INTERNATIONAL STANDARD

ISO/IEC 23002-3

First edition
2007-10-15

Information technology — MPEG video technologies —

Part 3: Representation of auxiliary video and supplemental information

Technologies de l'information — Technologies vidéo MPEG —

Partie 3: Représentation de vidéo auxiliaire et des informations complémentaires

IECNORM.COM : Click to view the full PDF of ISO/IEC 23002-3:2007

Reference number
ISO/IEC 23002-3:2007(E)

© ISO/IEC 2007

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.

IECNORM.COM : Click to view the full PDF of ISO/IEC 23002-3:2007

COPYRIGHT PROTECTED DOCUMENT

© ISO/IEC 2007

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
Web www.iso.org

Published in Switzerland

Contents

Page

IECNORM.COM : Click to view the full PDF of ISO/IEC 23002-3:2007

© ISO/IEC 2007 – All rights reserved

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 23002-3 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 8×8 inverse discrete cosine transform*

— *Part 2: Fixed-point 8×8 IDCT and DCT*

— *Part 3: Representation of auxiliary video and supplemental information*

IECNORM.COM : Click to view the full PDF of ISO/IEC 23002-3:2007

© ISO/IEC 2007 – All rights reserved

Introduction

In this part of ISO/IEC 23002, auxiliary video streams are data coded as video sequences and supplementing a primary video sequence. Depth maps and parallax maps are the first specified types of auxiliary video streams, relating to stereoscopic-view video content.

In this context, this part of ISO/IEC 23002 specifies syntax and semantics for conveying information describing the interpretation of auxiliary video streams.

Syntax for such information is specified herein as a stream of data referred to as a supplemental information (SI) message stream. Provisions for extensibility have been included, so that additional types of data can be defined in future extensions of the current SI message stream syntax by ISO/IEC.

An SI message stream can contain several concatenated SI messages, hence conveying various types of information. The auxiliary video SI (AVSI) is the only currently-defined type of SI (other than reserved SI message types that are reserved for future specification by ISO/IEC and are to be ignored by decoders if present). An AVSI message characterizes the interpretation of an auxiliary video sequence that accompanies a primary video sequence. For instance, an AVSI can indicate that the auxiliary video represents depth map information, and can provide parameters for the proper interpretation of the auxiliary video as such depth information. The means for identifying the primary video stream and the auxiliary video stream to which these messages pertain is a system-level issue that is outside the scope of this part of ISO/IEC 23002.

Although the auxiliary video SI is the only type of SI that is currently specified herein, the SI message format has been defined in a generic fashion so that it can potentially be used for purposes other than aiding in the interpretation of auxiliary video sequences. Any kind of data could potentially be carried in the SI message format.

This part of ISO/IEC 23002 addresses two types of auxiliary video and their supplemental information: depth and parallax. They can be used to create stereoscopic images for both glasses-based displays and auto-stereoscopic displays in a large variety of products, ranging from large, very high quality television sets to tiny mobile devices. Further information on depth, parallax and the relation between them can be found in the annexes of this part of ISO/IEC 23002.

Other auxiliary video types and supplemental information messages might be defined in future extensions of this part of ISO/IEC 23002.

IECNORM.COM : Click to view the full PDF of ISO/IEC 23002-3:2007

IECNORM.COM : Click to view the full PDF of ISO/IEC 23002-3:2007

Information technology — MPEG video technologies —

Part 3: Representation of auxiliary video and supplemental information

IECNORM.COM : Click to view the full PDF of ISO/IEC 23002-3:2007

1 Scope

This part of ISO/IEC 23002 defines auxiliary video streams as data coded as video sequences and supplementing a primary video sequence. Depth maps and parallax maps are the first specified types of auxiliary video streams, relating to stereoscopic-view video content.

In this context, this part of ISO/IEC 23002 specifies syntax and semantics for conveying information describing the interpretation of auxiliary video streams.

Syntax for such information is specified herein as a stream of data referred to as a supplemental information (SI) message stream. Provisions for extensibility have been included, so that additional types of data can be defined in future extensions of the current SI message stream syntax by ISO/IEC.

2 Terms and definitions

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

2.1
auxiliary picture
picture that supplements a *primary picture* that may be used in combination with other data specified by this part of ISO/IEC 23002 in the display process

NOTE Depth maps are an example of *auxiliary pictures.*

2.2
auxiliary video sequence
set of auxiliary pictures

2.3
bit depth
amount of data used to represent the value of an integer quantity (expressed in bits)

2.4
bitstream
sequence of bits that forms the coded representation, according to a specified format, of one or more *primary video* sequences, of one or more *auxiliary video sequences*, or of one or more supplemental information messages

2.5
bottom field
one of two *fields* that comprise a *frame*, each row of this *field* being spatially located immediately below a corresponding row of the *top field*

© ISO/IEC 2007 – All rights reserved

2.6
can
term used to refer to behaviour that is allowed or enabled, but not necessarily required

2.7
channel
collective term for luma or chroma

2.8
chroma, adj.
⟨sample array or single sample⟩ representing one of the two colour difference signals related to the primary colours

NOTE 1 The symbols used for a chroma array or sample are Cb and Cr.

NOTE 2 The term chroma is used rather than the term chrominance in order to avoid the implication of the use of linear light transfer characteristics that is often associated with the term chrominance.

2.9
depth
distance of an object relative to a given two-dimensional plane

2.10
field
assembly of alternate rows of a *frame*

NOTE A *frame* is composed of two *fields*: a *top field* and a *bottom field*.

2.11
frame
A *frame* contains an array of *luma* samples and two corresponding arrays of *chroma* samples. A *frame* consists of two *fields*, a *top field* and a *bottom field*

2.12
luma, adj.
⟨sample array or single sample⟩ representing the monochrome signal related to the primary colours

NOTE 1 The symbol or subscript used for luma is Y.

