INTERNATIONAL STANDARD

ISO/IEC 23002-1

First edition
2006-12-01

Information technology — MPEG video technologies —

Part 1: Accuracy requirements for implementation of integer-output 8×8 inverse discrete cosine transform

Technologies de l'information — Technologies vidéo MPEG —

Partie 1: Exigences d'exactitude pour l'implémentation de la transformation cosinus inverse discrète de sortie du nombre entier 8×8

Reference number
ISO/IEC 23002-1:2006(E)

Contents

Page

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 23002-1 was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information Technology*, Subcommittee SC 29, *Coding of audio, picture, multimedia and hypermedia information*.

ISO/IEC 23002 consists of the following parts, under the general title *Information technology — MPEG video technologies*:

— *Part 1: Accuracy requirements for implementation of integer-output 8x8 inverse discrete cosine transform*

The following part is under preparation:

— *Part 3: Auxiliary video data representation*

Information technology — MPEG video technologies —

Part 1:
Accuracy requirements for implementation of integer-output 8x8 inverse discrete cosine transform

1 Scope

A number of image and video coding related standards (see Bibliography) include a requirement for decoders to implement an integer-output 8x8 inverse discrete cosine transform (IDCT) for the generation of inverse-transformed sample differences with a nominal range from -2^B to $(2^B)-1$ for some integer number of bits B, where B is greater than or equal to 8. This part of ISO/IEC 23002 specifies conformance requirements for establishing sufficient accuracy in such an integer-output IDCT implementation. It is intended to be suitable for reference to establish partial or complete requirements for IDCT accuracy for conformance to other standards that require IDCT use.

The accuracy requirements specified in the main body of this part of ISO/IEC 23002 are essentially the same as those previously specified in [7], in Annex A of [1], and in Annex A of [5]. These requirements have been specified herein to resolve normative references to [7] in MPEG standards after its withdrawal and to provide improved clarity for the specification of IDCT accuracy requirements.

An additional requirement on encoded-bitstream intra refresh frequency was also previously specified in [7], establishing a requirement of bitstream conformance that each macroblock be intra-coded at least once within each series of 132 times that it is coded in a predicted picture without an intervening intra picture. That additional requirement is not specified in this part of ISO/IEC 23002, in order to confine its scope to the domain of decoder conformance specification.

Some allowances for modification of the specified accuracy requirements are made within this part of ISO/IEC 23002. Additional accuracy requirements that may be invoked by a referencing specification are specified in Annexes A and B.

2 Terms and definitions

For the purposes of this document, the following terms and definitions apply.

2.1
array

indexed collection of scalar values

NOTE Square parentheses are used to indicate the indexing of arrays. For example, *s*[5] denotes the entry at index 5 in the array *s*.

2.2
discrete cosine transform
DCT

transformation that produces an output matrix of transform coefficients from an input matrix of samples using some approximation of the process specified in 5.1 or 5.2

2.3

forward discrete cosine transform

discrete cosine transform

NOTE The addition of the word “forward” in the term is used when a contrast with the concept of an inverse discrete cosine transform is intended to be emphasized

2.4

inverse discrete cosine transform
IDCT

transformation that produces an output matrix of samples from an input matrix of transform coefficients using some approximation of the process specified in 5.3 or 5.4

2.5

matrix

array with a two-dimensional index

NOTE When the matrix represents samples in a spatial sampling grid, the first (left-most) component of the index is considered to be the vertical component of the index and the second (right-most) component of the index is considered to be the horizontal component of the index. For example, entry f[3][5] in a matrix f of samples would denote the entry at vertical position 3 and horizontal position 5. When the matrix represents transform coefficients, the first (left-most) component of the index is considered to represent a vertical frequency index and the second (right-most) component of the index is considered to represent a horizontal frequency index. For example, entry F[3][5] in a matrix F of transform coefficients would denote the entry with vertical frequency index 3 and horizontal frequency index 5.

2.6

sample

entry in a matrix that is the input of a discrete cosine transform or the output of an inverse discrete cosine transform

NOTE A sample is a scalar value.

2.7

scalar value

integer or real-valued number

2.8

transform coefficient

entry in a matrix that is the output of a forward discrete cosine transform or the input of an inverse discrete cosine transform

NOTE A transform coefficient is a scalar value.

3 Abbreviations and symbols

For purposes of this document, the following listed abbreviations and symbols apply.

3.1 Abbreviations

AME: absolute mean error

DC: referring to the entry in a matrix of transform coefficients associated with the index coordinate pair [0][0].

DCT: discrete cosine transform

IDCT: inverse DCT

MSE: mean squared error

OME: overall mean error

OMSE: overall mean squared error

PAE: peak absolute error

3.2 Symbols

π: Archimedes' constant 3,141 592 653 589 793 238 462 643 …

4 Conventions

4.1 Arithmetic operators

The following arithmetic operators are defined.

