INTERNATIONAL STANDARD

ISO/IEC 14492

First edition
2001-12-15

AMENDMENT 3
2012-12-01

Information technology — Lossy/lossless coding of bi-level images

AMENDMENT 3: Extension to colour coding

Technologies de l'information — Codage avec ou sans perte des images au trait

AMENDEMENT 3: Extension au codage de la couleur

Reference number
ISO/IEC 14492:2001/Amd.3:2012(E)

PDF disclaimer

This PDF file may contain embedded typefaces. In accordance with Adobe's licensing policy, this file may be printed or viewed but shall not be edited unless the typefaces which are embedded are licensed to and installed on the computer performing the editing. In downloading this file, parties accept therein the responsibility of not infringing Adobe's licensing policy. The ISO Central Secretariat accepts no liability in this area.

Adobe is a trademark of Adobe Systems Incorporated.

Details of the software products used to create this PDF file can be found in the General Info relative to the file; the PDF-creation parameters were optimized for printing. Every care has been taken to ensure that the file is suitable for use by ISO member bodies. In the unlikely event that a problem relating to it is found, please inform the Central Secretariat at the address given below.

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 in Switzerland

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.

Amendment 3 to ISO/IEC 14492:2001 was prepared jointly 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.88 (2000 E)/Amd.3.

INTERNATIONAL STANDARD
ITU-T RECOMMENDATION

Information technology – Lossy/lossless coding of bi-level images

Amendment 3

Extension to colour coding

1) Clause 2, Normative References

Append a new reference as follows (with the additions underlined):

– Recommendation ITU-T T.45 (2000), *Run-length colour encoding*.

2) Clause 4.2, Symbol definitions

Insert new symbols ***COLEXTFLAG***, ***CPCOMPLEN***, *CPDEFCOLS, CPEXCOLS,* ***CPNCOMP***, ***CPNVALS***, *GBCOLS,* ***GBCOMBOP***, *GBFGCOLID,* ***SBCOLS***, *SBCOLSECTSIZE and SBFGCOLID as follows (with the additions underlined)*:

(Symbols left untouched)

Clow Low-order 16 bits of C

COLEXTFLAG A parameter indicating whether the generic region segment is extended to represent a coloured bitmap

CONTEXT The values of the pixels in a template used in the generic or generic refinement decoding procedure

CPCOMPLEN The length (in bytes) of each component's value

CPDEFCOLS The default colour set

CPEXCOLS The colours defined in the colour palette segment

CPNCOMP The number of colour components

CPNVALS The number of colour values coded in this segment

(Symbols left untouched)

GB The prefix used for many of the variables associated with a generic (bitmap) region decoding procedure

GBCOLS An array containing the colours used in a generic region segment

GBCOMBOP The combination operator used in a generic region decoding procedure

GBFGCOLID The 4-byte integer indicating the foreground colour of the generic region segment

(Symbols left untouched)

SB The prefix used for many of the variables associated with a symbol (bitmap) region decoding procedure

SBCOLS An array of colours used in a text region segment

SBCOLSECTSIZE The colour section length

(Symbols left untouched)

SBDEFPIXEL The default for pixels in a text region

SBFGCOLID The three 1-byte integers indicating the foreground colour of the text region segment

(Symbols left untouched)

3) Clause 6.2.2, Input parameters

Add new rows of ***GBCOLS****,* ***GBCOMBOP*** *and* ***COLEXTFLAG*** *after* ***GBATY$_4$****, and add a new Note f), in Table 2 as follows (with the additions underlined):*

Table 2 – Parameters for the generic region decoding procedure

Name	Type	Size (bits)	Signed?	Description and restrictions
(Rows left untouched)				
GBATY$_4$	Integer	8	Y	The Y location of the adaptive template pixel A$_4$.[b)]
GBCOLS	Array of colours (colour palette)			An array containing the colours used in this generic region.[f)]
GBCOMBOP	Operator			The combination operator for this generic region. Shall take on the value REPLACE. [f)]
COLEXTFLAG	Integer	1	N	A parameter indicating whether the generic region segment is extended to represent coloured bitmap.
(Notes left untouched) [f)] Unused if **COLEXTFLAG** = 0.				