NOTE 2 The term luma is used rather than the term luminance in order to avoid the implication of the use of linear light transfer characteristics that is often associated with the term luminance.

2.13
parallax
apparent shift of an object due to a change in observer position

2.14
picture
collective term for a *field* or a *frame*

2.15
primary picture
picture to which an *auxiliary picture* is attached

2.16
primary video
video sequence to which an auxiliary video sequence is attached

IECNORM.COM : Click to view the full PDF of ISO/IEC 23002-3:2007

 © ISO/IEC 2007 – All rights reserved

2.17
raw byte sequence payload
RBSP
syntax structure containing an integer number of bytes

2.18
reserved
⟨values of a particular *syntax element*⟩ not to be used in *bitstreams* conforming to this part of ISO/IEC 23002, but may be used in future extensions of this part of ISO/IEC 23002 by ISO/IEC

2.19
stereoscopic pictures
pictures that provide a stereoscopic (*depth*) effect while viewing them on a stereoscopic display, for example constructed from a *primary video* and from an *auxiliary video sequence* representing its *depth* (or a parallax) map

2.20
shall
term used to express mandatory requirements for conformance to this part of ISO/IEC 23002

2.21
should
term used to refer to behaviour of an implementation that is encouraged to be followed under anticipated ordinary circumstances, but is not a mandatory requirement for conformance to this part of ISO/IEC 23002

2.22
syntax element
element of data represented in the *bitstream*

2.23
top field
one of two *fields* that comprise a *frame*, each row of this *field* being spatially located immediately above the corresponding row of the *bottom field*

2.24
video sequence
sequence of *pictures*

IECNORM.COM : Click to view the full PDF of ISO/IEC 23002-3:2007

3 Abbreviations

For the purpose of this document, the following abbreviations apply.

AVSI auxiliary video supplemental information

RBSP raw byte sequence payload

SI supplemental information

4 Conventions

4.1 Arithmetic operators

NOTE The mathematical operators used in this part of ISO/IEC 23002 are similar to those used in the C programming language. However, integer division and arithmetic shift operations are specifically defined. Numbering and counting conventions generally begin from 0.

© ISO/IEC 2007 – All rights reserved

The following arithmetic operators are defined as follows.

+ Addition

– Subtraction (as a two-argument operator) or negation (as a unary prefix operator)

* Multiplication

/ Integer division with truncation of the result toward zero. For example, 7/4 and (–7)/(–4) are truncated to 1 and (–7)/4 and 7/(–4) are truncated to –1.

÷ Used to denote division in mathematical equations where no truncation or rounding is intended.

$\frac{x}{y}$ Used to denote division in mathematical equations where no truncation or rounding is intended.

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

– 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 Relational operators

The following relational operators are defined as follows:

> Greater than

>= Greater than or equal to

< Less than

<= Less than or equal to

= =Equal to

!= Not equal to

When a relational operator is applied to a syntax element or variable that has been assigned the value "na" (not applicable), the value "na" is treated as a distinct value for the syntax element or variable. The value "na" is considered not to be equal to any other value.

4.3 Assignment operators

The following arithmetic operators are defined as follows:

= Assignment operator.

++ Increment, i.e., *x*++ is equivalent to *x* = *x* + 1; when used in an array index, evaluates to the value of the variable prior to the increment operation.

–– Decrement, i.e., *x*–– is equivalent to *x* = *x* – 1; when used in an array index, evaluates to the value of the variable prior to the decrement operation.

+= Increment by amount specified, i.e., x += 3 is equivalent to x = x + 3, and x += (-3) is equivalent to x = x + (-3).

–= Decrement by amount specified, i.e., x –= 3 is equivalent to x = x – 3, and x –= (-3) is equivalent to x = x – (-3).

IECNORM.COM : Click to view the full PDF of ISO/IEC 23002-3:2007

© ISO/IEC 2007 – All rights reserved

4.4 Variables, syntax elements, and tables

Syntax elements in the bitstream are represented in **bold** type. Each syntax element is described by its name (all lower case letters with underscore characters), and its descriptor for its method of coded representation. The decoding process behaves according to the value of the syntax element and to the values of previously decoded syntax elements. When a value of a syntax element is used in the syntax tables or the text, it appears in regular (i.e., not bold) type.

In some cases the syntax tables may use the values of other variables derived from syntax elements values. Such variables appear in the syntax tables, or text, named by a mixture of lower case and upper case letter and without any underscore characters. Variables starting with an upper case letter are derived for the decoding of the current syntax structure and all depending syntax structures. Variables starting with an upper case letter may be used in the decoding process for later syntax structures mentioning the originating syntax structure of the variable. Variables starting with a lower case letter are only used within the subclause in which they are derived.

In some cases, "mnemonic" names for syntax element values or variable values are used interchangeably with their numerical values. Sometimes "mnemonic" names are used without any associated numerical values. The association of values and names is specified in the text. The names are constructed from one or more groups of letters separated by an underscore character. Each group starts with an upper case letter and may contain more upper case letters.

NOTE The syntax is described in a manner that closely follows the C-language syntactic constructs.

Functions are described by their names, which are constructed as syntax element names, with left and right round parentheses including zero or more variable names (for definition) or values (for usage), separated by commas (if more than one variable).

Binary notation is indicated by enclosing the string of bit values by single quote marks. For example, '01000001' represents an eight-bit string having only its second and its last bits equal to 1.

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

Numerical values not enclosed in single quotes are decimal values.

A value equal to 0 represents a FALSE condition in a test statement. The value TRUE is represented by any other value different than zero.

4.5 Method of specifying syntax in tabular form

The syntax tables specify a superset of the syntax of all allowed bitstreams. Additional constraints on the syntax may be specified, either directly or indirectly, in other clauses.

