

# INTERNATIONAL STANDARD

**ISO/IEC  
14492**

First edition  
2001-12-15

**AMENDMENT 2**  
2003-12-15

---

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

AMENDMENT 2: Extension of adaptive  
templates for halftone coding

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

*AMENDEMENT 2: Extensions des modèles adaptatifs pour le codage  
des demi-teintes*

STANDARDSISO.COM : Click to view the PDF of ISO/IEC 14492:2001/AMD2:2003

---

---

---

Reference number  
ISO/IEC 14492:2001/Amd.2:2003(E)



© ISO/IEC 2003

**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/IEC 2003

All rights reserved. Unless otherwise specified, no part of this publication may be reproduced or utilized in any form or by any means, electronic 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.org](mailto:copyright@iso.org)  
Web [www.iso.org](http://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 2 to ISO/IEC 14492:2001 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.88 (2000)/Amd.2(06/2003).



INTERNATIONAL STANDARD  
ITU-T RECOMMENDATION

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

Amendment 2

Extension of adaptive templates for halftone coding

1) Subclause 4.2

a) *Eight symbols for additional adaptive template pixels (from  $A_5$  to  $A_{12}$ ) are added after  $A_4$  as follows (with the additions underlined):*

A<sub>1</sub>, A<sub>2</sub>, A<sub>3</sub>, A<sub>4</sub>, A<sub>5</sub>, A<sub>6</sub>, A<sub>7</sub>, A<sub>8</sub>, A<sub>9</sub>, A<sub>10</sub>, A<sub>11</sub>, A<sub>12</sub>

Adaptive template pixels in the generic region decoding procedure

b) *A new symbol "EXTTEMPLATE" is inserted after the symbol "EXRUNLENGTH" as follows (with the additions underlined):*

EXRUNLENGTH The length of a run of identical export flag values

EXTTEMPLATE A parameter indicating whether extended reference template is used in a generic region decoding procedure

c) *New 16 symbols "GBATX<sub>i</sub>" and "GBATY<sub>i</sub>" ( $i=\{5, \dots, 12\}$ ) are inserted after "GBATY<sub>4</sub>" as follows (with the additions underlined):*

GBATY<sub>4</sub> The Y location of adaptive template pixel 4 in a generic region decoding procedure

GBATX<sub>5</sub> The X location of adaptive template pixel 5 in a generic region decoding procedure

GBATY<sub>5</sub> The Y location of adaptive template pixel 5 in a generic region decoding procedure

GBATX<sub>6</sub> The X location of adaptive template pixel 6 in a generic region decoding procedure

GBATY<sub>6</sub> The Y location of adaptive template pixel 6 in a generic region decoding procedure

GBATX<sub>7</sub> The X location of adaptive template pixel 7 in a generic region decoding procedure

GBATY<sub>7</sub> The Y location of adaptive template pixel 7 in a generic region decoding procedure

GBATX<sub>8</sub> The X location of adaptive template pixel 8 in a generic region decoding procedure

GBATY<sub>8</sub> The Y location of adaptive template pixel 8 in a generic region decoding procedure

GBATX<sub>9</sub> The X location of adaptive template pixel 9 in a generic region decoding procedure

GBATY<sub>9</sub> The Y location of adaptive template pixel 9 in a generic region decoding procedure

GBATX<sub>10</sub> The X location of adaptive template pixel 10 in a generic region decoding procedure

GBATY<sub>10</sub> The Y location of adaptive template pixel 10 in a generic region decoding procedure

GBATX<sub>11</sub> The X location of adaptive template pixel 11 in a generic region decoding procedure

GBATY<sub>11</sub> The Y location of adaptive template pixel 11 in a generic region decoding procedure

GBATX<sub>12</sub> The X location of adaptive template pixel 12 in a generic region decoding procedure

GBATY<sub>12</sub> The Y location of adaptive template pixel 12 in a generic region decoding procedure

## 2) Subclause 6.2.2

In Table 2, new symbols "EXTTEMPLATE", "GBATX<sub>i</sub>" and "GBATY<sub>i</sub>" ( $i=\{5, \dots, 12\}$ ) are inserted, and the notes of the table are revised as follows (with the additions and revisions underlined):

