horizontal reference grid point from the origin of the second point.
YBrgn (bottom)	32	$0\text{-}(2^{32} - 1)$	Vertical reference grid point from the origin of the second point.
ZBrgn (back)	32	$0\text{-}(2^{32} - 1)$	Axial reference grid point from the origin of the second point.

A.2.5 Quantization component default (QCD), Rec. ITU-T T.800 | ISO/IEC 15444-1 extended

Function: Describes the quantization default used for compressing all components not defined by a QCC marker segment. The parameter values can be overridden for an individual component by a QCC marker segment in either the main or tile-part header.

Usage: Main and first tile-part header of a given tile. There shall be one and only one in the main header. There may be at most one for all tile-part headers of a tile. If there are multiple tile-parts for a tile, and this marker segment is present, it shall be found only in the first tile-part (*TPsot* = 0).

When used in the tile-part header, it overrides the main QCD and the main QCC for the specific component. Thus, the order of precedence is the following:

Tile-part QCC > Tile-part QCD > Main QCC > Main QCD

where the "greater than" sign, >, means that the greater overrides the lesser marker segment.

Length: Variable depending on the number of quantized elements.

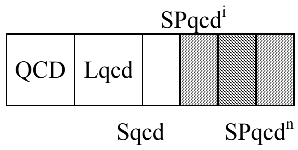


Figure A.7 – Quantization default style

QCD: Marker code. Table A.14 shows the size and values of the symbol and parameters for the quantization default marker segment.

Lqcd: Length of the marker segment in bytes (not including the marker). The value of this parameter is determined by the following equation:

$$Lqcd = \begin{cases} 4 + \text{number\_of\_subbands} & \text{no\_quantization} \\ 5 & \text{scalar\_quantization\_derived} \\ 5 + 2 \cdot \text{number\_of\_subbands} & \text{scalar\_quantization\_expounded} \end{cases} \qquad \text{(A-4)}$$

where *number_of_sub-bands* (depending on number of decomposition levels along X, Y and Z axis) is defined in the COD and COC marker segments, and *no_quantization*, *scalar_quantization_derived*, or *scalar_quantization_expounded* is signalled in the *Sqcd* parameter.

NOTE – The Lqcd can be used to determine how many quantization step sizes are present in the marker segment. However, there is not necessarily a correspondence with the number of sub-bands present because the sub-bands can be truncated with no requirement to correct this marker segment.

Sqcd: Quantization style for all components.

SPqcd^i: Quantization step size value for the ith sub-band in the defined order (see Annex B). The number of parameters is the same as the number of sub-bands in the tile-component with the greatest number of decomposition levels, N_L.

Table A.14 – Quantization default parameter values, extended

Parameter	Size (bits)	Values
QCD	16	0xFF5C
Lqcd	16	4-441
Sqcd	8	Rec. ITU-T T.800 | ISO/IEC 15444-1 Table A.28
SPqcd^i	variable	Rec. ITU-T T.800 | ISO/IEC 15444-1 Table A.28

A.2.6 Quantization component (QCC), Rec. ITU-T T.800 | ISO/IEC 15444-1 extended

Function: Describes the quantization used for compressing a particular component.

Usage: Main and first tile-part header of a given tile. Usage is optional in both the main and tile-part headers. No more than one per any given component may be present in either the main or tile-part headers. If there are multiple tile-parts in a tile, and this marker segment is present, it shall be found only in the first tile-part ($TPsot$ = 0).

Optional in both the main and tile-part headers. When used in the main header, it overrides the main QCD marker segment for the specific component. When used in the tile-part header, it overrides the main QCD, main QCC, and tile QCD for the specific component. Thus, the order of precedence is the following:

Tile-part QCC > Tile-part QCD > Main QCC > Main QCD

where the "greater than" sign, >, means that the greater overrides the lesser marker segment.

Length: Variable depending on the number of quantized elements.

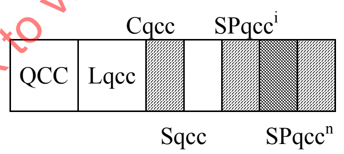


Figure A.8 – Quantization component syntax

QCC: Marker code. Table A.15 shows the size and values of the symbol and parameters for the quantization component marker segment.

Lqcc: Length of the marker segment in bytes (not including the marker). The value of this parameter is determined by the following equation:

$$Lqcc = \begin{cases} 5 + \text{number\_of\_subbands} & \text{no\_quantization AND } Csiz < 257 \\ 6 & \text{scalar\_quantization\_derived AND } Csiz < 257 \\ 6 + 2 \cdot \text{number\_of\_subbands} & \text{scalar\_quantization\_expounded AND } Csiz < 257 \\ 6 + \text{number\_of\_subbands} & \text{no\_quantization AND } Csiz \geq 257 \\ 7 & \text{scalar\_quantization\_derived AND } Csiz \geq 257 \\ 7 + 2 \cdot \text{number\_of\_subbands} & \text{scalar\_quantization\_expounded AND } Csiz \geq 257 \end{cases} \quad \text{(A-5)}$$

NOTE – The $Lqcc$ can be used to determine how many step sizes are present in the marker segment. However, there is not necessarily a correspondence with the number of sub-bands present because the sub-bands can be truncated with no requirement to correct this marker segment.

Cqcc: The index of the component to which this marker segment relates. The components are indexed 0, 1, 2, etc. (Either 8 or 16 bits depending on *Csiz* value.)

Sqcc: Quantization style for this component.

$SPqcc^i$: Quantization value for each sub-band in the defined order (see Annex D). The number of parameters is the same as the number of sub-bands in the tile-component with the greatest number of decomposition levels.

Table A.15 – Quantization component parameter values

Parameter	Size (bits)	Values
QCC	16	0xFF5D
Lqcc	16	5-443
Cqcc	8 16	0-255; if Csiz < 257 0-16383; if Csiz ≥ 257
Sqcc	8	Rec. ITU-T T.800 | ISO/IEC 15444-1 Table A.28
$SPqcc^i$	variable	Rec. ITU-T T.800 | ISO/IEC 15444-1 Table A.28

A.2.7 Component registration (CRG), Rec. ITU-T T.800 | ISO/IEC 15444-1 extended

Function: Allows specific registration of components with respect to each other. For coding purposes, the samples of components are considered to be located at reference grid points that are integer multiples of *XRsiz*, *YRsiz* and *ZRsiz* (see Annex B). However, this may be inappropriate for rendering the image. The CRG marker segment describes the "center of mass" of each component's samples with respect to the separation. This marker segment has no effect on decoding the codestream.

NOTE – This component registration offset is with respect to the image offset (*XOsiz*, *YOsiz* and *ZOsiz*) and the component separation ($XRsiz^i$, $YRsiz^i$ and $ZRsiz^i$). For example, the horizontal reference grid point for the left most samples of component *c* is $XRsiz^c \cdot \lfloor XOsiz / XRsiz^c \rfloor$. (Likewise for the vertical and axial direction.) The horizontal offset denoted in this marker segment is in addition to this offset.

Usage: Main header only. Only one CRG may be used in the main header and it is applicable for all tiles.

Length: Variable depending on the number of components.

CRG	Lcrg	$Xcrg^i$	$Ycrg^i$	$Zcrg^i$		$Xcrg^n$	$Ycrg^n$	$Zcrg^n$

Figure A.9 – Component registration syntax

CRG: Marker code. Table A.16 shows the size and values of the symbol and parameters for the component registration marker segment.

Lcrg: Length of the marker segment in bytes (not including the marker).

$Xcrg^i$: Value of the horizontal offset, in units of 1⁄65536 of the horizontal separation $XRsiz^i$, for the *i*th component. Thus, values range from 0⁄65536 (sample occupies its reference grid point) to $XRsiz^c$ (65535⁄65536) (just before the next sample's reference grid point). This value is repeated for every component.

$Ycrg^i$: Value of the vertical offset, in units of 1⁄65536 of the vertical separation $YRsiz^i$, for the *i*th component. Thus, values range from 0⁄65536 (sample occupies its reference grid point) to $YRsiz^c$ (65535⁄65536) (just before the next sample's reference grid point). This value is repeated for every component.

$Zcrg^i$: Value of the axial offset, in units of 1⁄65536 of the axial separation $ZRsiz^i$, for the *i*th component. Thus, values range from 0⁄65536 (sample occupies its reference grid point) to $ZRsiz^c$ (65535⁄65536) (just before the next sample's reference grid point). This value is repeated for every component.

Table A.16 – Component registration parameter values

Parameter	Size (bits)	Values
CRG	16	0xFF63
Lcrg	16	6-65534
$Xcrg^i$	16	0-65535
$Ycrg^i$	16	0-65535
$Zcrg^i$	16	0-65535

A.2.8 Extended capabilities (CAP), Rec. ITU-T T.801 (2002)/Amd.2 (05/2005) | ISO/IEC 15444-2:2004/Amd.2:2006

This Recommendation | International Standard requires the presence of the CAP marker segment in the main header, with the Pcap field signalling the use of Rec. ITU-T T.809 | ISO/IEC 15444-10 as described in Rec. ITU-T T.801 | ISO/IEC 15444-2. Table A.17 explains the usage of its associated $Ccap^i$ parameter for this Recommendation | International Standard.

NOTE – These extended capabilities were developed after the first edition of Rec. ITU-T T.801 | ISO/IEC 15444-2, and were specified in the amendment Rec. ITU-T T.801 (2002)/Amd.2 (05/2005) | ISO/IEC 15444-2:2004/Amd.2:2006.

For this Recommendation | International Standard it is required for the Rsiz field of the SIZ marker segment (see Rec. ITU-T T.801 | ISO/IEC 15444-2) to signal the presence of the CAP marker segment by enabling the second-most-significant bit. This prevents decoders that are unable to correctly handle Rec. ITU-T T.809 | ISO/IEC 15444-10 enabled codestreams from continuing. It is advisable that the CAP marker segment appears immediately after the SIZ marker segment, such that a decoder can easily recognize Rec. ITU-T T.809 | ISO/IEC 15444-10 enabled codestreams before encountering any extended marker segments.

Table A.17 – $Ccap^i$, extended

Values (bits) MSB LSB	Coding style
0000 0000 0000 0000 0000	No extended capabilities
	All other values reserved

Annex B

Image and compressed image data ordering, extension

(This annex forms an integral part of this Recommendation | International Standard.)

B.1 Introduction

In this annex and all of its subclauses, the flow charts and tables are normative only in the sense that they are defining an output that alternative implementations shall duplicate. This annex describes the various structural entities, and their organization in the codestream: components, tiles, sub-bands and their divisions.

This annex is the extension of Rec. ITU-T T.800 | ISO/IEC 15444-1 Annex B with volumetric coding functionality, i.e., from two to three spatial dimensions. The following subclauses will describe only the changed items to Rec. ITU-T T.800 | ISO/IEC 15444-1 Annex B that are needed to add the extra axial dimension. Unless explicitly noted, all descriptions and specifications of Rec. ITU-T T.800 | ISO/IEC 15444-1 remain valid.

B.2 Introduction to image data structure concepts

Update of Rec. ITU-T T.800 | ISO/IEC 15444-1 clause B.1.

Components no longer consist of two-dimensional-arrays of samples, but instead they consist of three-dimensional-arrays of samples. Each component, c, now has parameters $XRsiz^c$, $YRsiz^c$ and $ZRsiz^c$ which define the mapping between component samples and the reference grid points.

Each resolution level consists of either the [L|H|X][L|H|X][L|H|X] sub-bands (excluding the LLL sub-band) or the N_LLLL sub-band, thus changing the number of sub-bands per decomposition level, m, with $m < N_L$, from three to the range [2;7] (see Figure D.6).

Each sub-band has its own origin. The sub-band boundary conditions are unique for each of the [L|H|X][L|H|X][L|H|X] sub-bands.

B.3 Component mapping to the reference grid

Update of Rec. ITU-T T.800 | ISO/IEC 15444-1 clause B.2.