The following table lists examples of pseudo code used to describe the syntax. When **syntax_element** appears, it specifies that a syntax element is parsed from the bitstream and the bitstream pointer is advanced to the next position beyond the syntax element in the bitstream parsing process.

	Descriptor
/* A statement can be a syntax element with an associated syntax category and descriptor or can be an expression used to specify conditions for the existence, type, and quantity of syntax elements, as in the following two examples */	
syntax_element	u (n)

IECNORM.COM : Click to view the full PDF of ISO/IEC 23002-3:2007

conditioning statement	
/* A group of statements enclosed in curly brackets is a compound statement and is treated functionally as a single statement. */	
{	
statement	
statement	
…	
}	
/* A "while" structure specifies a test of whether a condition is true, and if true, specifies evaluation of a statement (or compound statement) repeatedly until the condition is no longer true */	
while(condition)	
statement	
/* A "do … while" structure specifies evaluation of a statement once, followed by a test of whether a condition is true, and if true, specifies repeated evaluation of the statement until the condition is no longer true */	
do	
statement	
while(condition)	
/* An "if … else" structure specifies a test of whether a condition is true, and if the condition is true, specifies evaluation of a primary statement, otherwise, specifies evaluation of an alternative statement. The "else" part of the structure and the associated alternative statement is omitted if no alternative statement evaluation is needed */	
if(condition)	
primary statement	
else	
alternative statement	

IECNORM.COM : Click to view the full PDF of ISO/IEC 23002-3:2007

© ISO/IEC 2007 – All rights reserved

/* A "for" structure specifies evaluation of an initial statement, followed by a test of a condition, and if the condition is true, specifies repeated evaluation of a primary statement followed by a subsequent statement until the condition is no longer true. */	
for(initial statement; condition; subsequent statement)	
primary statement	

4.6 Specification of syntax functions, categories, and descriptors

The functions presented here are used in the syntactical description. These functions assume the existence of a bitstream pointer with an indication of the position of the next bit to be read by the decoding process from the bitstream.

next_bits(n) provides the next bits in the bitstream for comparison purposes, without advancing the bitstream pointer. Provides a look at the next n bits in the bitstream with n being its argument.

read_bits(n) reads the next n bits from the bitstream and advances the bitstream pointer by n bit positions. When n is equal to 0, read_bits(n) is specified to return a value equal to 0 and to not advance the bitstream pointer.

The following descriptors specify the parsing process of each syntax element.

— b(8): byte having any pattern of bit string (8 bits). The parsing process for this descriptor is specified by the return value of the function read_bits(8).

— f(n): fixed-pattern bit string using n bits written (from left to right) with the left bit first. The parsing process for this descriptor is specified by the return value of the function read_bits(n).

— u(n): unsigned integer using n bits. The parsing process for this descriptor is specified by the return value of the function read_bits(n) interpreted as a binary representation of an unsigned integer with most significant bit written first.

5 Auxiliary video stream

5.1 Definition

An auxiliary video stream is a coded representation of an auxiliary video sequence (as defined in subclause 2). The SI messages currently specified herein relate to the interpretation of a single channel of the auxiliary video sequence. If the auxiliary video stream contains a luma channel and some additional channels (for example, chroma channels), the SI messages specified herein are to be interpreted as relating to the luma channel, and the content of the other channels may be ignored (for purposes of interpreting the data specified herein).

NOTE 1 This part of ISO/IEC 23002 currently supports auxiliary video sequences mapped on one channel only.

NOTE 2 The use of the chroma might be defined in future amendments to this part of ISO/IEC 23002.

NOTE 3 Identifying the auxiliary video stream and determining how it is represented is out of the scope of this part of ISO/IEC 23002. This is determined by the application and its relevant specifications. For example, ITU-T Rec. H.222.0 | ISO/IEC 13818-1 provides a descriptor for auxiliary video streams that contains this information.

The value of an auxiliary video sample is noted by the symbol m in this part of ISO/IEC 23002. The bit depth N of m shall be inferred from the decoding process of the auxiliary video stream.

IECNORM.COM : Click to view the full PDF of ISO/IEC 23002-3:2007

© ISO/IEC 2007 – All rights reserved

5.2 Use of supplemental information

An auxiliary video stream should be accompanied by a supplemental information (SI) RBSP (specified in subclause 6.1) containing at least one auxiliary video supplemental information (AVSI) message (specified in subclauses 6.1.2 and 6.2.2 through the parameter is_avsi).

If the SI RBSP contains more than one AVSI message:

- if the first one has a value of payloadType equal to 0 and if the second one has a value of payloadType equal to 1 and is immediately following the first one, then both AVSI messages are relevant and either one may be used. All but these first two AVSI messages shall be ignored.

NOTE Interpreting the same auxiliary video channel equally as depth or as parallax is reasonable for small depth effects because of the linear correspondence between them. See Annex A and Annex B.

- otherwise, all but the first AVSI message shall be ignored.

The use of more AVSI messages and the use of SI messages that are not AVSI messages are reserved for future extensions of this part of ISO/IEC 23002.

The sample values m of an auxiliary video picture are interpreted according to the value of payloadType in the AVSI message. Table 1 lists for each AVSI payload type and the corresponding type of auxiliary video. Values of payloadType other than those defined in this table are reserved for future extensions of this part of ISO/IEC 23002. If a decoder encounters an AVSI message with a reserved payload type, it shall ignore the AVSI message.

IECNORM.COM : Click to view the full PDF of ISO/IEC 23002-3:2007

Table 1 — Auxiliary video type codes

payloadType	Type of auxiliary video
0	Depth map
1	Parallax map
Other values	Reserved

This part of ISO/IEC 23002 does not specify how the supplemental information (SI) message stream is conveyed to the decoder or how the primary and auxiliary video streams are identified or represented. Such issues are outside the scope of this part of ISO/IEC 23002 and are determined by the application and its relevant specifications.

