INTERNATIONAL STANDARD

ISO/IEC 18477-5

First edition
2018-03

Information technology — Scalable compression and coding of continuous-tone still images —

Part 5: Reference software

Technologies de l'information — Compression échelonnable et codage d'images plates en ton continu —

Partie 5: Logiciel de référence

Reference number
ISO/IEC 18477-5:2018(E)

ISO copyright office
CP 401 • Ch. de Blandonnet 8
CH-1214 Vernier, Geneva
Phone: +41 22 749 01 11
Fax: +41 22 749 09 47
Email: copyright@iso.org
Website: www.iso.org

Published in Switzerland

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.

The procedures used to develop this document and those intended for its further maintenance are described in the ISO/IEC Directives, Part 1. In particular the different approval criteria needed for the different types of document should be noted. This document was drafted in accordance with the editorial rules of the ISO/IEC Directives, Part 2 (see www.iso.org/directives).

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. Details of any patent rights identified during the development of the document will be in the Introduction and/or on the ISO list of patent declarations received (see www.iso.org/patents).

Any trade name used in this document is information given for the convenience of users and does not constitute an endorsement.

For an explanation on the voluntary nature of standards, the meaning of ISO specific terms and expressions related to conformity assessment, as well as information about ISO's adherence to the World Trade Organization (WTO) principles in the Technical Barriers to Trade (TBT) see the following URL: www.iso.org/iso/foreword.html.

This document was prepared by Technical Committee ISO/IEC JTC 1, *Information technology*, Subcommittee SC 29, *Coding of audio, picture, multimedia and hypermedia information*.

A list of all parts in the ISO/IEC 18477 series can be found on the ISO website.

Introduction

ISO/IEC 18477, also known under the name "JPEG XT", is a series of extensions of ISO/IEC 18477-1, a compression system for continuous tone digital still images which is backwards compatible with Rec. ITU-T T.81 | ISO/IEC 10918-1. That is, legacy applications conforming to Rec. ITU-T T.81 | ISO/IEC 10918-1 will be able to reconstruct streams generated by an encoder conforming to the ISO/IEC 18477 series, though will possibly not be able to reconstruct such streams in full dynamic range, full quality or other features defined in the ISO/IEC 18477 series.

This document offers implementations of various parts of the ISO/IEC 18477 standard that demonstrate the features and capabilities of JPEG XT. Its purpose is to act as a guideline for implementations and as a reference for conformance testing. As such, the implementations are conforming to the part of Rec. ITU-T T.81 | ISO/IEC 10918-1 that has been standardized as ISO/IEC 18477-1, i.e. it implements the baseline, extended sequential and progressive Huffman coding modes of the legacy standard together with common extensions such as Rec. ITU-T T.871 | ISO/IEC 10918-5, commonly known as JFIF. In addition, the reference software implementations also cover all other parts of the ISO/IEC 18477 standard, i.e., IDR coding, HDR coding, lossless and near-lossless coding and coding of alpha channels.

This document includes the source code for reference implementations of the ISO/IEC 18477 series of standards, available at http://standards.iso.org/iso-iec/18477/-5/ed-1/en. They have been successfully complied and tested on Linux[1)] and Windows™[2)] operating systems at the time of writing.

Note that ISO/IEC 18477-1 does not include the arithmetic coding modes, the hierarchical coding modes and the lossless coding modes of Rec. ITU-T T.81 | ISO/IEC 10918-1.

1) Linux is an example of a suitable product available commercially. This information is given for the convenience of users of this document and does not constitute an endorsement by ISO or IEC of this product.

2) Windows is an example of a suitable product available commercially. This information is given for the convenience of users of this document and does not constitute an endorsement by ISO or IEC of this product.

Information technology — Scalable compression and coding of continuous-tone still images —

Part 5:
Reference software

1 Scope

This document provides reference implementations of multiple parts of the ISO/IEC 18477 series, also known under the name "JPEG XT". JPEG XT is designed primarily for compression of continuous-tone photographic content.

