

ISO/IEC 14492:2001-12 (E)

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

Contents

Page

Reference number INTERNATIONAL STANDARD 14492 First edition 2001-12-15 Information technology -- Lossy/lossless coding of bi-level images Technologies de l'information -- Codage avec/sans perte d'images à deux niveaux 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. or mechanical, including photocopying and microfilm, without permission in writing from either ISO at the address below or ISO's member body in the country of the requester. 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.ch Web www.iso.ch CONTENTS 0 Introductionviii

0.1	Interpretation and use of the requirements	viii
0.1.1	Subject matter for JBIG2 coding	viii
0.1.2	Relationship between segments and documents	ix
0.1.3	Structure and use of segments	ix
0.1.4	Internal representations	ix
0.1.5	Decoding results	xi
0.1.6	Decoding procedures	xi
0.2	Lossy coding	xii
0.2.1	Symbol coding	xii
0.2.2	Generic coding	xii
0.2.3	Halftone coding	xiii
0.2.4	Consequences of inadequate segmentation	xiii
1	Scope	1
2	Normative References	1
3	Terms and Definitions	1
4	Symbols and Abbreviations	3
4.1	Abbreviations	3
4.2	Symbol definitions	4
4.3	Operator definitions	10
5	Conventions	10
5.1	Typographic conventions	10
5.2	Binary notation	10
5.3	Hexadecimal notation	11
5.4	Integer value syntax	11
5.4.1	Bit packing	11
5.4.2	Multi-byte values	11
5.4.3	Bit numbering	11
5.4.4	Signedness	11
5.5	Array notation and conventions	11

5.6	Image and bitmap conventions	11
6	Decoding Procedures	12
6.1	Introduction to decoding procedures	12
6.2	Generic region decoding procedure	13
6.2.1	General description	13
6.2.2	Input parameters	13
6.2.3	Return value	13
6.2.4	Variables used in decoding	14
6.2.5	Decoding using a template and arithmetic coding	14
6.2.6	Decoding using MMR coding	18
6.3	Generic Refinement Region Decoding Procedure	19
6.3.1	General description	19
6.3.2	Input parameters	19
6.3.3	Return value	19
6.3.4	Variables used in decoding	20
6.3.5	Decoding using a template and arithmetic coding	20
6.4	Text Region Decoding Procedure	23
6.4.1	General description	23
6.4.2	Input parameters	23
6.4.3	Return value	24
6.4.4	Variables used in decoding	24
6.4.5	Decoding the text region	25
6.4.6	Strip delta T	28
6.4.7	First symbol instance S coordinate	28
6.4.8	Subsequent symbol instance S coordinate	28
6.4.9	Symbol instance T coordinate	29
6.4.10	Symbol instance symbol ID	29
6.4.11	Symbol instance bitmap	29
6.5	Symbol Dictionary Decoding Procedure	30
6.5.1	General description	30
6.5.2	Input parameters	30
6.5.3	Return value	30
6.5.4	Variables used in decoding	30
6.5.5	Decoding the symbol dictionary	32
6.5.6	Height class delta height	34
6.5.7	Delta width	34
6.5.8	Symbol bitmap	34
6.5.9	Height class collective bitmap	37
6.5.10	Exported symbols	37
6.6	Halftone Region Decoding Procedure	38
6.6.1	General description	38
6.6.2	Input parameters	38
6.6.3	Return value	39
6.6.4	Variables used in decoding	39
6.6.5	Decoding the halftone region	39
6.7	Pattern Dictionary Decoding Procedure	42
6.7.1	General description	42
6.7.2	Input parameters	42
6.7.3	Return value	42
6.7.4	Variables used in decoding	43
6.7.5	Decoding the pattern dictionary	43
7	Control Decoding Procedure	44
7.1	General description	44
7.2	Segment header syntax	45
7.2.1	Segment header fields	45
7.2.2	Segment number	45
7.2.3	Segment header flags	45
7.2.4	Referred-to segment count and retention flags	45
7.2.5	Referred-to segment numbers	47
7.2.6	Segment page association	47

7.2.7	Segment data length	47
7.2.8	Segment header example	47
7.3	Segment types	48
7.3.1	Rules for segment references	49
7.3.2	Rules for page associations	50
7.4	Segment syntaxes	50
7.4.1	Region segment information field	50
7.4.2	Symbol dictionary segment syntax	51
7.4.3	Text region segment syntax	56
7.4.4	Pattern dictionary segment syntax	66
7.4.5	Halftone region segment syntax	67
7.4.6	Generic region segment syntax	70
7.4.7	Generic refinement region syntax	72
7.4.8	Page information segment syntax	73
7.4.9	End of page segment syntax	76
7.4.10	End of stripe segment syntax	76
7.4.11	End of file segment syntax	76
7.4.12	Profiles segment syntax	76
7.4.13	Code table segment syntax	77
7.4.14	Extension segment syntax	77
7.4.15	Defined extension types	77
8	Page Make-up	78
8.1	Decoder model	78
8.2	Page image composition	78
Annex A - Arithmetic Integer Decoding Procedure		82
A.1	General description	82
A.2	Procedure for decoding values (except IAID)	82
A.3	The IAID decoding procedure	84
Annex B - Huffman Table Decoding Procedure		86
B.1	General description	86
B.2	Code table structure	86
B.2.1	Code table flags	87
B.2.2	Code table lowest value	87
B.2.3	Code table highest value	87
B.3	Assigning the prefix codes	87
B.4	Using a Huffman table	88
B.5	Standard Huffman tables	89
Annex C - Gray-scale Image Decoding Procedure		97
C.1	General description	97
C.2	Input parameters	97
C.3	Return value	97
C.4	Variables used in decoding	97
C.5	Decoding the gray-scale image	98
Annex D - File Formats		99
D.1	Sequential organisation	99
D.2	Random-access organisation	99
D.3	Embedded organisation	100
D.4	File header syntax	100
D.4.1	ID string	100
D.4.2	File header flags	100
D.4.3	Number of pages	100
Annex E - Arithmetic Coding		101

E.1	Binary encoding	101
E.1.1	Recursive interval subdivision	101
E.1.2	Coding conventions and approximations	101
E.2	Description of the arithmetic encoder	102
E.2.1	Encoder code register conventions	103
E.2.2	Encoding a decision (ENCODE)	103
E.2.3	Encoding a 1 or 0 (CODE1 and CODE0)	103
E.2.4	Encoding an MPS or LPS (CODEMPS and CODELPS)	104
E.2.5	Probability estimation	105
E.2.6	Renormalisation in the encoder (RENORME)	105
E.2.7	Compressed data output (BYTEOUT)	106
E.2.8	Initialisation of the encoder (INITENC)	107
E.2.9	Termination of encoding (FLUSH)	107
E.2.10	Minimisation of the compressed data	107
E.3	Arithmetic decoding procedure	109
E.3.1	Decoder code register conventions	111
E.3.2	Decoding a decision (DECODE)	111
E.3.3	Renormalisation in the decoder (RENORMD)	111
E.3.4	Compressed data input (BYTEIN)	111
E.3.5	Initialisation of the decoder (INITDEC)	114
E.3.6	Resynchronisation of the decoder	114
E.3.7	Resetting arithmetic coding statistics	115
E.3.8	Saving arithmetic coding statistics	115
Annex F - Profiles		116
Annex G - Arithmetic Decoding Procedure (Software Conventions)		119
Annex H - Datastream Example and Test Sequence		121
H.1	Datastream example	121
H.2	Test sequence for arithmetic coder	142
Bibliography		149
Annex I - Patents		147