5.3 Alignment of auxiliary and primary videos

The primary video and the auxiliary video might be spatially and/or temporally misaligned due to:

- interlaced/progressive mismatch,
- different spatial resolutions,
- different temporal resolutions.

Although the re-sampling process is voluntarily left open, the minimal constraints specified in this subclause should be met to ensure a correct matching of the primary and auxiliary samples.

NOTE 1 The spatial and temporal resolutions are those at the output of the decoding process, before any post-processing or rendering operation.

NOTE 2 For both primary and auxiliary video data, padding may have been required prior to encoding to ensure that the coded picture dimensions contain an integer number of macroblocks. Cropping the decoded pictures to the useful size is already specified by most video coding standards: display_horizontal_size and display_vertical_size in the picture display extension of MPEG-2 Video, frame_cropping_flag in the sequence header of MPEG-4 part 10…).

© ISO/IEC 2007 – All rights reserved

5.3.1 Frame/field alignment

Field/frame alignment is provided through the syntax elements aux_is_one_field, aux_is_bottom_field and aux_is_interlaced, which are part of AVSI messages as specified in subclause 6.1.2.3. Their semantics are defined in subclause 6.2.2.3.

5.3.2 Spatial alignment

Spatial alignment is provided through the two syntax elements position_offset_h and position_offset_v, which are part of AVSI messages as specified in subclause 6.1.2.3. Their semantics are defined in subclause 6.2.2.3.

5.3.3 Temporal alignment

The temporal synchronization between the primary and the auxiliary videos shall be conveyed by means beyond the scope of this part of ISO/IEC 23002 (for instance through time stamps in ITU-T Rec. H.222.0 | ISO/IEC 13818-1).

6 Supplemental information (SI)

6.1 Syntax

The data in the SI stream shall conform to the syntax constraints specified in this subclause.

si_rbsp(NumBytesInSI) {	Descriptor
NumBytesInRBSP = 0	
while(NumBytesInRBSP < NumBytesInSI)	
si_message( )	
}	

6.1.1 Supplemental information message syntax

si_message( ) {	Descriptor
payloadType = 0	
while(next_bits(8) = = 0xFF) {	
ff_byte /* equal to 0xFF */	f(8)
NumBytesInRBSP ++	
payloadType += 255	
}	
last_payload_type_byte	u(8)
NumBytesInRBSP ++	

IECNORM.COM : Click to view the full PDF of ISO/IEC 23002-3:2007

payloadType += last_payload_type_byte	
payloadSize = 0	
while(next_bits(8) = = 0xFF) {	
ff_byte /* equal to 0xFF */	f(8)
NumBytesInRBSP ++	
payloadSize += 255	
}	
last_payload_size_byte	u(8)
NumBytesInRBSP ++	
payloadSize += last_payload_size_byte	
si_payload(payloadType, payloadSize)	
NumBytesInRBSP += payloadSize	
}	

6.1.2 Supplemental information payload syntax

si_payload(payloadType, payloadSize) {	Descriptor
is_avsi = FALSE	
if(payloadType == 0 || payloadType == 1) {	
is_avsi = TRUE	
generic_params( )	
}	
if(payloadType == 0)	
depth_params( )	
else if(payloadType == 1)	
parallax_params( )	
else	
reserved_si_message(payloadType, payloadSize)	
}	

IECNORM.COM : Click to view the full PDF of ISO/IEC 23002-3:2007

 © ISO/IEC 2007 – All rights reserved

6.1.2.1 Depth map parameters syntax

depth_params( ) {	Descriptor
nkfar	u(8)
nknear	u(8)
}	

6.1.2.2 Parallax map parameters syntax

parallax_params( ) {	Descriptor
parallax_zero	u(16)
parallax_scale	u(16)
dref	u(16)
wref	u(16)
}	

6.1.2.3 Generic parameters syntax

generic_params( ) {	Descriptor
aux_is_one_field	u(1)
if (aux_is_one_field) {	
aux_is_bottom_field	u(1)
}	
else {	
aux_is_interlaced	u(1)
}	
reserved_generic_bits	f(6)
position_offset_h	u(8)
position_offset_v	u(8)
}	

IECNORM.COM : Click to view the full PDF of ISO/IEC 23002-3:2007

© ISO/IEC 2007 – All rights reserved

6.1.2.4 Reserved SI message syntax

reserved_si_message(payloadType, payloadSize) {	Descriptor
for(i = 0; i < payloadSize; i++)	
reserved_si_byte	b(8)
}	

6.2 Semantics

An SI RBSP contains one or more SI messages.

NumBytesInSI is the size of the SI RBSP in bytes. Some means for determining this value is necessary for decoding of the SI RBSP. The means for determining the value of NumBytesInSI is determined by the application and is outside the scope of this part of ISO/IEC 23002.

NOTE ITU-T Rec. H.222.0 | ISO/IEC 13818-1 provides a means for specifying the value of NumBytesInSI.

6.2.1 Supplemental information message semantics

Each SI message contains variables specifying the type payloadType and size payloadSize of the SI payload. The derived SI payload size payloadSize is specified in bytes and shall be equal to the number of bytes in the SI payload.

ff_byte is a byte equal to 0xFF identifying a need for a longer representation of the syntax structure that it is used within.

last_payload_type_byte is the last byte of the payload type of an SI message.

last_payload_size_byte is the last byte of the size of an SI message.

6.2.2 Supplemental information payload semantics

is_avsi is a boolean. If it is TRUE, it indicates that the current si_message() is an AVSI message.