+	Addition
–	Subtraction (as a two-argument operator) or negation (as a unary prefix operator)
*	Multiplication
x^y	Exponentiation. Specifies x to the power of y.
$\frac{x}{y}$	Division. Specifies division of x by y, producing a real-valued number result.

The use of 64-bit floating-point precision for the computation of arithmetic operations is acceptable for purposes of this part of this International Standard.

When an order of operations is not indicated explicitly by use of parenthesis, the following rules apply:

– exponentiation operations are considered to take place before multiplication and division;

– multiplication and division operations are considered to take place before addition and subtraction;

– multiplication and division operations in sequence are evaluated sequentially from left to right;

– addition and subtraction operations in sequence are evaluated sequentially from left to right.

4.2 Logical operators

A logical operator is defined as follows

$x ? y : z$ If the condition x is TRUE, evaluates to the value of y; otherwise, evaluates to the value of z.

4.3 Relational operators

The following relational operators are defined.

= =	Equal to
<	Less than

> Greater than

<= Less than or equal to

>= Greater than or equal to

The relational operators return the value TRUE if the expressed condition is fulfilled and otherwise return the value FALSE.

4.4 Bit-wise operators

The following bit-wise operators are defined.

& Bit-wise "and". When operating on integer arguments, operates on two's complement representations of the integer values. When operating on a binary argument that contains fewer bits than another argument, the shorter argument is extended by adding more significant bits equal to 0.

4.5 Assignment operators

An assignment operator is defined as follows.

x = *y* Assigns the variable *x* equal to the value of *y*.

The assignment operator is considered to take effect after all other operations in an equation have been completed.

4.6 Mathematical functions

The following mathematical functions are defined.

$\sqrt{x}$ is the square root of an argument *x*.

$$\text{abs}(\ x\ ) = \begin{cases} x & ; \quad x >= 0 \\ -x & ; \quad x < 0 \end{cases}$$

$$\text{clip3}(\ x, y, z\ ) = \begin{cases} x & ; \quad z < x \\ y & ; \quad z > y \\ z & ; \quad \text{otherwise} \end{cases}$$

cos(*x*) the trigonometric cosine function operating on an argument *x* in units of radians.

floor(*x*) the greatest integer less than or equal to *x*.

$$\max(\ x, y\ ) = \begin{cases} x & ; \quad x >= y \\ y & ; \quad x < y \end{cases}$$

$\text{round}_{\varepsilon}(\ x\ ) = \text{sign}(\ x\ ) * \text{floor}(\ \text{abs}(\ x\ ) + 0.5 + \varepsilon\ )$

$$\text{sign}(\ x\ ) = \begin{cases} 1 & ; \quad x >= 0 \\ -1 & ; \quad x < 0 \end{cases}$$

The use of 64-bit floating-point precision for the computation of mathematical functions is acceptable for purposes of this part of this International Standard.

4.7 Range notation

The following notation is used to specify a range of values.

x = *y* .. *z* *x* takes on integer values starting from *y* to *z* inclusive, with *x*, *y*, and *z* being integer numbers.

4.8 Hexadecimal notation

Hexadecimal notation, indicated by prefixing the hexadecimal number by "0x", may be used when the number of bits used to represent an integer value is an integer multiple of 4. For example, 0x41 represents an eight-bit string having only its second-most-significant and its least-significant bits equal to 1.

5 IDCT accuracy specification

5.1 Ideal real-valued 8x8 forward DCT

An ideal real-valued 8x8 forward DCT is defined as a transformation of an 8x8 input matrix of samples *f*[*y*][*x*] with *x* = 0..7 and *y* = 0..7 into an 8x8 output matrix of transform coefficients *F*[*v*][*u*] with *u* = 0..7 and *v* = 0..7 as specified by the following (unitary) equation.

$$F[v][u] = \sum_{x=0}^{7}\sum_{y=0}^{7} f[y][x] * \left(\frac{(u==0)?\sqrt{2}:2}{4}\right) * \cos\left(\frac{(2*x+1)*u*\pi}{16}\right) * \left(\frac{(v==0)?\sqrt{2}:2}{4}\right) * \cos\left(\frac{(2*y+1)*v*\pi}{16}\right) \quad (1)$$

5.2 Ideal integer-valued 8x8 forward DCT

An ideal integer-valued 8x8 forward DCT is defined as a transformation of an 8x8 input matrix of transform coefficients *f*[*y*][*x*] with *x* = 0..7 and *y* = 0..7 into an 8x8 output matrix of samples *F*[*v*][*u*] with *u* = 0..7 and *v* = 0..7 as specified by Equation 1 as specified in subclause 5.1, followed by rounding each of the resulting samples as follows.