2 Normative references

The following documents are referred to in the text in such a way that some or all of their content constitutes requirements of this document. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

ISO/IEC 18477-1, *Information technology — Scalable compression and coding of continuous-tone still images — Part 1: Scalable compression and coding of continuous-tone still images*

ISO/IEC 18477-3, *Information technology — Scalable compression and coding of continuous-tone still images — Part 3: Box file format*

ISO/IEC 18477-6, *Information technology — Scalable compression and coding of continuous-tone still images — Part 6: IDR Integer Coding*

ISO/IEC 18477-7, *Information technology — Scalable compression and coding of continuous-tone still images — Part 7: HDR Floating-Point Coding*

ISO/IEC 18477-8, *Information technology — Scalable compression and coding of continuous-tone still images — Part 8: Lossless and near-lossless coding*

ISO/IEC 18477-9, *Information technology — Scalable compression and coding of continuous-tone still images — Part 9: Alpha channel coding*

3 Terms and definitions

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

ISO and IEC maintain terminological databases for use in standardization at the following addresses:

— IEC Electropedia: available at http://www.electropedia.org/

— ISO Online browsing platform: available at http://www.iso.org/obp

3.1
codestream
sequence of bytes that conforms to or is to be checked for conformance with the codestream syntax specified in ISO/IEC 18477-1 and/or ISO/IEC 18477-3

3.2
decoder
embodiment of the decoding process specified in the ISO/IEC 18477 series or a process embodiment that is to be tested for conformance to the ISO/IEC 18477 series

3.3
encoder
process that produces *codestreams* (3.1) that conform to ISO/IEC 18477-1 or ISO/IEC 18477-3 or that are to be tested for conformance to the ISO/IEC 18477 series

3.4
pfm format
format similar to *ppm* (3.6) for carrying floating-point based colour images

Note 1 to entry: It is further specified in ISO/IEC 18477-4.

3.5
pnm format
superset of the *pfm* (3.4) and *ppm format* (3.6)

3.6
ppm format
portable pixmap format for carrying three-component integer sample based colour images specified in ISO/IEC 18477-4

4 Abbreviated terms

HDR high dynamic range

IDR intermediate dynamic range

LDR low dynamic range

PPM portable pixmap format

PFM portable floating-point format

TMO tone mapping operator

5 Conventions

5.1 Conformance language

This document consists of normative and informative text.

Normative text is that text which expresses mandatory requirements. The word "shall" is used to express mandatory requirements strictly to be followed in order to conform to this document and from which no deviation is permitted. A conforming implementation is one that fulfils all mandatory requirements.

Informative text is text that is potentially helpful to the user, but not indispensable and can be removed, changed or added editorially without affecting interoperability. All text in document is normative, with the following exceptions: the Introduction, any parts of the text that are explicitly labelled as "informative" and statements appearing with the preamble "NOTE" and behaviour described using the word "should". The word "should" is used to describe behaviour that is encouraged but is not required for conformance to this document.

The keywords "may" and "need not" indicate a course of action that is permissible in a conforming implementation.

The keyword "reserved" indicates a provision that is not specified at this time, shall not be used and may be specified in the future. The keyword "forbidden" indicates "reserved" and in addition indicates that the provision will never be specified in the future.

5.2 Typesetting

Regular face fonts as this text describe informative text that provides instructions, comments or details for the reader.

`Monospaced text as this paragraph indicates program input or output as necessary to either run the software or as generated by the software on the console.`

6 Reference software

6.1 Purpose

The purpose of this document is to provide the following.

— Reference decoder software capable of decoding codestreams that conform to ISO/IEC 18477-1, ISO/IEC 18477-3, ISO/IEC 18477-6, ISO/IEC 18477-7, ISO/IEC 18477-8 and/or ISO/IEC 18477-9.

— Sample encoder software capable of producing codestreams that conform to one or multiple parts of the ISO/IEC 18477 series.

