COVERING NOTE

GENERAL SECRETARIAT INTERNATIONAL TELECOMMUNICATION UNION

Geneva, 16 December 2004

ITU – TELECOMMUNICATION
STANDARDIZATION SECTOR

Subject: Erratum 1 (12/2004) to

ITU-T Recommendation T.88 (02/2000), Amendment 1 (06/2003), *Information technology – Lossy/lossless coding of bi-level images, Encoder*

1) Clause 10

*In the last sentence, replace "*7.4*" with "*7.4.1*".*

2) Clause 11

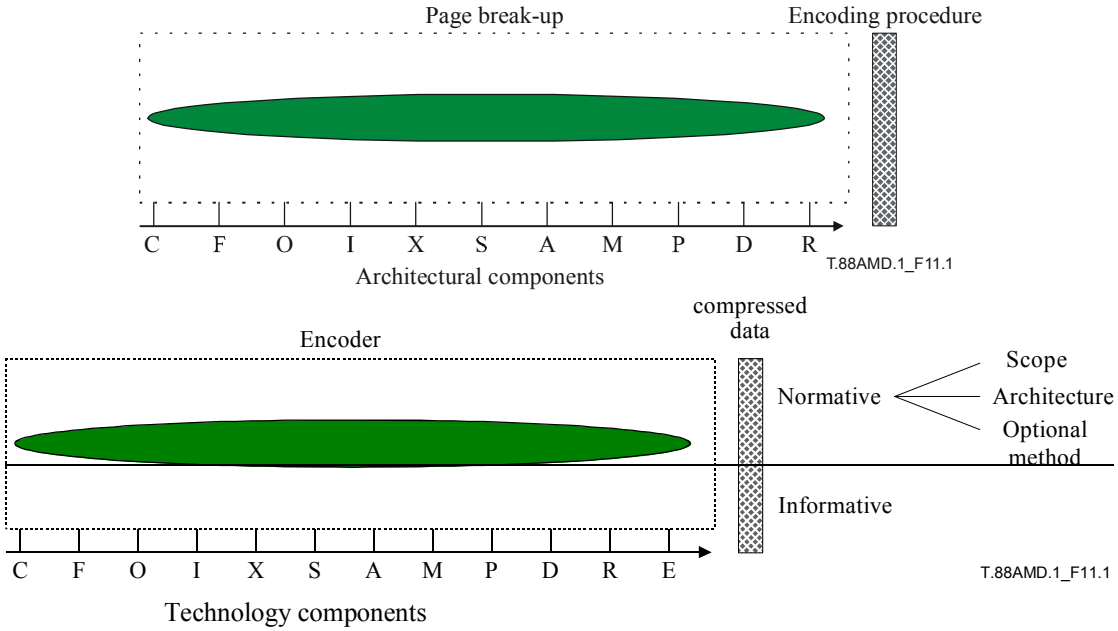
a) Modify clauses 11 *and* 11.1 *as follows:*

11 Page break-up

The page break-up ("Front end") procedures in this clause ~~9~~ are conceptually the inverse of the page make-up ("Back end") procedures already described in clause 8. However, page~~Page~~ break-up also requires additional page and document decomposition steps prior to encoding.

11.1 Page break-up architecture~~Encoder model~~

This clause describes the JBIG2 ~~encoding architecture~~encoder break-up defined by compliant, but optional, technical 'components' (with a range of 'algorithms' possible to implement each of these components). These JBIG2 page break-up~~encoding~~ components are a set of processing steps labelled: Capture, Filter, Orient (de-skew), Identify, eXtract, Screen, Align (register), Match, Post-match, Dictionary (optimize), and Refine ~~and Encode (bitstream generation)~~. An example sequence of this component set is illustrated in the ~~Encoder~~ Architectural~~e~~ Components figure below as the horizontal axis with abbreviated labels C F O I X S A M P D R ~~E~~ (leading from input on the left to a compressed data stream on the right). The vertical dimension above each label represents the range of possible algorithms that may be used to implement each component. The horizontal band illustrates an example JBIG2 compliant ~~encoding~~ page break-up method, using some algorithm for each architectural component ~~of this JBIG2 encoding~~ and spanning overall ~~these~~ components.



A compliant JBIG2 encoder need not include all architectural components, nor use them in exactly the above sequence.

b) *In 11.2, replace "*11.2.12*" with "*11.2.11*".*

c) *In 11.2.1, add a bracket as follows:*

... (e.g., a bitmap rasterized from a document created using a typical word processor), ...

d) *In 11.2.4, replace "*line-arts*" with "*line-art*".*

e) *In 11.2.10, replace "*any other symbols*" with "*more than one mark*".*

f) *Delete clause 11.2.12.*

3) Annex J

a) *Modify the second bullet as follows:*

– minimum resolution of 300 dpi (to reduce matching error and accommodate a number of encoder system simplifications, which would otherwise be required for 200-dpi images);

b) *In the second paragraph, modify last sentence as follows:*

Selecting one of each method ~~will~~ should result in reasonable encoder performance.