6.2.2.1 Depth parameters semantics

NOTE 1 Depth maps are one type of auxiliary video data, addressing stereoscopic video applications. The stereoscopic view-rendering from the primary video and the depth map is left open to implementers. Depending on the desired precision and complexity level, it can be done using exact calculations or approximations. Some examples are given in the non-normative Annex A.

nkfar represents the parameter kfar, which is inferred as follows:

$$kfar = \frac{nkfar}{16} \quad (7\text{-}1)$$

nknear represents the parameter knear, which is inferred as follows:

$$knear = \frac{nknear}{64} \quad (7\text{-}2)$$

IECNORM.COM : Click to view the full PDF of ISO/IEC 23002-3:2007

 © ISO/IEC 2007 – All rights reserved

A depth value z_p is represented by an unsigned N-bit value m, as generically specified in clause 5. z_p should non-normatively be inferred as follows:

$$z_{\mathrm{p}} = \frac{m}{2^N}(knear * W + kfar * W) - kfar * W \quad (7\text{-}3)$$

kfar and knear specify the range of the depth information respectively behind and in front of the picture relatively to W. W represents the screen width at the receiver side. W and z_p is expressed using the same distance units.

NOTE 2 W is receiver-dependent and can also be viewed as an arbitrary scaling control on the range of depth to be displayed.

NOTE 3 Negative (respectively positive) values of zp should correspond to a position behind the display (respectively in front of the display). If zp equals zero, the corresponding position should be on the display.

NOTE 4 In general it is recommended kfar has a greater value than knear, to accommodate the need for a depth range behind the picture that is larger than the range in front of the picture. For instance nkfar = 128 and nknear = 128 correspond to kfar = 8 and knear = 2.

6.2.2.2 Parallax parameters semantics

NOTE 1 Parallax maps are one type of auxiliary video data. As depth maps, they address stereoscopic video applications. The parallax data is transmitted as the reference parallax for a reference spectator. To correctly interpret a parallax value from its representation, some metadata are needed - including the parameters of the reference spectator. For optimal advantage of this type of representation, the reference spectator set-up should be as close as possible to what is expected to be the real situation when presenting the given footage. Background information and examples can be found in the non-normative Annex B.

parallax_zero defines the value for which the parallax is null.

parallax_scale is a scaling factor that defines the dynamic range of the decoded parallax values.

wref is the reference spectator's monitor width given in cm.

dref is the reference spectator's viewing distance given in cm.

Furthermore the reference spectator's eye-distance xB_{ref} is fixed and its value is 65 mm.

A parallax value p_{ref} for a reference stereoscopic set-up is represented by an unsigned N-bit value m, as generically specified in subclause 5.1. p_{ref} should non-normatively be inferred as follows:

$$p_{ref} = \frac{(m - parallax\_zero)}{2^N} * \frac{parallax\_scale}{256} * \frac{wref}{8} \quad (7\text{-}4)$$

Negative (respectively positive) values of p_{ref} indicate a left shift (respectively a right shift). If p_{ref} equals zero, there is no sample position shift.

NOTE 2 The calculated parallax p_{ref} is normalized for the reference spectator with his eye-distance xB_{ref}, his viewing-distance dref and monitor width wref. So at the receiver side the actual parallax value p has to be transformed according to the actual configuration (W,D,xB). The calculation of p is left open to implementers. Depending on the desired precision and complexity level, it can be done using exact calculations or approximations (some examples are given in the non-normative Annex B).

NOTE 3 The factor of 1/8 used with W_{ref} makes sure that the realistic parallaxes can be coded using mid-ranged values. For the nominal values for parallax_zero and parallax_scale, and a bit depth of 8, this scaling leads to a parallax range that is 12 % of screen width.

NOTE 4 As a realistic example: parallax_zero = 2^{N-1}, parallax_scale = 256, dref = 300 and wref = 100 represent a reference monitor with a width of 1m, a viewing distance of 3m, and a symmetric dynamic range for depth relatively to the monitor.

IECNORM.COM : Click to view the full PDF of ISO/IEC 23002-3:2007

6.2.2.3 Generic parameters semantics

NOTE 1 The syntax elements of generic_params() allow a precise alignment of the auxiliary video with the primary one (for instance in case of sub-sampling), as specified in subclause 5.3.

aux_is_one_field is a boolean. If it is TRUE, it indicates that the auxiliary picture relates only to one field of the primary picture. Otherwise, it indicates that auxiliary picture relates to both fields of the primary picture.

aux_is_bottom_field is a boolean. If it is TRUE, the auxiliary video data corresponds only to the bottom field of the primary video. If FALSE, the auxiliary video data corresponds only to the top field of the primary video. If aux_is_one_field is FALSE, aux_is_bottom_field is not applicable.

aux_is_interlaced is a boolean. If it is TRUE, any spatial re-sampling operation on the auxiliary video should be field-based. If it is FALSE, any spatial re-sampling operation on the auxiliary video should be frame-based. If aux_is_one_field is TRUE, aux_is_interlaced is inferred to be TRUE.

position_offset_h and **position_offset_v** respectively represent horizontal and vertical position offsets of the auxiliary video data expressed in $1/16^{th}$ sample position in the primary video spatial sampling grid (see Figure 1). The latter grid is that of primary frame pictures if primary_is_interlaced is FALSE, that of primary field pictures if primary_is_interlaced is TRUE.

NOTE 2 When an auxiliary video has a lower spatial resolution than its primary video, a filtering and decimation process has been performed, causing shifting. Hence the usefulness of position_offset_h and position_offset_v.

A bitstream conforming to this International Standard should respect the following constraints:

If the auxiliary and the primary video have the same width, then position_offset_h should be zero.

If aux_is_one_field is FALSE, and if the auxiliary and the primary video frames have the same height, then position_offset_v should be set to zero.

If aux_is_one_field is TRUE, and if the auxiliary frames have half the height of the primary video ones, then position_offset_v should be set to zero.