The use of the reference software is not required for making an implementation of an encoder or decoder in conformance to any of the ISO/IEC 18477 series. Requirements established in ISO/IEC 18477-1, ISO/IEC 18477-3, ISO/IEC 18477-6, ISO/IEC 18477-7, ISO/IEC 18477-8 and ISO/IEC 18477-9 take precedence over the behaviour of the reference software.

6.2 Examples of use

Some examples of use for the reference decoder software implementations are as follows:

— as an illustration of how to perform the decoding processes specified in one or multiple parts of the ISO/IEC 18477 series;

— as the starting basis for the implementation of a decoder that conforms to one or multiple parts of the ISO/IEC 18477 series;

— for testing the conformance of a decoder implementation of one or multiple parts of the ISO/IEC 18477 series with the procedures specified in ISO/IEC 18477-4. Details on reference testing can be found in ISO/IEC 18477-4;

— for (non-exhaustive) testing of the conformance of a codestream (or file) to the constraints specified in ISO/IEC 18477-3, ISO/IEC 18477-6, ISO/IEC 18477-7, ISO/IEC 18477-8 or ISO/IEC 18477-9.

NOTE 1 Attempting to decode a codestream under testing with a reference software implementation implements only a non-exhaustive test for conformance. The lack of detection of any conformance violation by any reference software implementation can therefore not be considered as a definite proof that the codestream under testing conforms to all constraints required for it to be conforming to one of the ISO/IEC 18477 standards.

Some examples of use for a reference encoder software are as follows:

— as an illustration of how to implement an encoding process that produces codestreams that are, depending on the settings of the software, conforming to one or multiple members of the software of the ISO/IEC 18477 series;

— as starting point for an implementation of an encoder that conforms to one or multiple members of the ISO/IEC 18477 series;

— as a means of generating codestreams conforming to one or multiple parts of the ISO/IEC 18477 series for testing purposes;

— as a means of demonstrating and evaluating examples of the quality that can be achieved by an encoding process that conforms to multiple parts of the ISO/IEC 18477 series.

NOTE 2 However, no guarantee of the quality that will be achieved by an encoder is provided by its conformance to one or multiple parts of the ISO/IEC 18477 series as the conformance is only defined in terms of specific constraints imposed on the syntax of the generated codestream. In particular, while sample encoder software implementations may suffice to provide some illustrative examples of which quality can be achieved within the ISO/IEC 18477 series, they provide neither an assurance of minimum guaranteed image encoding quality nor maximum achievable image encoding quality.

NOTE 3 Similarly, the computation resource characteristics in terms of program or data memory usage, execution speed, etc. of sample software encoder or decoder implementations cannot be construed as a representative of the typical, minimal or maximal computational resource characteristics to be exhibited by implementations of some parts of the ISO/IEC 18477 series.

6.3 General

The reference software implementations for the ISO/IEC 18477 series are provided at http://standards.iso.org/iso-iec/18477/-5/ed-1/en.

— The file "reference1367abcd89.zip" contains a reference implementation for ISO/IEC 18477-1, ISO/IEC 18477-3, ISO/IEC 18477-6, ISO/IEC 18477-7 profiles A, B, C and D, ISO/IEC 18477-8 and ISO/IEC 18477-9. Unpacking and compilation of this software is explained in Annex A and guidance on how to use this software is given in Annex B.

— The file "reference7b.zip" contains a reference implementation for ISO/IEC 18477-7 profiles B. Unpacking and compilation of this software is explained in Annex C and guidance on how to use this software is given in Annex D.

Annex A
(informative)

Unpacking and compiling the reference software for ISO/IEC 18477-3 and ISO/IEC 18477-6 to ISO/IEC 18477-9

The source code of the software is provided in a ZIP archive at http://standards.iso.org/iso-iec/18477/-5/ed-1/en. Unpacking a ZIP file is operating system specific. Under POSIX compliant operating systems, open a command line window and enter

```
unzip reference1367abcd89.zip
```

This will unpack all components of the software into the current directory.