The reference grid becomes a three-dimensional space of points with the indices from (0, 0, 0) to ($Xsiz - 1$, $Ysiz - 1$, $Zsiz - 1$). An "image area" is then defined on the reference grid by the dimensional parameters, $(Xsiz, Ysiz, Zsiz)$ and ($XOsiz$, $YOsiz$, $ZOsiz$). Specifically, the image area on the reference grid is defined by its reference grid points at location $(XOsiz, YOsiz, ZOsiz)$ and ($Xsiz - 1$, $Ysiz - 1$, $Zsiz - 1$).

The samples of component c are now at integer multiples of $(XRsiz^c, YRsiz^c, ZRsiz^c)$ on the reference grid. Each component domain is a sub-sampled version of the reference grid with the (0, 0, 0) coordinate as common point for each component.

Thus, the samples of component c are mapped to a cuboid with corner coordinates (x_0, y_0, z_0) and ($x_1 - 1$, $y_1 - 1$, $z_1 - 1$), where x_0, y_0, x_1 and y_1 are given by Rec. ITU-T T.800 | ISO/IEC 15444-1 Equation B-1 and z_0 and z_1 as:

$$z_0 = \left\lceil \frac{ZOsiz}{ZRsiz^c} \right\rceil \qquad z_1 = \left\lceil \frac{Zsiz}{ZRsiz^c} \right\rceil \tag{B-1}$$

Thus, the three dimensions of component c are given by:

$$(width, height, depth) = (x_1 - x_0, y_1 - y_0, z_1 - z_0) \tag{B-2}$$

The parameters $Zsiz$, $ZOsiz$ and $ZRsiz^c$ are all defined in the NSI marker segment (see A.2.1).

B.4 Image area division into tiles and tile-components

Update of Rec. ITU-T T.800 | ISO/IEC 15444-1 clause B.3.

The reference grid is partitioned into a regular sized three dimensional array of tiles. The tile size and tiling offset are defined, on the reference grid, by dimensional pairs *(XTsiz, YTsiz, ZTsiz)* and *(XTOsiz, YTOsiz, ZTOsiz)*, respectively. *ZTsiz* and *ZTOsiz* are parameters from the NSI marker segment.

Every tile is *XTsiz* reference grid points wide, *YTsiz* reference grid points high and *ZTsiz* reference grid points deep. The upper left front corner of the first tile (tile 0) is offset from the upper left front corner of the reference grid by *(XTOsiz, YTOsiz, ZTOsiz)*. The tiles are numbered in raster order (i.e., from left to right, top to bottom, front to back). This number is the tile index.

The tile grid offsets *(XTOsiz, YTOsiz, ZTOsiz)* are constrained to be not larger than the image area offsets. This is expressed by Rec. ITU-T T.800 | ISO/IEC 15444-1 Equation B-3 and additionally by the following range:

$$0 \le ZTOsiz \le ZOsiz \tag{B-3}$$

Also, the tile size plus the tile offset shall be greater than the image area offset. This ensures that the first tile (tile 0) will contain at least one reference grid point from the image area. This is expressed by Rec. ITU-T T.800 | ISO/IEC 15444-1 Equation B-4 and additionally by the following range:

$$ZTsiz + ZTOsiz > ZOsiz \tag{B-4}$$

The number of tiles in the horizontal (X) direction (*numXtiles*) and the vertical (Y) direction (*numYtiles*) are given by Rec. ITU-T T.800 | ISO/IEC 15444-1 Equation B-5. The number of tiles for the axial (Z) direction (*numZtiles*) is the following:

$$numZtiles = \left\lceil \frac{Zsiz - ZTOsiz}{ZTsiz} \right\rceil \tag{B-5}$$

For the purpose of this description, it is useful to have tiles indexed in terms of horizontal, vertical and axial position. Let p_x be the horizontal index of a tile and p_y be the vertical index of a tile, then p_z represents the axial index of a tile, ranging from 0 to (*numZtiles* – 1). The following expression redefines p_x and p_y (Rec. ITU-T T.800 | ISO/IEC 15444-1 Equation B-6) and defines p_z:

$$\begin{aligned} p_x &= \mathrm{mod}(\mathrm{mod}(t, numXtiles \cdot numYtiles), numXtiles) \\ p_y &= \left\lfloor \frac{\mathrm{mod}(t, numXtiles \cdot numYtiles)}{numXtiles} \right\rfloor \\ p_z &= \left\lfloor \frac{t}{numXtiles \cdot numYtiles} \right\rfloor \end{aligned} \tag{B-6}$$

As for the coordinates of a particular tile on the reference grid, these are described by the following equations:

$$\begin{aligned} tx_0(p_x, p_y, p_z) &= \max(XTOsiz + p_x \cdot XTsiz, XOsiz) \\ ty_0(p_x, p_y, p_z) &= \max(YTOsiz + p_y \cdot YTsiz, YOsiz) \\ tz_0(p_x, p_y, p_z) &= \max(ZTOsiz + p_z \cdot ZTsiz, ZOsiz) \\ tx_1(p_x, p_y, p_z) &= \min(XTOsiz + (p_x + 1) \cdot XTsiz, Xsiz) \\ ty_1(p_x, p_y, p_z) &= \min(YTOsiz + (p_y + 1) \cdot YTsiz, Ysiz) \\ tz_1(p_x, p_y, p_z) &= \min(ZTOsiz + (p_z + 1) \cdot ZTsiz, Zsiz) \end{aligned} \tag{B-7}$$

Where $tx_0(p_x, p_y, p_z)$, $ty_0(p_x, p_y, p_z)$ and $tz_0(p_x, p_y, p_z)$ are the coordinates of the upper left front corner of the tile and $tx_1(p_x, p_y, p_z) - 1$, $ty_1(p_x, p_y, p_z) - 1$ and $tz_1(p_x, p_y, p_z) - 1$ are the coordinates of the lower right back corner of the tile. We will often drop the tile's coordinates in referring to a specific tile and refer instead to coordinates (tx_0, ty_0, tz_0) and $(tx_1\ ty_1, tz_1)$.

Thus, the dimensions of a tile in the reference grid are:

$$(tx_1 - tx_0, ty_1 - ty_0, tz_1 - tz_0) \tag{B-8}$$

Within the domain of image component i, the coordinates of the upper left front hand sample are given by (tcx_0, tcy_0, tcz_0) and the coordinates of the lower right back hand sample are given by (tcx_1, tcy_1, tcz_1), where tcx_0, tcy_0, tcx_1 *and* tcy_1 are described in Rec. ITU-T T.800 | ISO/IEC 15444-1 Equation B-12 and tcz_0 and tcz_1 are given by the following equation:

$$tcz_0 = \left\lceil \frac{tz_0}{ZRsiz^i} \right\rceil \quad tcz_1 = \left\lceil \frac{tz_1}{ZRsiz^i} \right\rceil \qquad \text{(B-9)}$$

So, the dimensions of the tile-component in the reference grid are:

$$\left(tcx_1 - tcx_0, tcy_1 - tcy_0, tcz_1 - tcz_0\right) \qquad \text{(B-10)}$$

B.5 Transformed tile-component division into resolution levels and sub-bands

Update of Rec. ITU-T T.800 | ISO/IEC 15444-1 clause B.5.

NOTE – Although Rec. ITU-T T.809 | ISO/IEC 15444-10 (JP3D) allows a different number of decomposition levels for each of the three dimensions (horizontal denoted by X, vertical denoted by Y and axial denoted by Z), it is fundamentally different from Rec. ITU-T T.801 | ISO/IEC 15444-2 Annex F: "Arbitrary Decomposition of Tile-components". The decomposition of a tile-component described in this Recommendation | International Standard is NOT arbitrary.

Each tile-component is wavelet transformed with N_{LX} decomposition levels in the horizontal direction, N_{LY} decomposition levels in the vertical direction and N_{LZ} decomposition levels in the axial direction as described in Annex D. The direction with the highest number of decomposition levels also determines the number of resolution levels. Thus, with $N_L = max(N_{LX}, N_{LY}, N_{LZ})$ as defined in D.3.2, there are $(N_L + 1)$ distinct resolution levels, denoted $r = 0, 1,..., N_L$. Each resolution level, r, is represented by the n[L|X][L|X][L|X] band, with $n = N_L - r$, where the actual sub-band type is determined by the number of decompositions in each of the directions (see D.4.1). For example, when N_{LX} is 3, N_{LY} is 3 and N_{LZ} is 2, then at $r = 0$, the lowest resolution level is represented by the 3LLX band, while at, $r = 1$, the resolution level is represented by the 2LLL band. This clause describes the dimensions of this reduced resolution.

The given tile-component's coordinates with respect to the reference grid at a particular resolution level, r, yield upper left front hand sample coordinates, (trx_0, try_0, trz_0) and lower right back hand sample coordinates, $(trx_1 - 1, try_1 - 1, trz_1 - 1)$, where:

$$\begin{aligned} trx_0 &= \left\lceil \frac{tcx_0}{2^{\min(N_L - r, N_{LX})}} \right\rceil & trx_1 &= \left\lceil \frac{tcx_1}{2^{\min(N_L - r, N_{LX})}} \right\rceil \\ try_0 &= \left\lceil \frac{tcy_0}{2^{\min(N_L - r, N_{LY})}} \right\rceil & try_1 &= \left\lceil \frac{tcy_1}{2^{\min(N_L - r, N_{LY})}} \right\rceil \\ trz_0 &= \left\lceil \frac{tcz_0}{2^{\min(N_L - r, N_{LZ})}} \right\rceil & trz_1 &= \left\lceil \frac{tcz_1}{2^{\min(N_L - r, N_{LZ})}} \right\rceil \end{aligned} \qquad \text{(B-11)}$$

In a similar manner, the tile coordinates may be mapped into any particular sub-band, b, yielding upper left front hand sample coordinates (tbx_0, tby_0, tbz_0) and lower right back hand sample coordinates $(tbx_1–1, tby_1–1, tbz_1–1)$ where:

$$\begin{aligned} tbx_0 &= \left\lceil \frac{tcx_0 - (2^{nx_b - 1} \cdot xo_b)}{2^{nx_b}} \right\rceil & tbx_1 &= \left\lceil \frac{tcx_1 - (2^{nx_b - 1} \cdot xo_b)}{2^{nx_b}} \right\rceil \\ tby_0 &= \left\lceil \frac{tcy_0 - (2^{ny_b - 1} \cdot yo_b)}{2^{ny_b}} \right\rceil & tby_1 &= \left\lceil \frac{tcy_1 - (2^{ny_b - 1} \cdot yo_b)}{2^{ny_b}} \right\rceil \\ tbz_0 &= \left\lceil \frac{tcz_0 - (2^{nz_b - 1} \cdot zo_b)}{2^{nz_b}} \right\rceil & tbz_1 &= \left\lceil \frac{tcz_1 - (2^{nz_b - 1} \cdot zo_b)}{2^{nz_b}} \right\rceil \end{aligned} \qquad \text{(B-12)}$$

where nx_b, ny_b and nz_b, as defined in Equation B-13, represent the respective decomposition levels for the horizontal (X), vertical (Y) and axial (Z) directions associated with sub-band b. The quantities (xo_b, yo_b, zo_b) are given in Table B.1.

$$nx_b = \min(N_L - r + 1, N_{LX})$$
$$ny_b = \min(N_L - r + 1, N_{LY}) \quad \text{(B-13)}$$
$$nz_b = \min(N_L - r + 1, N_{LZ})$$

Table B.1 – Quantities *(xo_b, yo_b, zo_b)* for sub-band *b*

Sub-band type	xo_b	yo_b	zo_b
[L|X][L|X][L|X]	0	0	0
H[L|X][L|X]	1	0	0
[L|X]H[L|X]	0	1	0
HH[L|X]	1	1	0
[L|X][L|X]H	0	0	1
H[L|X]H	1	0	1
[L|X]HH	0	1	1
HHH	1	1	1

For each sub-band, these coordinates define the tile boundaries in distinct sub-band domains. Furthermore, the dimensions of each sub-band are given by:

$$\left(tbx_1 - tbx_0, tby_1 - tby_0, tbz_1 - tbz_0\right) \quad \text{(B-14)}$$

B.6 Division of resolution levels into precincts

Update of Rec. ITU-T T.800 | ISO/IEC 15444-1 clause B.6.