IECNORM.COM : Click to view the full PDF of ISO/IEC 23002-3:2007

 © ISO/IEC 2007 – All rights reserved

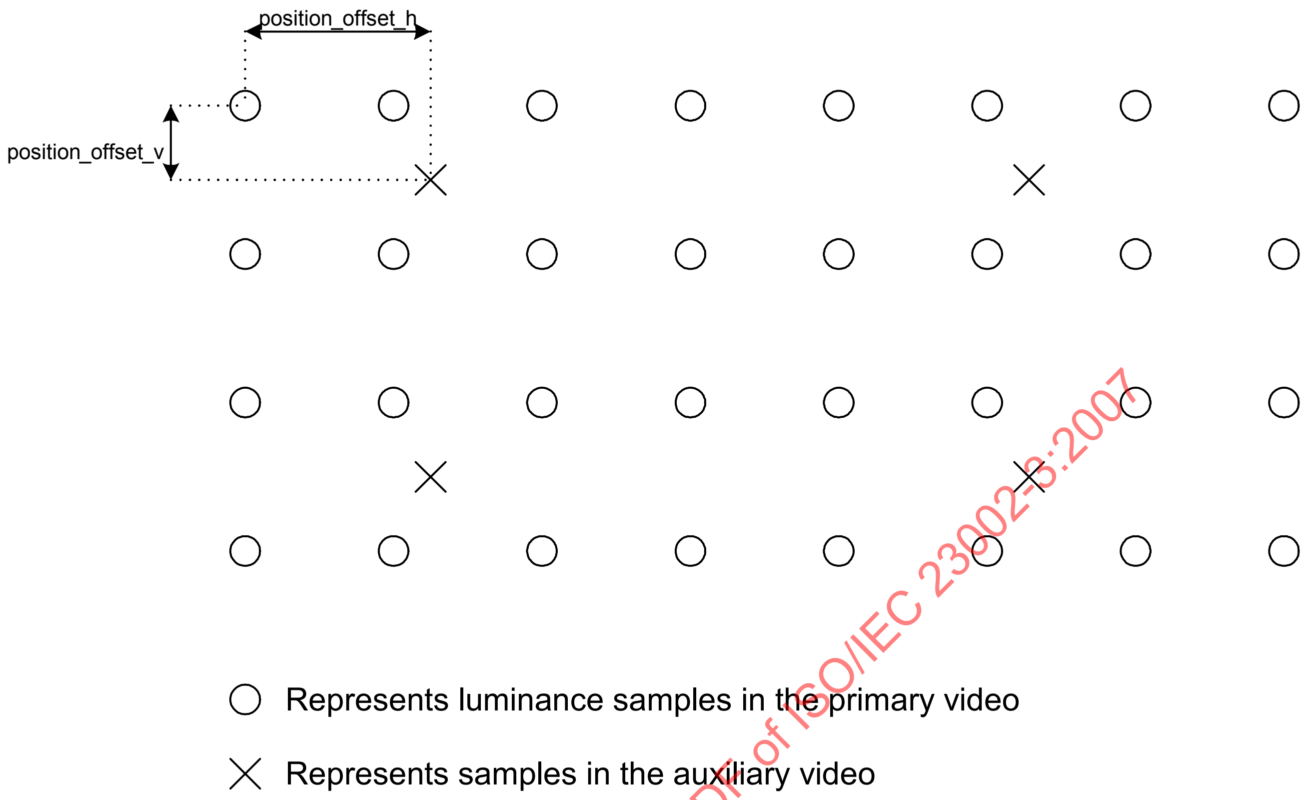

Figure 1 — Position offset between the conventional 2D video and the auxiliary video.

IECNORM.COM : Click to view the full PDF of ISO/IEC 23002-3:2007

NOTE 3 In the example depicted in Figure 1, the auxiliary video data is sub-sampled by a factor of 4 horizontally and a factor of 2 vertically, compared to the spatial resolution of the primary video. The offsets due to the sub-sampling filters correspond here to: position_offset_h = 20 and position_offset_v = 8 (expressed in $1/16^{th}$ sample position, i.e. respectively 1.25 sample position and 0.5 sample position).

6.2.2.4 Reserved SI message syntax

This message consists of data reserved for future backward-compatible use by ISO/IEC. Encoders conforming to this part of ISO/IEC 23002 shall not send reserved SI messages until and unless the use of such messages has been specified by ISO/IEC. Decoders conforming to this part of ISO/IEC 23002 that encounter reserved SI messages shall discard their content without effect on the decoding process, except as may be specified in a future revised version of this part of ISO/IEC 23002 by ISO/IEC.

reserved_si_message_payload_byte is a byte reserved for uses to be determined in the future by ISO/IEC.

© ISO/IEC 2007 – All rights reserved

Annex A
(informative)

Usage of depth map streams for stereo and auto-stereoscopic viewing

A.1 Introduction

This Annex gives background information on depth maps and their usage.

Given a picture I and a depth map z at the receiver side, a new picture can be created by shifting the viewpoint. The resulting sample position shift on the display is called "screen parallax". This allows generating different images for the left and right eye of the viewer, giving the impression of a 3D scene with a depth effect. Systems that require the viewer to wear glasses are called "stereoscopic" viewing systems, whereas other systems are called "auto-stereoscopic".

The following subclauses give examples of screen parallax calculation from depth.

A.2 Deriving Screen Parallax from Depth — Geometric Approach

In this non-normative subclause the geometrical relation between depth and parallax is calculated, independently of any coding and representation.

A.2.1 Exact non-linear formulation

Figure A.1 illustrates the geometry on a horizontal plane. The origin of the coordinate system is taken at the display. The problem can be seen as the x-axis being a frosted glass and how the object is mapped onto the frosted glass as seen from the 2 different viewpoints.

The z-coordinate points towards the viewer. The central image is assumed to be viewed by the left-eye which is located at (0,D). A right-eye image can now be created by selecting a new viewpoint at location (x_B;D). Figure A.1 shows that the same object point (x_p,z_p) is shifted to the right on the display. This effect is called screen parallax.