To compile the software, follow these steps:

— For POSIX compliant operating systems, change into the directory the electronic attachment was unpacked into, then enter on the command line

```
./configure
```

```
make
```

This assumes that a POSIX compliant shell is available and the GNU compiler (make, gcc compiler and linker) are installed on the system. The reference software will then be built in the current directory and a binary named “jpeg” will be created.

— For Microsoft Windows™[2], the Visual Studio[4] VS2010™ or VS2013™ compiler suite provides another option for compling the software. A VS2010 solution file allowing loading and compiling the project can be found in the directory “vs10.0/jpeg”. A solution file for VS2013 can be found in the directory “vs12.0/jpeg”.

The compiler will generate a command line tool without any graphical interface that compresses images represented in pnm (Picture AnyMap) into JPEG XT and expands JPEG XT images into pnm-files. pnm-files can either represent integer colour data, using the .ppm file extension or floating-point data using the .pfm extension. The .ppm files contain three (or one) big-endian integers per pixel and the .pfm files three (or one) 32-bit IEC big-endian floating-point integers. Converting these formats to other formats is out of the scope of this document.

NOTE Some implementations of PPM or PFM readers use little-endian encoding or save the image bitmap upside down, i.e. with the bottom line of the image included as first data in the file. Such encodings are not supported by this software.

Annex B
(informative)

Using the reference software for ISO/IEC 18477-3 and ISO/IEC 18477-6 to ISO/IEC 18477-9

B.1 General

This annex describes the usage of the software contained in the file reference1367abcd89.zip and prepared and compiled with the instructions given in Annex A.

The software can both compress images to JPEG XT files and expand such files back to pnm. The compressor and decompressor are both contained in the same executable named "jpeg". If a quality parameter (see below) is included on the command line or the command line parameters request lossless coding, the executable encodes images. If only an input and an output file are provided, the software attempts to expand the input file to the output file.

The compressor requires either one or two input images, both encoded in the ppm format. The first image is the HDR or IDR image, i.e. the original image that is to be compressed. The second image, to be provided as argument to the `-ldr` command line parameter, defines an 8-bit LDR image that defines the base image legacy implementations of Rec. ITU-T T.81 | ISO/IEC 10918-1 will fall back to.

If the `-ldr` command line parameter is omitted, the software itself creates an 8-bit tonemapped version of the input. It is noted that the images generated by the simple built-in operator are not ideal and external tonemappers are preferable for better results. Encoding the LDR/HDR image depends on the base image, containing the LDR image and an extension layer that allows the decoder to reconstruct an approximate version of the original image from the LDR image and the extension layer. Both base image and extension layer are controlled by independent quantizers, both of which are controlled by the command line through "quality" parameters.

The compressor also allows specifying separate quality parameters for the tonemapped image and the original image. Alternatively, a very simple built-in rate-allocation can provide a reasonable default for the quality selection between tonemapped and original image compressor.

The command line parameters for the compressors are as follows:

```
jpeg [options] source target
```

where `options` is a set of command line options, all starting with a dash, `source` is the source image to be expanded or compressed and `target` is the output file the compressor or expander provides its output in. Command line switches control the operations of the software. They are listed in the following subclauses.

Decoding from an JPEG XT file to a ppm or pfm file requires only the input and output file name:

```
jpeg [options] source target
```

The only command line option recognized for decoding is `-c`, which disables the transformation from YcbCr to RGB.

B.2 Options defining the quality of the base and full image and of the alpha channel

The following options control the quality of the encoded images.

`-q quality`	selects the encoding mode and defines the quality of the base image
`-Q quality`	defines the quality of the extension layer
`-quality q`	uses a profile and part specific weighting between base and extension layer quality
`-l`	enables lossless coding without a residual image by an int-to-int DCT, also requires -c and -q 100 for true lossless
`-rl`	enforces a int-to-int lossless DCT in the residual domain, for lossless coding enabled by -Q 100

B.3 Options defining the base image

The following options control the base image and its creation.