Consider a particular tile-component and resolution level whose bounding sample coordinates in the reduced resolution image domain are (trx_0, try_0, trz_0) and $(trx_1{-}1, try_1{-}1, trz_1{-}1)$, as already described. Analogue to the method described in Rec. ITU-T T.800 | ISO/IEC 15444-1 clause B.6, the three-dimensional tile-component resolution level is partitioned into precincts, using trx_0, trx_1, try_0 and try_1, and additionally also trz_0 and trz_1. The precinct is anchored at (0, 0, 0), so that the upper left front hand corner of any given precinct in the partition is located at integer multiples of (2^{PPx}, 2^{PPy}, 2^{PPz}) where *PPx*, *PPy* and *PPz* are signalled in the COD or COC marker segments. As for *PPx* and *PPy*, *PPz* may be different for each tile-component and resolution level. *PPz* must be at least 1 for all resolution levels, *r*, except when $r = 0$ where it is allowed to be zero.

The number of precincts which span the tile-component at resolution level, *r*, is given by Rec. ITU-T T.800 | ISO/IEC 15444-1 Equation B-16 and by the following equation:

$$numprecinctsdeep = \begin{cases} \left\lceil \dfrac{trz_1}{2^{PPz}} \right\rceil - \left\lfloor \dfrac{trz_0}{2^{PPz}} \right\rfloor & trz_1 > trz_0 \\ 0 & trz_1 = trz_0 \end{cases} \quad \text{(B-15)}$$

Even if Rec. ITU-T T.800 | ISO/IEC 15444-1 Equation B-16 or Equation B-15 of this Recommendation | International Standard indicate that both *numprecinctswide*, *numprecinctshigh*, and *numprecinctsdeep* are non-zero, some, or all, precincts may still be empty as explained in Rec. ITU-T T.800 | ISO/IEC 15444-1 clause B.6. The precinct index runs from 0 to *numprecincts* –1 where *numprecincts* = *numprecinctswide·numprecinctshigh·numprecinctsdeep* in raster order (i.e., from left to right, top to bottom, front to back). The index is used in determining the order of appearance, in the codestream, of packets corresponding to each precinct, as explained in Rec. ITU-T T.800 | ISO/IEC 15444-1 clause B.12.

B.7 Division of sub-bands into code-blocks

Update of Rec. ITU-T T.800 | ISO/IEC 15444-1 clause B.7.

The sub-bands are partitioned into rectangular 3D code-blocks for the purpose of coefficient modelling and coding. The size of each code-block is determined from three parameters, *xcb*, *ycb* and *zcb*, which are signalled in the COD or COC marker segments. The code-block size for each sub-band at a particular resolution level is determined as $2^{xcb'}$ by $2^{ycb'}$ by $2^{zcb'}$ where *xcb'* and *ycb'* are described by Rec. ITU-T T.800 | ISO/IEC 15444-1 Equations B-17 and B-18 and *zcb'* is given by:

$$zcb' = \begin{cases} \min(zcb, PPz - 1) & r > 0 \\ \min(zcb, PPz) & r = 0 \end{cases} \quad \text{(B-16)}$$

These equations reflect the fact that the code-block size is constrained both by the precinct size and the code-block size, whose parameters, *xcb*, *ycb* and *zcb*, are identical for all sub-bands in the tile-component. Like the precinct, the code-block partition is anchored at (0, 0, 0). Thus, all boundaries of code-blocks in the code-block partition are located at $x = g_x 2^{xcb'}$, $y = g_y 2^{ycb'}$ and $z = g_z 2^{zcb'}$ where g_x, g_y and g_z are integers.

B.8 Packets

Update of Rec. ITU-T T.800 | ISO/IEC 15444-1 clause B.9.

All compressed image data representing a specific tile, layer, component, resolution level and precinct appears in the codestream in a contiguous segment called a packet. Packet data is aligned at 8-bit (one byte) boundaries.

As defined in Annex D, resolution level $r = 0$ contains the sub-band coefficients from the N_LLLL band, where N_L is the number of decomposition levels as defined in clause D.3.2. Each subsequent resolution level, $r > 0$, contains the sub-band coefficients from the *n*[L|H|X][L|H|X][L|H|X] sub-bands, *n*LLL excluded, as defined in Annex D, where $n = N_L - r + 1$. There are $(N_L + 1)$ resolution levels for a tile-component with N_L decomposition levels.

The compressed image data in a packet is ordered such that the contribution from the LLL, XLL, LXL, LLX, LXX, XLX, XXL, HLL, HXL, HLX, HXX, LHL, XHL, LHX, XHX, HHL, HHX, LLH, XLH, LXH, XXH, HLH, HXH, LHH, XHH and HHH sub-bands appear in that order (i.e., Morton scanning order). Within each sub-band, the codeblock contributions appear in raster order, confined to the bounds established by the relevant precinct. Resolution level $r = 0$ contains only the N_LLLL band and resolution levels $r > 0$ can only contain some of the N_L[L|H|X][L|H|X][L|H|X] bands, excluded N_LLLL. Only those codeblocks that contain samples from the relevant sub-band, confined to the precinct, have any representation in the packet.

Packet data is introduced by a packet header whose syntax is described in Rec. ITU-T T.800 | ISO/IEC 15444-1 clause B.10 and is followed by a packet body containing the actual code-bytes contributed by each of the relevant code-blocks. The order defined above is followed in constructing both the packet header and the packet body.

B.9 Packet header information coding

Update of Rec. ITU-T T.800 | ISO/IEC 15444-1 clause B.10.

B.9.1 Tag trees

Rec. ITU-T T.800 | ISO/IEC 15444-1 clause B.10.2 describes two dimensional tag trees. For the purpose of a three-dimensional extension, three-dimensional tag trees are required.

A 3D tag tree is a way of representing a three-dimensional array of non-negative integers in a hierarchical way. It successively creates reduced resolution levels of this three-dimensional array, forming a tree. At every node of this tree, the minimum integer of the top (up to eight) nodes below is recorded. The notation, $q_i(m_x, m_y, m_z)$, is the value at the node that is $m_x th$ from the left, $m_y th$ from the top and $m_z th$ from the front, at the *ith* level. Level 0 is the lowest level of the tag tree; it contains the top node.

See Rec. ITU-T T.800 | ISO/IEC 15444-1 clause B.10.2 for further information on how to actually encode and decode values with the tag tree. The description given there is independent of the number of actual dimensions.

B.9.2 Order of information within a packet

Update of Rec. ITU-T T.800 | ISO/IEC 15444-1 clause B.10.8.

The following is the packet header information order for one packet of a specific layer, tile-component, resolution level and precinct.

```
bit for zero or non-zero length packet
  for each subband ([L|H|X][L|H|X][L|H|X])
    for all code-blocks in this subband confined to the relevant precinct, in raster order
      code-block inclusion bits (if not previously included then tag tree, else one bit)
      if code-block included
        if first instance of code-block
          zero bit-planes information
        number of coding passes included
        increase of code-block length indicator (Lblock)
        for each codeword segment
          length of codeword segment
```

B.10 Progression order

Update of Rec. ITU-T T.800 | ISO/IEC 15444-1 clause B.12.

For a given tile-part, the packets contain all compressed image data from a specific layer, a specific component, a specific resolution level, and a specific precinct. The order in which these packets are found in the codestream is called the progression order. The ordering of the packets can progress along four axes: layer, component, resolution level and precinct.

It is possible that components have a different number of resolution levels. In this case, the resolution level that corresponds to the N_LLLL sub-band is the first resolution level ($r = 0$) for all components. The indices are synchronized from that point on.

B.10.1 Progression order determination

This clause describes the algorithms that define the five possible progression orders. They are basically identical to those described in Rec. ITU-T T.800 | ISO/IEC 15444-1 clause B.12.1, but extended to three dimensions. The lines printed in bold indicate the additions necessary for the third dimension.

The COD marker segments signal which of the five progression orders are used (see Rec. ITU-T T.800 | ISO/IEC 15444-1 clause A.6.1). The progression order can also be overridden with the POC marker segment (see Rec. ITU-T T.800 | ISO/IEC 15444-1 clause A.6.6) in any tile-part header. For each of the possible progression orders, the mechanism to determine the order in which packets are included is described below.

B.10.1.1 Layer-resolution level-component-position progression

See Rec. ITU-T T.800 | ISO/IEC 15444-1 clause B.12.1.1.

B.10.1.2 Resolution level-layer-component-position progression

See Rec. ITU-T T.800 | ISO/IEC 15444-1 clause B.12.1.2.

B.10.1.3 Resolution level-position-component-layer progression

Resolution level-position-component-layer progression for three dimensions is defined as the interleaving of the packets in the following order:

for each $r = 0,\ldots, N_{max}$
 for each $z = tz_0,\ldots, tz_1 - 1$,
  for each $y = ty_0,\ldots, ty_1 - 1$,
   for each $x = tx_0,\ldots, tx_1 - 1$,
    for each $i = 0,\ldots, Csiz - 1$
     if ((z divisible by $ZRsiz(i) \cdot 2^{PPz(r,i)+N_L(i)-r}$) OR (($z = tz_0$) AND ($trz_0 \cdot 2^{N_L(i)-r}$ NOT divisible by $2^{PPz(r,i)+N_L(i)-r}$)))
      if ((y divisible by $YRsiz(i) \cdot 2^{PPy(r,i)+N_L(i)-r}$) OR (($y = ty_0$) AND ($try_0 \cdot 2^{N_L(i)-r}$ NOT divisible by $2^{PPy(r,i)+N_L(i)-r}$)))
       if ((x divisible by $XRsiz(i) \cdot 2^{PPx(r,i)+N_L(i)-r}$) OR (($x = tx_0$) AND ($trx_0 \cdot 2^{N_L(i)-r}$ NOT divisible by $2^{PPx(r,i)+N_L(i)-r}$)))
        for the next precinct, k, if one exists,
         for each $l = 0,\ldots, L - 1$
          packet for component i, resolution level r, layer l, and precinct k

In the above, k can be obtained from:

$$k = \left\lfloor \frac{\left\lceil \frac{x}{XRsiz(i)\cdot 2^{N_L - r}} \right\rceil}{2^{PPx(r,i)}} \right\rfloor - \left\lfloor \frac{trx_0}{2^{PPx(r,i)}} \right\rfloor + numprecinctswide(r,i) \cdot \left(\left\lfloor \frac{\left\lceil \frac{y}{YRsiz(i)\cdot 2^{N_L - r}} \right\rceil}{2^{PPy(r,i)}} \right\rfloor - \left\lfloor \frac{try_0}{2^{PPy(r,i)}} \right\rfloor \right)$$

$$+ numprecinctswide \cdot numprecinctshigh \cdot \left(\left\lfloor \frac{\left\lceil \frac{z}{ZRsiz(i)\cdot 2^{N_L - r}} \right\rceil}{2^{PPz(r,i)}} \right\rfloor - \left\lfloor \frac{trz_0}{2^{PPz(r,i)}} \right\rfloor \right) \quad \text{(B-17)}$$

To use this progression, *XRsiz*, *YRsiz* and *ZRsiz* values must be powers of two for each component.

B.10.1.4 Position-component-resolution level-layer progression

Position-component-resolution level-layer progression is defined as the interleaving of the packets in the following order:

for each $z = tz_0,\ldots, tz_1 - 1$,
  for each $y = ty_0,\ldots, ty_1 - 1$,
    for each $x = tx_0,\ldots, tx_1 - 1$,
      for each $i = 0,\ldots,$ Csiz - 1
        for each $r = 0,\ldots, N_{max}$ where N_L is the number of decomposition levels for component i,
          if ((z divisible by $ZRsiz(i) \cdot 2^{PPz(r,i)+N_L(i)-r}$) OR (($z = tz_0$) AND ($trz_0 \cdot 2^{N_L(i)-r}$ NOT divisible by $2^{PPz(r,i)+N_L(i)-r}$)))
            if ((y divisible by $YRsiz(i) \cdot 2^{PPy(r,i)+N_L(i)-r}$) OR (($y = ty_0$) AND ($try_0 \cdot 2^{N_L(i)-r}$ NOT divisible by $2^{PPy(r,i)+N_L(i)-r}$)))
              if ((x divisible by $XRsiz(i) \cdot 2^{PPx(r,i)+N_L(i)-r}$) OR (($x = tx_0$) AND ($trx_0 \cdot 2^{N_L(i)-r}$ NOT divisible by $2^{PPx(r,i)+N_L(i)-r}$)))
                for the next precinct, k, if one exists,
                  for each $l = 0,\ldots, L - 1$
                    packet for component i, resolution level r, layer l, and precinct k

In the above, k can be obtained from Equation B-17. To use this progression, *XRsiz*, *YRsiz* and *ZRsiz* values must be powers of two for each component.