Let x_L denote the coordinate of the left-eye view and let x_R denote the coordinate of the right-eye view for that object on the display. Using equal triangles, the following equation applies:

$$\frac{x_L}{D} = \frac{x_p}{D - z_p} \qquad \text{(A-1)}$$

And:

$$\frac{x_R - x_B}{D} = \frac{x_p - x_B}{D - z_p} \qquad \text{(A-2)}$$

IECNORM.COM : Click to view the full PDF of ISO/IEC 23002-3:2007

 © ISO/IEC 2007 – All rights reserved

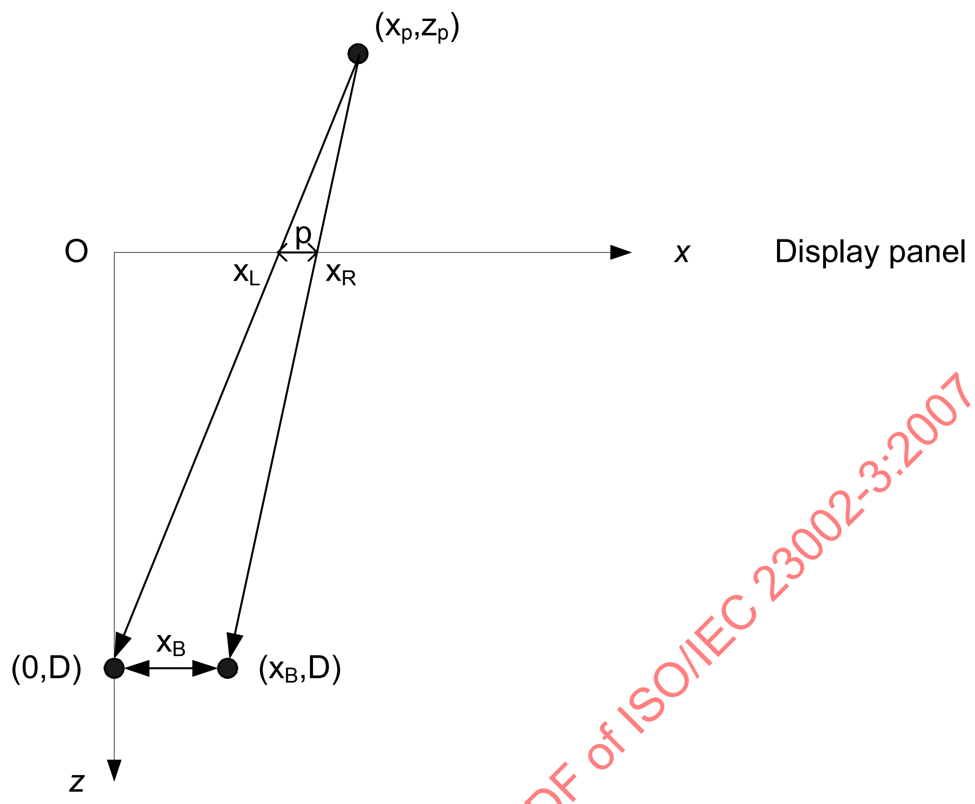

Figure A.1 — Stereoscopic viewing

IECNORM.COM : Click to view the full PDF of ISO/IEC 23002-3:2007

From (A-1) and (A-2):

$$\frac{x_L - x_R + x_B}{D} = \frac{x_B}{D - z_p} \qquad \text{(A-3)}$$

So the parallax $p=x_R-x_L$ is given by:

$$\boxed{p = x_B\left(1 - \frac{D}{D - z_p}\right)} \qquad \text{(A-4)}$$

Equation (A-4) indicates that, in its exact formulation, parallax is a non-linear function of the z-coordinate.

A.2.2 Linear approximation

In many applications, the amount of depth rendering z_p is relatively small compared to the viewing distance D, allowing a linear approximation for equation (A-4). This linear approximation may yield an advantage when implementing the algorithm in a computer or any other electronic device.

$$\boxed{p \approx -\frac{x_B}{D} z_p}. \qquad \text{(A-5)}$$

© ISO/IEC 2007 – All rights reserved

Figure A.2 compares the approximate relation of equation (A-5) with the exact relation of equation (A-4). A typical eye separation of $x_B = 6$ cm was selected as baseline and a typical living room viewing distance of $D = 250$ cm was used to generate Figure A.2. In such conditions, for comfortable viewing, objects are usually displayed less than 50 cm from the display. In this range, the distortion in depth due to the linear relation between parallax and depth is rather limited.