`-ldr file`	specifies a separate file containing the base layer for encoding
`-g gamma`	defines the exponent for the gamma for the LDR domain, or rather, for mapping HDR to LDR. A suggested value is 2.4 for mapping scRGB to sRBG. This option is only used if -ldr is missing to generate an LDR image from the HDR input. Use -g 0 for automatic tonemapping.
`-gf file`	defines the inverse one-point L-nonlinearity on decoding from a file This file contains one (ASCII encoded) digit per line, 256*2^h lines in total, where h is the number of refinement bits. Each line contains an (integer) output value the corresponding input is mapped to.
`-e exposure`	defines a manual exposure value for part 7 profile B. Auto-exposure, i.e. not using this parameter, is recommended.
`-ae factor`	defines an auto-exposure value for part 7 profile B. By default, this factor is 0.6.

B.4 Options controlling the colourspace

The following options control how out of gamut colours and gamut extension is handled.

`-c`	disables the RGB to YcbCr decorrelation transformation
`-xyz`	indicates that the HDR image is in the XYZ colourspace. Note that the image is not **converted** to this space, but is assumed to be encoded in this space.
`-cxyz`	similar to above, but uses the dedicated C transformation to implement a XYZ colourspace conversion.
`-ncl`	disables clamping of out-of-gamut colours. This is automatically enabled for lossless.

B.5 Options controlling the generation of additional JPEG XT specific scans

The following options control the generation of additional scans including additional data to enhance the bit precision. All parts and profiles that include a residual codestream, most notably all profiles of ISO/IEC 18477-7 except profile D, require the specification of the `-r` command line option to trigger the computation of a residual.

`-r`	enables the residual codestream for HDR and lossless coding, requires -q and -Q to define base and enhancement layer quality
`-r12`	uses a 12 bit residual image instead of an 8 bit residual image
`-R bits`	specifies refinement bits for the base image. This works like `-r` but in the DCT domain
`-rR bits`	specifies refinement bits for the residual image

B.6 Options controlling the profile and part

By default, the encoder creates an ISO/IEC 18477-1 conforming file. If the `-r`, `-l` or `-R` options are given, the part it complies to depends on additional parameters. With integer (ppm) input files, the output is conforming to ISO/IEC 18477-6. With `-r -Q` 100 or with `-l -q` 100 `-c`, the output is conforming to ISO/IEC 18477-8. For floating-point input, the profile is specified by an additional command line parameter.

`-profile X`	encodes according to profile X, X = a, b, c, d as defined in the JPEG XT standard. Without any parameter, the encoder tries to figure out the profile itself.

B.7 Options controlling the scan generation and entropy coding

The following parameters control the entropy coding and generation of scan patterns for JPEG and JPEG XT scans. Especially, the `-h` parameter (optimize Huffman tables) is recommended and for most configurations, even necessary as the legacy JPEG standard does not provide examples for default tables for most extended modes.

`-h`	optimizes the Huffman tables
`-v`	uses progressive instead of sequential encoding, available for all coding schemes (-r,-a,-l and default)
`-qv`	uses a simplified scan pattern for progressive that only AC from DC bands and may improve the performance
`-rv`	encodes the residual image in progressive coding mode.
`-rs`	encodes the residual image in sequential (rather than the modified residual)coding mode
`-ro`	disables the DCT in the residual domain, quantizes spatially for near-lossless coding

B.8 Options controlling the quantizer

The following parameters control the quantizer of both the base and the extension layer.

`-qt n` defines the quantization table. The following tables are currently defined:

- n = 0 the default tables from Annex K of the JPEG standard (default)
- n = 1 a completely flat table that should be PSNR-optimal
- n = 2 a MS-SSIM optimized table
- n = 3 the table suggested by ImageMagick
- n = 4 a HSV-PSNR optimized table
- n = 5 the table from Klein, Silverstein and Carney: Relevance of human vision to JPEG-DCT compression (1992)
- n = 6 the table from Watson, Taylor, Borthwick: DCTune perceptual optimization of compressed dental X-Rays (1997)
- n = 7 the table from Ahumada, Watson, Peterson: A visual detection model for DCT coefficient quantization (1993)
- n = 8 the table from Peterson, Ahumada and Watson: An improved detection model for DCT coefficient quantization (1993)