Table 2 – Parameters for the generic region decoding procedure

Name	Type	Size (bits)	Signed?	Description and restrictions
<i>... (Leave untouched) ...</i>				
<u>TPGDON</u>	Integer	1	N	Whether typical prediction is used. <sup>a)</sup>
<u>EXTTEMPLATE</u>	Integer	1	N	Whether extended reference template is used. <sup>e)</sup>
<i>... (Leave untouched) ...</i>				
<u>GBATY<sub>4</sub></u>	Integer	8	Y	The Y location of the adaptive template pixel A <sub>4</sub> . <sup>b)</sup>
<u>GBATX<sub>5</sub></u>	Integer	8	Y	The X location of the adaptive template pixel A <sub>5</sub> . <sup>c)</sup>
<u>GBATY<sub>5</sub></u>	Integer	8	Y	The Y location of the adaptive template pixel A <sub>5</sub> . <sup>d)</sup>
<u>GBATX<sub>6</sub></u>	Integer	8	Y	The X location of the adaptive template pixel A <sub>6</sub> . <sup>d)</sup>
<u>GBATY<sub>6</sub></u>	Integer	8	Y	The Y location of the adaptive template pixel A <sub>6</sub> . <sup>d)</sup>
<u>GBATX<sub>7</sub></u>	Integer	8	Y	The X location of the adaptive template pixel A <sub>7</sub> . <sup>d)</sup>
<u>GBATY<sub>7</sub></u>	Integer	8	Y	The Y location of the adaptive template pixel A <sub>7</sub> . <sup>d)</sup>
<u>GBATX<sub>8</sub></u>	Integer	8	Y	The X location of the adaptive template pixel A <sub>8</sub> . <sup>d)</sup>
<u>GBATY<sub>8</sub></u>	Integer	8	Y	The Y location of the adaptive template pixel A <sub>8</sub> . <sup>d)</sup>
<u>GBATX<sub>9</sub></u>	Integer	8	Y	The X location of the adaptive template pixel A <sub>9</sub> . <sup>d)</sup>
<u>GBATY<sub>9</sub></u>	Integer	8	Y	The Y location of the adaptive template pixel A <sub>9</sub> . <sup>d)</sup>
<u>GBATX<sub>10</sub></u>	Integer	8	Y	The X location of the adaptive template pixel A <sub>10</sub> . <sup>d)</sup>
<u>GBATY<sub>10</sub></u>	Integer	8	Y	The Y location of the adaptive template pixel A <sub>10</sub> . <sup>d)</sup>
<u>GBATX<sub>11</sub></u>	Integer	8	Y	The X location of the adaptive template pixel A <sub>11</sub> . <sup>d)</sup>
<u>GBATY<sub>11</sub></u>	Integer	8	Y	The Y location of the adaptive template pixel A <sub>11</sub> . <sup>d)</sup>
<u>GBATX<sub>12</sub></u>	Integer	8	Y	The X location of the adaptive template pixel A <sub>12</sub> . <sup>d)</sup>
<u>GBATY<sub>12</sub></u>	Integer	8	Y	The Y location of the adaptive template pixel A <sub>12</sub> . <sup>d)</sup>
<sup>a)</sup> Unused if <u>MMR</u> = 1 <sup>b)</sup> Unused if <u>MMR</u> = 1 or <u>GBTEMPLATE</u> ≠ 0 <sup>c)</sup> Unused if <u>USESPLIT</u> = 0 or <u>MMR</u> = 1 <sup>d)</sup> Used only if <u>MMR</u> = 0 and <u>GBTEMPLATE</u> = 0 and <u>EXTTEMPLATE</u> = 1 <sup>e)</sup> Used only if <u>MMR</u> = 0 and <u>GBTEMPLATE</u> = 0				

## 3) Subclause 6.2.5.3

a) The identification number of Figure 3 is changed to "Figure 3(a)", and its caption is revised (with the additions and revisions underlined):