4) Clause 6.2.4, Variables used in decoding

Add a new row of GBFGCOLID after CONTEXT, and add a new Note b) in Table 4 as follows (with the additions underlined):

Table 4 – Variables used in the generic region decoding procedure

Name	Type	Size (bits)	Signed?	Description and restrictions
(Rows left untouched)				
CONTEXT	Integer	16	N	The values of the pixels in the template.[a)]
GBFGCOLID	Integer	32	N	The 4-byte integer indicating the colour palette ID of the foreground colour for the generic region segment.[b)]
(Notes left untouched) [b)] Unused if **COLEXTFLAG** = 0.				

5) New clause 6.2.7, Colour extension of generic region segment

Add a new clause 6.2.7 after clause 6.2.6, as follows:

6.2.7 Colour extension of generic region segment

If **COLEXTFLAG** is 1, after the decoding procedure described in 6.2.5 or 6.2.6, read GBFGCOLID which is a 4-byte integer indicating the colour palette ID, and set the colour specified by **GBCOLS**[GBFGCOLID] to the segment's foreground colour.

The background colour of the segment of which **COLEXTFLAG** is 1 is regarded as transparent.

NOTE – GBFGCOLID (the foreground colour information) has a fixed 4-byte field and is put at the end of the region segment only when **COLEXTFLAG** is 1. The decoder can find the address of GBFGCOLID by subtracting 4 from the segment data length recorded in the segment header.

6) Clause 6.4.2, Input parameters

*Revise the description and restrictions of **SBCOMBOP**, add new rows of **COLEXTFLAG** and **SBCOLS** after **SBRATY$_2$**, and new Notes e) and f) in Table 9 as follows (with the additions underlined):*

Table 9 – Parameters for the text region decoding procedure

Name	Type	Size (bits)	Signed?	Description and restrictions
(Rows left untouched)				
SBCOMBOP	Operator			The combination operator for this text region. May take on the values OR, AND, XOR, XNOR and REPLACE.[e]
(Rows left untouched)				
SBRATY$_2$	Integer	8	Y	The Y location of the adaptive template pixel RA$_2$.[c]
COLEXTFLAG	Integer	1	N	A parameter indicating whether the generic region segment is extended to represent coloured bitmap.
SBCOLS	Array of colours			An array containing the colours used in this text region.[f]
(Notes left untouched) [e] REPLACE operator is used if and only if **COLEXTFLAG** = 1. [f] Unused if **COLEXTFLAG** = 0.				

7) Clause 6.4.4, Variables used in decoding

Add new rows of SBCOLSECTSIZE and SBFGCOLID after HO$_I$, and Note b) after Note a) in Table 11 as follows (with the additions underlined):

Table 11 – Variables used in the text region decoding procedure

Name	Type	Size (bits)	Signed?	Description and restrictions
(Rows left untouched)				
HO$_I$	Integer	32	N	The height of *IBO$_I$*.[a]
SBCOLSECTSIZE	Integer	32	N	The colour section length.[b]
SBFGCOLID	Array of integers			An array of colour palette ID, indicating the colour of each symbol instance.[b]
[a] Unused if **SDREFINE** = 0. [b] Unused if **COLEXTFLAG** = 0.				

8) Clause 6.4.5, Decoding the text region

Insert Figure AMD3-1 after Figure 17 as follows:

Initial STRIPT values
First strip
Second strip
…
Last strip
Colour section

Figure AMD3-1 – Coded structure of a text region extended for colour text

9) Clause 6.4.5, Decoding the text region

Insert Figure AMD3-2 after Figure 20:

Colour palette IDs
Colour section length

Figure AMD3-2 – Coded structure of colour section

10) Clause 6.4.5, Decoding the text region

Insert new Notes after Figure AMD3-2:

NOTE 3bis – If **COLEXTFLAG** is 1 then a colour section, which specifies the colours of all symbol instances, is put at the end of the region as shown in Figure AMD3-1. The colour section consists of two parts as shown in Figure AMD3-2: colour palette IDs and colour section length. The colour palette IDs, which is encoded in the data structure based on Rec. ITU-T T.45, contains the information that specifies the colour of every symbol instances in the segment. The colour section length is a 4-byte field that contains the length of the colour section, in bytes.