`-rqt` defines the quantization table for the residual stream in the same way

`-dz` improved deadzone quantizer, may help to improve the R/D performance

`-oz` optimizes quantizer, may help to improve the R/D performance

B.9 Options controlling the subsampling of components

The following parameters control the subsampling of components. Unlike other implementations, the reference software **does not** specify the subsampling by means of the MCU sizes, but directly by giving the subsampling factors. **By default, the reference software uses 444 subsampling.** 420 chroma subsampling is, for example, defined by the parameter `-s 1x1, 2x2, 2x2,` 422 subsampling by `-s 1x1, 2x1, 2x1`.

`-s WxH,…` defines subsampling factors for all components. Note that these are NOT MCU sizes. Default is 1x1,1x1,1x1 (444 subsampling). 1x1, 2x2, 2x2 is the 420 subsampling often used.

`-sr WxH,...` defines subsampling in the residual domain

B.10 Options controlling the generation of the alpha channel

The following set of parameters control the encoding of the alpha channel for ISO/IEC 18477-9 conforming coding. For alpha channels with more than 8 bit precision, the **-ar** command line option should be enabled to trigger the generation of a residual alpha channel. The alpha channel is defined by a one component pgm file that contains the sample values of the channel scaled to the sample precision of the file. For an 8-bit channel, 255 is the maximum opacity, 0 is full transparency.

-al file	specifies a one-component pgm/pfm file that contains an alpha component or the code will write the alpha component to. This demo code DOES NOT implement compositing of alpha and background.
-am mode	specifies the mode of the alpha: 1 (regular) 2 (premultiplied) 3 (matte-removal)
-ab r,g,b	specifies the matte (background) colour for mode 3 as RGB triple
-ar	enables residual coding for the alpha channel, required if the alpha channel is larger than 8bpp
-ar12	uses a 12-bit residual for the alpha channel
-aR bits	sets refinement bits in the alpha base codestream
-arR bits	sets refinement bits in the residual alpha codestream
-adz	enables the deadzone quantizer for the alpha channel
-aoz	enables the quantization optimization for the alpha channel
-aq qu	specifies a quality for the alpha base channel (usually the only one)
-aQ qu	specifies a quality for the alpha extension layer
-aqt n	specifiesthe quantization table for the alpha channel, the parameter n is selected as for the -qt option.
-arqt n	specifies the quantization table for residual alpha
-aquality q	specifies a combined quality for both
-aol	enables open loop coding for the alpha channel
-alo	disables the DCT in the residual alpha channel, quantizes spatially.
-all	enables lossless DCT for alpha coding
-adr	includes the de-ringing filter for the alpha channel

B.11 Options controlling the generation of the inverse TMO

The following parameters control how the inverse tone mapping lookup table for ISO/IEC 18477-6, ISO/IEC 18477-7 profile C and ISO/IEC 18477-8 shall be performed.

-sp	uses separate LUTs for each component
-md	uses the median instead of the centre of mass for constructing the inverse TMO of profile C.
-ct	uses the centre of mass instead of the median for constructing the inverse TMO of profile C.
-sm iter	uses <iter> iterations to smooth out the histogram for inverse-TMO based algorithms. Default is not to smooth the histogram.

B.12 Miscellaneous options

The following options control miscellaneous encoder options for all profiles and all parts. In specific, `-ol` controls whether the encoder uses an open-loop or closed-loop coding scheme. By default, the encoder is closed-loop, i.e. predicts residuals by going through a full encoder-decoder cycle. This is ideal for ISO/IEC 18477-6, ISO/IEC 18477-8 and ISO/IEC 18477-7 profile C. For ISO/IEC 18477-7 profiles A and B, open loop coding is recommended and the `-ol` command line option should be given.