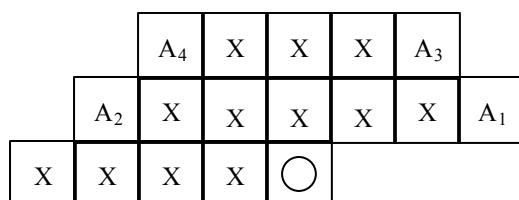


Figure 3(a) – Template when GBTEMPLATE = 0 and EXTTEMPLATE = 0, showing the AT pixels at their nominal locations

b) New Figure 3(b) is inserted as follows:

		A <sub>11</sub>	A <sub>4</sub>	A <sub>2</sub>	A <sub>5</sub>	A <sub>9</sub>	
	A <sub>12</sub>	A <sub>3</sub>	X	X	X	A <sub>6</sub>	A <sub>10</sub>
A <sub>8</sub>	A <sub>7</sub>	A <sub>1</sub>	X	○			

**Figure 3(b) – Template when GBTEMPLATE = 0 and EXTTEMPLATE = 1, showing the AT pixels at their nominal locations**

c) The second paragraph in subclause 6.2.5.3 is revised as follows (with the additions and revisions underlined):

Figure 3(a) shows the template which shall be used when **GBTEMPLATE** is 0 and **EXTTEMPLATE** is 0. Figure 3(b) shows the template which shall be used when **GBTEMPLATE** is 0 and **EXTTEMPLATE** is 1. Figure 4 shows the template which shall be used when **GBTEMPLATE** is 1. Figure 5 shows the template which shall be used when **GBTEMPLATE** is 2. Figure 6 shows the template which shall be used when **GBTEMPLATE** is 3. In each of these figures, the pixel denoted by a circle corresponds to the pixel to be coded and is not part of the template. The pixels denoted by 'X' correspond to ordinary pixels in the template. The pixels denoted A<sub>1</sub>-A<sub>12</sub> are special pixels in the template. They are denoted "adaptive" or AT pixels. These pixels are special in that their locations are not fixed, but can be placed at different locations. See 6.2.5.4 for a description of AT pixels. The legends A<sub>1</sub>-A<sub>12</sub> indicate the AT pixels 1 to 12. The pixels' actual locations are specified as parameters to this decoding procedure; Figures 3-6 show the nominal locations of these AT pixels for each template.

#### 4) Subclause 6.2.5.4

a) The second paragraph is revised as follows (with the additions and revisions underlined):

The pixels that are allowed to change are called AT pixels. Their nominal locations are indicated by 'A<sub>1</sub>', 'A<sub>2</sub>', 'A<sub>3</sub>', 'A<sub>4</sub>', 'A<sub>5</sub>', 'A<sub>6</sub>', 'A<sub>7</sub>', 'A<sub>8</sub>', 'A<sub>9</sub>', 'A<sub>10</sub>', 'A<sub>11</sub>', and 'A<sub>12</sub>' in Figures 3(a), 3(b), 4, 5 and 6. Note that some templates have fewer than sixteen AT pixels. In general, an AT pixel can be located anywhere in the field shown in Figure 7, not including the current pixel. Hence, there is the possibility to use an effective template size of 15, 14, 13, 12, 11, 10, 9, 8, 7, 6, 5, or 4 pixels by having the moved location of the AT pixel overlap a regular template pixel. The actual locations of the AT pixels for any invocation of this decoding procedure are specified as parameters to the decoding procedure. The location of the pixel A<sub>1</sub> is given by (GBATX<sub>1</sub>, GBATY<sub>1</sub>). If **GBTEMPLATE** is 0, then:

- the location of the pixel A<sub>2</sub> is given by (GBATX<sub>2</sub>, GBATY<sub>2</sub>);
- the location of the pixel A<sub>3</sub> is given by (GBATX<sub>3</sub>, GBATY<sub>3</sub>);
- and the location of the pixel A<sub>4</sub> is given by (GBATX<sub>4</sub>, GBATY<sub>4</sub>).