B.10.1.5 Component-position-resolution level-layer progression

Component-position-resolution level-layer progression is defined as the interleaving of the packets in the following order:

for each $i = 0,\ldots,$ Csiz - 1
  for each $z = tz_0,\ldots, tz_1 - 1$,
    for each $y = ty_0,\ldots, ty_1 - 1$,
      for each $x = tx_0,\ldots, tx_1 - 1$,
        for each $r = 0,\ldots, N_{max}$ where N_L is the number of decomposition levels for component i,
          if ((z divisible by $ZRsiz(i) \cdot 2^{PPz(r,i)+N_L(i)-r}$) OR (($z = tz_0$) AND ($trz_0 \cdot 2^{N_L(i)-r}$ NOT divisible by $2^{PPz(r,i)+N_L(i)-r}$)))
            if ((y divisible by $YRsiz(i) \cdot 2^{PPy(r,i)+N_L(i)-r}$) OR (($y = ty_0$) AND ($try_0 \cdot 2^{N_L(i)-r}$ NOT divisible by $2^{PPy(r,i)+N_L(i)-r}$)))
              if ((x divisible by $XRsiz(i) \cdot 2^{PPx(r,i)+N_L(i)-r}$) OR (($x = tx_0$) AND ($trx_0 \cdot 2^{N_L(i)-r}$ NOT divisible by $2^{PPx(r,i)+N_L(i)-r}$)))
                for the next precinct, k, if one exists,
                  for each $l = 0,\ldots, L - 1$
                    packet for component i, resolution level r, layer l, and precinct k

In the above, k can be obtained from Equation B-17.

Annex C

Coefficient bit modelling

(This annex forms an integral part of this Recommendation | International Standard.)

C.1 Introduction

In this annex and all of its subclauses, the flow charts and tables are normative only in the sense that they are defining an output that alternative implementations shall duplicate. This annex formally extends Rec. ITU-T T.800 | ISO/IEC 15444-1 Annex D with volumetric coding functionality.

This annex defines the modelling and scanning of transform coefficient bits.

Code-blocks (see Annex B) are decoded a bit-plane at a time starting from the most significant bit-plane with a non-zero element to the least significant bit-plane. For each bit-plane in a code-block, a special code-block scan pattern is used for each of three coding passes. Each coefficient bit in the bit-plane appears in only one of the three coding passes called significance propagation, magnitude refinement, and cleanup. For each pass contexts are created which are provided to the arithmetic coder, CX, along with the bit stream, CD (see Rec. ITU-T T.800 | ISO/IEC 15444-1 clause C.3).

C.2 Code-block scan pattern within code-blocks, extended

Each bit-plane of a code-block is scanned in a particular order. The 3D code-block is processed in stripes, each consisting of four rows (or all remaining rows if less than four) and spanning the width of the 3D code-block. Each stripe is processed column by column from top to bottom and from left to right. The complete 3D code-block is consequently scanned slice by slice. Within a slice, Rec. ITU-T T.800 | ISO/IEC 15444-1 is followed.

C.3 Context model updates

For the context modelling, the model described in Rec. ITU-T T.800 | ISO/IEC 15444-1 Annex D is modified with respect to the significance propagation and cleanup coding passes.

The context vector for a given current coefficient is the binary vector consisting of the significance states of its 8 nearest-neighbour coefficients in the XY plane, as shown in Rec. ITU-T T.800 | ISO/IEC 15444-1 Figure D.2. Any nearest neighbour lying outside the current coefficient's code-block is regarded as insignificant (i.e., it is treated as having a zero significance state) for the purpose of creating a context vector for decoding the current coefficient.

Rather than using Rec. ITU-T T.800 | ISO/IEC 15444-1 Table D.1, the relevant contexts are specified by Table C.1 below.

Table C.1 – Contexts for the significance propagation and cleanup coding passes

Sub-bands with primary orientation of L[L|H|X][L|H|X], X[L|H][L|H|X] or XX[L|H]			Sub-bands with primary orientation of H[L|X][L|H|X]			Sub-bands with primary orientation of HH[L|H|X]		Context label[a]
$\sum H_i$	$\sum V_i$	$\sum D_i$	$\sum H_i$	$\sum V_i$	$\sum D_i$	$\sum(H_i+V_i)$	$\sum D_i$	
2	x[b]	x	x	2	x	x	≥ 3	8
1	≥ 1	x	≥ 1	1	x	≥ 1	2	7
1	0	≥ 1	0	1	≥ 1	0	2	6
1	0	0	0	1	0	≥ 2	1	5
0	2	x	2	0	x	1	1	4
0	1	x	1	0	x	0	1	3
0	0	≥ 2	0	0	≥ 2	≥ 2	0	2
0	0	1	0	0	1	1	0	1
0	0	0	0	0	0	0	0	0

[a] Note that the context labels are indexed only for identification convenience in this Recommendation | International Standard. The actual identifiers used is a matter of implementation.

[b] x = "don't care".

Annex D

Discrete wavelet transformation of tile-components

(This annex forms an integral part of this Recommendation | International Standard.)

D.1 Introduction

In this annex and all of its subclauses, the flow charts and tables are normative only in the sense that they define an output that alternative implementations shall duplicate.

This annex describes the three-dimensional forward discrete wavelet transformation applied to one tile-component and specifies the inverse three-dimensional discrete wavelet transformation used to reconstruct the tile-component [3], [4] and [5]. This annex formally extends Rec. ITU-T T.800 | ISO/IEC 15444-1 Annex F with volumetric coding functionality.

D.2 Tile-component parameters

Consider the tile-component defined by the coordinates tcx_0, tcx_1, tcy_0, tcy_1, tcz_0 and tcz_1 given in Rec. ITU-T T.800 | ISO/IEC 15444-1 Equation B-12 and in Equation B-9 of this Recommendation | International Standard. Then the coordinates (x,y,z) of the tile-component (with sample values $I(x,y,z)$) lie in the range defined by:

$$tcx_0 \le x < tcx_1,\ tcy_0 \le y < tcy_1 \text{ and } tcz_0 \le z < tcz_1 \quad \text{(D-1)}$$

D.3 Discrete wavelet transformations

D.3.1 Low-pass and high-pass filtering (informative)

See Rec. ITU-T T.800 | ISO/IEC 15444-1 clause F.2.1.

D.3.2 Decomposition levels

Each tile-component is transformed into a set of three-dimensional sub-band signals (called sub-bands), each representing the activity of the signal in various frequency bands, at various spatial resolutions. N_{LX} denotes the number of decomposition levels in the horizontal direction, N_{LY} denotes the number of decomposition levels in the vertical direction and N_{LZ} denotes the number of decomposition levels in the axial direction. For the purpose of this Recommendation | International Standard, we define $N_L = max(N_{LX}, N_{LX}, N_{LX})$.

D.3.3 Discrete wavelet filters (informative)

See Rec. ITU-T T.800 | ISO/IEC 15444-1 clause F.2.3.

D.4 Inverse discrete wavelet transformation

D.4.1 The IDWT procedure

The inverse discrete wavelet transformation (IDWT) transforms a set of sub-bands, $a_b(u_b,v_b,w_b)$ into a DC-level shifted tile-component, $I(x, y, z)$ (IDWT procedure). The IDWT procedure (see Figure D.1) also takes as input the set of parameters N_{LX}, N_{LY} and N_{LZ}, which represent the number of decomposition levels in each of the three dimensions and are signalled in the COD or COC markers (see A.2.2 and A.2.3).

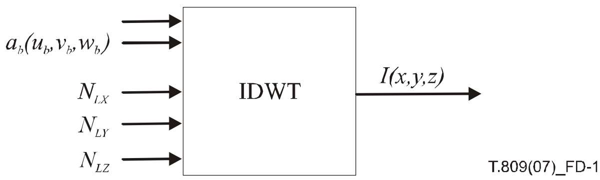


Figure D.1 – Inputs and outputs of the IDWT procedure

The sub-bands are labelled in the following way: an index *lev* corresponding to the decomposition level, followed by three letters of which each letter is either L, H or X. Thus a sub-band label is given as *lev*[L|H|X][L|H|X][L|H|X].

The sub-band b = *lev*[L|X][L|X][L|X] corresponds to a downsampled version of the sub-band *(lev* – 1*)*[L|X][L|X][L|X] which has been low-pass filtered for those directions (horizontal, vertical and/or axial) that are denoted by the L letter. If at decomposition level, *(lev* – 1*)*, a specific direction is not low-pass filtered (i.e., the letter X was used), it may no longer be low-pass filtered in any of the subsequent higher decomposition levels, $n > (lev - 1)$. Thus, when the number of decompositions in direction d (with d being either X, Y or Z), N_{Ld}, is less than *lev*, the decomposition type of that respective dimension is X; otherwise, it is L. The sub-band b = 0LLL corresponds to the original tile-component.

Similarly to Rec. ITU-T T.800 | ISO/IEC 15444-1 clause F.3.1, the sub-bands are signalled in the codestream, in the order given below:

N_LLLL, N_LXLL, N_LLXL, N_LLLX, N_LLXX, N_LXLX, N_LXXL, N_LHLL, N_LHXL, N_LHLX, N_LHXX, N_LLHL, N_LXHL, N_LLHX, N_LXHX, N_LHHL, N_LHHX, N_LLLH, N_LXLH, N_LLXH, N_LXXH, N_LHLH, N_LHXH, N_LLHH, N_LXHH, N_LHHH, $(N_L{-}1)$HLL, $(N_L{-}1)$HXL, $(N_L{-}1)$HLX, $(N_L{-}1)$HXX, $(N_L{-}1)$LHL, $(N_L{-}1)$XHL, $(N_L{-}1)$LHX, …, 1XHX, 1HHL, 1HHX, 1LLH, 1XLH, 1LXH, 1XXH, 1HLH, 1HXH, 1LHH, 1XHH, 1HHH.

Note that, from the list of sub-bands given above, only those sub-bands that exist, given the parameters N_{LX}, N_{LY} and N_{LZ}, will be present in the codestream. These are the exact sub-bands necessary to fully reconstruct the original tile-component.

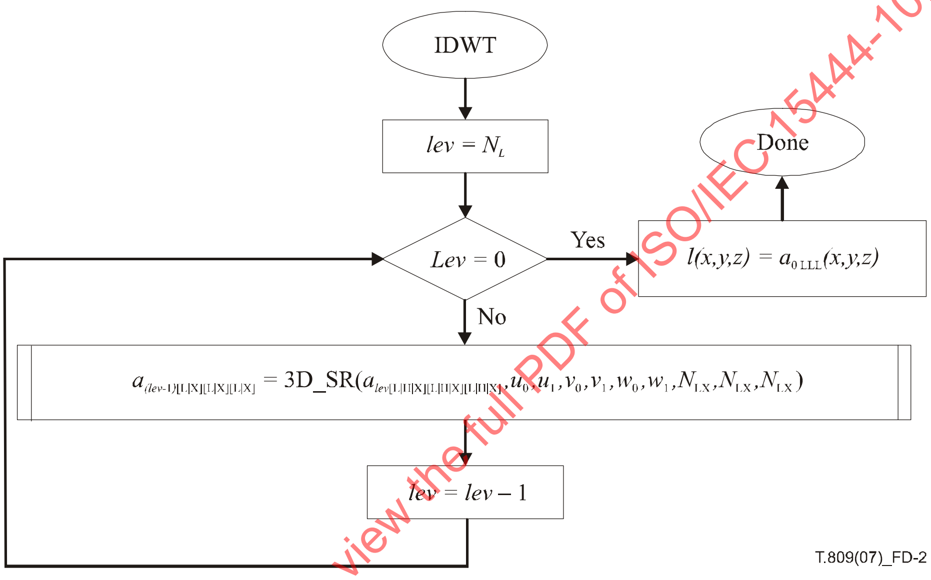


Figure D.2 – IDWT procedure

As in Rec. ITU-T T.800 | ISO/IEC 15444-1 clause F.3.1, the IDWT procedure starts with the initialization of the variable *lev* (the current decomposition level) to N_L. The 3D_SR procedure (see D.4.2) is performed at every level *lev*, where the level *lev* decreases at each iteration, until N_L iterations are performed. The 3D_SR procedure is iterated over the *lev*[L|X][L|X][L|X] sub-band produced at each iteration.