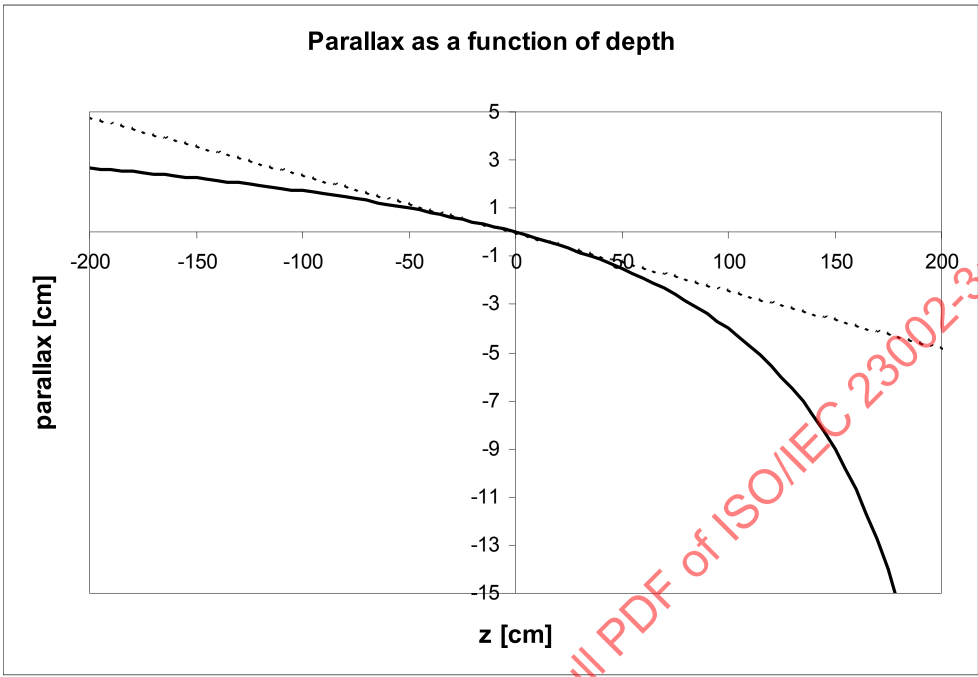

Figure A.2 — Comparison of approximation (broken line) with the exact (solid line) relation between screen parallax and depth

IECNORM.COM : Click to view the full PDF of ISO/IEC 23002-3:2007

A.3 Deriving screen parallax from a coded depth value

A.3.1 Exact non-linear formulation

The parallax as a function of the representation value *m* can be solved for the non-linear case as follows:

$$p = x_B \left(\frac{1}{\frac{W}{D}\left(\frac{m}{2^N}(knear + kfar) - kfar\right) - 1} + 1 \right) \quad \text{(A-6)}$$

A.3.2 Linear formulation

The linear approximation becomes:

$$p \approx -x_B \frac{W}{D}\left(\frac{m}{2^N}(knear + kfar) - kfar\right) \quad \text{(A-7)}$$

 © ISO/IEC 2007 – All rights reserved

A.3.3 Screen parallax expressed in pixels

Let N_{pix} be the number of horizontal pixels of the display, the parallax p_{pix} in units of pixels is expressed as follows:

$$p_{\text{pix}} = \frac{pN_{\text{pix}}}{W} \qquad \text{(A-8)}$$

Using the approximation of equation (A-7) the expression becomes:

$$p_{\text{pix}} \approx -x_{\text{B}} \frac{N_{\text{pix}}}{D} \left(\frac{m}{2^N} (knear + kfar) - kfar \right) \qquad \text{(A-9)}$$

IECNORM.COM : Click to view the full PDF of ISO/IEC 23002-3:2007

© ISO/IEC 2007 – All rights reserved

Annex B
(informative)

Reference spectator and parallax computation examples

B.1 Introduction

This Annex gives two examples showing how to calculate a screen parallax p at the receiver side from a transmitted value p_{ref} (as defined in subclause 6.2.2.2 of this part of ISO/IEC 23002). All notations and definitions from subclause 6.2.2.2 and Annex A are reused here.

Let us consider a reference setup according to Figure A.1, with reference values D_{ref}, W_{ref} and p_{ref}. At the receiver side, ones knows the actual values of D and W, retrieves D_{ref} and W_{ref} through the SI message defined in clause 6.2.2.2 and gets the value of p_{ref} by decoding the auxiliary video stream. As in Annex A, the problem is to calculate the screen parallax p to be applied.

IECNORM.COM : Click to view the full PDF of ISO/IEC 23002-3:2007

B.2 General case

B.2.1 Exact non-linear formulation

Using (Equation A-4):

$$p_{ref} = x_B \left(1 - \frac{D_{ref}}{D_{ref} - z_{ref}} \right) \qquad \text{(B-1)}$$

This can also be written as:

$$z_{ref} = \frac{D_{ref} \cdot p_{ref}}{p_{ref} - x_B} \qquad \text{(B-2)}$$

At the receiver side, the depth is scaled according to the actual display width W:

$$z = \frac{W}{W_{ref}} * z_{ref} \qquad \text{(B-3)}$$

So the screen parallax is expressed as follows:

$$\boxed{p = x_B \left(1 - \frac{D}{D - (\frac{D_{ref}}{W_{ref}} * \frac{W \cdot p_{ref}}{p_{ref} - x_B})} \right)} \qquad \text{(B-4)}$$

 © ISO/IEC 2007 – All rights reserved

B.2.2 Linear formulation

If the depth is small compared to the viewing distance, then:

$$p \approx -p_{ref} * \frac{dref}{wref} * \frac{W}{D} * \frac{x_B}{p_{ref} - x_B} \qquad \text{(B-5)}$$

B.3 Specific examples

B.3.1 Doubling screen width, depth and viewing distance

In this example, dimensions are doubled compared to the reference set-up. In order to keep the perspective correct everything must be scaled accordingly. Because the spectator's eye distance remains the same, the parallax p to be applied is p_{ref}, as depicted in Figure B.1.

Of course any global scaling by a factor α would give the same result:

$$p = x_B \left(1 - \frac{\alpha D_{ref}}{\alpha D_{ref} - (\frac{D_{ref}}{W_{ref}} * \frac{\alpha W_{ref} \cdot p_{ref}}{p_{ref} - x_B})} \right) = p_{ref} \qquad \text{(B-6)}$$

B.3.2 Doubling depth while keeping the same viewing distance

The case when zooming width and depth but not viewing distance is depicted by Figure B.2.

In this example the viewing distance D is equal to D_{ref}, so:

$$p = x_B \left(\frac{1}{1 - \frac{W_{ref}}{W} * \frac{p_{ref} - x_B}{p_{ref}}} \right) \qquad \text{(B-7)}$$

If moreover the depth is small compared to the viewing distance, then (see Equation B-5):

$$p = \frac{W}{W_{ref}} * p_{ref} \qquad \text{(B-8)}$$

IECNORM.COM : Click to view the full PDF of ISO/IEC 23002-3:2007

© ISO/IEC 2007 – All rights reserved