c) *Modify* "J.1 List of JBIG2 encoding components and corresponding algorithms" *as follows:*

Architectural component		~~Norm/ Inform~~	Component method	Method reference (Found in J.2)
Scanned	Capture (rasterize) – Resolution – Striping	~~N~~	Digitize from sensor to bi-tonal bitmap – 300 dpi – 2 or more	 – None – None
	Filter – Fly-speck remove – Quantization error remove	~~N~~	 – Isolated mismatched pixels method – Single protruding pixel method	 – Figure 3 of [J1] – Page 217 of [J1]
	Orient (de-skew)	~~N~~	Adjacent mark-based slope detection, Hough transform	– Pages 357-372 of [J2]
	Identify (Region) – Segment – Classify	~~N~~	 – Recursive x-y cut – Texture analyzer	 – Pages 372-384 of [J2] – Pages 385-388 of [J2]
	Extract – Isolate (black shapes) – Truncate (by size) – Force (to avoid generic region encoding)	~~N~~	 – 8 connected boundary trace/rubout-based region-fill approach – Min/max of shape w & h – Not required (default)	 – Pages 320-325 of [J2], [J3] – [J3] – None
	Screen (means of speeding up match)	~~N~~	Comparison of height, width and/or distribution of black pixels	– Pages 332-333 of [J2]
	Align (register)	~~N~~	Alignment of symbols using centroid	– Pages 332-333 of [J2]
	Match – Lossy/Lossless – Lossless	~~N~~	 – Weighted XOR (WXOR) plus CSIS (combined size-independent strategy)[b] /CTM (compression-based template matching)[c] – XOR (Hamming distance = 0)	 – Pages 325-332 of [J2], [J3] and [J4] – None
	Post-match – Symbol optimize – Symbol encoding – Symbol placement optimization	~~N~~	 – 'Best' dictionary symbol shape (not just simple average of the marks) – Direct encoding (no refinement) – Align symbol bottoms	 – None – None – None
	Dictionary – Singletons – Page spanning	~~N~~	 – In dictionary[d] (default) – Sequentially incremental[e] (default)	 – None – None
	Refine	~~N~~	Arithmetic (inverse of JBIG2 decoder process)	– [J4], ITU-T Rec. T.88 | ISO/IEC 14492
	Encode (bitstream generation)	~~N~~	Arithmetic or Huffman (inverse of JBIG2 decoder process)	– ITU-T Rec. T.88 | ISO/IEC 14492

Architectural component		~~Norm/ Inform~~	Component method	Method reference (Found in J.2)
Generated	Capture (rasterize) – at page or character level – Resolution – Striping	~~N~~	Rasterize from generated to bi-tonal bitmap – at page or character level – 300 dpi – 2 or more	 – None – None
	Filter	~~N~~	N/A	N/A
	Orient (de-skew)	~~N~~	N/A	N/A
	~~Region~~ Identify (Region) – Segment – Classify	~~N~~	 – Recursive x-y cut – Texture analyzer	 – Pages 372-384 of [J2] – Pages 385-388 of [J2]
	Extract – Isolate (black shapes) – Truncate (by size) – Force (to avoid generic region encoding)	~~N~~	 – 8 connected boundary trace/rub-based region-fill approach – Min/max of shape w & h – Not required (default)	 – Pages 320-325 of [J2], [J3] – [J3] – None
	Screen (means of speeding up match)	~~N~~	Comparison of height, width and/or distribution of black pixels	Pages 332-333 of [J2]
	Align (register)	~~N~~	Identical (Bounding box is identical)	None
	Match[a)]	~~N~~	XOR (Hamming distance = 0)	None
	Post-match – Symbol optimize – Symbol encoding – Symbol placement optimization	~~N~~	 – N/A – Direct encoding (no refinement) – N/A	 – N/A – None – N/A
	Dictionary – Singletons – Page spanning	~~N~~	 – In dictionary[d)] (default) – Sequentially incremental[e)] (default)	 – None – None
	Refine	~~N~~	N/A	N/A
	Encode (bitstream generation)	~~N~~	Arithmetic or Huffman (inverse of JBIG2 decoder process)	– ITU-T Rec. T.88 | ISO/IEC 14492

a) For some cases of JBIG2 encoding from Generated data, it may be practical to also eliminate the Extract, Screen~~ing~~ and Align ~~Registration~~ components by implementing the Match~~ing~~ component before ~~rasterization, which is typically part of~~ the Capture (rasterize) component.

b) Add CSIS to reduce substitution errors of WXOR.

c) Select WXOR+CSIS for less computation than CTM.

d) Minimizing number of Singletons in the dictionary will reduce dictionary memory requirements, and optimizing the sizes (h, w) of Singletons in the dictionary may increase the compression.

e) Random page access is ~~best~~ enabled if, for example, one uses a single multi-page-spanning 'common' dictionary, plus multiple 'page unique' dictionaries.