Finally, the sub-band $a_{0LLL}(u_{0LLL}, u_{1LLL}, v_{0LLL}, v_{1LLL}, w_{0LLL}, w_{1LLL})$ is the output array $I(x, y, z)$.

As defined in Equation B-12, the indices of sub-band coefficients for a given sub-band b lie in the range defined by:

$$tbx_0 \le u_b < tbx_1,\ \ tby_0 \le v_b < tby_1 \text{ and } tbz_0 \le w_b < tbz_1 \qquad \text{(D-2)}$$

D.4.2 The 3D_SR procedure

The 3D_SR procedure performs a reconstruction of sub-band $a_{(lev-1)[L|X][L|X][L|X]}(u,v,w)$ from the given sub-bands $a_{lev[L|H|X][L|H|X][L|H|X]}(u,v,w)$. The total number of coefficients of the reconstructed $(lev{-}1)$[L|X][L|X][L|X] sub-band is equal to the sum of the total number of coefficients of the sub-bands used as input to the 3D_SR procedure.

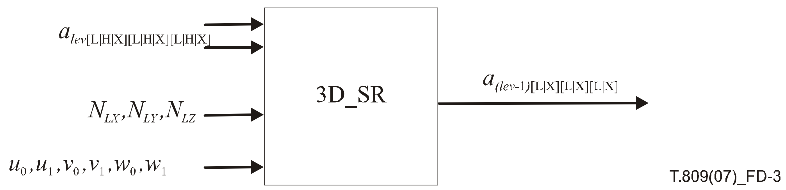


Figure D.3 – Inputs and outputs of the 3D_SR procedure

The sub-bands $a_{lev[L|H|X][L|H|X][L|H|X]}(u,v,w)$ are merged into the array $a(u,v,w)$, using the TO_ARRAY procedure. This temporary array is used to store the intermediate results and is constantly updated as the 3D_SR procedure is executed. Subsequently, the 3D_SR procedure reconstructs each direction, d, for which $lev < N_{Ld} \leq N_L$, with *d being X, Y or Z*, using the HOR_SR, VER_SR and AXIAL_SR procedures respectively. The end result is the sub-band $a_{(lev-1)[L|X][L|X][L|X]}(u,v,w)$. Figure D.4 describes the 3D_SR procedure in detail.

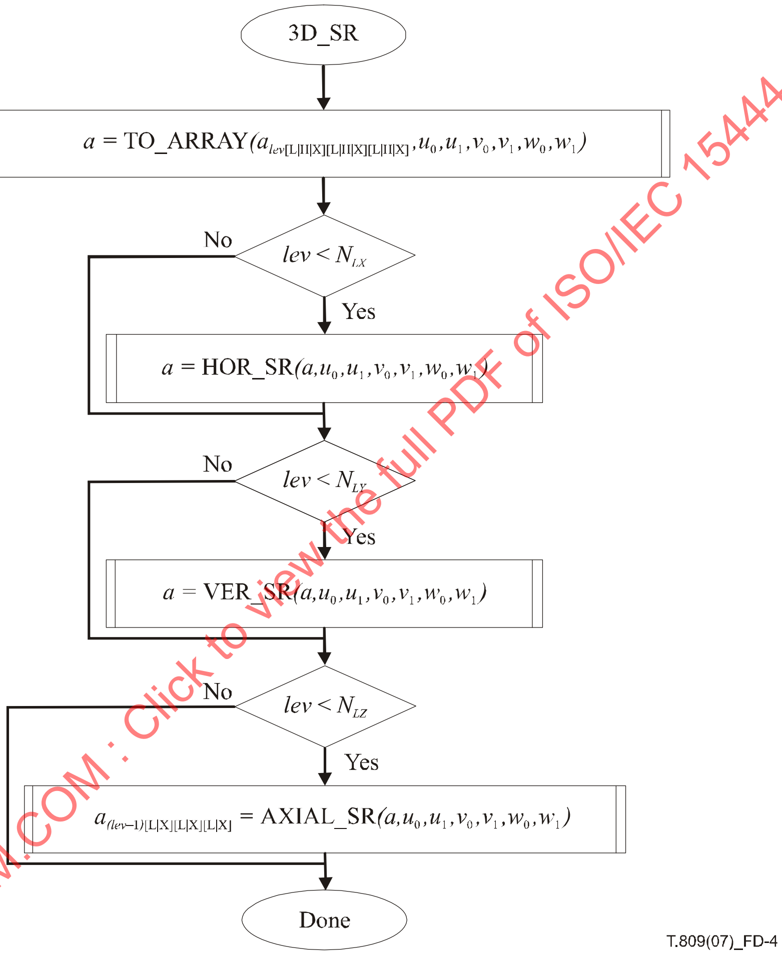


Figure D.4 – 3D_SR procedure

D.4.3 The TO_ARRAY procedure

The TO_ARRAY procedure takes all sub-bands of decomposition level *lev* and merges them into a three-dimensional array $a(u,v,w)$ as shown in Figure D.6. It takes as input the sub-bands $a_{lev[L|H|X][L|H|X][L|H|X]}(u,v,w)$ and the horizontal, vertical and axial extent of the coefficients of sub-band $a_{(lev-1)[L|X][L|X][L|X]}$, given as u_0, u_1, v_0, v_1, w_0 and w_1. The output array takes the horizontal, vertical and axial extent of the coefficients as indicated by $u_0 \leq u < u_1$, $v_0 \leq v < v_1$ and $w_0 \leq w < w_1$.

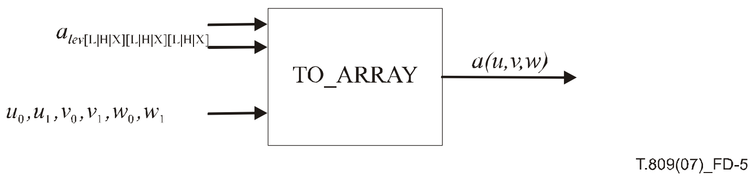


Figure D.5 – Inputs and outputs of the TO_ARRAY procedure

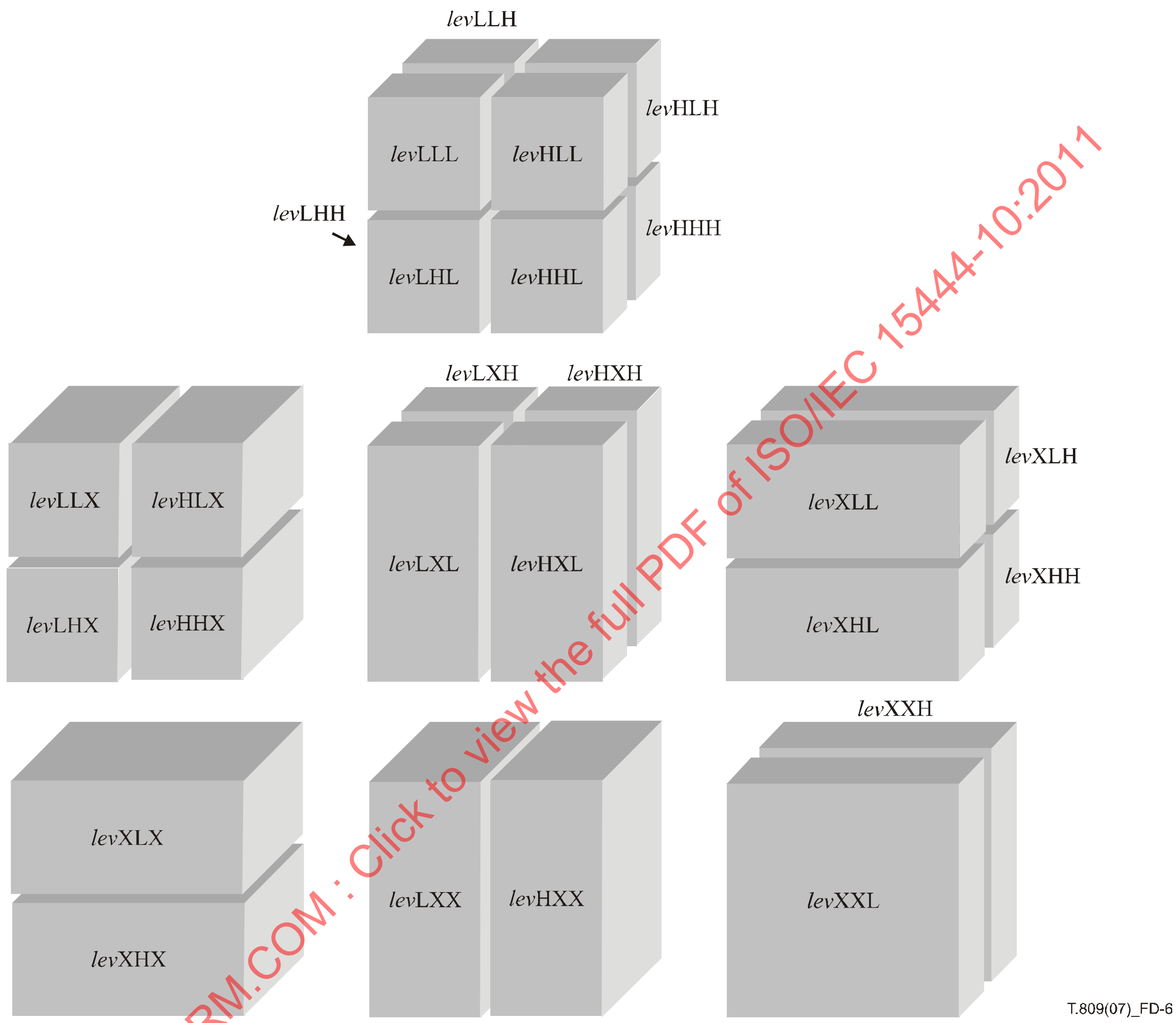


Figure D.6 – All possible sub-band configurations for 3D DWT

D.4.4 The 1D_INTERLEAVE procedure

As illustrated in Figure D.7, the 1D_INTERLEAVE procedure interleaves the low-pass and high-pass coefficients of a wavelet transform. It takes as input a one-dimensional signal, Y(n), and rearranges the coefficient values within the signal by interleaving them. The values of i_0 and i_1 used by the 1D_INTERLEAVE procedure represent the start and the end of the signal respectively. The way the signal is interleaved to form the output is described by the 1D_INTERLEAVE procedure, given in Figure D.8.