Additionally, if **GBTEMPLATE** is 0 and **EXTTEMPLATE** is 1, then:

- the location of the pixel A<sub>5</sub> is given by (GBATX<sub>5</sub>, GBATY<sub>5</sub>);
- the location of the pixel A<sub>6</sub> is given by (GBATX<sub>6</sub>, GBATY<sub>6</sub>);
- the location of the pixel A<sub>7</sub> is given by (GBATX<sub>7</sub>, GBATY<sub>7</sub>);
- the location of the pixel A<sub>8</sub> is given by (GBATX<sub>8</sub>, GBATY<sub>8</sub>);
- the location of the pixel A<sub>9</sub> is given by (GBATX<sub>9</sub>, GBATY<sub>9</sub>);
- the location of the pixel A<sub>10</sub> is given by (GBATX<sub>10</sub>, GBATY<sub>10</sub>);
- the location of the pixel A<sub>11</sub> is given by (GBATX<sub>11</sub>, GBATY<sub>11</sub>);
- the location of the pixel A<sub>12</sub> is given by (GBATX<sub>12</sub>, GBATY<sub>12</sub>).

b) Note 2 is revised as follows (with the additions and revisions underlined):

NOTE 2 – The indices of the AT pixels in Figures 3(a) and 3(b) correspond to the expected goodness. If moving only one AT pixel from the nominal location shown in Figure 3(a), it is advisable to move A<sub>4</sub>. The next pixel to move is A<sub>3</sub> and so on.

c) *Table 5 is revised as follows (with the additions and revisions underlined):*

**Table 5 – The nominal values of the AT pixel locations**

GBTEMPLATE		0	0	1	2	3
EXTTEMPLATE		1	0	0	0	0
GBATX <sub>1</sub>	GBATY <sub>1</sub>	<u>-2</u>	<u>0</u>	3	-1	3
GBATX <sub>2</sub>	GBATY <sub>2</sub>	<u>0</u>	<u>-2</u>	-3	-1	NA
GBATX <sub>3</sub>	GBATY <sub>3</sub>	<u>-2</u>	<u>-1</u>	2	-2	NA
GBATX <sub>4</sub>	GBATY <sub>4</sub>	<u>-1</u>	<u>-2</u>	-2	-2	NA
GBATX <sub>5</sub>	GBATY <sub>5</sub>	1	<u>-2</u>	<u>NA</u>	<u>NA</u>	<u>NA</u>
GBATX <sub>6</sub>	GBATY <sub>6</sub>	2	<u>-1</u>	<u>NA</u>	<u>NA</u>	<u>NA</u>
GBATX <sub>7</sub>	GBATY <sub>7</sub>	<u>-3</u>	<u>0</u>	<u>NA</u>	<u>NA</u>	<u>NA</u>
GBATX <sub>8</sub>	GBATY <sub>8</sub>	<u>-4</u>	<u>0</u>	<u>NA</u>	<u>NA</u>	<u>NA</u>
GBATX <sub>9</sub>	GBATY <sub>9</sub>	2	<u>-2</u>	<u>NA</u>	<u>NA</u>	<u>NA</u>
GBATX <sub>10</sub>	GBATY <sub>10</sub>	3	<u>-1</u>	<u>NA</u>	<u>NA</u>	<u>NA</u>
GBATX <sub>11</sub>	GBATY <sub>11</sub>	<u>-2</u>	<u>-2</u>	<u>NA</u>	<u>NA</u>	<u>NA</u>
GBATX <sub>12</sub>	GBATY <sub>12</sub>	<u>-3</u>	<u>-1</u>	<u>NA</u>	<u>NA</u>	<u>NA</u>

NOTE – NA means that the parameter has no nominal value.