$$F'[v][u] = \mathrm{round}_{\varepsilon}(\,F[v][u]\,) \quad (2)$$

where

$$\varepsilon = \begin{cases} 10^{-10} & \text{for } [v][u] \in \{\ [0][0], [0][4], [4][0], \text{or } [4][4]\ \}; \\ 0 & \text{otherwise.} \end{cases}$$

5.3 Ideal real-valued 8x8 inverse DCT

An ideal real-valued 8x8 inverse DCT is defined as a transformation of an 8x8 input matrix of transform coefficients *F*[*v*][*u*] with *u* = 0..7 and *v* = 0..7 into an 8x8 output matrix of samples *f*[*y*][*x*] with *x* = 0..7 and *y* = 0..7 as specified by the following (unitary) equation.

$$f[y][x] = \sum_{u=0}^{7}\sum_{v=0}^{7} F[v][u] * \left(\frac{(u==0)?\sqrt{2}:2}{4}\right) * \cos\left(\frac{(2*x+1)*u*\pi}{16}\right) * \left(\frac{(v==0)?\sqrt{2}:2}{4}\right) * \cos\left(\frac{(2*y+1)*v*\pi}{16}\right) \quad (3)$$

5.4 Ideal integer-valued 8x8 inverse DCT

An ideal integer-valued 8x8 inverse DCT is defined as a transformation of an 8x8 input matrix of transform coefficients *F*[*v*][*u*] with *u* = 0..7 and *v* = 0..7 into an 8x8 output matrix of samples *f*[*y*][*x*] with *x* = 0..7 and *y* = 0..7 as specified by Equation 3 as specified in subclause 5.3, followed by rounding each of the resulting samples as follows.

$$f'[y][x] = \mathrm{round}_{0}(\,f[y][x]\,) \quad (4)$$

5.5 Integer-output IDCT accuracy testing procedure and requirements

The IDCT under test is considered conforming to the main body of this part of ISO/IEC 23002 if it passes the tests specified in 5.5.1 to 5.5.2.

Unless specified otherwise by a referencing specification, the value of B shall be considered equal to 8.

The tests using pseudo-random input data specified in subclause 5.5.1 shall be performed.

The test of all-zero input behaviour specified in subclause 5.5.2 shall be performed.

5.5.1 Tests using pseudo-random input data

For the sets of values of the variables L, H, and Q specified below in this subclause, the test using pseudo-random input data specified in subclause 5.5.1.1 shall be performed using the pseudo-random number generator specified in subclause 5.5.1.2. The accuracy requirements for the tests using pseudo-random input data are as specified in subclause 5.5.1.3.

Unless specified otherwise by a referencing specification, the sets of values of the variables L, H, and Q shall be as follows.

– $L = 2^B$ and $H = (2^B)–1$ and $Q = 10\ 000$

– $L = 5*2^{(B-8)}$ and $H = 5*2^{(B-8)}$ and $Q = 10\ 000$

– $L = 300*2^{(B-8)}$ and $H = 300*2^{(B-8)}$ and $Q = 10\ 000$

5.5.1.1 Test procedure using pseudo-random input data for a set of variables *L*, *H*, and *Q*

The operations to be performed for a set of variables L, H, and Q are as follows.

1. Initialize the pseudo-random number generator specified in subclause 5.5.1.2 and use it to generate $64*Q$ pseudo-random sample values $r[i]$ for $i = 0..64*Q–1$.
2. For $x = 0..7$ and $y = 0..7$, set $p[y][x] = 0$.
3. For each value of z in the range $0..Q–1$, perform the following operations.
   a. For $x = 0..7$ and $y = 0..7$, set $f_z[y][x] = r[\ 64 * z + 8 * y + x\ ]$.
   b. Apply the process specified in subclause 5.2 to the integer-valued 8x8 sample matrix f_z to produce a corresponding integer-valued 8x8 transform coefficient matrix F'_z. At least 64 bits of floating-point accuracy shall be used for the transformation process of Equation 1.
   c. Set $F''_z[v][u] = \text{clip3}(\ –2^{(B+3)}, (2^{(B+3)})–1, F'_z[v][u]\ )$ for $u = 0..7$ and $v = 0..7$.
   d. Apply the process specified in subclause 5.4 to the integer-valued 8x8 transform coefficient matrix F''_z to produce a corresponding integer-valued 8x8 sample matrix g'_z. At least 64 bits of floating-point accuracy shall be used for the transformation process of Equation 3.
   e. For $x = 0..7$ and $y = 0..7$, set $g''_z[y][x] = \text{clip3}(\ –2^B, (2^B)–1, g'_z[y][x]\ )$.
   f. Apply the IDCT under test to the integer-valued 8x8 transform coefficient matrix F''_z to produce a corresponding integer-valued 8x8 sample matrix h'_z for testing.
   g. For $x = 0..7$ and $y = 0..7$, set $h''_z[y][x] = \text{clip3}(\ –2^B, (2^B)–1, h'_z[y][x]\ )$.
   h. For $x = 0..7$ and $y = 0..7$, set $p[y][x] = \max(\ p[y][x], \text{abs}(\ h''_z[y][x] – g''_z[y][x]\ )\ )$.
4. For $x = 0..7$ and $y = 0..7$, compute the 8x8 matrices d and e as specified by