`-z mcus`	defines the restart interval size, zero disables it
`-N`	enables noise shaping of the prediction residual. This only applies to 18477-8 encoding (lossless and near-lossless)
`-ol`	open loop encoding, residuals are based on original, not reconstructed
`-dr`	includes the optional de-ringing (Gibbs Phenomenon) filter on encoding
`-epsn eps`	defines the numerator normalizer for profile B encoding, defaults to 1e-7
`-epsd eps`	defines the denominator normalizer for profile B encoding, defaults to 1e-7

Annex C
(informative)

Unpacking and compiling the reference software for ISO/IEC 18477-7 profile B

The source code of the software is provided in a ZIP archive at http://standards.iso.org/iso-iec/18477/-5/ed-1/en. Unpacking a ZIP file is operating system specific. Under POSIX compliant operating systems, open a command line window and enter

```
unzip reference7b.zip
```

This will unpack all components of the software into the current directory.

To compile the software, follow these steps.

— For POSIX compliant operating systems, change into the directory the electronic attachment was unpacked into, then enter on the command line

```
./configure
```

```
make
```

This assumes that a POSIX compliant shell is available and the GNU compiler (make, gcc compiler and linker) are installed on the system. The reference software will then be built in the current directory and a binary named “jpeg” will be created.

— For Microsoft Windows™[2] the Visual Studio[4] VS2010™ or VS2013™ compiler suite offers another possibility to compile the software. A VS solution file allowing loading and compiling the project can be found in the directory "vs2010/" and "vs2013/".

— The compiler will generate a command line tool without any graphical interface that compresses images represented in pnm (Picture AnyMap) into JPEG XT and expands JPEG XT images into pnm-files. pnm-files can either represent integer colour data using the .ppm file extension or floating-point data using the .pfm extension. The .ppm files contain three (or one) big-endian integers per pixel, the .pfm files three (or one) 32-bit IEC big-endian floating-point integers. Converting these formats to other formats is out of the scope of this document.

NOTE Some implementations of PPM or PFM readers use little-endian encoding or save the image bitmap upside down, i.e. with the bottom line of the image included as first data in the file. Such encodings are not supported by this software.

Annex D
(informative)

Using the reference software for ISO/IEC 18477-7 profile B

D.1 General

This annex describes the usage of the software contained in the file reference7b.zip and prepared and compiled with the instructions given in Annex C.

The software can both compress images to JPEG XT files and expand such files back to pnm. The compressor and decompressor are both contained in the same executable named "jpeg". If a quality parameter (see below) is included on the command line or the command line parameters request lossless coding, the executable encodes images. If only an input and an output file are provided, the software attempts to expand the input file to the output file.

The compressor requires either one or two input images, both encoded in the ppm format. The first image is the HDR or IDR image, i.e. the original image that is to be compressed. The second image, to be provided as argument to the `-ldr` command line parameter, defines an 8-bit LDR image that defines the base image legacy implementations of Rec. ITU-T T.81 | ISO/IEC 10918-1 will fall back to.

If the `-ldr` command line parameter is omitted, the software itself creates an 8-bit tonemapped version of the input. It is noted that the images generated by the simple built-in operator are not ideal, and external tonemappers are preferable for better results. Encoding the LDR/HDR image depends on the base image, containing the LDR image and an extension layer that allows the decoder to reconstruct an approximate version of the original image from the LDR image and the extension layer. Both base image and extension layer are controlled by independent quantizers, both of which are controlled by the command line through "quality" parameters.

The compressor also allows specifying separate quality parameters for the tonemapped image and the original image. Alternatively, a very simple built-in rate-allocation can provide a reasonable default for the quality selection between tonemapped and original image compressor.

The command line parameters for the compressors are as follows:

```
jpeg [options] source target
```

where `options` is a set of command line options, all starting with a dash, `source` is the source image to be expanded or compressed and the `target` is the output file the compressor or expander provides its output in. Command line switches control the operations of the software. They are listed in the following subclauses.

Decoding from an JPEG XT file to a ppm or pfm file requires only the input and output file name:

```
jpeg [options] source target
```

The only command line option recognized for decoding is `-c`, which disables the transformation from YcbCr to RGB.