Figure D.7 – Parameters of the 1D_INTERLEAVE procedure

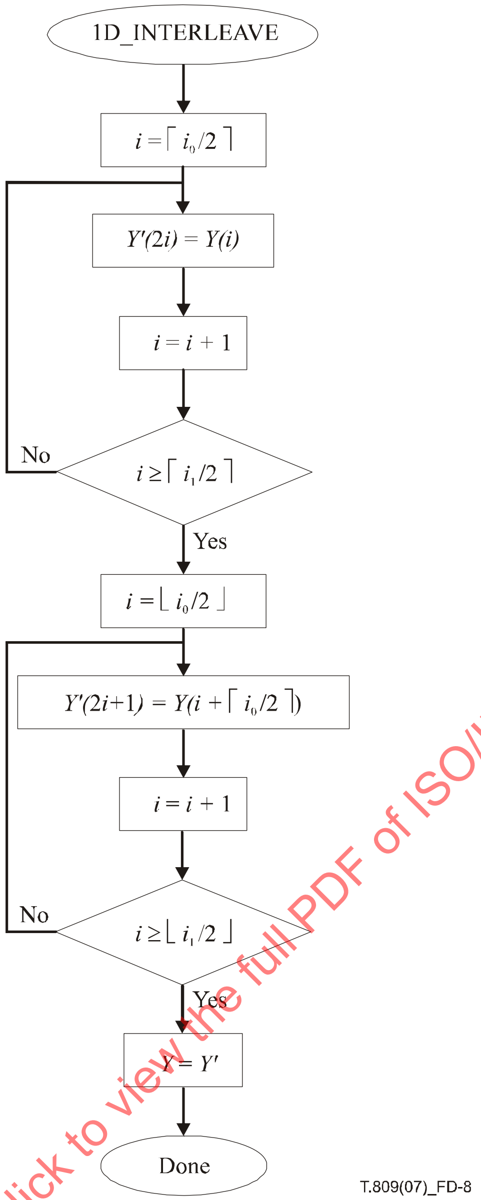


Figure D.8 – 1D_INTERLEAVE procedure

D.4.5 The HOR_SR procedure

The HOR_SR procedure performs an interleave operation and a horizontal sub-band reconstruction of a three-dimensional array of coefficients. It takes as input a three-dimensional array $a(u,v,w)$, the horizontal, vertical and axial extent of its coefficients as indicated by $u_0 \leq u < u_1$, $v_0 \leq v < v_1$ and $w_0 \leq w < w_1$ (see Figure D.9) and produces as output a horizontally filtered version of the input array, row by row and slice by slice.

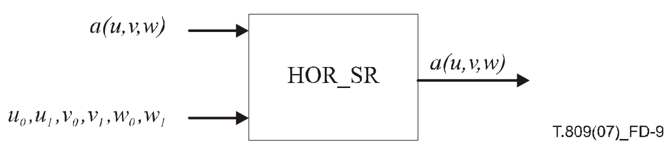


Figure D.9 – Inputs and outputs of the HOR_SR procedure

As illustrated in Figure D.10, the HOR_SR procedure applies the one-dimensional sub-band reconstruction (1D_SR procedure) to each row of the input array $a(u,v,w)$, and stores the result back in each row.

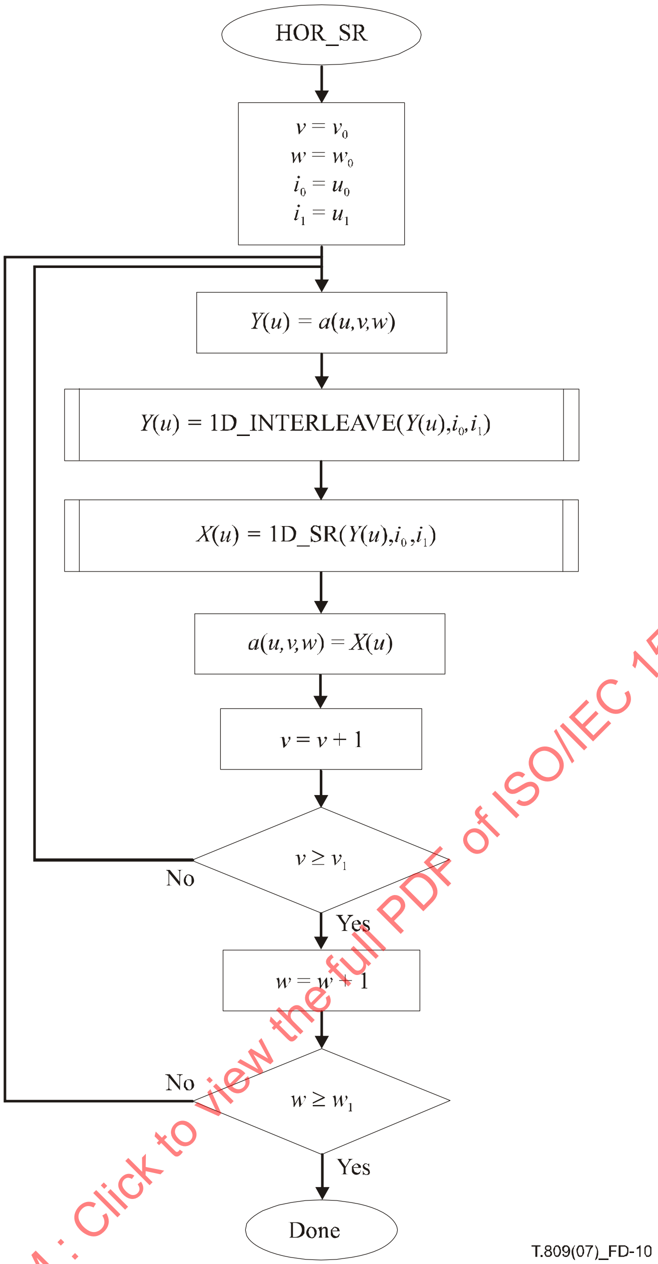


Figure D.10 – HOR_SR procedure

D.4.6 The VER_SR procedure

The VER_SR procedure performs an interleave operation and a vertical sub-band reconstruction of a three-dimensional array of coefficients. It takes as input a three-dimensional array *a(u,v,w)*, the horizontal, vertical and axial extent of its coefficients as indicated by $u_0 \leq u < u_1$, $v_0 \leq v < v_1$ and $w_0 \leq w < w_1$ (see Figure D.11) and produces as output a vertically filtered version of the input array, column by column and slice by slice.

As illustrated in Figure D.12, the VER_SR procedure applies the one-dimensional sub-band reconstruction (1D_SR procedure) to each column of the input array *a(u,v,w)*, and stores the result back in each column.

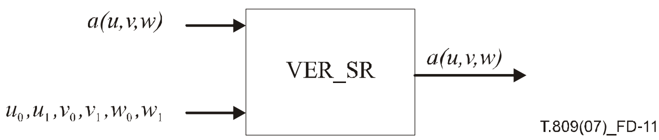


Figure D.11 – Inputs and outputs of the VER_SR procedure

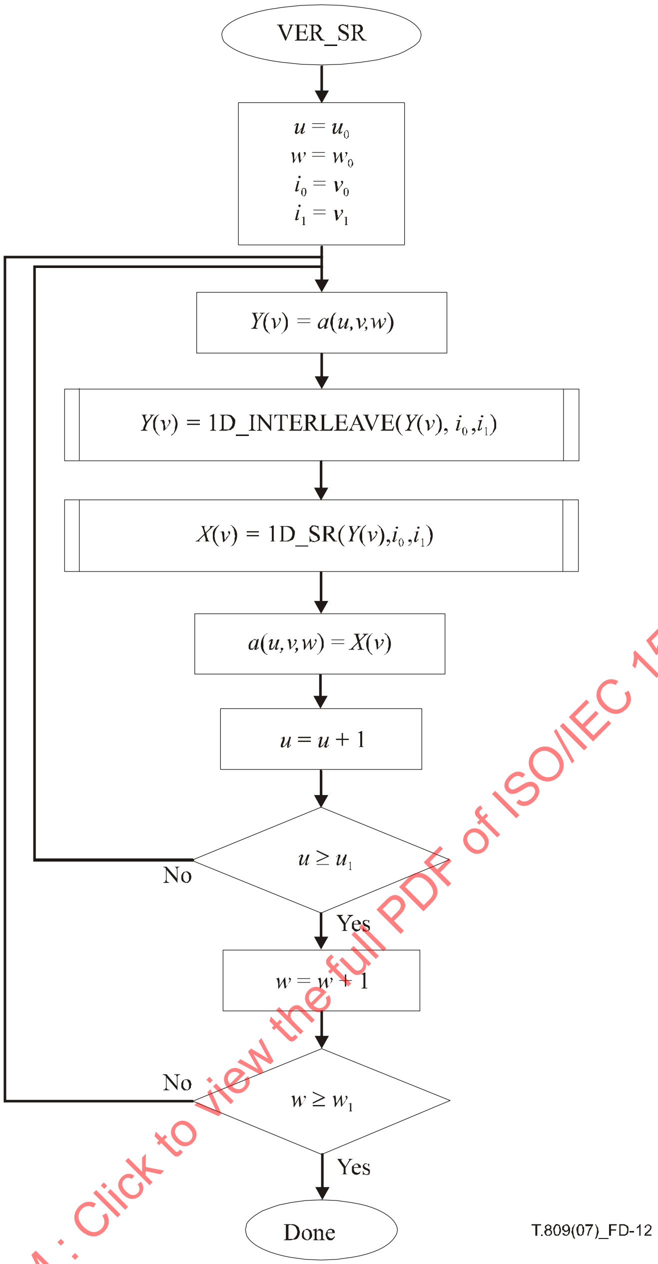


Figure D.12 – VER_SR procedure

D.4.7 The AXIAL_SR procedure

The AXIAL_SR procedure performs an interleave operation and an axial sub-band reconstruction of a three-dimensional array of coefficients. It takes as input a three-dimensional array *a(u,v,w)*, the horizontal, vertical and axial extent of its coefficients as indicated by $u_0 \leq u < u_1$, $v_0 \leq v < v_1$ and $w_0 \leq w < w_1$ (see Figure D.13) and produces as output an axially filtered version of the input array, row by row and column by column.

As illustrated in Figure D.14, the AXIAL_SR procedure applies the one-dimensional sub-band reconstruction (1D_SR procedure) to each depth of the input array *a(u,v,w)*, and stores the result back in each depth.

Figure D.13 – Inputs and outputs of the AXIAL_SR procedure

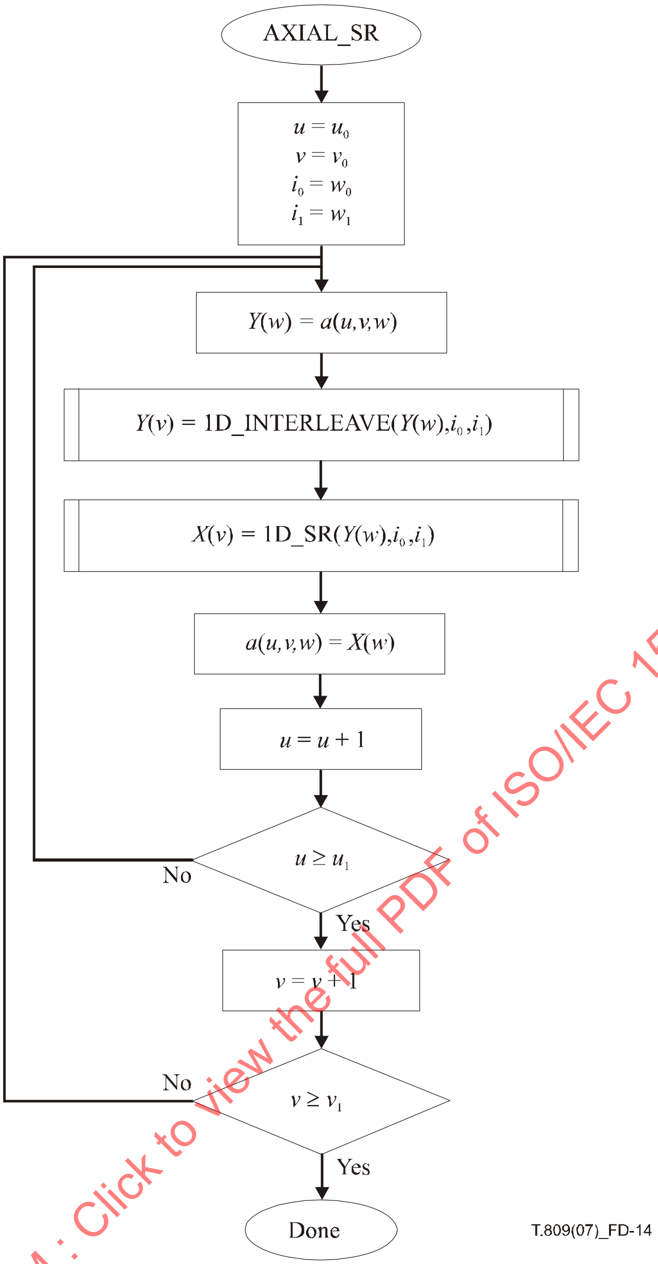


Figure D.14 – AXIAL_SR procedure

D.4.8 The 1D_SR procedure

The 1D_SR procedure used to perform the sub-band reconstruction is given in Rec. ITU-T T.800 | ISO/IEC 15444-1 clause F.3.6. Also, all subsequent procedures and specifications, needed directly or indirectly by the 1D_SR procedure, are given in Rec. ITU-T T.800 | ISO/IEC 15444-1 Annex F.