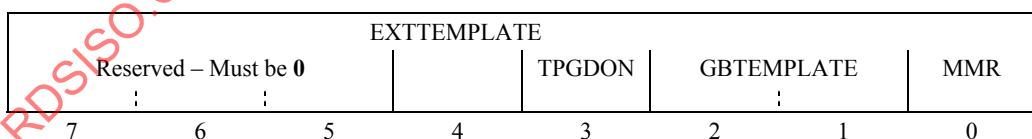
## 5) Subclause 6.2.5.7

*The step d) ii) is changed as follows (with the additions and revisions underlined):*

- d) If **LTP** = **0** then, from left to right, decode each pixel of the current row of **GBREG**. The procedure for each pixel is as follows:
  - i) If **USESskip** is **1** and the pixel in the bitmap **SKIP** at the location corresponding to the current pixel is **1**, then set the current pixel to **0**.
  - ii) Otherwise:
    - Place the template given by parameters **GBTEMPLATE**, **GBATX<sub>1</sub>** through **GBATX<sub>12</sub>** and **GBATY<sub>1</sub>** through **GBATY<sub>12</sub>** so that the current pixel is aligned with the location denoted by a circle in the figure describing the appearance of the template with identifier **GBTEMPLATE**.

## 6) Subclause 7.4.6.2

a) *Figure 46 is replaced by the following figure:*



**Figure 46 – Generic region segment flags field structure**

## 7) Subclause 7.4.6.2

*The notations of Bit 4 are added for EXTTEMPLATE, and reserved bits are changed to Bits 5-7 as follows (with the additions and revisions underlined):*

### Bit 4 EXTTEMPLATE

This field specifies whether extended reference template is used.

**Bits 5-7** Reserved; must be zero.

### 8) Subclause 7.4.6.3

a) *The first paragraph is changed as follows (with the additions and revisions underlined):*

This field is only present if **MMR** is **0**. If **GBTEMPLATE** is **0** and **EXTTEMPLATE** is **0**, it is an eight-byte field, formatted as shown in Figure 47(a) and as described below.

b) *The identification number of Figure 47 is changed to Figure 47(a) as follows (with the revision underlined):*

GBATX <sub>1</sub>	GBATY <sub>1</sub>	GBATX <sub>2</sub>	GBATY <sub>2</sub>	GBATX <sub>3</sub>	GBATY <sub>3</sub>	GBATX <sub>4</sub>	GBATY <sub>4</sub>
--------------------	--------------------	--------------------	--------------------	--------------------	--------------------	--------------------	--------------------

**Figure 47(a) – Generic region AT flags field structure when GBTEMPLATE is 0 and EXTTEMPLATE is 0**

c) *New Figure 47(b) is inserted immediately after as follows:*

GBATX <sub>1</sub>	GBATY <sub>1</sub>	GBATX <sub>2</sub>	GBATY <sub>2</sub>	•	•	•	GBATX <sub>12</sub>	GBATY <sub>12</sub>
--------------------	--------------------	--------------------	--------------------	---	---	---	---------------------	---------------------

T.88Amd.2\_F47(b)

**Figure 47(b) – Generic region AT flags field structure when GBTEMPLATE is 0 and EXTTEMPLATE is 1**

d) *The last paragraph is revised as follows (with the additions and revisions underlined):*

If **GBTEMPLATE** is **0** and **EXTTEMPLATE** is **1**, it is a 32-byte field, formatted as shown in Figure 47(b) and as described below.

Byte 0 **GBATX<sub>1</sub>**

Byte 1 **GBATY<sub>1</sub>**

Byte 2 **GBATX<sub>2</sub>**

Byte 3 **GBATY<sub>2</sub>**

Byte 4 **GBATX<sub>3</sub>**

Byte 5 **GBATY<sub>3</sub>**

Byte 6 **GBATX<sub>4</sub>**

Byte 7 **GBATY<sub>4</sub>**

Byte 8 **GBATX<sub>5</sub>**

Byte 9 **GBATY<sub>5</sub>**

Byte 10 **GBATX<sub>6</sub>**

Byte 11 **GBATY<sub>6</sub>**

Byte 12 **GBATX<sub>7</sub>**

Byte 13 **GBATY<sub>7</sub>**

Byte 14 **GBATX<sub>8</sub>**

Byte 15 **GBATY<sub>8</sub>**

Byte 16 **GBATX<sub>9</sub>**

Byte 17 **GBATY<sub>9</sub>**

Byte 18 **GBATX<sub>10</sub>**

Byte 19 **GBATY<sub>10</sub>**