NOTE 3ter – The background colour of the segment of which **COLEXTFLAG** is 1 is regarded as transparent.

11) Clause 6.4.5, Decoding the text region

Change the procedures itemized by 3) and 4) to 4) and 5) respectively, and insert a new procedure 3) as follows (with the modifications and additions underlined):

3) If **COLEXTFLAG** is 1, decode the colour section as described in 6.4.12.

(Procedures left untouched)

4) Decode each strip as follows:

(Procedures left untouched)

5) After all the strips have been decoded, the current contents of SBREG are the results that shall be obtained by every decoder, whether it performs this exact sequence of steps or not.

12) Clause 6.4.5, Decoding the text region

In the sub-procedure 4)-c), which is previously itemized by 3)-c), change the sub-sub-procedures itemized by ix), x) and xi) to x), xi) and xii) respectively, and insert a new sub-sub-procedure ix) after procedure viii), as follows (with the modifications and additions underlined):

ix) If **COLEXTFLAG** is 1, set the colour specified by **SBCOLS**[SBFGCOLID[NINSTANCES]] to the foreground colour of the symbol instance bitmap IB_I.

x) Draw IB_I into SBREG. Combine each pixel of IB_I with the current value of the corresponding pixel in SBREG, using the combination operator specified by **SBCOMBOP**. Write the results of each combination into that pixel in SBREG.

xi) Update CURS as follows:

(Procedures left untouched)

xii) Set:

NINSTANCES = NINSTANCES + 1

13) Clause 6.4.5, Decoding the text region

Insert a new Note at the end of 6.4.5, as follows:

NOTE 6 – The colour palette (**SBCOLS**) is created by concatenating the default colour set (CPDEFCOLS) and the additional colours (CPEXCOLS) defined in the colour palette segment referred to by this segment.

14) New clause 6.4.12, Colour section

Add a new clause 6.4.12 after clause 6.4.11.5, as follows:

6.4.12 Colour section

If **COLEXTFLAG** is 1, the colour information recorded in the colour section needs to be decoded as shown in this clause.

6.4.12.1 Colour section length

The decoding procedure of the colour section begins with achieving its top address by subtracting the SBCOLSECTSIZE from the segment data length defined in the segment header. SBCOLSECTSIZE is a 4-byte field at the end of the text region segment data part. The decoder can find the end of the data part by reading the segment data length in the segment header.

6.4.12.2 Colour palette IDs

The field of the colour palette IDs is decoded in accordance with Rec. ITU-T T.45 and the result is stored in SBFGCOLID. The number of elements in SBFGCOLID is equal to the number of symbol instances in the text region segment (**SBNUMINSTANCES**).

NOTE – If the codestream has the coloured text region segment, the ITU-T T.45 for colour palette IDs in the colour section is restricted as follows:

- The number of colour components is 1 (NCOMP = 1 in the header of ITU-T T.45 codestream).
- The length of each colour component's value is 1-byte (COMPLEN = 1 in the header of ITU-T T.45 codestream).
- The number of colour values is equal to SBNUMINSTANCES (NVALS = SBNUMINSTANCES in the header of ITU-T T.45 codestream).

15) New clause 6.8, Colour palette decoding procedure

Add a new clause 6.8 after clause 6.7.5, as follows:

6.8 Colour palette decoding procedure

6.8.1 General description

This decoding procedure is used to decode a set of colours; these colours can then be used by generic and text region decoding procedures.

6.8.2 Input parameters

The colour palette segment requires no input parameter.