D.5 Forward transformation (informative)

D.5.1 The FDWT procedure (informative)

The forward discrete wavelet transformation (FDWT) transforms DC-level shifted tile-component samples $I(x,y,z)$ into a set of sub-bands with coefficients $a_b(u_b,v_b,w_b)$ (FDWT procedure). The FDWT procedure (see Figure D.15) also takes as input the number of decomposition levels in each of the three dimensions and are signalled in the COD or COC markers (see A.2.2 and A.2.3).

As in Rec. ITU-T T.800 | ISO/IEC 15444-1 clause F.4.1, the FDWT procedure starts with the initialization of the variable *lev* (the current decomposition level) to 1 and assigns the three-dimensional array $I(x,y,z)$ to sub-band $a_{0LLL}(u_{0LLL},u_{1LLL},v_{0LLL},v_{1LLL},w_{0LLL},w_{1LLL})$. The 3D_SD procedure (see D.5.2) is performed at every level *lev*, where the level *lev* increases by one at each iteration, until N_L iterations are performed. The 3D_SD procedure is iterated over the $(lev - 1)$[L|X][L|X][L|X] sub-band produced at each iteration.

The complete FDWT procedure is described in detail in Figure D.16.

Similar to what is described in Rec. ITU-T T.800 | ISO/IEC 15444-1 Annex F, the generated sub-bands are signalled in the codestream in the order given below:

N_LLLL, N_LXLL, N_LLXL, N_LLLX, N_LLXX, N_LXLX, N_LXXL, N_LHLL, N_LHXL, N_LHLX, N_LHXX, N_LLHL, N_LXHL, N_LLHX, N_LXHX, N_LHHL, N_LHHX, N_LLLH, N_LXLH, N_LLXH, N_LXXH, N_LHLH, N_LHXH, N_LLHH, N_LXHH, N_LHHH, $(N_L - 1)$HLL, $(N_L - 1)$HXL, $(N_L - 1)$HLX, $(N_L - 1)$HXX, $(N_L - 1)$LHL, $(N_L - 1)$XHL, $(N_L - 1)$LHX, …, 1XHX, 1HHL, 1HHX, 1LLH, 1XLH, 1LXH, 1XXH, 1HLH, 1HXH, 1LHH, 1XHH, 1HHH.

Note that, from the list of sub-bands given above, only those sub-bands that exist, given the parameters N_{LX}, N_{LY} and N_{LZ}, will be present in the codestream. These are the exact sub-bands necessary to fully reconstruct the original tile-component. Also note that N_L[L|X][L|X][L|X] is the only [L|X][L|X][L|X] type sub-band that is signalled in the codestream.

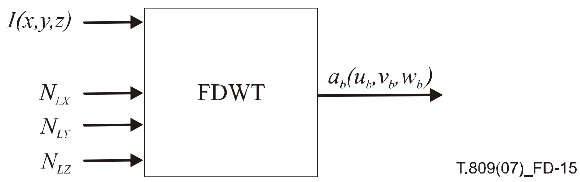


Figure D.15 – Inputs and outputs of the FDWT procedure

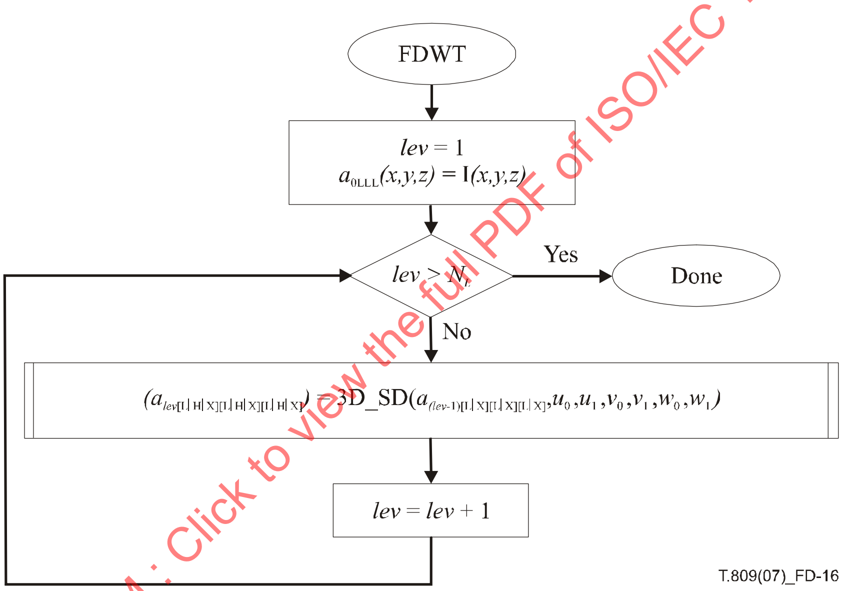


Figure D.16 – FDWT procedure

D.5.2 The 3D_SD procedure (informative)

The 3D_SD procedure performs a decomposition of a three-dimensional array of coefficients or samples $a_{(lev-1)[L|X][L|X][L|X]}(u,v,w)$ into a number of groups of sub-band coefficients $a_{lev[L|H|X][L|H|X][L|H|X]}(u,v,w)$, depending on the number of decompositions N_{LX}, N_{LY} and N_{LZ}.

The total number of coefficients of the sub-band $(lev-1)$[L|X][L|X][L|X] is equal to the sum of the total number of coefficients of the sub-bands lev[L|H|X][L|H|X][L|H|X] resulting from the 3D_SD procedure.

Figure D.17 describes the input and output parameters of the 3D_SD procedure.

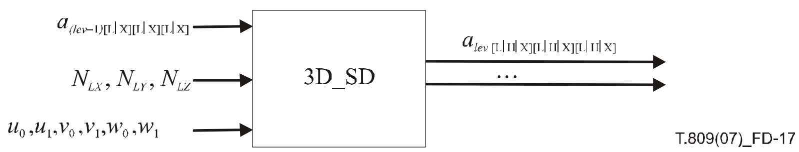


Figure D.17 – Inputs and outputs of the 3D_SD procedure

The 3D_SR procedure (see Figure D.18) decomposes each direction, *d*, for which $lev < N_{Ld} \leq N_L$, with *d being X, Y or Z,* using the HOR_SD, VER_SD and AXIAL_SD procedures respectively. Subsequently, the three-dimensional array *a(u,v,w)* is split into the different constructed sub-bands $a_{lev[L|H|X][L|H|X][L|H|X]}(u,v,w)$. Of these sub-bands, the *lev*[L|X][L|X][L|X] sub-band is used in the next iteration of the FDWT procedure for further decomposition (as long as less than N_L iterations occurred). The other sub-bands are further processed for signalling in the codestream.

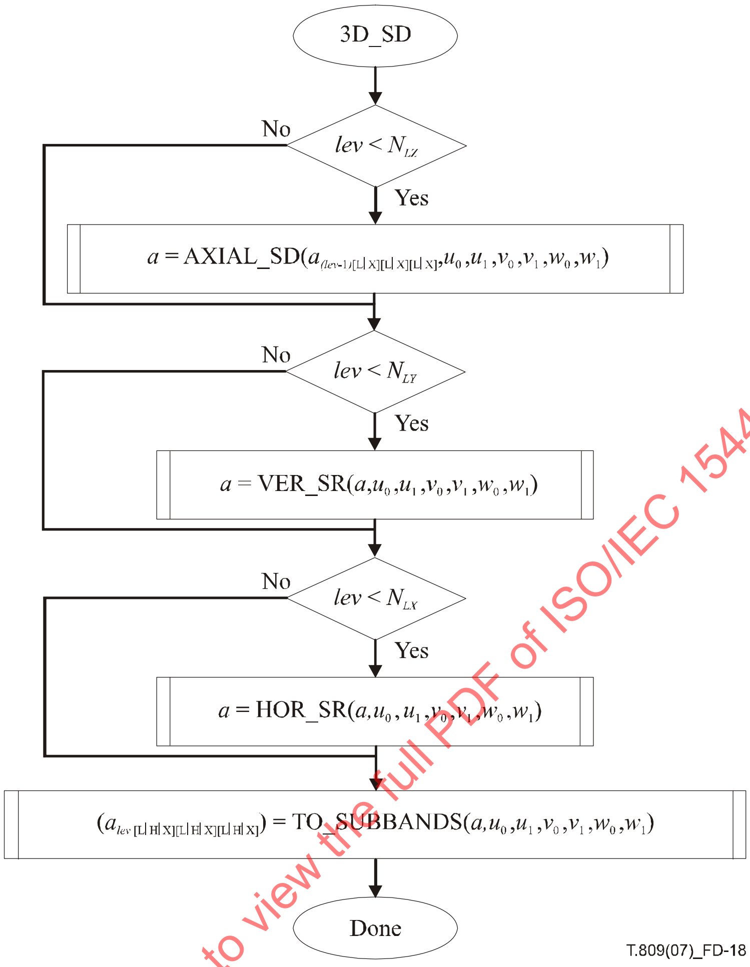


Figure D.18 – 3D_SD procedure

D.5.3 The TO_SUBBANDS procedure (informative)

The TO_SUBBANDS procedure takes a three-dimensional array *a(u,v,w)* as input and returns the sub-bands as indicated in Figure D.6 as output. The array *a(u,v,w)* has the horizontal, vertical and axial extent of the coefficients as indicated by $u_0 \leq u < u_1$, $v_0 \leq v < v_1$ and $w_0 \leq w < w_1$.

D.5.4 The AXIAL_SD procedure (informative)

The AXIAL_SD procedure performs an axial sub-band decomposition of a three-dimensional array of coefficients. It takes as input a three-dimensional array *a(u,v,w)*, the horizontal, vertical and axial extent of its coefficients as indicated by $u_0 \leq u < u_1$, $v_0 \leq v < v_1$ and $w_0 \leq w < w_1$ (see Figure D.19) and produces as output an axially filtered version of the input array, row by row and column by column. After the decomposition, a deinterleave operation is performed.

As illustrated in Figure D.20, the AXIAL_SD procedure applies the one-dimensional sub-band decomposition (1D_SD procedure) to each depth of the input array *a(u,v,w)*, and stores the result back in each depth.