$$d[y][x] = \frac{1}{Q}\sum_{z=0}^{Q-1}\left(h''_z[y][x] - g''_z[y][x]\right)$$

$$e[y][x] = \frac{1}{Q}\sum_{z=0}^{Q-1}\left(h''_z[y][x] - g''_z[y][x]\right)^2$$

5. Compute the values of *m* and *n* as specified by

$$m = \frac{1}{64}\sum_{x=0}^{7}\sum_{y=0}^{7} d[y][x]$$

$$n = \frac{1}{64}\sum_{x=0}^{7}\sum_{y=0}^{7} e[y][x]$$

6. The accuracy requirements specified in subclause 5.5.1.3 shall be fulfilled.

7. Set *r*[*i*] = – *r*[*i*] for *i* = 0..64**Q*–1 and repeat steps 2 through 5 above. The requirements tested in step 6 above shall again be fulfilled.

5.5.1.2 Process for pseudo-random number generation

Consider the variables *i, j*, *k,* and *s* to be 32-bit integer variables represented using the two's complement integer representation method. The following process is specified for the production of pseudo-random number values *r*[*i*] for *i* = 0..64**Q*–1.

1. Set *s* = 1.

2. For *i* = 0 to 64**Q*–1, perform the following operations.

   a. Set *s* = (*s* * 1103515245) + 12345.

   b. Set *j* = *s* & 0x7ffffffe.

   c. Set *k* = floor$\left(\left(\frac{j}{2^{31}-1}\right) * (L + H + 1)\right)$.

   d. Set *r*[*i*] = *k* – *L*.

Arithmetic operations that cause the value of *s* to exceed its 32-bit range of signed two's complement integer values are specified to result in interpretation of the least significant 32 bits of the result as the signed two's complement result.

5.5.1.3 Accuracy requirements for the tests using pseudo-random input data

Unless specified otherwise by a referencing specification, the following requirements shall be fulfilled for each test using pseudo-random input data.

1. For *x* = 0..7 and *y* = 0..7, the value of *p*[*y*][*x*], which may be referred to as the peak absolute error (PAE) for location [*y*][*x*] for the pseudo-random data test, shall be less than or equal to *P*. Unless specified otherwise by a referencing specification, the value of *P* is specified as follows.

   a. If *Q* is less than or equal to 10 000, *P* shall be equal to 1.

   b. Otherwise, *P* shall be equal to 2.

2. For *x* = 0..7 and *y* = 0..7, the value of abs(*d*[*y*][*x*]), which may be referred to as the absolute mean error (AME) for location [*y*][*x*], shall be less than or equal to *D*. Unless specified otherwise by a referencing specification, the value of *D* is specified as follows.

   a. If *Q* is less than or equal to 10 000, *D* shall be equal to 0.015.

   b. Otherwise, *D* shall be equal to 2.

3. For x = 0..7 and y = 0..7, the value of $e[y][x]$, which may be referred to as the mean squared error (MSE) for location $[y][x]$, shall be less than or equal to E. Unless specified otherwise by a referencing specification, the value of E is specified as follows.

   a. If Q is less than or equal to 10 000, E shall be equal to 0.06.

   b. Otherwise, E shall be equal to 4.

4. The value of abs(m), which may be referred to as the absolute value of the overall mean error (OME), shall be less than or equal to M. Unless specified otherwise by a referencing specification, the value of M is specified as follows.

   a. If Q is less than or equal to 10 000, M shall be equal to 0.0015.

   b. Otherwise, M shall be equal to 2.

5. The value of n, which may be referred to as the overall mean squared error (OMSE), shall be less than or equal to N. Unless specified otherwise by a referencing specification, the value of N is specified as follows.

   a. If Q is less than or equal to 10 000, N shall be equal to 0.02.

   b. Otherwise, N shall be equal to 4.

5.5.2 Test of all-zero input behaviour

The following test shall be performed.

1. For u = 0..7 and v = 0..7, set $F''[v][u]$ = 0.

2. Apply the integer-output IDCT under test to the 8x8 transform coefficient matrix F'' to produce a corresponding integer-valued 8x8 sample matrix h' for testing.

3. The following requirement shall be fulfilled: For x = 0..7 and y = 0..7, the value of $h'[y][x]$ shall be equal to 0.