The parameters to this decoding procedure are shown in Table AMD3-1.

Table AMD3-1 – Parameters for the colour palette decoding procedure

Name	Type	Size (bits)	Signed?	Description and restrictions
CPNCOMP	Integer	8	N	The number of colour components.
CPCOMPLEN	Integer	8	N	The length (in bytes) of each component's value. Make take on the values 1, 2 or 4.
CPNVALS	Integer	8	N	The number of colour values coded in this segment.

6.8.3 Return value

The variable whose value is the result of this decoding procedure is shown in Table AMD3-2.

Table AMD3-2 – Return value from the colour palette decoding procedure

Name	Type	Size (bits)	Signed?	Description and restrictions
CPDEFCOLS	Array of colours			The default colour set.
CPEXCOLS	Array of colours			The colours defined in this segment.

6.8.4 Decoding the colour palette

The data part shall be stored as a sequence of the colour value (CVAL) that consists of **CPNCOMP** fields, and each field shall consist of **CPCOMPLEN** octets.

The result of decoding is stored in CPEXCOLS, and the end of the data part shall be determined when the number of CVAL is equal to **CPNVALS**.

6.8.5 Default colour set

Thirty-two colours are defined and registered in CPDEFCOLS (default colour set), as shown in Table AMD3-3.

Table AMD3-3 – Definition of default colour set

ID	Colour name	Colour value (RGB)	Colour value (Hex)
0	Black	0, 0, 0	#000000
1	Gray	128, 128, 128	#808080
2	Silver	192, 192, 192	#c0c0c0
3	White	255, 255, 255	#ffffff
4	Red	255, 0, 0	#ff0000
5	Lime	0, 255, 0	#00ff00
6	Blue	0, 0, 255	#0000ff
7	Yellow	255, 255, 0-	#ffff00
8	Aqua	0, 255, 255	#00ffff
9	Fuchsia	255, 0, 255	#ff00ff
10	Maroon	128, 0, 0	#800000
11	Green	0, 128, 0	#008000
12	Navy	0, 0, 128	#000080
13	Olive	128, 128, 0	#808000
14	Teal	0, 128, 128	#008080
15	Purple	128, 0, 128	#800080
16	Orange	288, 168, 0	#ffa500
17		204, 204, 0	#cccc00
18		153, 0, 0	#990000
19		0, 204, 0	#00cc00
20		0, 153, 0	#009900
21		204, 204, 0	#cccc00
22		153, 153, 0	#999900
23		102, 0, 0	#660000
24		0, 0, 204	#0000cc
25		0, 0, 153	#000099
26		204, 0, 204	#cc00cc
27		153, 0, 153	#990099
28		0, 204, 204	#00cccc
29		0, 153, 153	#009999

Table AMD3-3 – Definition of default colour set

ID	Colour name	Colour value (RGB)	Colour value (Hex)
30		102, 102, 102	#666666
31		153, 153, 153	#999999

NOTE 1 – The default colour set in Table AMD3-3 includes the standard colours defined in the Cascading Style Sheets Level 2 Revision 1 (CSS 2.1) Specification (http://www.w3.org/TR/CSS2/).

NOTE 2 – If there is no region segment in which the colour palette ID or the largest element of the colour palette IDs is equal to or smaller than 32, the elements recorded in the colour palette segment are never used in any region segment. Therefore, the colour palette sections are not always required even if the documents include segments with colour extension.

16) Clause 7.3, Segment types

Add a new row of "54 Colour palette" after "53 Tables" as follows (with the additions underlined):

53 Tables – see 7.4.13

54 Colour palette – see 7.4.16

17) Clause 7.3.1, Rules for segment references

Revise the 3rd rule as follows (with the revision underlined):

- A segment of type "intermediate text region", "immediate text region" or "immediate lossless text region" (type 4, 6 or 7) may refer to any number of segments of type "symbol dictionary" and to up to eight segments of type "tables". Additionally, it may refer to any number of segments of type "colour palette segment", if it has **COLEXTFLAG** = 1 in its region segment flags.