Figure D.19 – Inputs and outputs of the AXIAL_SD procedure

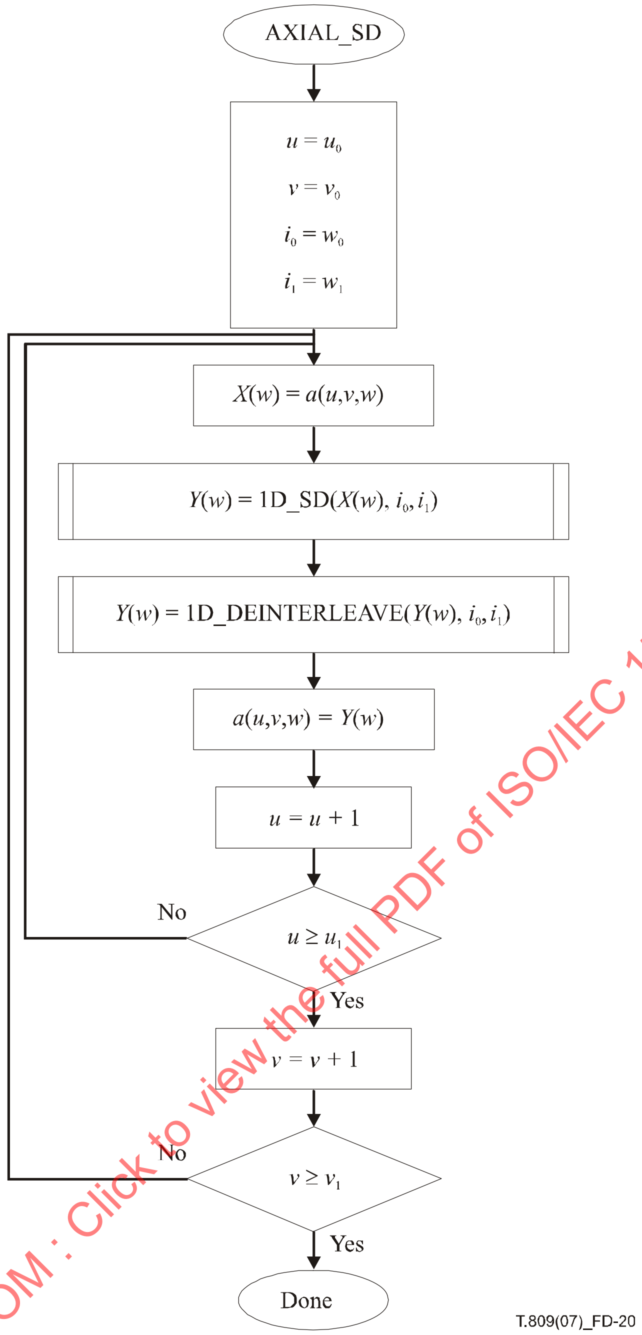


Figure D.20 – AXIAL_SD procedure

D.5.5 The VER_SD procedure (informative)

The VER_SD procedure performs a vertical sub-band decomposition of a three-dimensional array of coefficients. It takes as input a three-dimensional array *a(u,v,w)*, the horizontal, vertical and axial extent of its coefficients as indicated by $u_0 \leq u < u_1$, $v_0 \leq v < v_1$ and $w_0 \leq w < w_1$ (see Figure D.21) and produces as output a vertically filtered version of the input array, column by column and slice by slice. After the decomposition, a deinterleave operation is performed.

As illustrated in Figure D.22, the VER_SD procedure applies the one-dimensional sub-band decomposition (1D_SD procedure) to each column of the input array *a(u,v,w)*, and stores the result back in each column.

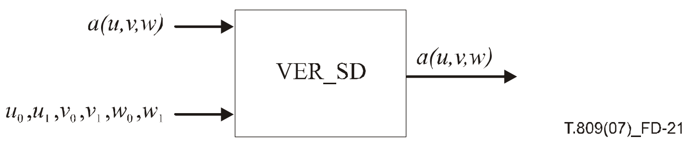


Figure D.21 – Inputs and outputs of the VER_SD procedure

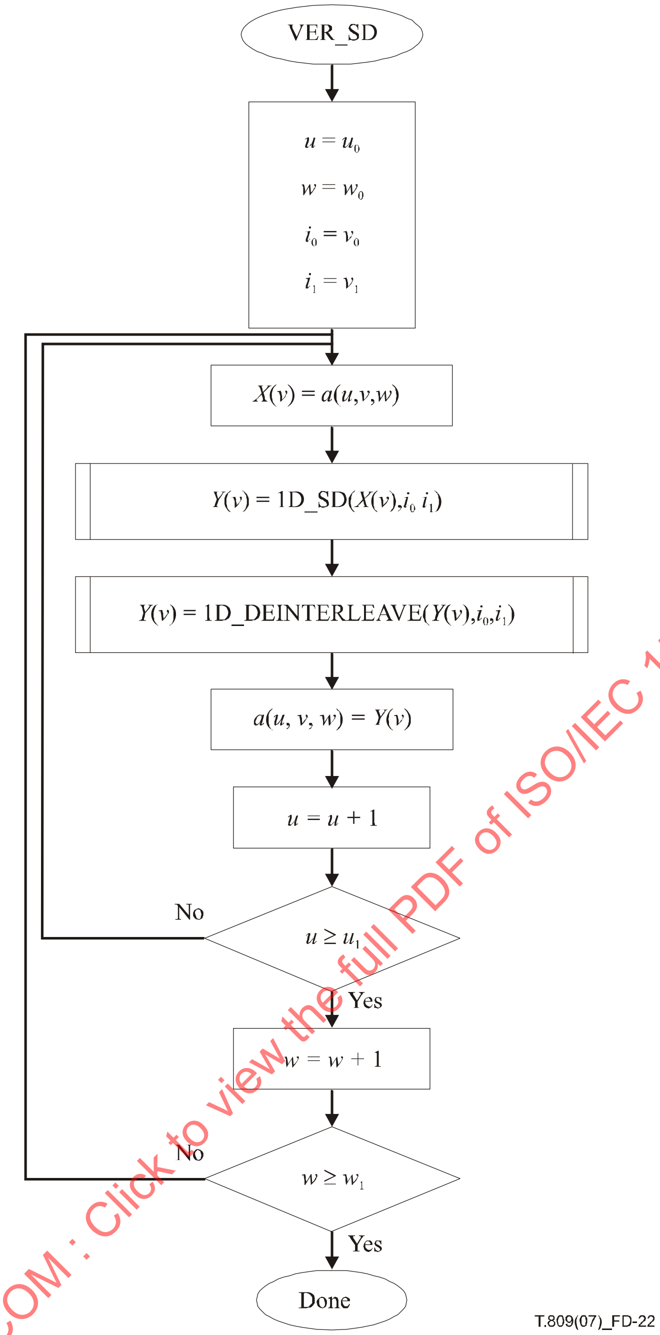


Figure D.22 – VER_SD procedure

D.5.6 The HOR_SD procedure (informative)

The HOR_SD procedure performs a horizontal sub-band decomposition of a three-dimensional array of coefficients. It takes as input a three-dimensional array *a(u,v,w)*, the horizontal, vertical and axial extent of its coefficients as indicated by $u_0 \leq u < u_1$, $v_0 \leq v < v_1$ and $w_0 \leq w < w_1$ (see Figure D.23) and produces as output a horizontally filtered version of the input array, row by row and slice by slice. After the decomposition, a deinterleave operation is performed.

As illustrated in Figure D.24, the HOR_SD procedure applies the one-dimensional sub-band decomposition (1D_SD procedure) to each row of the input array *a(u,v,w)*, and stores the result back in each row.

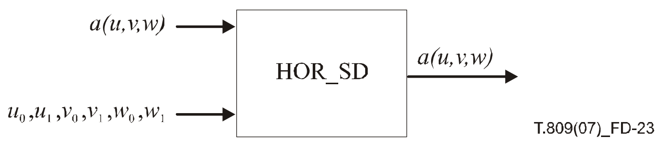


Figure D.23 – Inputs and outputs of the HOR_SD procedure

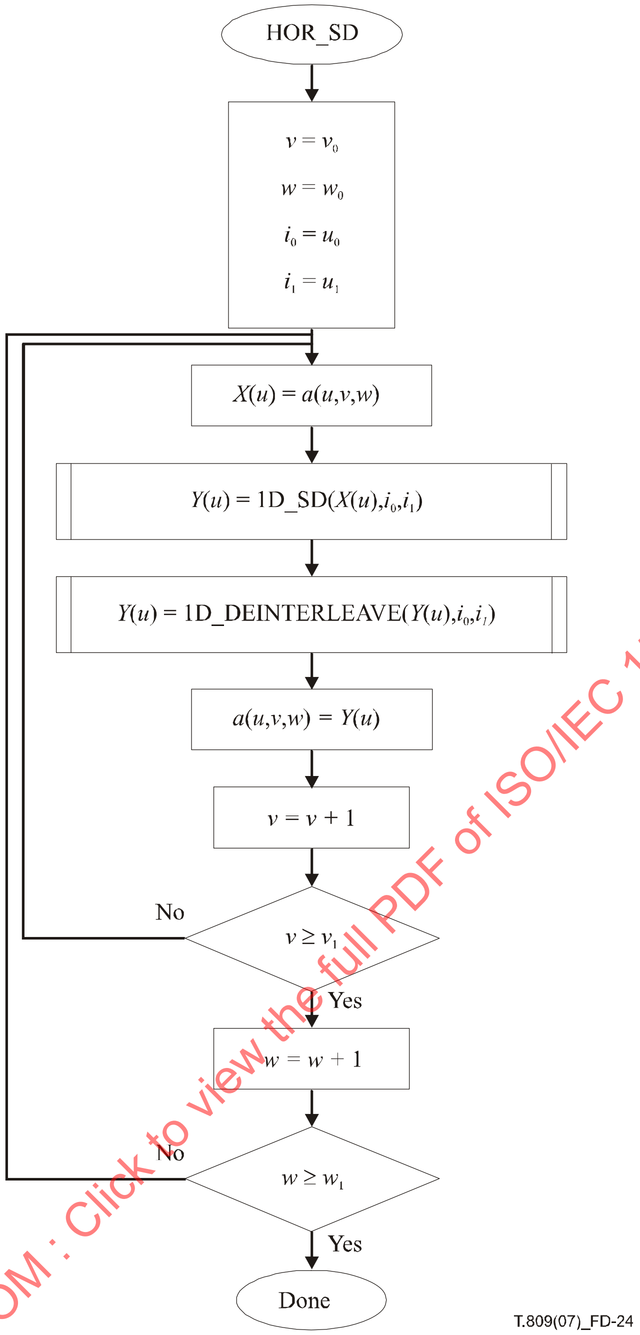


Figure D.24 – HOR_SD procedure

D.5.7 The 1D_DEINTERLEAVE procedure (informative)

As illustrated in Figure D.25, the 1D_DEINTERLEAVE procedure deinterleaves the low-pass and high-pass coefficients of a wavelet transform. It takes as input a one-dimensional signal, X(n), and rearranges the coefficient values within the signal by deinterleaving them. The values of i_0 and i_1 used by the 1D_DEINTERLEAVE procedure represent the start and the end of the signal respectively. The way the signal is deinterleaved to form the output is described by the 1D_DEINTERLEAVE procedure, given in Figure D.26.

Figure D.25 – Parameters of the 1D_DEINTERLEAVE procedure

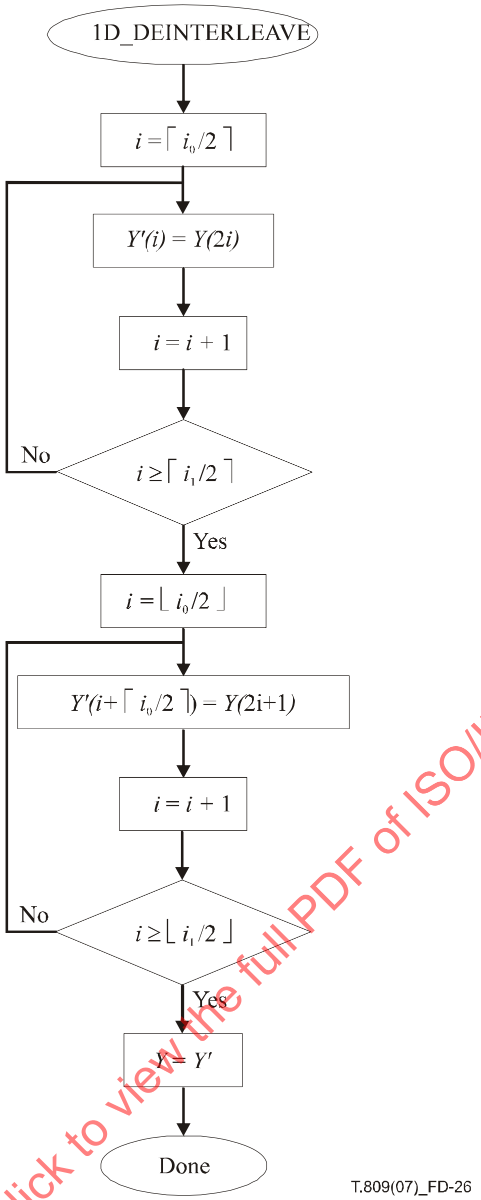


Figure D.26 – 1D_DEINTERLEAVE procedure

D.5.8 The 1D_SD procedure (informative)

The 1D_SD procedure used to perform the sub-band decomposition is given in Rec. ITU-T T.800 | ISO/IEC 15444-1 clause F.4.6. Also, all subsequent procedures and specifications, needed directly or indirectly by the 1D_SD procedure, are given in Rec. ITU-T T.800 | ISO/IEC 15444-1 Annex F.