18) Clause 7.3.1, Rules for segment references

Revise the 6th rule as follows (with the revision underlined):

- A segment of type "intermediate generic region", "immediate generic region" or "immediate lossless generic region" (type 36, 38 or 39) must not refer to any other segment. If it has **COLEXTFLAG** = 1 in its region segment flags, however, it may refer to any number of segments of the type "colour palette segment".

19) Clause 7.3.1, Rules for segment references

Insert a new rule at the end of 7.3.1, as follows:

- A segment of type "colour palette" (type 54) must not refer to any other segments.

20) Clause 7.4.1.5, Region segment flags

Replace Figure 29 with:

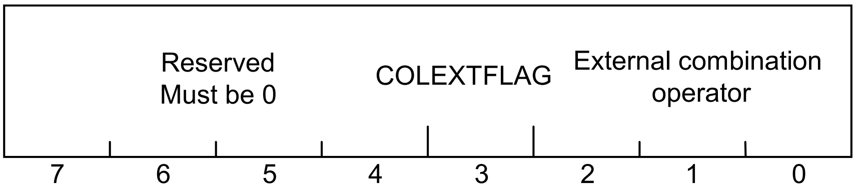

Figure 29 – Region segment flags field structure

21) Clause 7.4.1.5, Region segment flags

Insert a new Note after Note 2 as follows:

NOTE 3 – If the colour extension flag (**COLEXTFLAG**) is equal to 1, the external combination operator must be REPLACE.

22) Clause 7.4.1.5, Region segment flags

*Add a new notation of Bit 3 for **COLEXTFLAG** (Colour Extension Flag), and change the reserved bits to Bits 4-7 as follows (with the additions and revisions underlined):*

Bit 3 Colour extension flag (**COLEXTFLAG**)

This field specifies whether the region segment is extended to represent coloured bitmap.

Bits 4-7 Reserved; must be **0**.

23) Clause 7.4.3.2, Decoding a text region segment

*Add new rows of **COLEXTFLAG** and **SBCOLS** below **SBRATY$_2$** in Table 31 as follows (with the additions underlined):*

Table 31 – Parameters used to decode a text region segment

Name	Value
(Rows left untouched)	
SBRATY$_2$	See 7.4.3.1.3.
COLEXTFLAG	A parameter indicating whether the generic region segment is extended to represent coloured bitmap.
SBCOLS	Concatenation of the default colour set and the additional colours from the colour palette segments referred to by this segment.

24) Clause 7.4.6.4, Decoding a generic region segment

*Add new rows of **COLEXTGLAF** and **GBCOLS** below **GBATY$_2$** in Table 34 as follows (with the additions underlined):*

Table 34 – Parameters used to decode a generic region segment

Name	Value
(Rows left untouched)	
GBATY$_2$	See 7.4.6.3.
COLEXTFLAG	A parameter indicating whether the generic region segment is extended to represent coloured bitmap.
GBCOLS	Concatenation of the default colour set and the additional colours from the colour palette segments referred to by this segment.

25) Clause 7.4.8.5, Page segment flags

Replace Figure 53 with:

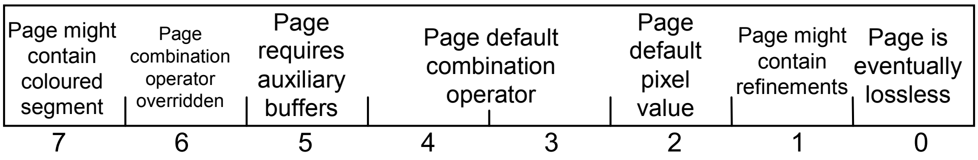
Page might contain coloured segment
Page combination operator overridden
Page requires auxiliary buffers
Page default combination operator
Page default pixel value
Page might contain refinements
Page is eventually lossless
7
6
5
4
3
2
1
0

Figure 53 – Page segment flags field structure